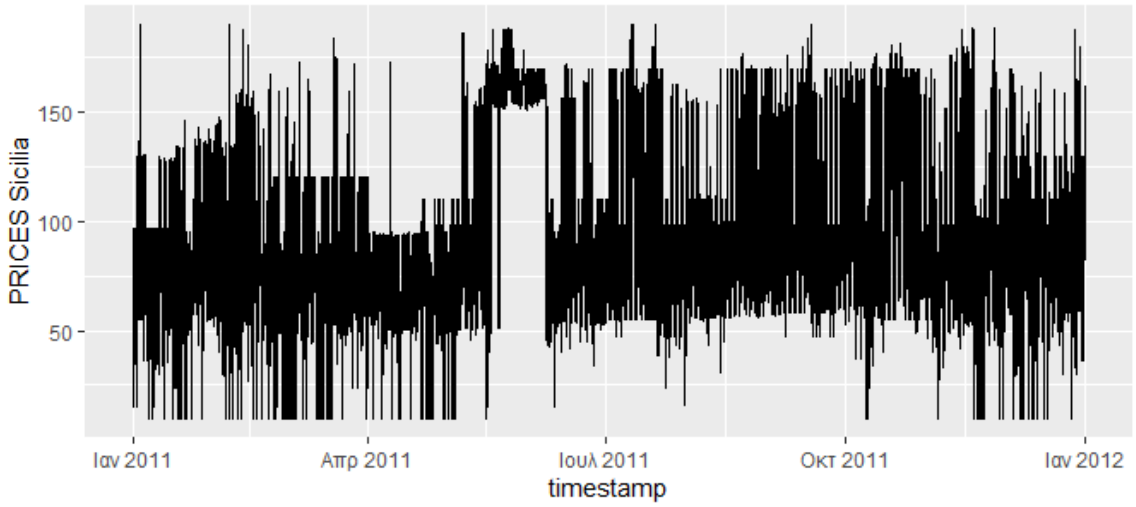
```
[1] 2067
```

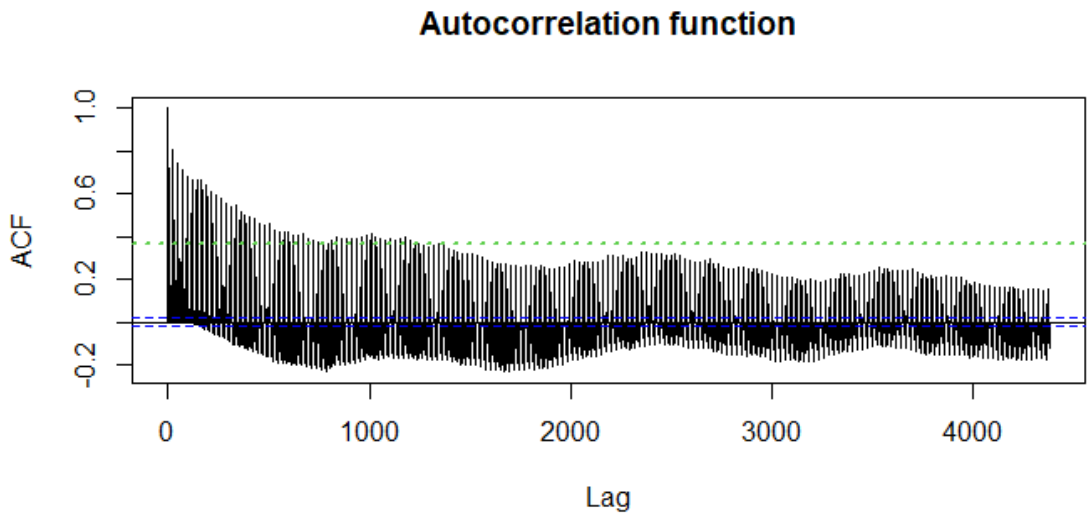
```
# A tibble: 1 x 1
  timestamp
  <dttm>
1 2011-03-28 02:59:59
> |
```

produce the timeseries graph

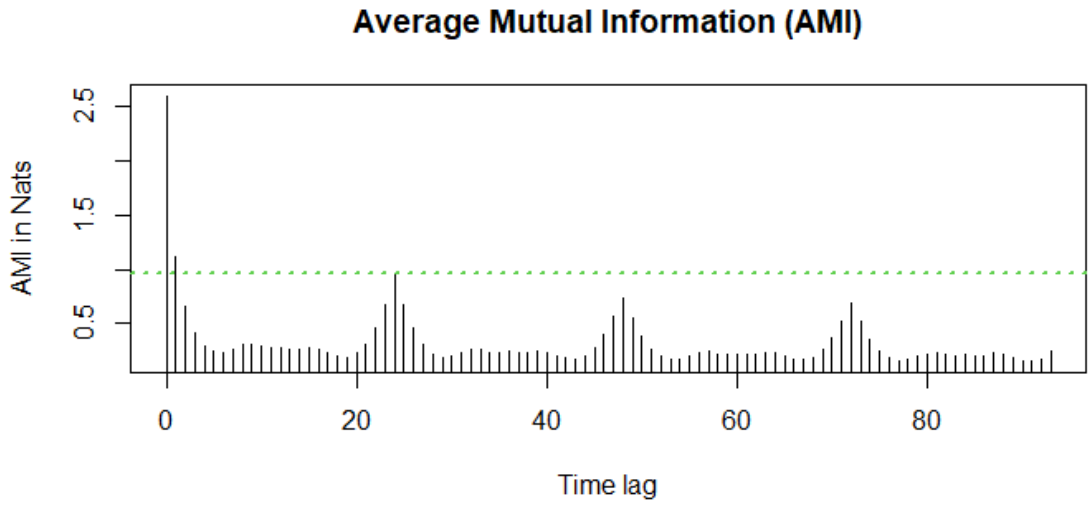


calculate the proper time delay, we prefer ami beacuse usually is smaller

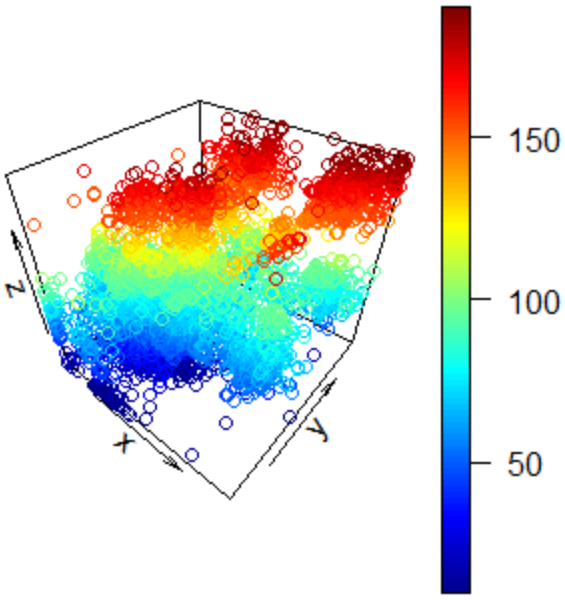
[1] 5



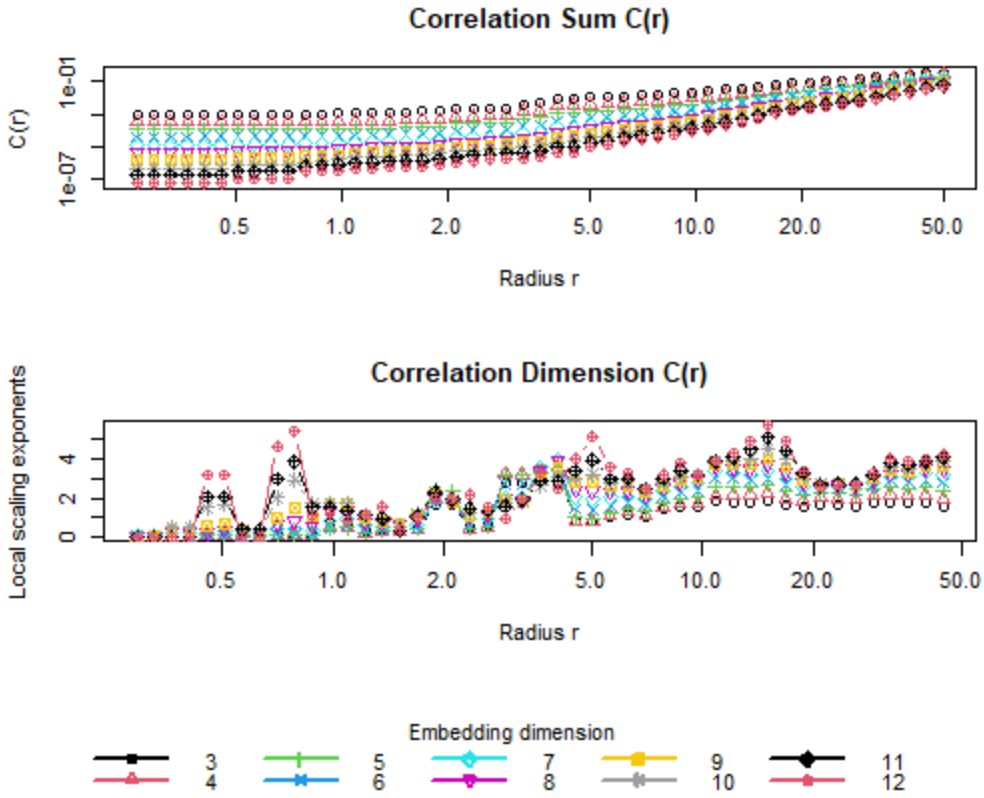
[1] 2

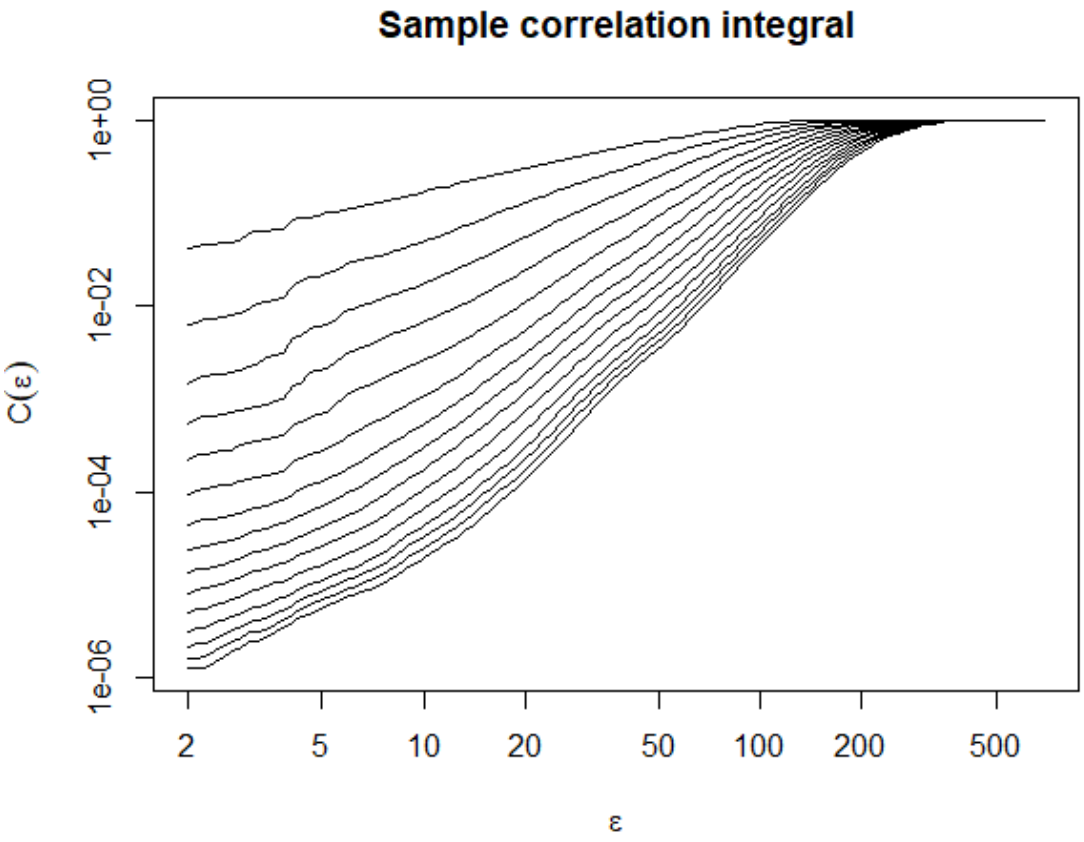


produce the phase portrait graph

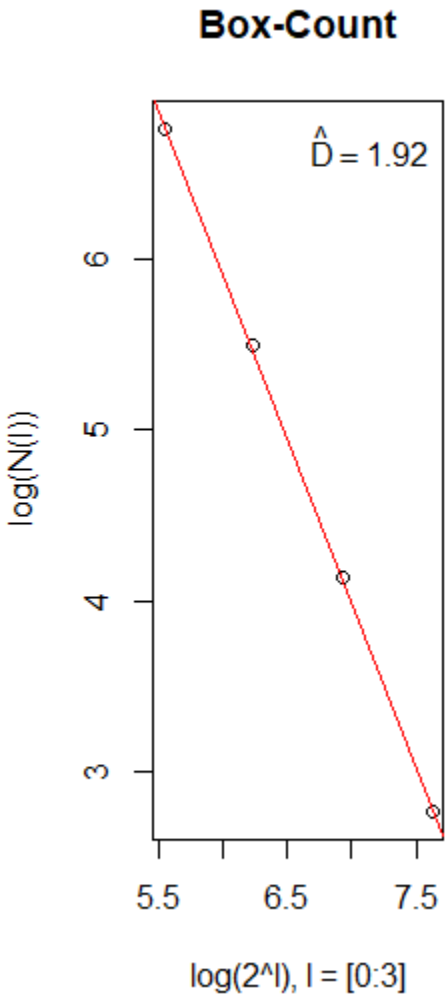


calculate and produce the correlation integrals' graph

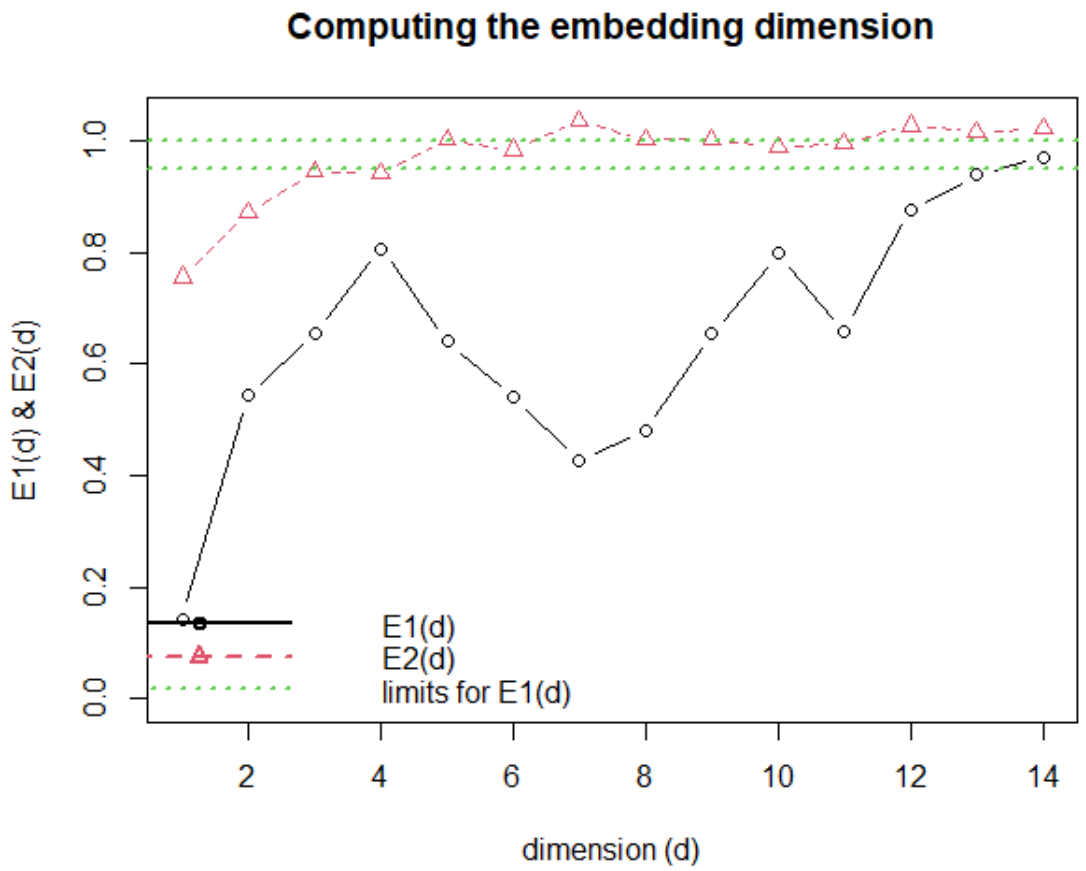

Correlation Sum C(r)
C(r)
1e-01
1e-07
0.5
1.0
2.0
5.0
10.0
20.0
50.0
Radius r
Correlation Dimension C(r)
Local scaling exponents
0
2
4
0.5
1.0
2.0
5.0
10.0
20.0
50.0
Radius r
Embedding dimension
3
4
5
6
7
8
9
10
11
12


Sample correlation integral
C(ε)
1e+00
1e-02
1e-04
1e-06
2
5
10
20
50
100
200
500
ε

[1] 1.919714



produce the correlation dimension vs. embedding dimension graph and estimate the fractal dimension, the minimum embedding dimension and the essential embedding dimension



calculate Kolmogorov entropy and (optional) produce the relevant graph

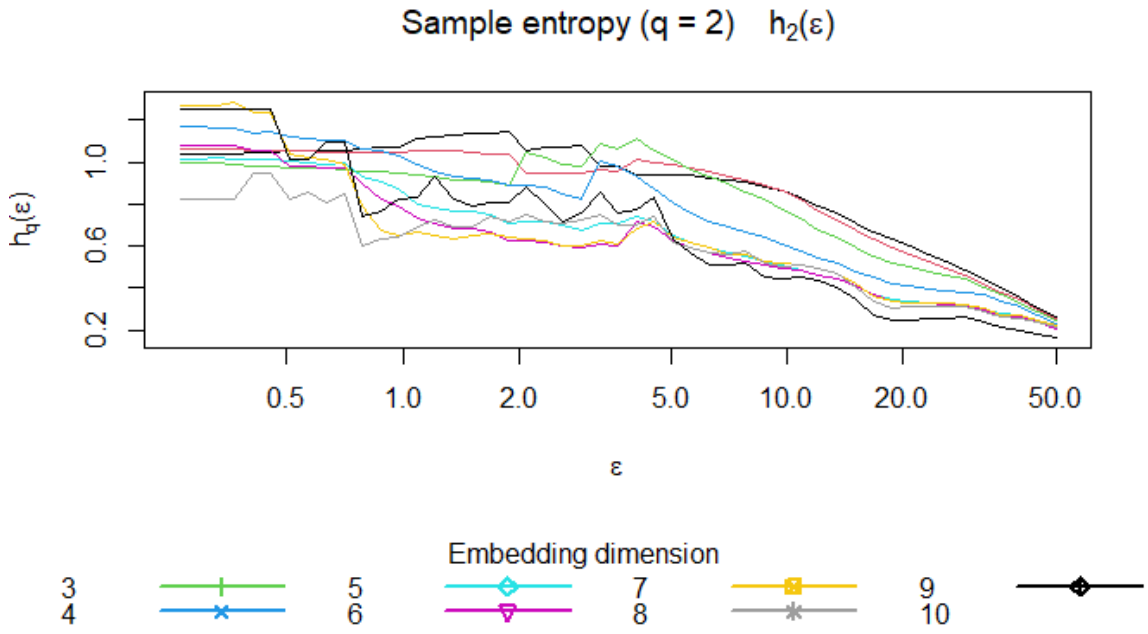
```
$sample.entropy
          50 44.8234892524193 40.182903775235 36.0227591100049
3  0.2641453        0.3059517       0.3573787        0.4005146
4  0.2569363        0.2958846       0.3446784        0.3811782
5  0.2457155        0.2841385       0.3330691        0.3672939
6  0.2307521        0.2677599       0.3126637        0.3401602
7  0.2104720        0.2409963       0.2711341        0.2814767
8  0.2083508        0.2363349       0.2619081        0.2726667
9  0.2182279        0.2469259       0.2693881        0.2749214
10 0.2116332        0.2364240       0.2524616        0.2600356
11 0.1633748        0.1835647       0.1999241        0.2125204
   32.2933151161959 28.949981260716 25.9527834779489 23.2658862258842
3         0.4446016       0.4898669        0.5262522        0.5602974
4         0.4233621       0.4648386        0.4933682        0.5257497
5         0.4070489       0.4408280        0.4591988        0.4810472
6         0.3665322       0.3808008        0.3878425        0.3981696
7         0.2978605       0.3122692        0.3242605        0.3328586
8         0.2948666       0.3105662        0.3201579        0.3273029
9         0.3012186       0.3190207        0.3263928        0.3298500
10        0.2912779       0.3116166        0.3124901        0.3153179
11        0.2421413       0.2604491        0.2562744        0.2573003
   20.8571640238786 18.6978173492054 16.7620282999161 15.0266519070008
8           1.0779321
9           1.2704625
10          0.8266786
11          1.2527630

$embedding.dims
[1]  3  4  5  6  7  8  9 10 11

$entr.order
[1] 2

$radius
 [1] 50.0000000 44.8234893 40.1829038 36.0227591 32.2933151 28.9499813
 [7] 25.9527835 23.2658862 20.8571640 18.6978173 16.7620283 15.0266519
[13] 13.4709394 12.0762902 10.8260292  9.7052081  8.7004258  7.7996689
[19]  6.9921675  6.2682669  5.6193119  5.0375433  4.5160054  4.0484623
[25]  3.6293242  3.2535795  2.9167357  2.6147654  2.3440582  2.1013773
[31]  1.8838213  1.6887889  1.5139482  1.3572088  1.2166967  1.0907318
[37]  0.9778081  0.8765754  0.7858234  0.7044669  0.6315333  0.5661505
[43]  0.5075368  0.4549914  0.4078861  0.3656576  0.3278009  0.2938636
[49]  0.2634399
```

```
attr(,"class")
[1] "sampleEntropy"
attr(,"time.lag")
[1] 2
attr(,"id")
[1] "df$`PRICES Sicilia`"
attr(,"theiler.window")
[1] 100
> 
```

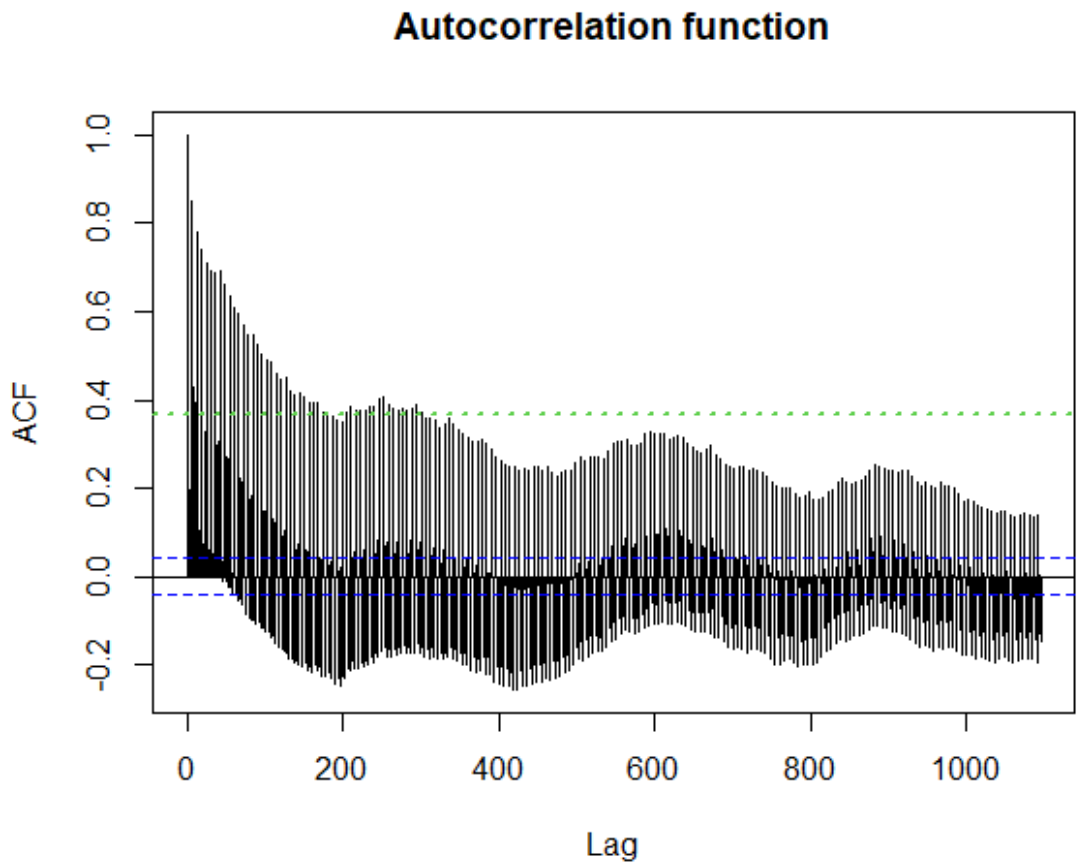


Convert the power-price timeseries from hourly sampling (8160 points) to 4-hour sampling timeseries (2190 points), averaging the 4 points to be replaced for each new sample. Then follow the previous procedure.

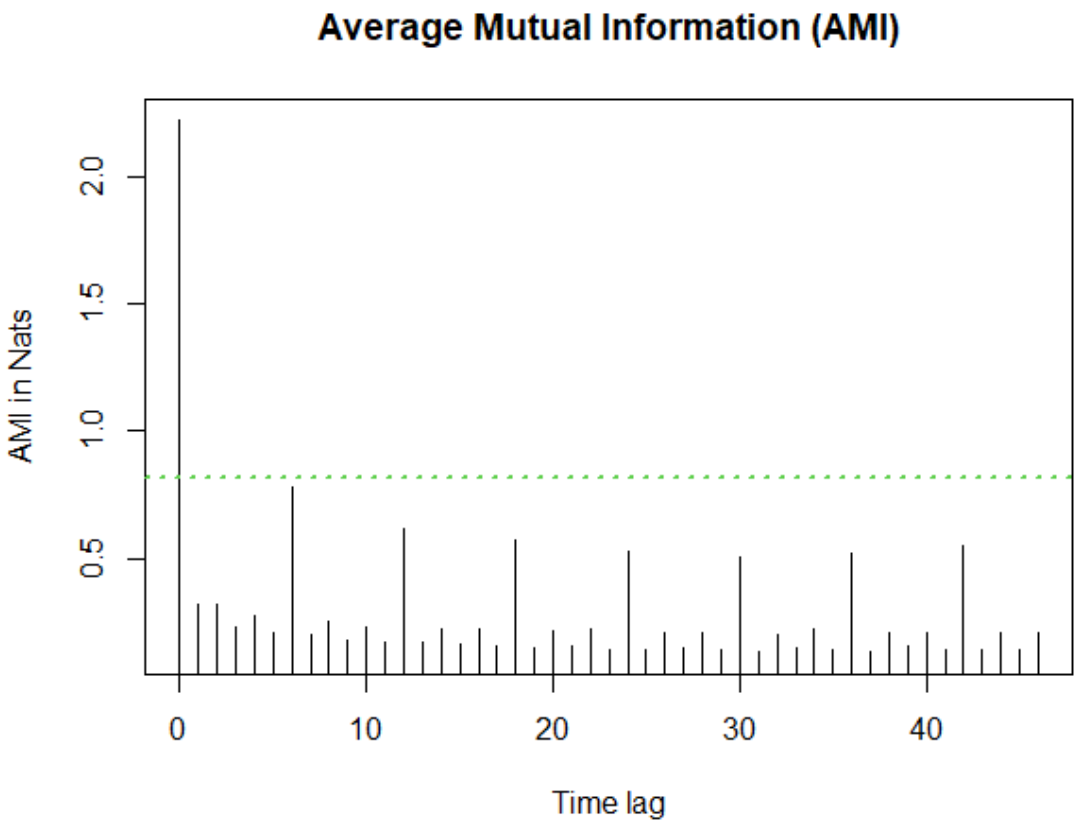
```
        rn      new
   1:    1  57.8500
   2:    2  44.2575
   3:    3  77.0125
   4:    4  79.0025
   5:    5  90.0025
  ---
2186: 2186  53.1775
2187: 2187  97.2550
2188: 2188  92.0150
2189: 2189 121.3075
2190: 2190  93.7700
> 
```

calculate the proper time delay, we prefer ami beacuse usually is smaller

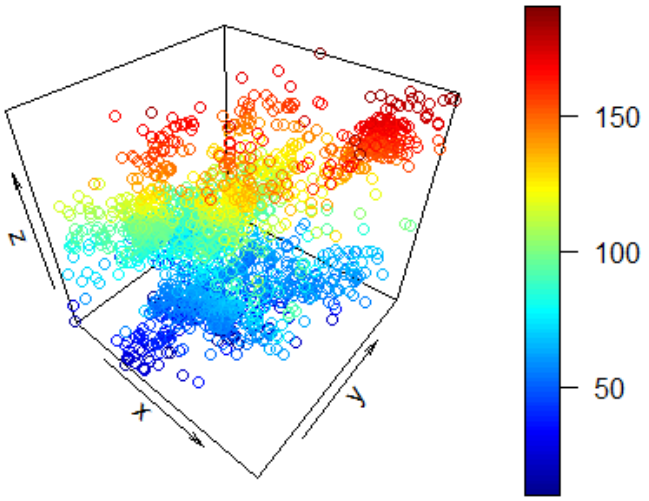
[1] 2


Autocorrelation function
ACF
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
0
200
400
600
800
1000
Lag

[1] 1

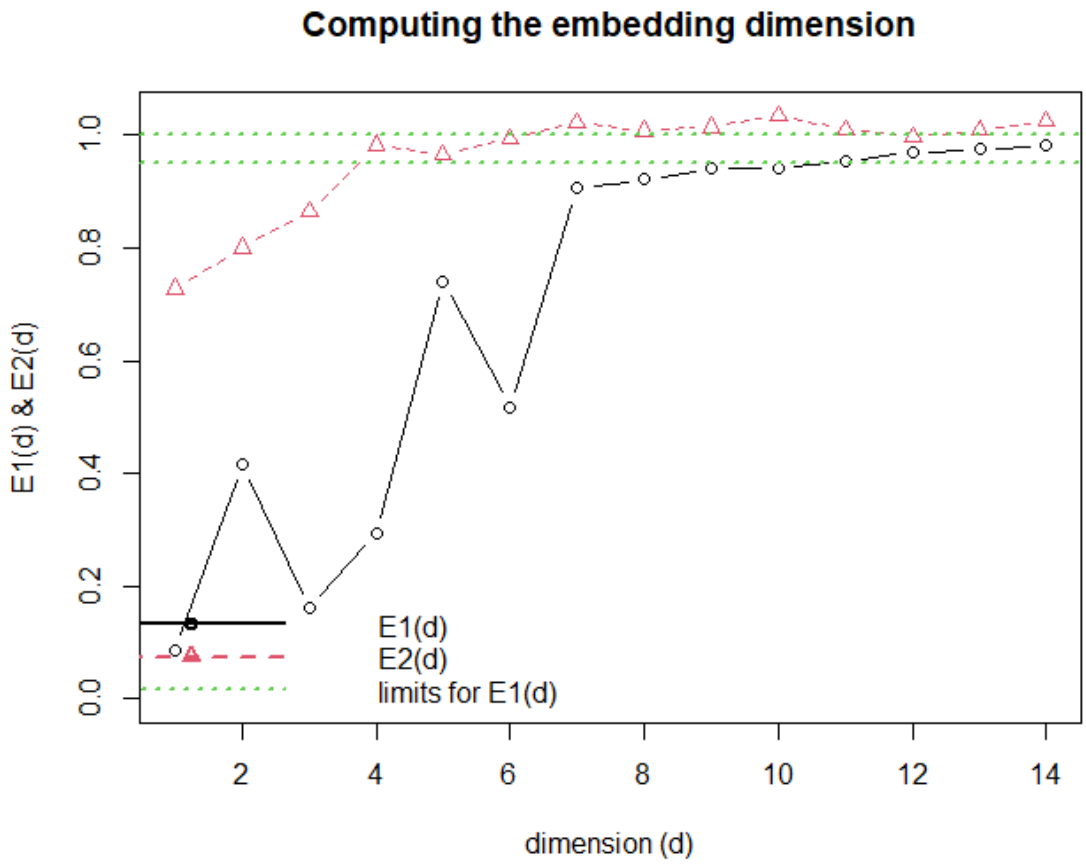


produce the phase portrait graph



produce the correlation dimension vs. embedding dimension graph and estimate the fractal dimension, the minimum embedding dimension and the essential embedding dimension

[1] 11



calculate Kolmogorov entropy and (optional) produce the relevant graph

```
$sample.entropy
          20
0  1.2292706
1  1.0046236
2  0.7942027
3  0.6429231
4  0.4996131
5  0.4255500
6  0.2901325
7  0.2896806
8  0.2938041
9  0.2784254
10 0.2660969
11 0.2661351

$embedding.dims
 [1]  0  1  2  3  4  5  6  7  8  9 10 11

$entr.order
[1] 2

$radius
[1] 20

attr(,"class")
[1] "sampleEntropy"
attr(,"time.lag")
[1] 1
attr(,"id")
[1] "newdf$new"
attr(,"theiler.window")
[1] 100
> |
```