Proposed solution to: Exercise 5 Classification, trees and clustering

In Appendix you will find a time series with a data record for each time step. This record or feature vector consists of three elements X1, X2 and Y= Alarm type. X1 and X2 specifies the observed value for a given alarm level. This data set is going to be your training set for classification.

Use a spreadsheet or similar to:

1. Calculate the average value for X1 and X2

7,18181818 8,24848485

2. Calculate the correlation between X1 and X2 (cross correlation)

0,58510877 or almost 59%

3. Calculate the linear regression between X1 and X2. What are the weights (coefficients)?

W0 (crossing point)	2,69037465
W1(gradient)	0,5445174

4. What is the co-variance between X1 and X2?

7,7384573

5. Based on 3 and 4 how well could X1 serve as a predictor for X2?

It would produce unreliable results due to the low covariance and mediocre fit

6. Calculate moving average over span s= 3 for X1. Repeat the exercise for s= 6 for X1. Compare with the original time series for X1. How well can the moving average be used as a predictor?

t	s= 3	s=6	X1	(X1-f(X1,3))**2	(X1-f(X1,6))**2
1	#I/T	#I/T	7,5		
2	#I/T	#I/T	2,2		
3	4,3	#I/T	3,2	#I/T	
4	3,2	#I/T	4,2	0,01	
5	3,9	#I/T	4,3	1,21	
6	4,33333333	4,31666667	4,5	0,36	#I/T
7	3,66666667	3,43333333	2,2	4,551111111	4,48027778
8	3,13333333	3,51666667	2,7	0,934444444	0,53777778
9	3,13333333	3,73333333	4,5	1,867777778	0,96694444

10	5,23333333	4,45	8,5	28,80111111	22,7211111
11	7,53333333	5,33333333	9,6	19,06777778	26,5225
12	8,53333333	5,83333333	7,5	0,001111111	4,69444444
13	7,2	6,21666667	4,5	16,26777778	1,77777778
14	5,33333333	6,43333333	4	10,24	4,91361111
15	5,7	7,11666667	8,6	10,67111111	4,69444444
16	7,6	7,4	10,2	20,25	9,50694444
17	10,3333333	7,83333333	12,2	21,16	23,04
18	11,9666667	8,83333333	13,5	10,02777778	32,1111111
19	13,7666667	10,6833333	15,6	13,20111111	45,7877778
20	13,1333333	11,7333333	10,3	12,01777778	0,14694444
21	11,9	11,9333333	9,8	11,11111111	3,73777778
22	9,46666667	11,6166667	8,3	12,96	13,2011111
23	7,9	10,5166667	5,6	14,95111111	36,2002778
24	7,23333333	9,56666667	7,8	0,01	7,38027778
25	6,03333333	7,75	4,7	6,417777778	23,6844444
26	6,76666667	7,33333333	7,8	3,121111111	0,0025
27	8,6	7,91666667	13,3	42,68444444	35,6011111
28	8,16666667	7,1	3,4	27,04	20,4002778
29	7,43333333	7,1	5,6	6,587777778	2,25
30	5,63333333	7,11666667	7,9	0,217777778	0,64
31	6,43333333	7,3	5,8	0,027777778	1,73361111
32	6,16666667	6,8	4,8	2,667777778	6,25
33	7,66666667	6,65	12,4	38,85444444	31,36

Prediction

T	s= 3	X1	s=6
1	#I/T	7,5	#I/T
2	#I/T	2,2	#I/T
3	4,3	3,2	#I/T

4	3,2	4,2	#I/T
5	3,9	4,3	#I/T
6	4,33333333	4,5	4,31666667
7	3,66666667	2,2	3,43333333
8	3,13333333	2,7	3,51666667
9	3,13333333	4,5	3,73333333
10	5,23333333	8,5	4,45
11	7,53333333	9,6	5,33333333
			5,83333333
			6,21666667
			6,43333333
			7,11666667
			7,4

With s=3

We see that for the values t=1, t=2 there is no way we can calculate the average. We need 3 values. This yield an average that is a prediction for t=4. Hence we can compare the value of t=4 with the moving average of the 3 former steps.

The estimate for t=4 is 4,3 and the actual value is 4,2

The estimate for t=5 is 3,2 while the actual value is 4,3

The estimate for t=6 is 3,9 while the actual value is 4,5

The estimate for t=7 is 4,3 while the actual value is 2,2

The prediction quality is poor

With s=6

The predicted value for t =6 is 4,3 while the actual value is 2,2

The predicted value for t=7 is 3,4 while the actual value is 2,7

The predicted value for t=8 is 3,7 while the actual value is 8,7

The prediction quality is poor

7. Calculate the autocorrelation with a delay of t=1, t=3 and t=10 for X1. What does this tell you about seasonality? How can you use the autocorrelation to predict the value at t+1?

	d=1	d=3	d=10
Correlation	0,53313532	0,34469259	-0,27428899

The history of X1 is almost as good a predictor as X2 when d=1. Otherwise the history yields little help.

8. K-NN

t	X1	X2
34	5	8
35	10	7
36	20	1

What are the likely alarm states at these future intervals when k=5

New records at t	X1	X2	state						
34	5	8	Imminent	2	Pending	2	Small	1	P(Not small|t=34) = 80%
35	10	7	Imminent	1	Pending	4	Small	0	P(Pending|t=35) = 80%
36	20	1	Imminent	3	Pending	2	Small	0	P(Imminent|t=36) = 60%

9. Use the K-means method to do the same thing. Run one iteration and check with predefined categorizations. How many negatives do you find. What is the weakness of the method?

k-NN	K = 3	Prototypes in red		1st iteration	

t	x1	x2	COG Small		COG pending		COG Imminent		COG Small	COG pend
1	7,5	8,5	4,3	5,8	7,5	8,5	8,6	13,2	0	
2	2,2	7,3	4,3	5,8	7,5	8,5	8,6	13,2	6,66	29
3	3,2	4,3	3,25	6,55	7,5	8,5	8,6	13,2	5,065	36
4	4,2	6,7	2,15	3,6167	7,5	8,5	8,6	13,2	13,7094444	14
5	4,3	5,8	1,5875	2,5792	7,5	8,5	8,6	13,2	17,7314236	17
6	4,5	3,5	1,5875	2,5792	5,9	7,15	8,6	13,2	9,33059028	15,
7	2,2	2,5	1,5219	1,2698	5,9	7,15	8,6	13,2	1,97326606	35,

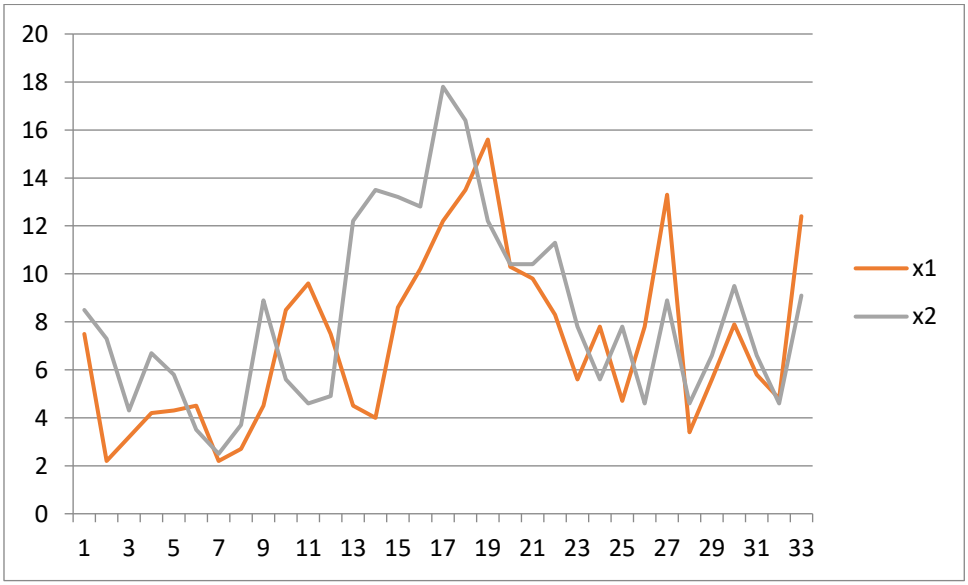
8	2,7	3,7	0,7444	0,994	5,9	7,15	8,6	13,2	11,1471306	22,
9	4,5	8,9	0,5741	1,649	5,9	7,15	8,6	13,2	67,990087	5,0
10	8,5	5,6	0,5741	1,649	3,467	5,35	8,6	13,2	78,4309411	25,39
11	9,6	4,6	0,5741	1,649	3,467	5,35	8,55	9,4	90,1759897	38,18
12	7,5	4,9	0,5741	1,649	3,467	5,35	6,05	4,6667	58,5376564	16,47
13	4,5	12,2	0,5741	1,649	3,467	5,35	3,3875	2,3917	126,736733	47,99
14	4	13,5	0,5741	1,649	1,992	4,3875	3,3875	2,3917	152,183413	87,0
15	8,6	13,2	0,5741	1,649	1,198	3,5775	3,3875	2,3917	197,841434	147,3
16	10,2	12,8	0,5741	1,649	1,198	3,5775	2,3975	3,1183	217,003629	166,0
17	12,2	17,8	0,5741	1,649	1,198	3,5775	2,0996	2,6531	396,017448	323,3
18	13,5	16,4	0,5741	1,649	2,233	3,5629	2,0996	2,6531	384,672066	291,7
19	15,6	12,2	0,5741	1,649	2,248	2,8518	2,0996	2,6531	337,102545	265,6
20	10,3	10,4	0,5741	1,649	2,231	1,8815	2,0996	2,6531	171,173983	137,6
21	9,8	10,4	0,5741	1,649	2,231	1,8815	2,0666	2,1755	161,698045	129,8
22	8,3	11,3	0,5741	1,649	2,231	1,8815	1,6952	1,7965	152,832045	125,5
23	5,6	7,8	0,5741	1,649	1,17	1,4646	1,6952	1,7965	63,0949342	59,76
24	7,8	5,6	0,5741	1,649	1,17	1,4646	0,9119	1,1996	67,8246286	61,05
25	4,7	7,8	0,5741	1,649	0,897	0,7065	0,9119	1,1996	54,8582467	64,7
26	7,8	4,6	0,7534	1,3499	0,897	0,7065	0,9119	1,1996	60,2174781	62,81
27	13,3	8,9	1,0692	0,7437	0,897	0,7065	0,9119	1,1996	216,117673	220,9
28	3,4	4,6	1,0692	0,7437	0,897	0,7065	1,5791	1,1222	20,3035261	21,42
29	5,6	6,6	1,0692	0,7437	0,897	0,7065	0,4979	0,5722	54,8242074	56,85
30	7,9	9,5	0,741	0,816	0,897	0,7065	0,4979	0,5722	126,663368	126,3
31	5,8	6,6	0,741	0,816	0,8	0,9279	0,4979	0,5722	59,0482794	57,1
32	4,8	4,6	0,741	0,816	0,55	0,6273	0,4979	0,5722	30,7942003	33,84

33	12,4	9,1	0,5541	0,5416	0,55	0,6273	0,4979	0,5722	213,571561	212,2
			0,5541	0,5416	1,079	0,8106	0,4979	0,5722		

Start all over again with COGs calculated from step 33

10. Create a classification tree between alarm states Imminent and not Imminent.

Appendix:



t	x1	x2	Alarm state
1	7,5	8,5	Pending
2	2,2	7,3	Pending
3	3,2	4,3	Small
4	4,2	6,7	Pending
5	4,3	5,8	Small
6	4,5	3,5	Small
7	2,2	2,5	Small
8	2,7	3,7	Small
9	4,5	8,9	Imminent
10	8,5	5,6	Pending
11	9,6	4,6	Pending
12	7,5	4,9	Pending
13	4,5	12,2	Imminent
14	4	13,5	Imminent
15	8,6	13,2	Imminent
16	10,2	12,8	Imminent
17	12,2	17,8	Imminent

18	13,5	16,4	Imminent
19	15,6	12,2	Imminent
20	10,3	10,4	Pending
21	9,8	10,4	Imminent
22	8,3	11,3	Imminent
23	5,6	7,8	Imminent
24	7,8	5,6	Imminent
25	4,7	7,8	Small
26	7,8	4,6	Pending
27	13,3	8,9	Imminent
28	3,4	4,6	Small
29	5,6	6,6	Small
30	7,9	9,5	Pending
31	5,8	6,6	Pending
32	4,8	4,6	Pending
33	12,4	9,1	Imminent