

2021년 5월 4일 랩미팅

백지윤

Done

- 지수 평활법 (ES)
- ES - Rnn
- (개인적인 공부) AlexNet

참고

<https://www.youtube.com/watch?v=noFCkN6gXZ4&list=PLpIPLT0Pf7lqSuMx237SHRdLd5ZA4AQwd&index=8>

을 통해서 지수 평활법을 공부하였습니다. 지수평활법 관련 내용은 모두 해당 강의의 *ppt* 를 참고하여 제가 복습해 본 것입니다.

지수 평활법

시계열 데이터 예측 방법은 ?

- 시계열회귀분석
- **지수평활법**
- 박스-젠킨스 시계열분석법
- 다양한 머신러닝 방법

구간 평균법 예 (N=3)

t	관측값	구간 평균치	예측값
0			
1	20		
2	60		
3	25	35.00	
4	70	51.67	
5	28	41.00	
6	72	56.67	
1			56.67
2			56.67

(20+60+25)/3=35

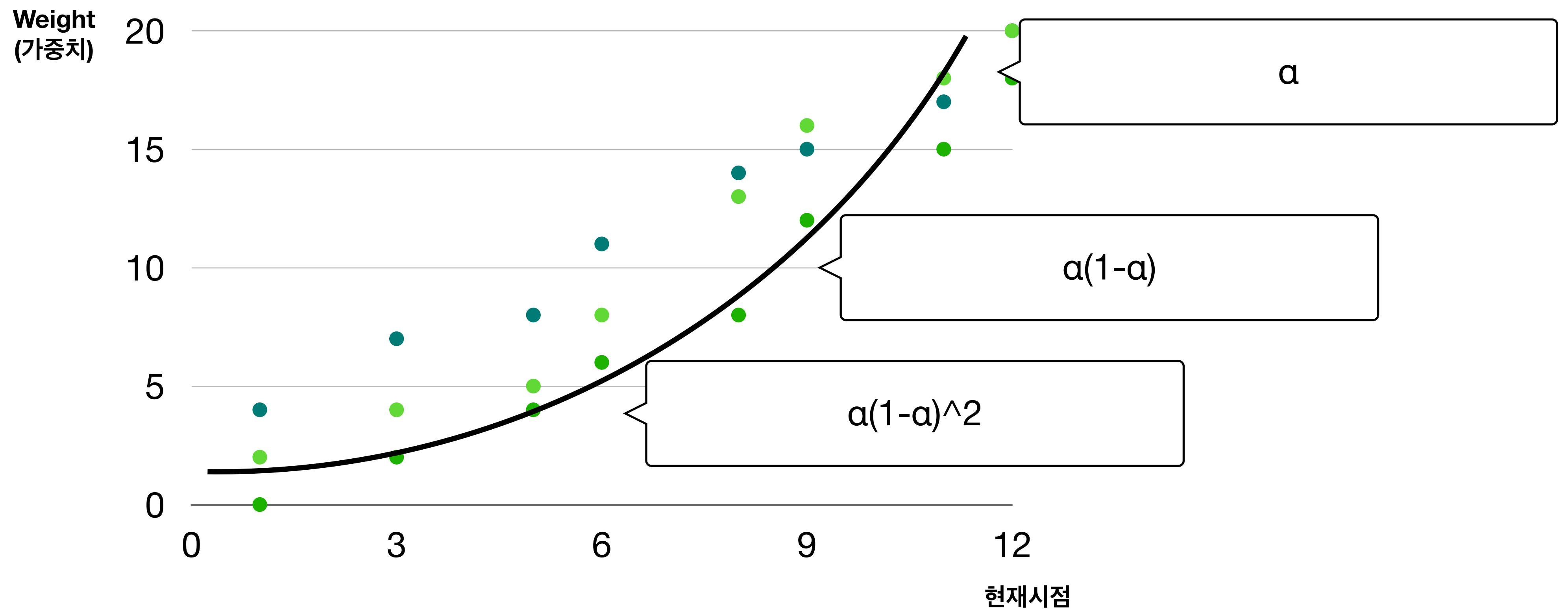
(60+25+70)/3=51.67

지수 평활법

- 단순 평균이 아닌 가중 평균을 이용
- 지수분포 모양에 근거한 가중치 결정
- 즉, 최근 데이터에 보다 많은 가중치를 두고 과거로 갈수록 가중치가 줄어듦

단순지수 평활법

- 모든 과거데이터 포함하여 계산
- 최근 데이터에 보다 큰 가중치 부여



단순 지수 평활법 요약

1. L_0 계산

$$L_0 = 1/n \left(\sum_{i=1}^n D_i \right)$$

2. $L_{\tau+1}$ 계산 , $\tau=0, \dots, N$

$$L_{\tau+1} = \alpha D_{t+1} + (1 - \alpha) L_{\tau}$$

3. 예측

$$F_{\tau+1} = L_{\tau}$$

$$F_{\tau+N} = L_{\tau}$$

단순지수 평활법 예 ($\alpha = 0.1$)

t	관측값	Level	Forecast
0		45.83	$L_0 = (20 + 60 + 25 + 70 + 28 + 72) / 6 = 45.83$
1	20	43.25	$L_1 = \alpha (20) + (1 - \alpha)(45.83) = 43.25$
2	60	44.93	
3	25	42.93	
4	70	45.64	
5	28	43.88	
6	72	46.69	
1			46.69
2			46.69

단순지수 평활법의 한계

- 트렌드가 있는 데이터에 적합하지 않음
- 계절적 변동이 있는 데이터에 적합하지 않음
- 미래시점에 관계없이 예측값 모두 동일

이중지수 평활법, 삼중지수 평활법

- 하이퍼 파라미터가 2개 / 3개

$S = \text{SEASON}$, $U = \text{DOUBLE SEASON}$

Non-seasonal models:

$$l_t = \alpha y_t + (1 - \alpha)l_{t-1} \quad (1)$$

Single seasonality models:

$$\begin{aligned} l_t &= \alpha y_t / s_t + (1 - \alpha)l_{t-1} \\ s_{t+K} &= \beta y_t / l_t + (1 - \beta)s_t \end{aligned} \quad (2)$$

Double seasonality models:

$$\begin{aligned} l_t &= \alpha y_t / (s_t u_t) + (1 - \alpha)l_{t-1} \\ s_{t+K} &= \beta y_t / (l_t u_t) + (1 - \beta)s_t \\ u_{t+L} &= \gamma y_t / (l_t s_t) + (1 - \gamma)u_t \end{aligned} \quad (3)$$

$$S_i = Y_i / L_M$$

es-RNN 학습 과정

$$s = \text{SEASON}$$

$$S_i = Y_i / L_M$$

$$l_t = \alpha \left(\frac{y_t}{s_{t-m}} \right) + (1 - \alpha) l_{t-1} b_{t-1} \quad (1)$$

$$b_t = \beta \left(\frac{l_t}{l_{t-1}} \right) + (1 - \beta) b_{t-1} \quad (2)$$

$$s_t = \gamma \frac{y_t}{l_{t-1} b_{t-1}} + (1 - \gamma) s_{t-m} \quad (3)$$

$$\hat{y}_{t+h} = l_t b_t^h s_{t-m+h_m}^+ \quad (4)$$

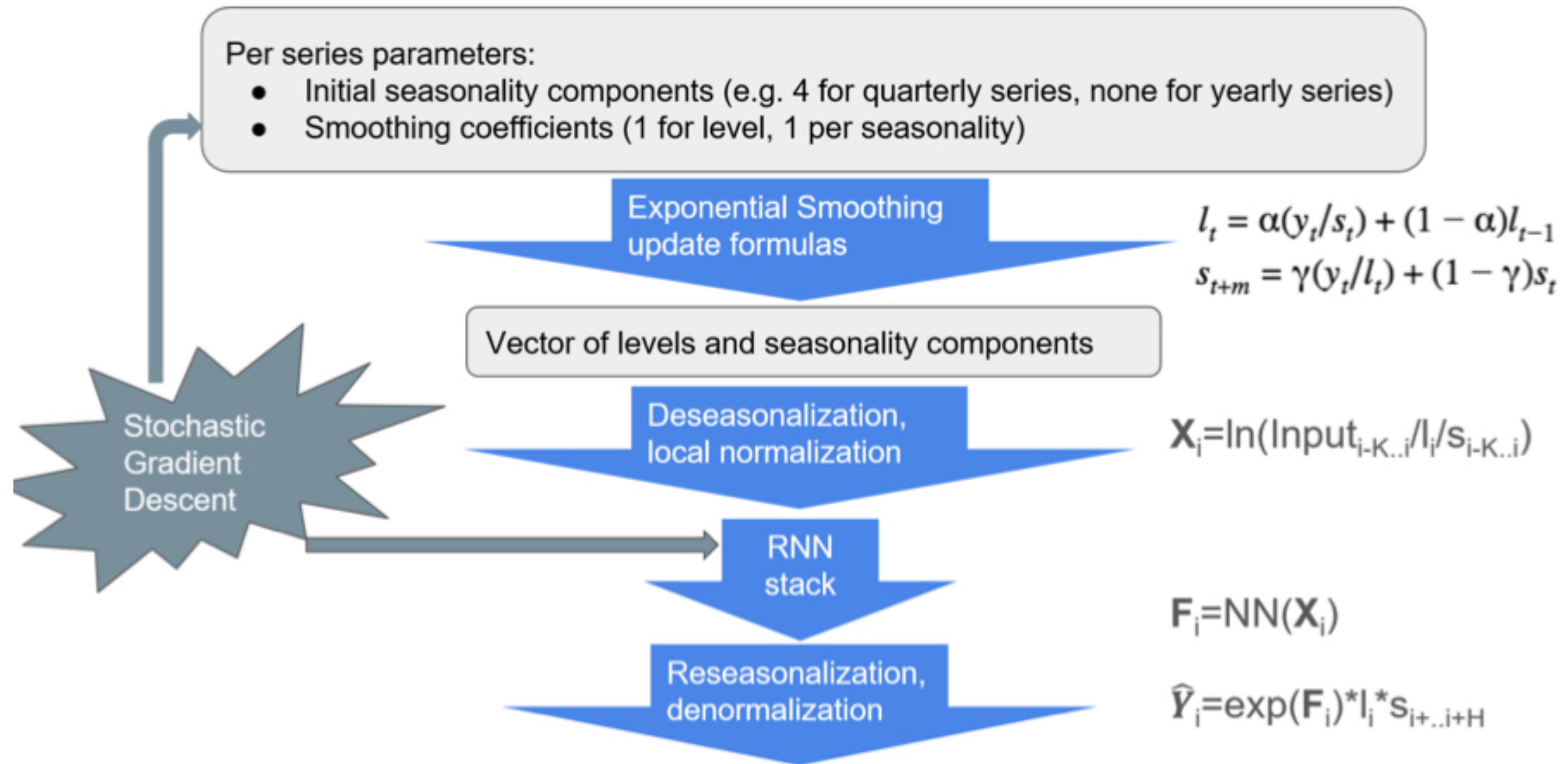
model as follows:

$$\hat{y}_{t+1 \dots t+h} = RNN(X_t) * l_t * s_{t+1 \dots t+h} \quad (5)$$

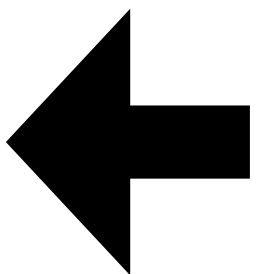
$$x_i = \frac{y_i}{l_t s_i} \quad i=1 \sim t \quad (6)$$

ized de-seasonalized features of which a scalar component x_t is calculated

Dataflow



es-RNN 모델 구성



Time Frame	Dilations	LSTM Size
Monthly	(1, 3), (6, 12)	50
Quarterly	(1, 2), (4, 8)	40
Yearly	(1, 2), (2, 6)	30

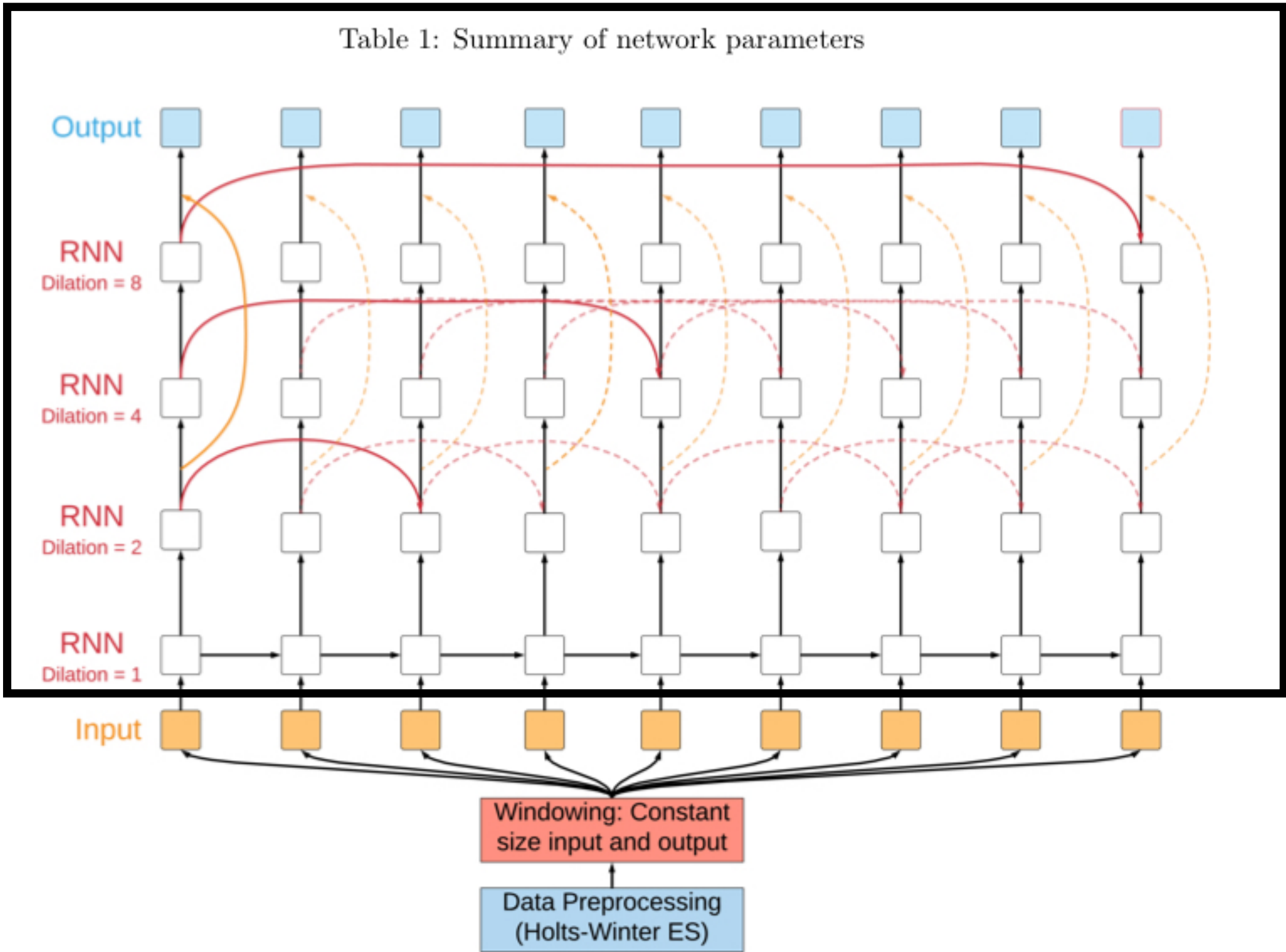
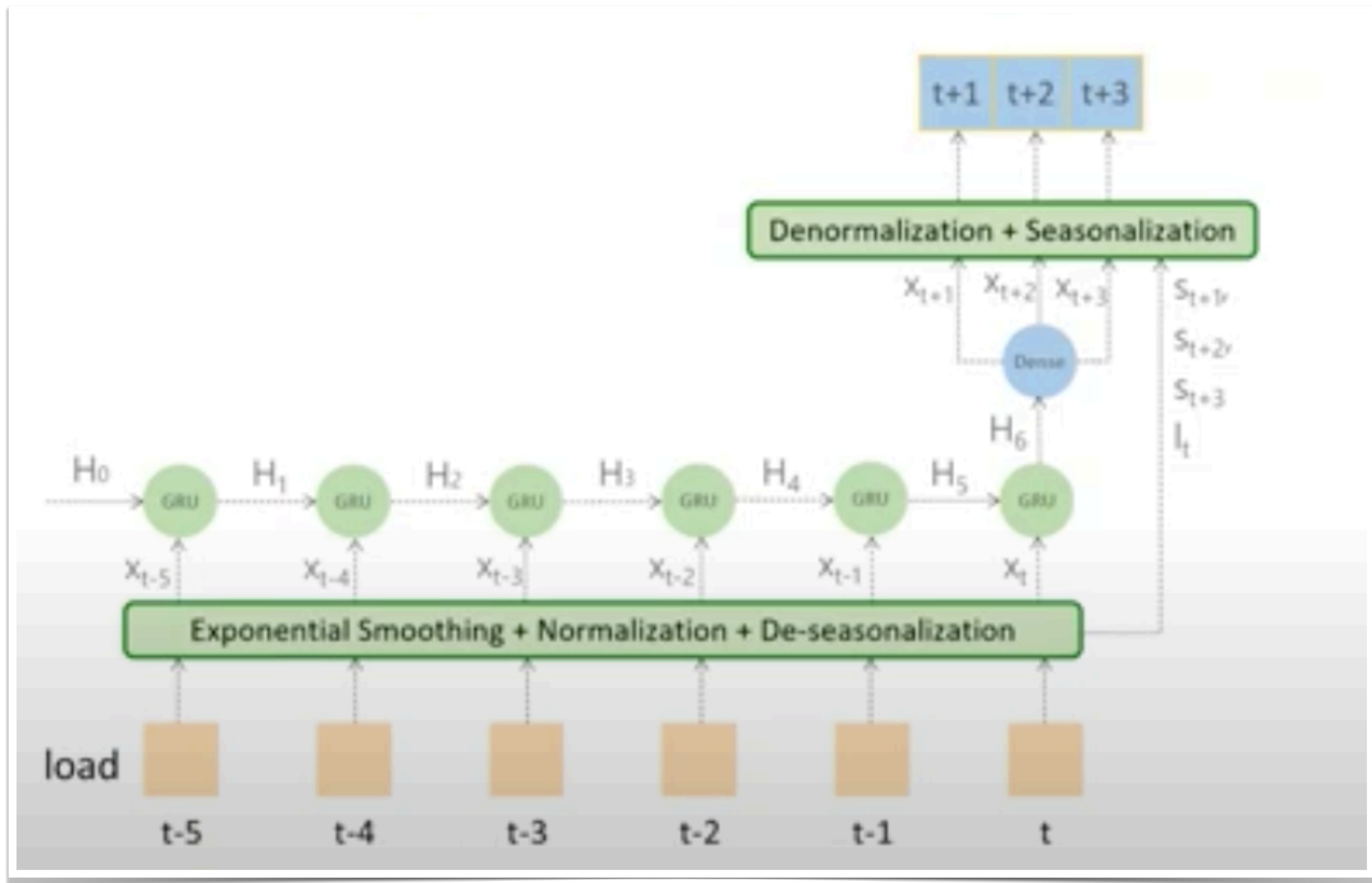


Figure 1: NN Architecture by Smyl et al. (2018)



es - RNN loss function

- Loss : pinball loss

forecast errors are calculated back in the linear space. To counter it, a pinball loss with a τ value a bit smaller than 0.5 (typically 0.45-0.49) was used. The pinball loss is defined as follows:

$$\begin{aligned} L_t &= (y_t - \hat{y}_t)\tau, \text{ if } y_t \geq \hat{y}_t \\ &= (\hat{y}_t - y_t)(1 - \tau), \text{ if } \hat{y}_t > y_t \end{aligned} \tag{7}$$

es - RNN 성과

- ~2018년까지는 세계적인 시계열 경진대회 M4 Competition 에서 통계 기반 모델이 우세
- 2019년 es-RNN 이 M4 Competition 1등을 차지
- N-BEATS 모델이 2020년 es-RNN 의 성능을 뛰어넘음

N-BEATS : NEURAL BASIS EXPANSION ANALYSIS FOR INTERPRETABLE TIME SERIES FORECASTING, ICLR 2020

참고

<A hybrid method of Exponential Smoothing and Recurrent Neural Networks for time series forecasting>

<Fast ES-RNN : A GPU Implementation of the ES-RNN Algorithm>