INDR 450/550

Spring 2022

Lecture 12: Regression for Time Series (4)

March 23, 2022

Fikri Karaesmen

Announcements

- Class Exercise at the end of lecture today. If you are participating online, please upload your document under Course Contents/Class Exercises
- HW 2 available with a deadline of April 4 (Labs 3 and 4).
- Exam scheduled.
- The first four labs were uploaded. Please follow them.
  - Next HW based on lab2 and lab3

Regression for Time Series: K – Nearest Neighbours

- Comparison to a non-parametric method: k-nearest neigbours (KNN) regression
- Let's say we would like to make a prediction for the point x_{1t} using the data available up to time $t-1$.
- We identify the K-nearest points to x_{1t} among $\{x_{11}, x_{12}, ...x_{1,t-1}\}$. Let us say that the nearest points are those in the set $\mathcal{N}_I$. Then our prediction is simply:

$$\hat{y}_t = \frac{\sum_{x_{i\tau} \in \mathcal{N}_I} y_{\tau+1}}{K}$$

- If we take $K = 1$, our prediction is given by the response to the nearest predictor in the training set.
- If we take $K = 10$ we are smoothing the prediction over the 10 nearest predictors.

Regression for Time Series: K – Nearest Neighbours: Example

- To implement 5-nn we search in months 1 to 46, the 5 nearest neigbours of 190.
  - Month 10: 190, Month 11: 192
  - Month 11: 192, Month 12: 192
  - Month 12: 192, Month 13: 147
  - Month 24: 188, Month 25: 139
  - Month 36: 184, Month 37: 151

- The 5-NN prediction is then (192+192+147+139+151)/5 = 164.2

10	190	0
11	192	2
12	192	2
13	147	43
14	133	57
15	163	27
16	150	40
17	129	61
18	131	59
19	145	45
20	137	53
21	138	52
22	168	22
23	176	14
24	188	2
25	139	51
26	143	47
27	150	40

Regression for Time Series: K – Nearest Neighbours: Example

- Note that the similarity could be based on a vector of features.
- For instance, we note that the demand in periods (45,46,47) was (143, 160, 190)
- We look for the nearest neigbour of (143, 160, 190) in the past data measured in terms of Euclidean distance (there could be other distance measures).
- The nearest neigbour in terms of Euclidean distance is months (33,34,35). The production in those months was (143, 151, 177). The production in month 36 is: 184.
    - This becomes our prediction for month 48:

$$\hat{y}_{48} = 184$$

Regression for Time Series: K – Nearest Neighbours: Example

- We could take distance-based weights for different neighbours
- Let us consider 2-NN
  - The nearest neigbour is months (33,34,35), with a Euclidean distance of: 19.72. Production in month 36 is 184
  - The next nearest neigbour is months (8,9,10) with a Euclidean distance of 22.20. The production in month 11 is 192.
  - We then take as our prediction for month 48:

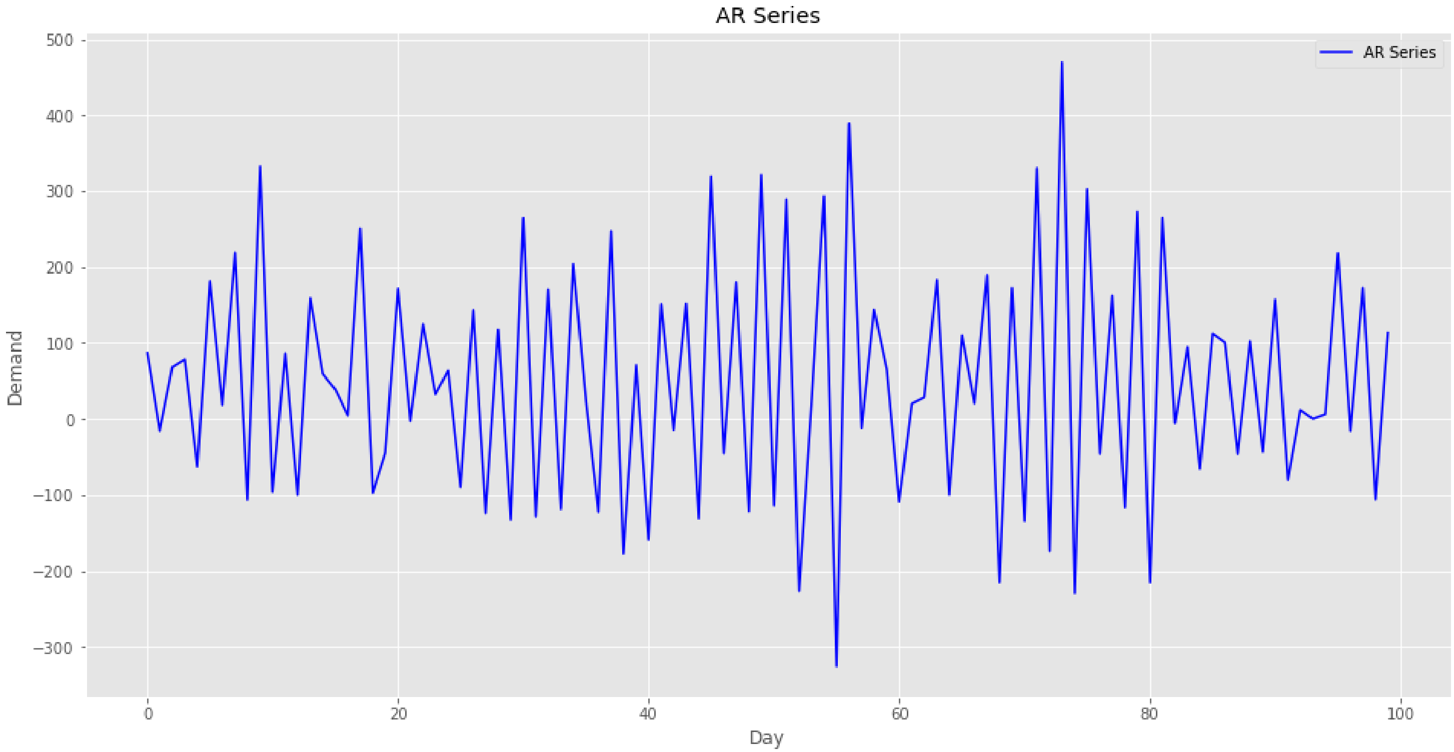
$$\hat{y}_{48} = \frac{\left(\frac{1}{19.72}\right)184 + \left(\frac{1}{22.20}\right)192}{\left(\frac{1}{19.72}\right) + \left(\frac{1}{22.20}\right)}$$

Regression for Time Series: K – Nearest Neighbours: Example

- Note that there are many options, the number of neigbours is a parameter, the weighting function is a parameter. For time series, the feature set itself is a parameter (previous month, previous three months, previous six-months etc.)
- Sometimes rather than averaging over a fixed number of neighbours, it's more sensible to average over only those neighbours that are within a reasonable distance.
    - The threshold distance then becomes a parameter.
    - Nadaraya-Watson Kernel Regression is an example.

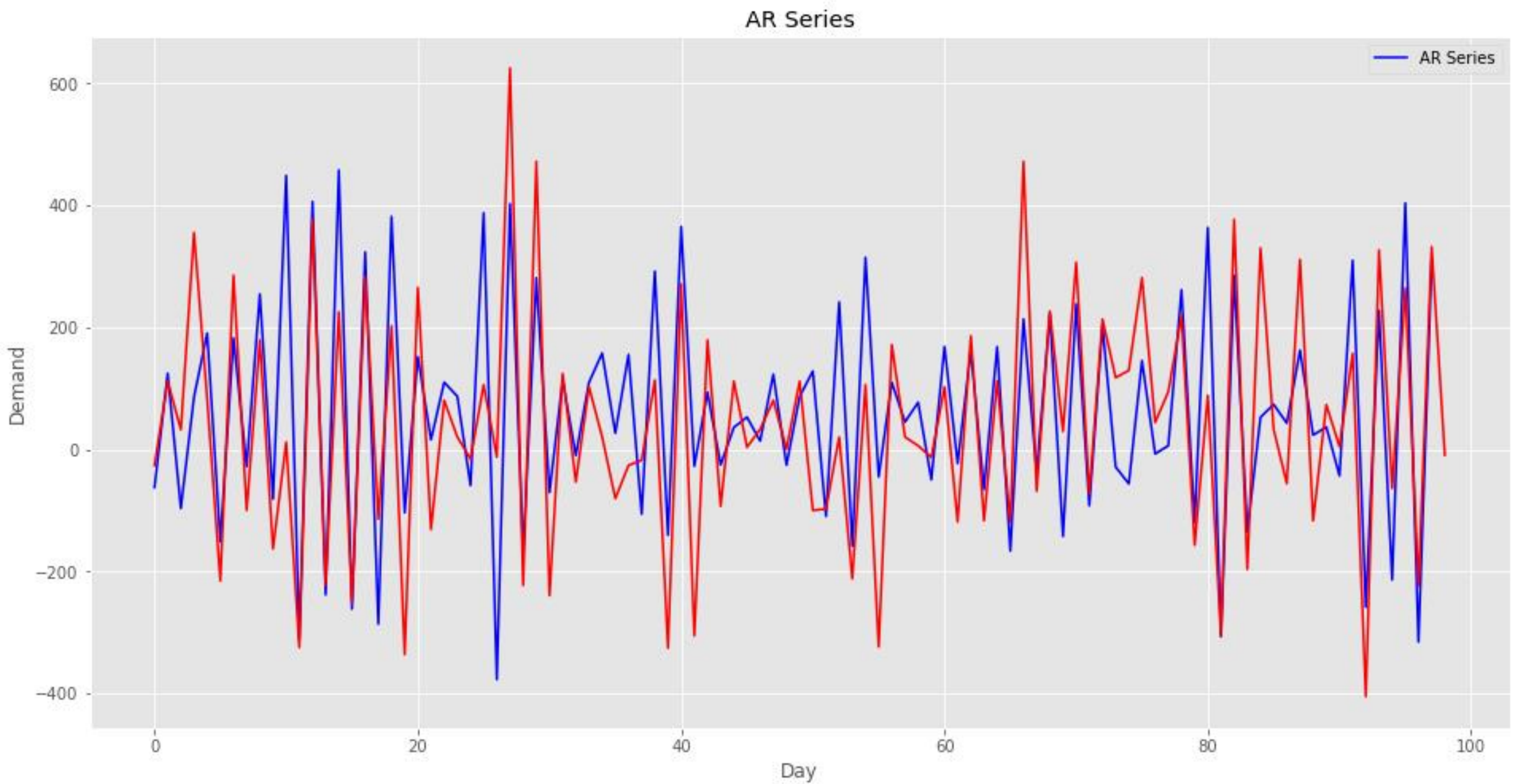
Regression for Time Series: K – Nearest Neighbours: Example

- I simulated an AR(1) process: $Y_t = \mathrm{c} + \phi_1 \, Y_{t-1} + \epsilon_t$
  - c=100; σ=100; ϕ_1=-0.9 .



Regression for Time Series: K – Nearest Neighbours: Example

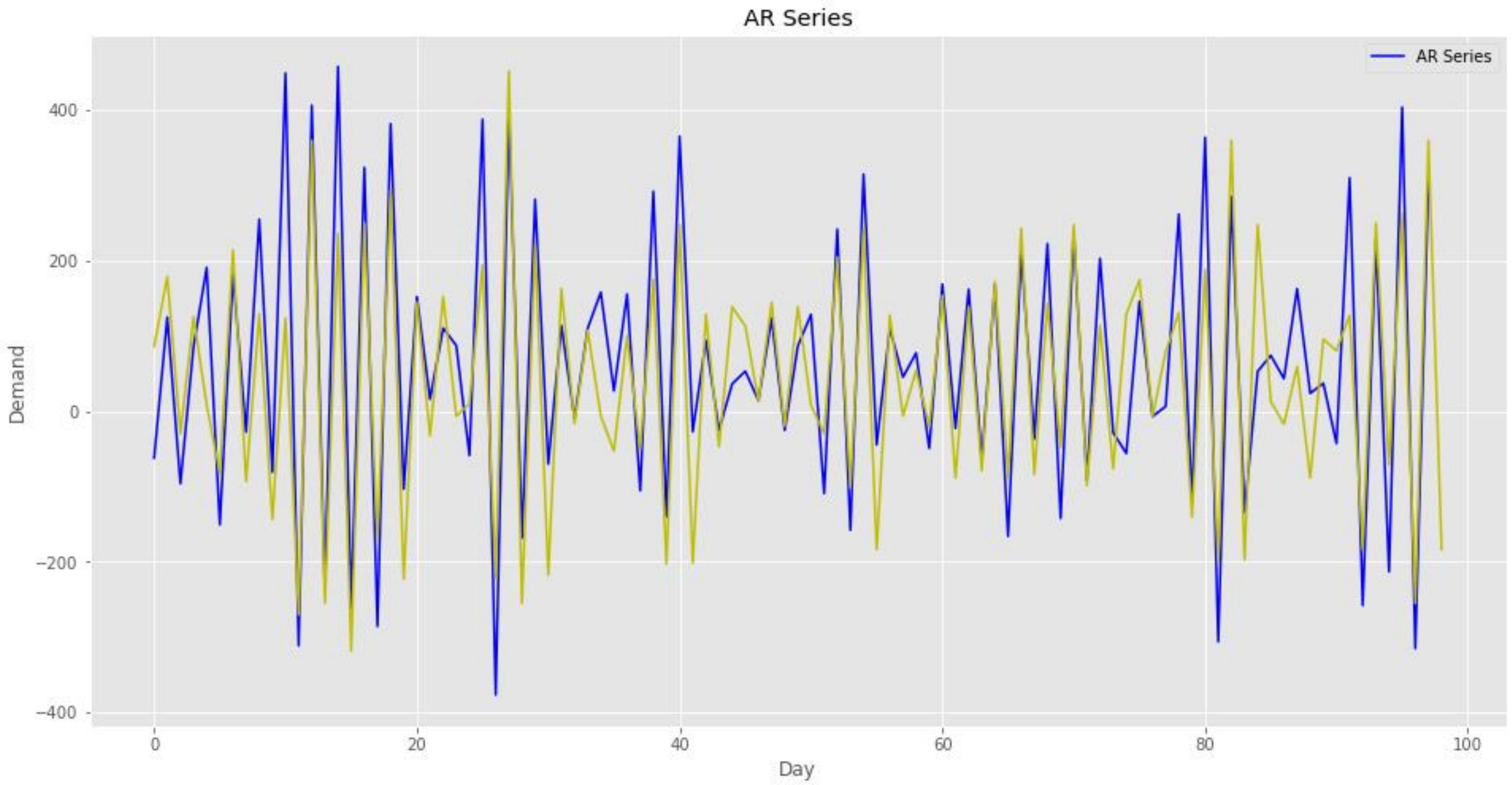
- Let's experiment with K=1, K=5 and K=10 and perform some one step ahead forecasts.
- Training set first 400 observations, test set: last 100 observations
- K=1 forecasts (in red)



MSE KNN (1) = 19135.90
RMSE KNN (1) = 138.33
MAE KNN (1)= 106.27
MAPE KNN (1) = 0.059

Regression for Time Series: K – Nearest Neighbours: Example

- K=10 forecasts (in yellow)



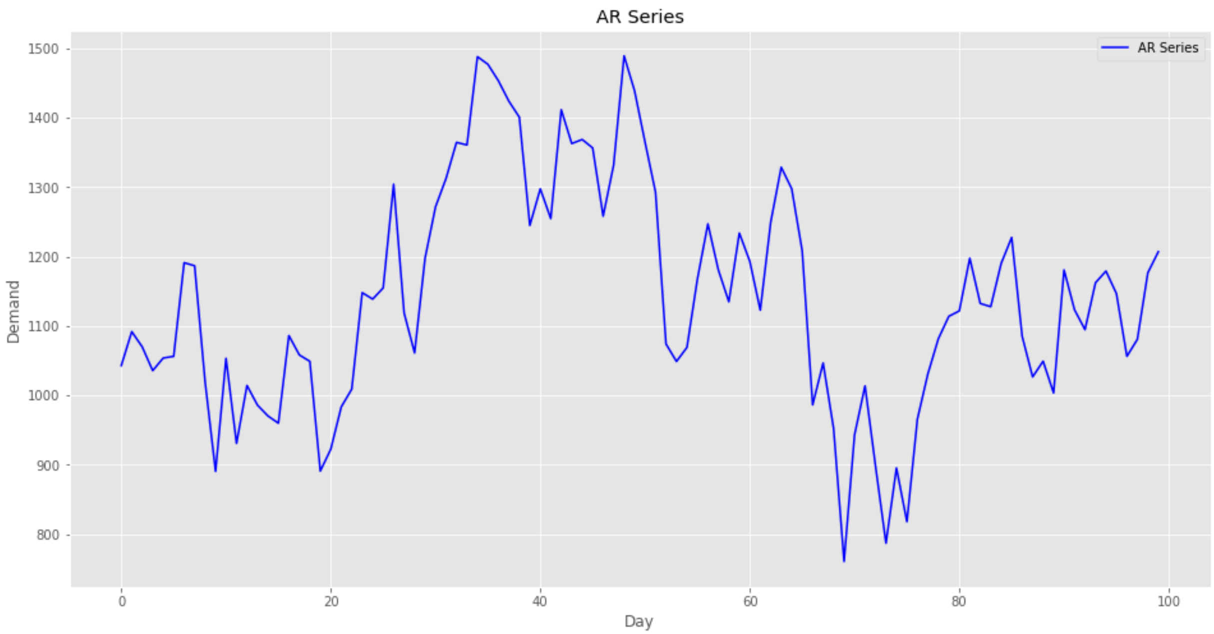
MSE KNN (10) = 9525.89
RMSE KNN (10) = 97.60
MAE KNN (10)= 78.12
MAPE KNN (10) = 0.093

Regression for Time Series: K – Nearest Neighbours: Example

- And now an AR(1) process with a positive coefficient:

$$Y_t = c + \phi_1 \, Y_{t-1} + \epsilon_t$$

  - c=100; σ=100; ϕ_1=0.9 .

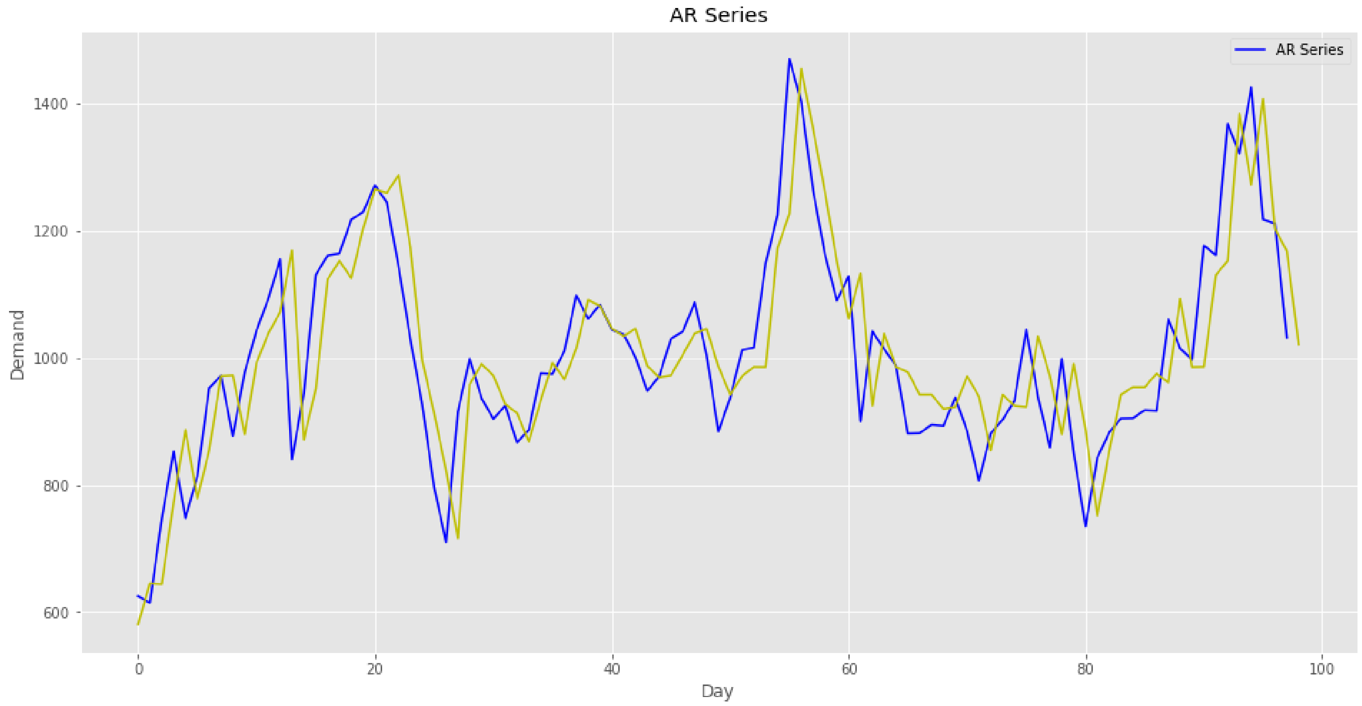


Regression for Time Series: K – Nearest Neighbours: Example

$$Y_t = c + \phi_1 Y_{t-1} + \epsilon_t$$

- c=100; σ=100; ϕ_1=0.9 .

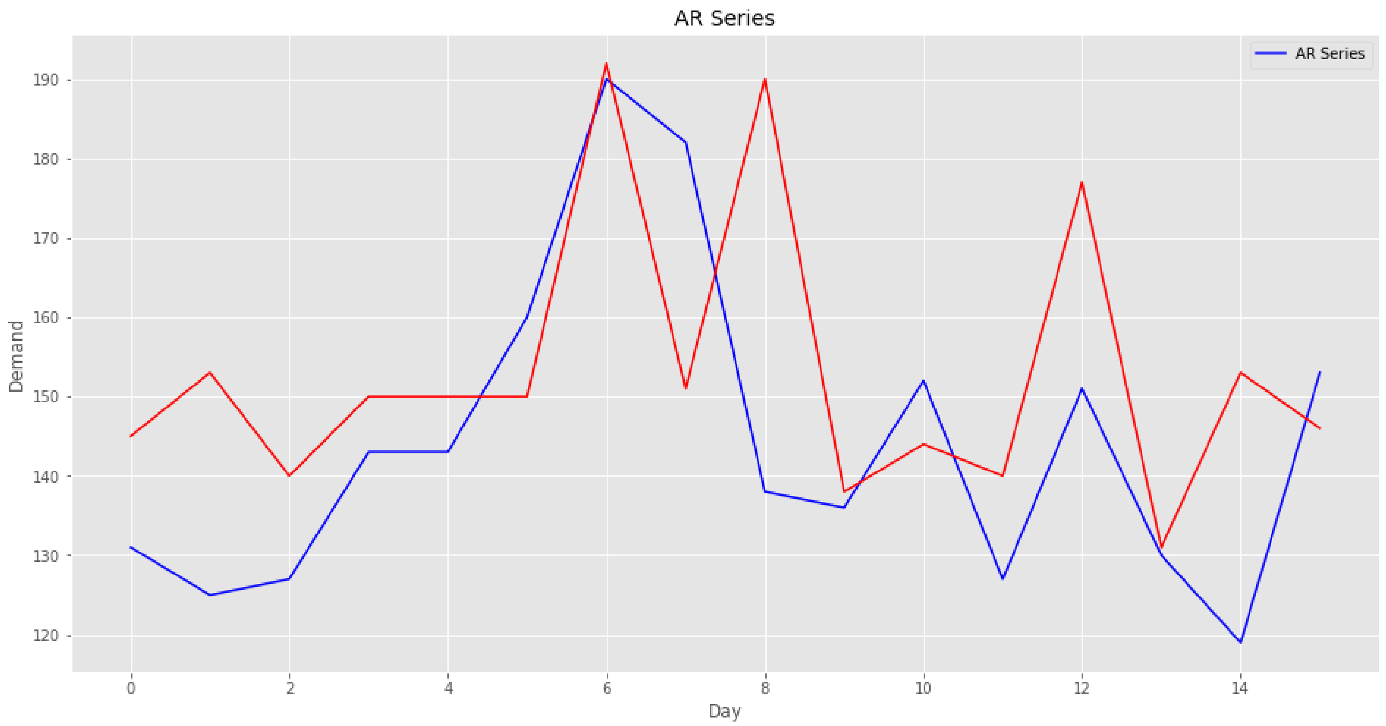
- K=10 forecasts



MSE KNN (10) = 19876.23
RMSE KNN (10) = 140.98
MAE KNN (10)= 84.56
MAPE KNN (10) = 0.076

Regression for Time Series: K – Nearest Neighbours: Example

- Some trials with Australian Beer data: seasonal series with insignificant trend.
- Train data: first 40 months, test data last 16 months




```
MSE KNN (1) =  445.9375
RMSE KNN (1) =  21.117232299712004
MAE KNN (1)=  15.9375
MAPE KNN (1) =  0.01938820577666877
MSE KNN (5) =  20834.570871089654
RMSE KNN (5) =  144.34185419028555
MAE KNN (5)=  87.75288308560324
MAPE KNN (5) =  0.1250975894701112
MSE KNN (10) =  19876.233190335715
RMSE KNN (10) =  140.9830954062781
MAE KNN (10)=  84.56145649098855
MAPE KNN (10) =  0.11581031328187803
```

Regression for Time Series: K – Nearest Neighbours:

- May work surprisingly well but a lot of parametrization to check.
- Validation is crucial.
- Here's some useful property for later.
  - If we take K to be not too small, our prediction is an average (or a distance weighted average) of some predictors (the K-nearest neigbour responses). This gives us a discrete probability distribution to work with.

Regression for Time Series: K – Nearest Neighbours: Example

- With 5-nn, we had found:
  - Month 10: 190, Month 11: 192
  - Month 11: 192, Month 12: 192
  - Month 12: 192, Month 13: 147
  - Month 24: 188, Month 25: 139
  - Month 36: 184, Month 37: 151
- The 5-NN prediction is then (192+192+147+139+151)/5 = 164.2
- We can also postulate that $P(Y_{48}=192) = 2/5$, $P(Y_{48}=139) = P(Y_{48}=147) = P(Y_{48}=151) = 1/5$.

10	190	0
11	192	2
12	192	2
13	147	43
14	133	57
15	163	27
16	150	40
17	129	61
18	131	59
19	145	45
20	137	53
21	138	52
22	168	22
23	176	14
24	188	2
25	139	51
26	143	47
27	150	40

Regression for Time Series: Classification Problems

- Linear regression is great tool but it is not appropriate for certain qualitative responses.
- For a slow selling item, the sales would likely be zero or one ($Y = 0$ or 1).
- Based on credit card usage data, a customer may default or not ($Y = 0$ or 1).
- The state of a patient is categorized as high risk, medium risk or low risk ($Y = 1$, 2 or 3).
- The appropriate question is typically of the form:

$$P(\text{no sale in given period} | \text{past observations})$$

Regression for Time Series: Classification Problems

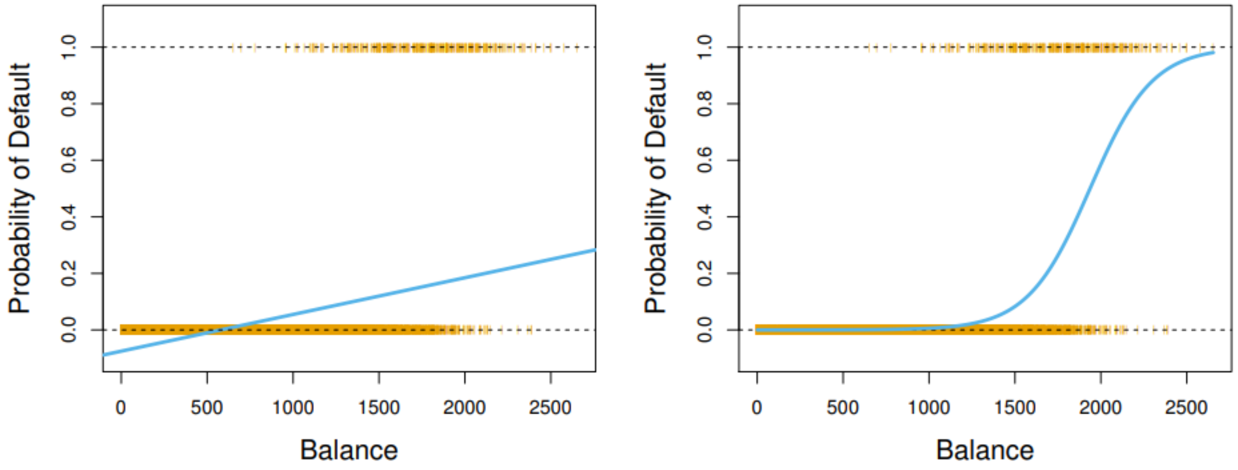


FIGURE 4.2. *Classification using the* `Default` *data.* Left: *Estimated probability of* `default` *using linear regression. Some estimated probabilities are negative! The orange ticks indicate the 0/1 values coded for* `default`*(*`No` *or* `Yes`*).* Right: *Predicted probabilities of* `default` *using logistic regression. All probabilities lie between* 0 *and* 1.

Regression for Time Series: Classification Problems- Logistic Regression

- We'll then try to predict the probability of an event $P(Y = 1|X)$. We use the shortcut:

$$p(X) = P(Y = 1|X)$$

and we use a logistic function to express this probability:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

- After some algebraic manipulation, we can write:

$$\frac{p(X)}{1 - p(X)} = e^{\beta_0+\beta_1 X}$$

Regression for Time Series: Classification Problems- Logistic Regression

- Taking logs on both sides we get:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X$$

- The left hand side is known as the log-odds ratio also known as logit. The logistic regression model has a logit that is linear in the predictor X.

Regression for Time Series: Classification Problems- Logistic Regression

- To estimate the parameters β_0 and β_1 in:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

  we use maximum likelihood estimation.

- Once we find the maximum likelihood estimators $\hat{\beta}_0$ and $\hat{\beta}_1$, we can estimate the probability:

$$p(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1 + e^{\hat{\beta}_0+\hat{\beta}_1 X}}$$

Regression for Time Series: Classification Problems- Logistic Regression

- Let’s take a look at the default prediction example from ISL

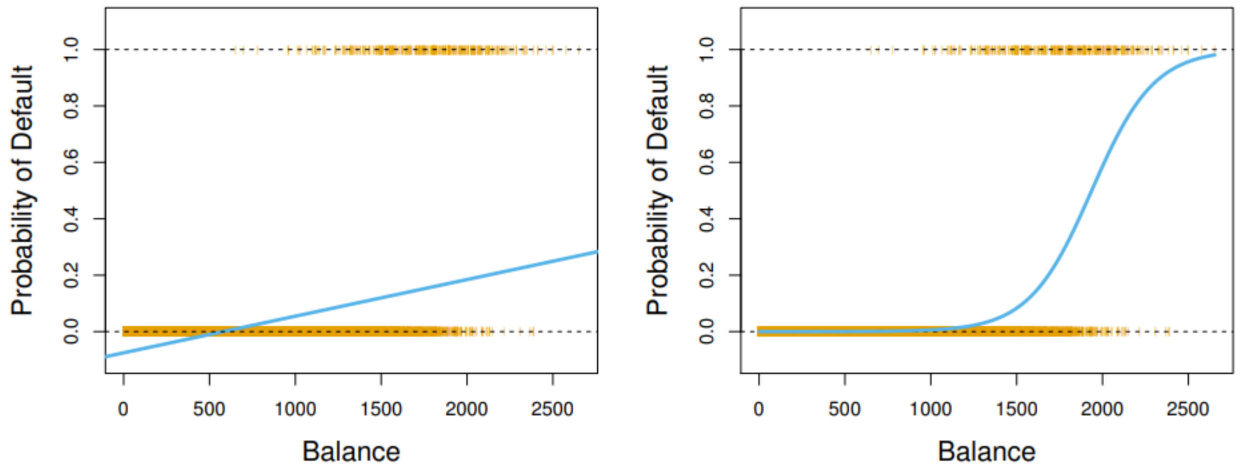


FIGURE 4.2. *Classification using the* `Default` *data.* Left: *Estimated probability of* `default` *using linear regression. Some estimated probabilities are negative! The orange ticks indicate the 0/1 values coded for* `default`*(*`No` *or* `Yes`*).* Right: *Predicted probabilities of* `default` *using logistic regression. All probabilities lie between* 0 *and* 1.

There is a clear relationship between the credit card balance of the customer and the probability of default.

Regression for Time Series: Classification Problems- Logistic Regression

- Let's take a look at the default prediction example from ISL

	Coefficient	Std. error	z-statistic	p-value
Intercept	-10.6513	0.3612	-29.5	<0.0001
balance	0.0055	0.0002	24.9	<0.0001

TABLE 4.1. *For the* Default *data, estimated coefficients of the logistic regression model that predicts the probability of* default *using* balance. *A one-unit increase in* balance *is associated with an increase in the log odds of* default *by* 0.0055 *units.*

For instance, for a customer with a balance of $1000, we predict the probability of default as:

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}} = \frac{e^{-10.6513+0.0055\times 1{,}000}}{1+e^{-10.6513+0.0055\times 1{,}000}} = 0.00576,$$

Regression for Time Series: Classification Problems- Logistic Regression

- We can also use binary (dummy) predictors. Let's check whether the card holder being a student has any effect on the probability of default.

	Coefficient	Std. error	z-statistic	p-value
Intercept	-3.5041	0.0707	-49.55	<0.0001
student[Yes]	0.4049	0.1150	3.52	0.0004

TABLE 4.2. *For the* Default *data, estimated coefficients of the logistic regression model that predicts the probability of* default *using student status. Student status is encoded as a dummy variable, with a value of 1 for a student and a value of 0 for a non-student, and represented by the variable* student[Yes] *in the table.*

$$\widehat{\Pr}(\texttt{default=Yes}|\texttt{student=Yes}) = \frac{e^{-3.5041+0.4049\times 1}}{1+e^{-3.5041+0.4049\times 1}} = 0.0431,$$

$$\widehat{\Pr}(\texttt{default=Yes}|\texttt{student=No}) = \frac{e^{-3.5041+0.4049\times 0}}{1+e^{-3.5041+0.4049\times 0}} = 0.0292.$$

Regression for Time Series: Logistic Regression

- We can extend the model directly to p predictors X_1, X_2, ,,,, X_p. Let $\mathbf{X} = (X_1, X_2, ..., X_n)$:

$$p(\mathbf{X}) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\ldots+\beta_p X_p}}$$

- And use maximum likelihood estimation to find $\hat{\beta}_0$ and $\hat{\beta}_1$, $\ldots\hat{\beta}_p$.

	Coefficient	Std. error	z-statistic	p-value
`Intercept`	−10.8690	0.4923	−22.08	<0.0001
`balance`	0.0057	0.0002	24.74	<0.0001
`income`	0.0030	0.0082	0.37	0.7115
`student[Yes]`	−0.6468	0.2362	−2.74	0.0062

TABLE 4.3. *For the* `Default` *data, estimated coefficients of the logistic regression model that predicts the probability of* `default` *using* `balance`, `income`, *and student status. Student status is encoded as a dummy variable* `student[Yes]`, *with a value of 1 for a student and a value of 0 for a non-student. In fitting this model,* `income` *was measured in thousands of dollars.*

Regression for Time Series: Logistic Regression

- There are versions of logistic regressions for multiple categories of responses (i.e. Not only 1 or 0 but 0 or 1 or 2, good,medium, bad, terrible etc.)
- But the better tool for multiple categories of responses is linear discriminant analysis (LDA)
  - Better stability properties with multiple predictors.
- For our purposes, we'll stick to logistic regression (and to binary classification)
  - The principles are similar when there are more than two categories

Regression for Time Series: Logistic Regression – Classification Errors

- At the end of the day, we are usually asked to classify the data (label it 0 or 1). The probabilities must be converted to 0's and 1's.
- This does not appear to be hard to do. If $p(X)>1/2$, we convert it to 1 and otherwise we convert it to 0.
  - But we don't have take a threshold of 1/2. Depending on the decision we might prefer 9/10 or ¼ or anything else.
- There are two types of classification errors
  - An individual who defaults is incorrectly classified as 'non-default'.
  - An individual who does not default is incorrectyl classified as 'default'
- For decision making purposes, it's important to determine the rates of both errors.