Time Series Forecasting

6조

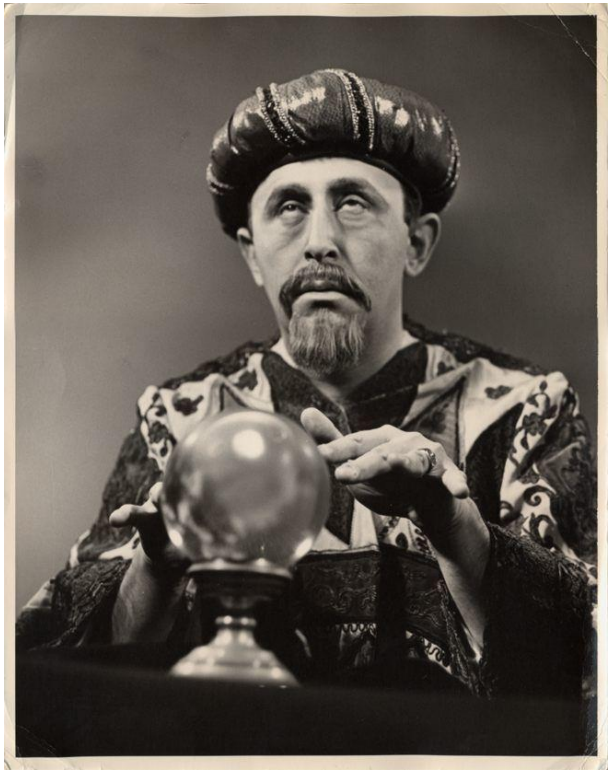
윤빈나 윤재경 정재원

KU-BIG

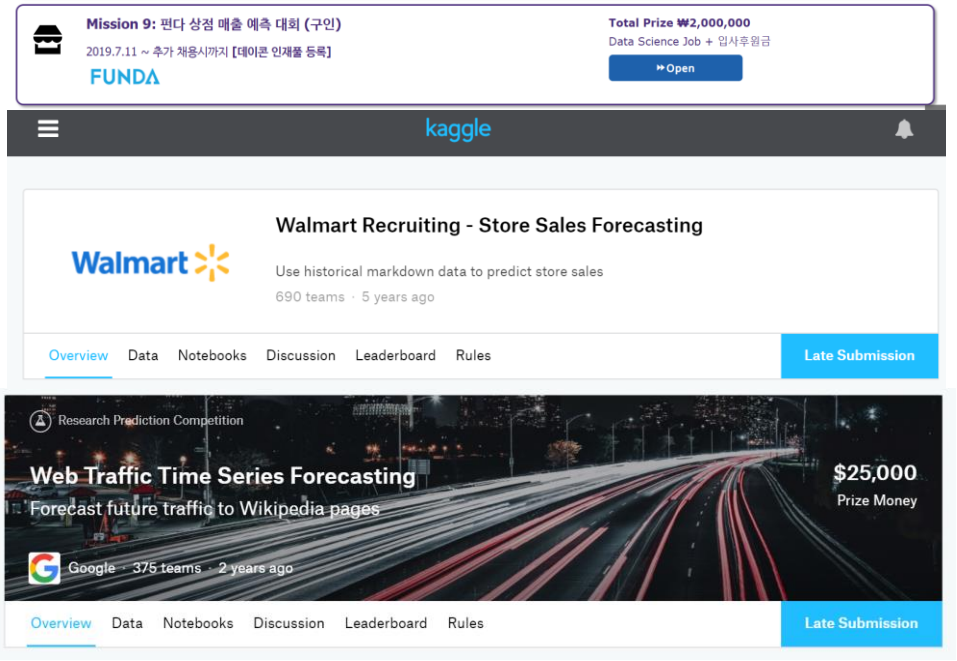
Index

KU-BIG

Time Series Forecasting

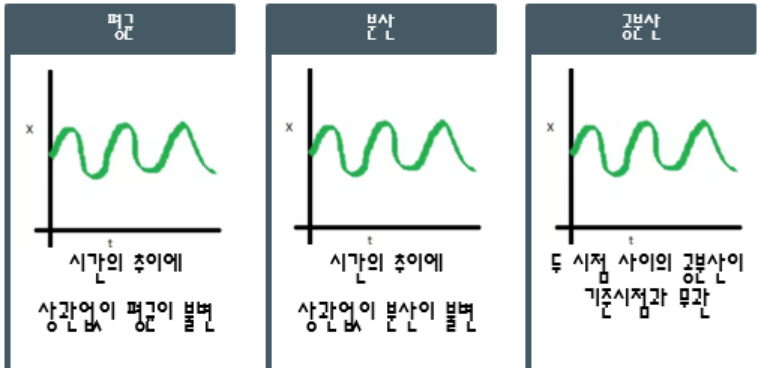


Time Series Forecasting

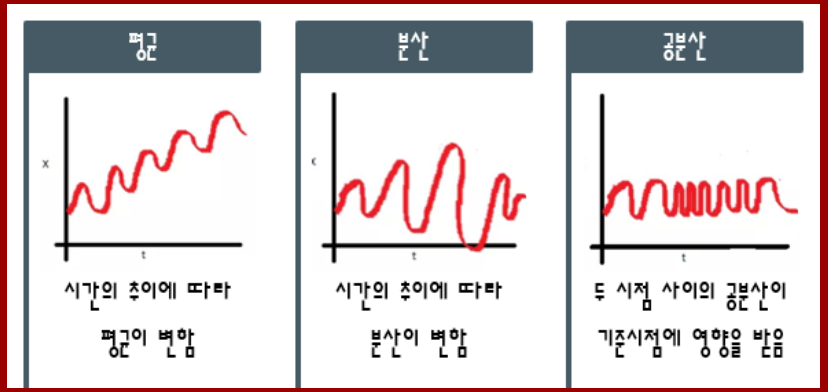


Model 1 – ARIMA (Auto-Regressive Integrated Moving-Average)

Stationary Series



Non-Stationary Series



Model 1 – ARIMA (Auto-Regressive Integrated Moving-Average)

AR (Auto-Regression) :
이전 관측값이 이후 관측값에 영향을 주는 모형

$$AR(1) : X_t = \phi X_{t-1} + \epsilon_t$$

I (Integrated) : 누적을 의미

MA (Moving-Average) :
관측값이 이전의 연속적인 오차항의 영향을 받는 모형

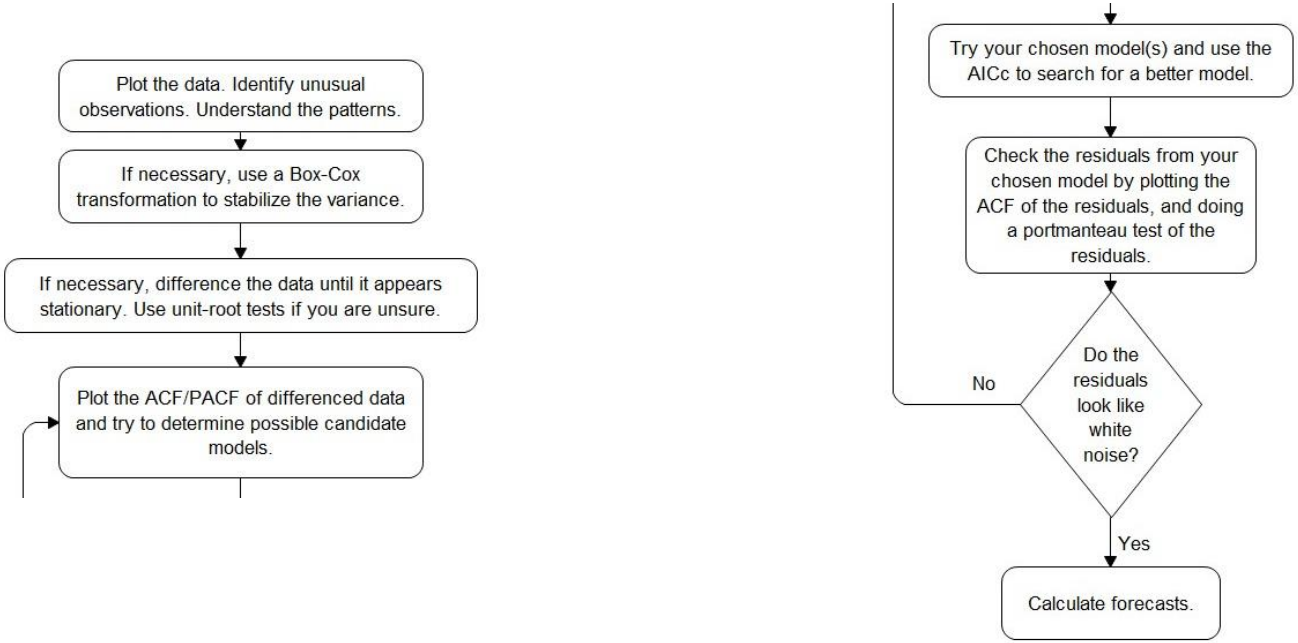
$$MA(1) : X_t = \epsilon_t - \beta_1 \epsilon_{t-1}$$

ARIMA(p, d, q)

$$\hat{y}_t = \mu + \phi_1 y_{t-1} + \ldots + \phi_p y_{t-p} - \beta_1 \epsilon_{t-1} - \ldots - \beta_q \epsilon_{t-q}$$

- $\mu = constant$
- $\phi_1 y_{t-1} + \ldots + \phi_p y_{t-p} : ARterms(laggedvaluesofy)$
- $-\beta_1 \epsilon_{t-1} - \ldots - \beta_q \epsilon_{t-q} : MAterms(laggedvaluesofy)$
- $\hat{y}_t = Y_t, \ if \ d = 0$
- $\hat{y}_t = Y_t - Y_{t-1}, \ if \ d = 1$
- $\hat{y}_t = (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2}), \ if \ d = 2$

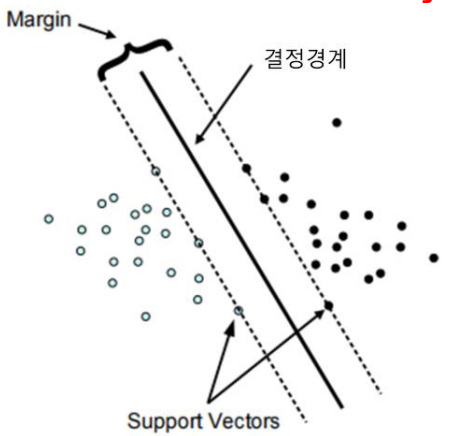
Model 1 – ARIMA (Auto-Regressive Integrated Moving-Average)



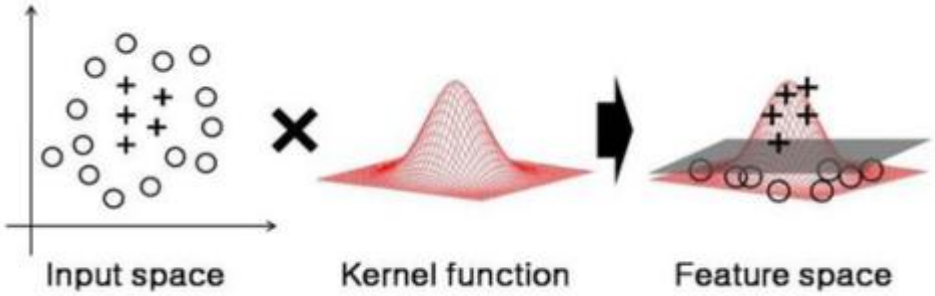
Model 2 – SVR (Support Vector Regression)

Support Vectors

Maximize the Margin!

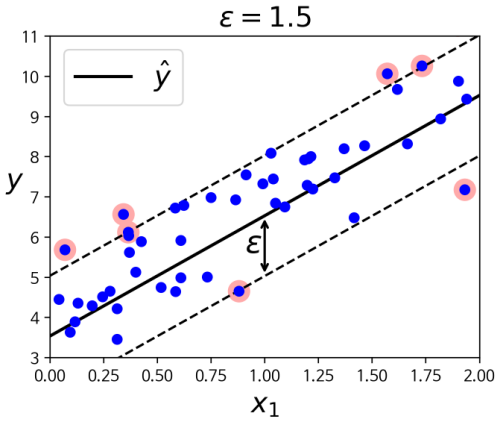


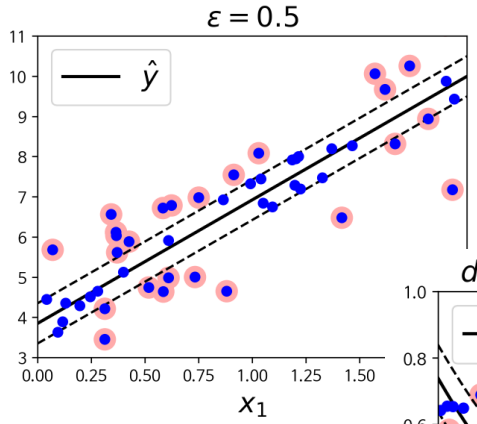
Kernel Function



Model 2 – SVR (Support Vector Regression)

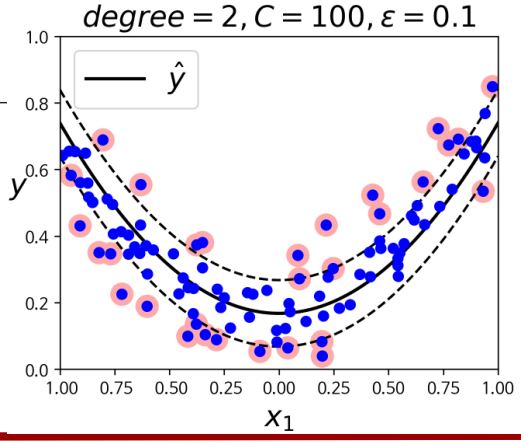
Linear SVR

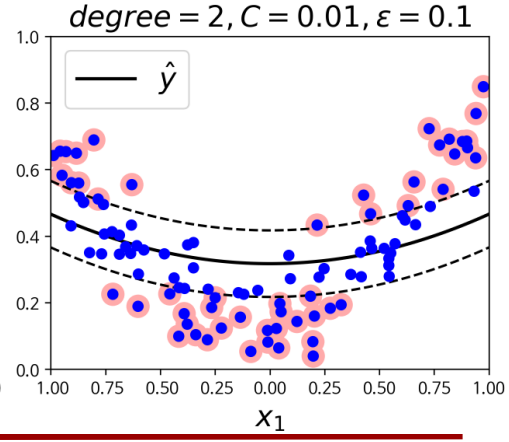


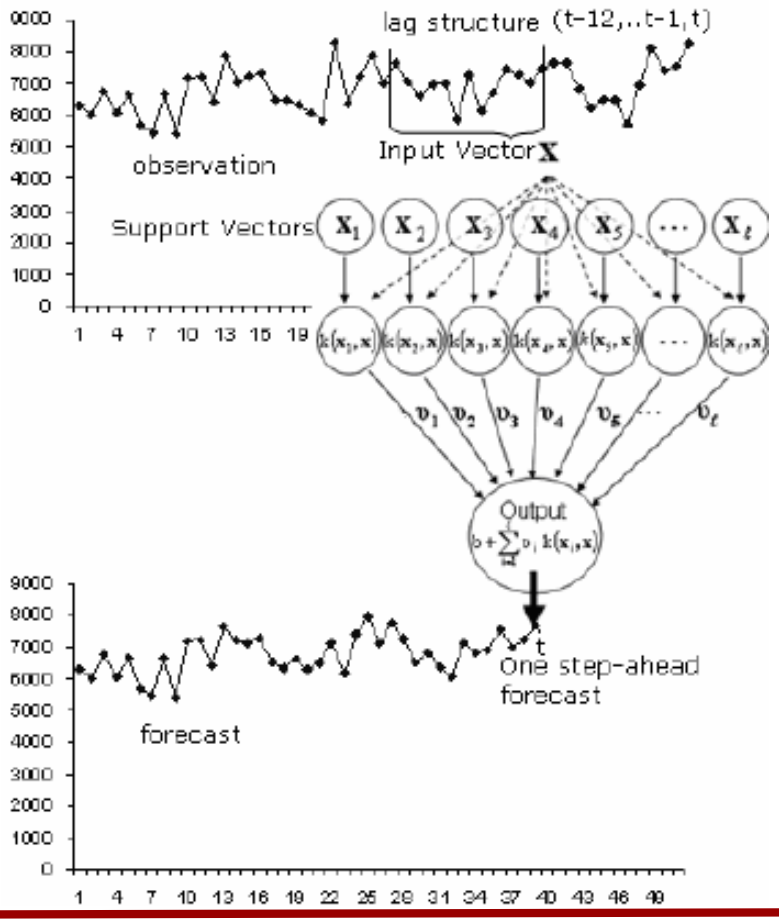


As many as possible points in the Epsilon-Margin!

Non-linear SVR







Time Series Forecasting Using SVR

Model3 - RNN based models

We can process a sequence of vectors **x** by applying a **recurrence formula** at every time step:

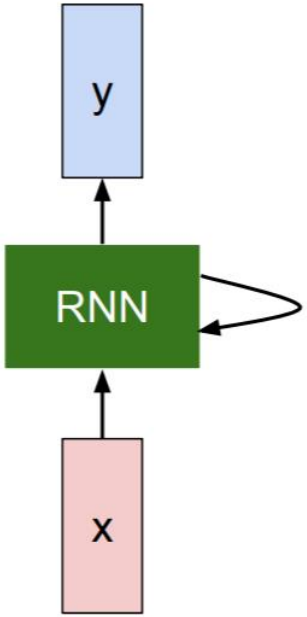
$$h_t = f_W(h_{t-1}, x_t)$$

new state

some function with parameters W

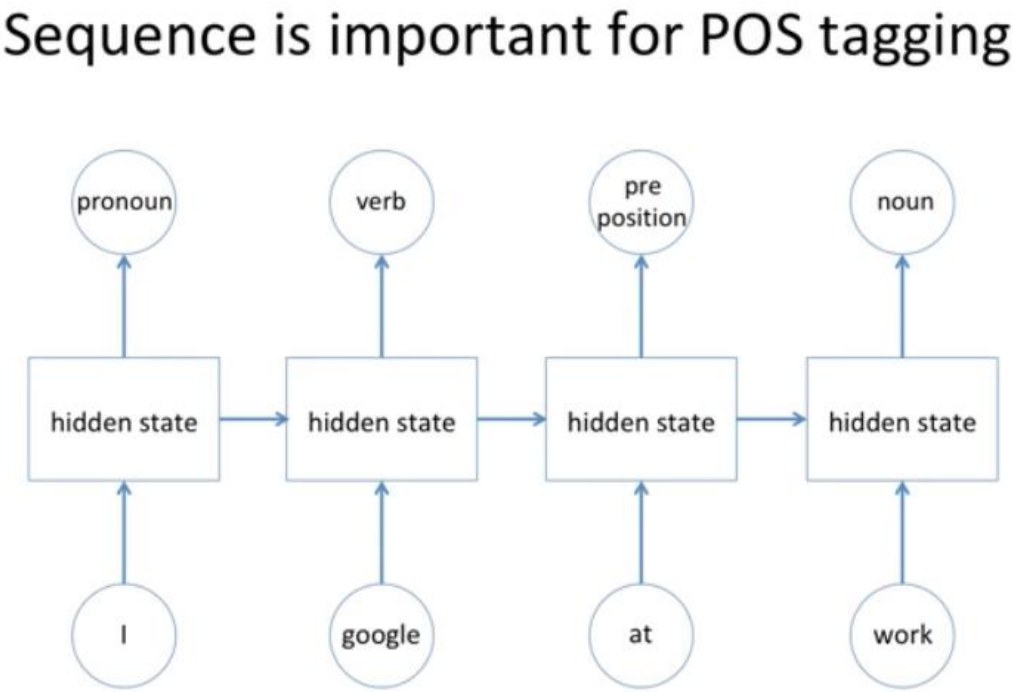
old state

input vector at some time step

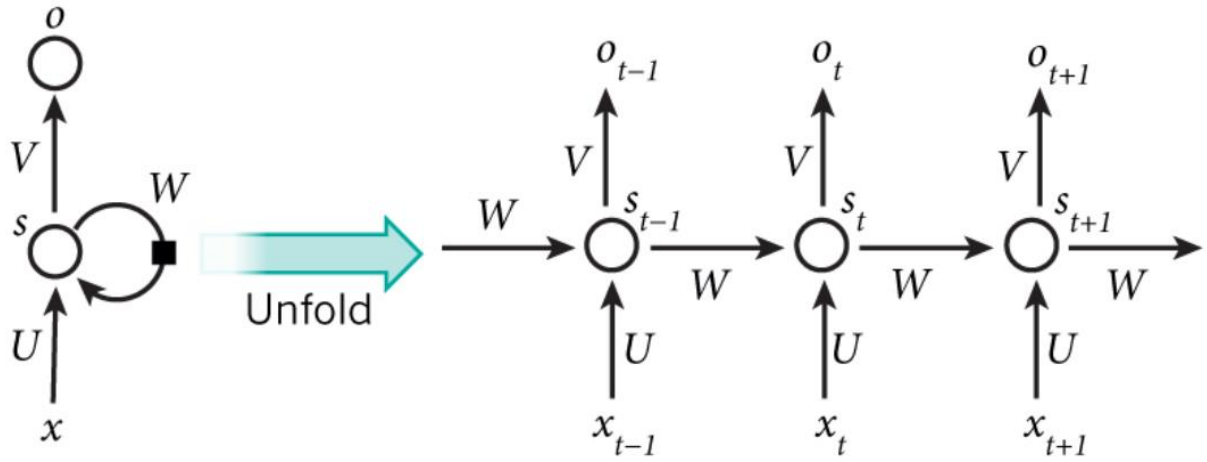


출처 : http://cs231n.stanford.edu/slides/2019/cs231n_2019_lecture10.pdf

Model3 - RNN based models



Model3 - RNN based models

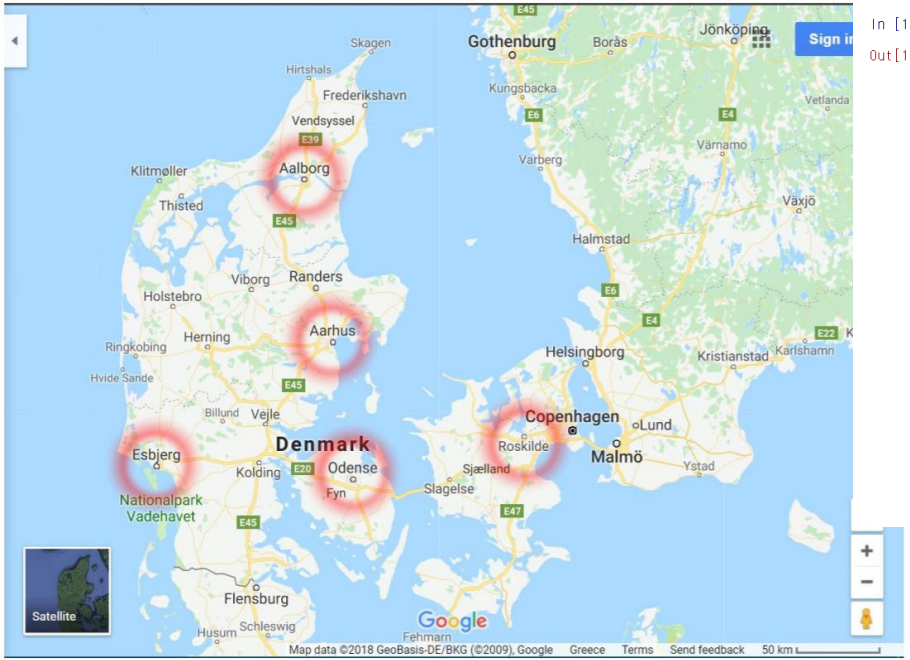


$$s_t = \tanh(Ux_t + Ws_{t-1})$$

$$\hat{o}_t = softmax(Vs_t)$$

출처 : https://aikorea.org/blog/rnn-tutorial-1/

Example - forecasting Odense weather



In [10]:
```
df.head()
```

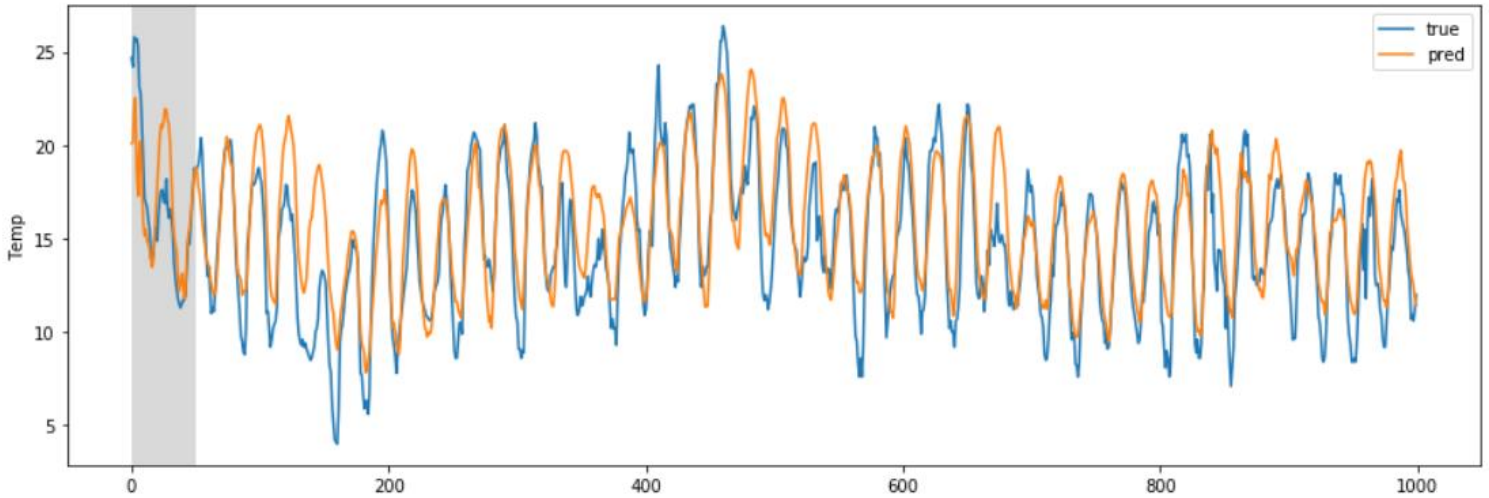
Out[10]:

	Aalborg				Aarhus				Esbjerg	
	Temp	Pressure	WindSpeed	WindDir	Temp	Pressure	WindSpeed	WindDir	Temp	Pressure
DateTime										
1980-03-01 11:00:00	5.000000	1007.766667	10.2	280.000000	5.0	1008.300000	15.4	290.0	6.083333	NaN
1980-03-01 12:00:00	5.000000	1008.000000	10.3	290.000000	5.0	1008.600000	13.4	280.0	6.583333	NaN
1980-03-01 13:00:00	5.000000	1008.066667	9.7	290.000000	5.0	1008.433333	15.4	280.0	6.888889	NaN
1980-03-01 14:00:00	4.333333	1008.133333	11.1	283.333333	5.0	1008.266667	14.9	300.0	6.222222	NaN
1980-03-01 15:00:00	4.000000	1008.200000	11.3	280.000000	5.0	1008.100000	17.0	290.0	5.555556	NaN

출처 : https://github.com/Hvass-Labs/TensorFlow-Tutorials/blob/master/23_Time-Series-Prediction.ipynb

Example - forecasting Odense weather

```
In [79]: plot_comparison(start_idx=200, length=1000, train=False)
```



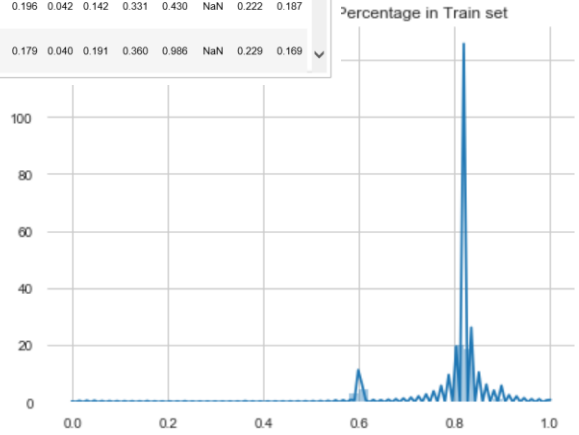
출처 : https://github.com/Hvass-Labs/TensorFlow-Tutorials/blob/master/23_Time-Series-Prediction.ipynb

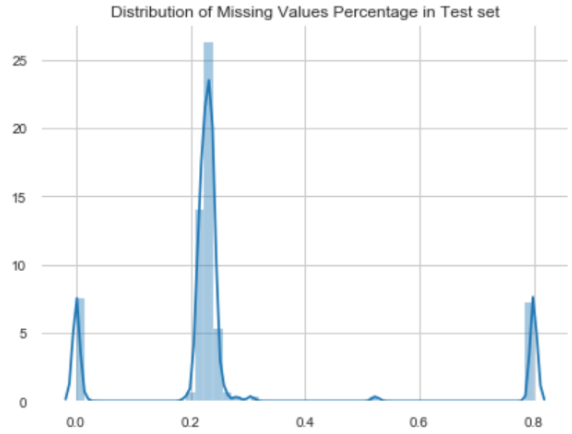
Data 1 - 전력수요예측

```
In [19]: data2.set_index('Time')
```

Out[19]:

Time	X692	X1272	X553	X1299	X4	X598	X1003	X1010	X1216	X1047	...	X1164	X656	X1287	X324	X767	X1271	X1288	X58	X1033	X1163
2018-03-02 11:00	0.242	0.174	0.532	0.084	NaN	0.273	0.245	0.276	0.164	0.210	...	0.348	0.498	0.272	0.036	0.213	0.475	0.078	NaN	0.172	0.200
2018-03-02 12:00	0.298	0.176	0.445	0.089	NaN	0.167	0.243	0.269	0.163	0.309	...	0.214	0.462	0.251	0.041	0.254	0.407	0.105	NaN	0.203	0.227
2018-03-02 13:00	0.200	0.156	0.474	0.093	NaN	0.145	0.421	0.518	0.158	0.233	...	0.184	0.533	0.195	0.040	0.147	0.354	0.184	NaN	0.232	0.253
2018-03-02 14:00	0.238	0.206	0.403	0.126	NaN	0.131	0.209	0.252	0.148	0.265	...	0.231	0.316	0.196	0.042	0.142	0.331	0.430	NaN	0.222	0.187
2018-03-02 15:00	0.204	0.320	0.445	0.206	NaN	0.164	0.249	0.237	0.144	0.232	...	0.235	0.411	0.179	0.040	0.191	0.360	0.986	NaN	0.229	0.169





Data 2 - Funda 상점매출예측

Out[2]:

	store_id	transacted_date	card_id	installment_term	amount	#	per_amount
0	0	2016-06-01	6	0	12571.428571	4	3142.857143
1	0	2016-06-02	49	0	40571.428571	7	5795.918367
2	0	2016-06-03	36	0	18142.857143	3	6047.619048
3	0	2016-06-04	119	0	31714.285714	7	4530.612245
4	0	2016-06-05	66	0	10428.571429	3	3476.190476
5	0	2016-06-06	102	0	17285.714286	4	4321.428571
6	0	2016-06-09	288	0	35000.000000	9	3888.888889
7	0	2016-06-10	317	8	53000.000000	8	6625.000000
8	0	2016-06-11	423	0	64428.571429	9	7158.730159
9	0	2016-06-12	214	0	51857.142857	4	12964.285714
10	0	2016-06-13	476	0	48714.285714	8	6089.285714
11	0	2016-06-15	330	0	11428.571429	5	2285.714286
12	0	2016-06-16	282	0	24857.142857	4	6214.285714
13	0	2016-06-17	375	0	12428.571429	5	2485.714286
14	0	2016-06-18	483	0	23428.571429	6	3904.761905

Out[5]:

	store_id	card_id	card_company	transacted_date	transacted_time	installment_term	region	type_of_business	amount
0	0	0	b	2016-06-01	13:13	0	NaN	기타 미용업	1857.142857
1	0	1	h	2016-06-01	18:12	0	NaN	기타 미용업	857.142857
2	0	2	c	2016-06-01	18:52	0	NaN	기타 미용업	2000.000000
3	0	3	a	2016-06-01	20:22	0	NaN	기타 미용업	7857.142857
4	0	4	c	2016-06-02	11:06	0	NaN	기타 미용업	2000.000000
5	0	5	c	2016-06-02	13:09	0	NaN	기타 미용업	2000.000000
6	0	6	f	2016-06-02	15:33	0	NaN	기타 미용업	2000.000000
7	0	7	a	2016-06-02	17:18	0	NaN	기타 미용업	7857.142857
8	0	8	c	2016-06-02	18:30	0	NaN	기타 미용업	2000.000000
9	0	9	a	2016-06-02	19:56	0	NaN	기타 미용업	1857.142857
10	0	10	f	2016-06-02	22:26	0	NaN	기타 미용업	22857.142857
11	0	11	c	2016-06-03	12:43	0	NaN	기타 미용업	2000.000000
12	0	12	b	2016-06-03	19:05	0	NaN	기타 미용업	1428.571429
13	0	13	e	2016-06-03	21:46	0	NaN	기타 미용업	14714.285714
14	0	14	a	2016-06-04	11:15	0	NaN	기타 미용업	2000.000000
15	0	15	c	2016-06-04	12:58	0	NaN	기타 미용업	3571.428571
16	0	16	h	2016-06-04	13:42	0	NaN	기타 미용업	1428.571429
17	0	17	a	2016-06-04	14:32	0	NaN	기타 미용업	2000.000000

Project Plan