طرق التنبؤ الإحصائي

(الجزء الثاني)

تأليف د. عدنان ماجد عبد الرحمن بري

استاذ الأنظمة العشوائية الحركية

جامعة الملك سعود (سابقا)

بسم الله الرحمن الرحيم

الحمد لله رب العالمين و الصلاة و السلام على اشرف خلق الله سيدنا و نبينا محمد و على آله و صحبه أجمعين و بعد.

هذا الجزء الثاني من كتاب طرق التنبؤ الإحصائي و الذي أخذ وقت طويل جدا للإنتهاء وذلك لعدم وجود برنامج يحوي الدوال المستخدمة غير برامج مكلفة جدا و بعيدة عن طائل الطلاب مثل SAS و SPSS أو MATLAB وغيرها.

نظام R و هو من البرامج القوية جدا و الذي يقوم بتطويرة مجموعة كبيرة جدا من الأكاديميين و الباحثين في جميع جامعات العالم تقريبا و الذي هو مجاني (المصدر المفتوح Open Source) و الذي أصبح الأداه الأكثر إستخدام في الجامعات و مراكز الأبحاث و حتى في البنوك و الأعمال الإقتصادية و قد تفوق بكثير على البرامج السابقة.

هذا الجزء الثاني موجه لطلاب الدراسات العليا و يتطرق للمواضيع التالية:

(1) تحليل التدخل Intervention Analysis

(2) نماذج دالة التحويل Transfer Function Models

(3) نماذج المتسلسلات الزمنية المتعددة Multivariate Time Series Models و نماذج المتسلسلات الزمنية الموجهه Vector Time Series Models

(4) نماذج فضاء الحالة ومرشح كالمن State Space Models and Kalman Filtering

(5) نماذج ARCH ونماذج GARCH وتطبيقاتها في التنبؤ المالي Finantial Time Series Forecasting

(6) الشبكات العصبية Neural Networks وإستخدامها في التنبؤ الإحصائي.

هذا و ارجوا من الله التوفيق.

المؤلف

المحتويات:

(1) تحليل التدخل Intervention Analysis 4

(2) نماذج دالة التحويل Transfer Function Models 22

(3) نماذج المتسلسلات الزمنية المتعددة Multivariate Time Series Models.
و نماذج المتسلسلات الزمنية الموجهه Vector Time Series Models 74

(4) نماذج فضاء الحالة ومرشح كالمن State Space Models and Kalman
Filtering 106

(5) نماذج ARCH ونماذج GARCH وتطبيقاتها في التنبؤ المالي Finantial
Time Series Forecasting 143

(6) الشبكات العصبية Neural Networks وإستخدامها في التنبؤ الإحصائي.
.....211

ملحق الهشاشة Volatility 252

ملحق مكتبة dlm في R 255

المراجع ... 258

الفصل الأول

تحليل التدخل وكشف الخوارج

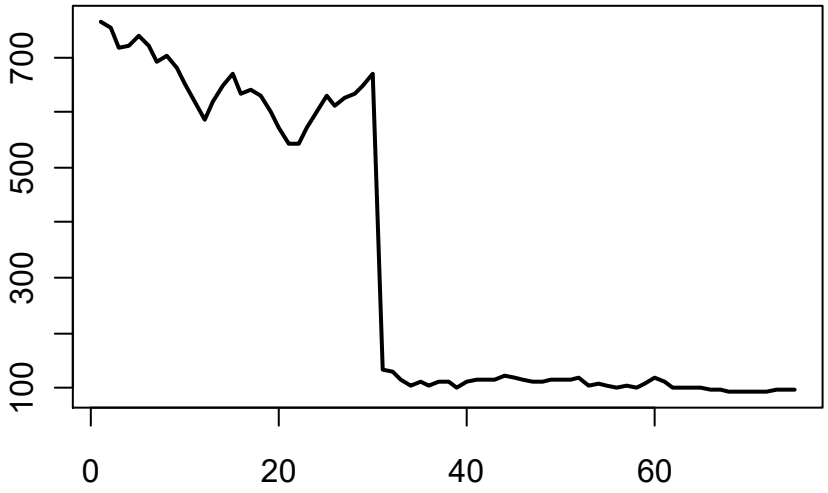
Intervention Analysis and Outliers Detection

يستخدم تحليل التدخل عندما يطرأ تغير على المتسلسلة الزمنية كأن يتغير مستواها بصفة دائمة أو مؤقتة أو تدريجيا ويحدث ذلك نتيجة تغير في منهجية أو قرار أو أحداث اخرى تدعى تدخلات Interventions فمثلا المتسلسلة الزمنية التي تمثل سعر الإقفال اليومي بالريال لسهم بنك الرياض مثلا حدث لة تغيير عندما قامت هيئة سوق المال بتقسيم قيمة السهم لفئات أقل.

(سعر الإقفال لسهم بنك الرياض بالريال - المصدر هيئة السوق المالية السعودية)

التاريخ	إقفال	التاريخ	إقفال	التاريخ	إقفال
3/1/2006	765	4/1/2006	613.25	5/1/2006	116.75
3/2/2006	753	4/2/2006	626	5/2/2006	115
3/4/2006	716	4/3/2006	632	5/3/2006	115
3/5/2006	720	4/4/2006	647	5/4/2006	117.25
3/6/2006	740	4/5/2006	669	5/6/2006	105.75
3/7/2006	719	4/8/2006	134.5	5/7/2006	108
3/8/2006	690	4/9/2006	129	5/8/2006	106
3/9/2006	703	4/10/2006	116.25	5/9/2006	102.5
3/11/2006	682	4/11/2006	104.75	5/10/2006	103
3/12/2006	650	4/12/2006	111.25	5/11/2006	99.5
3/13/2006	618	4/15/2006	103.5	5/13/2006	109.25
3/14/2006	588	4/16/2006	112	5/14/2006	118.25
3/15/2006	617	4/17/2006	113	5/15/2006	112
3/16/2006	647	4/18/2006	101.75	5/16/2006	101
3/18/2006	668	4/19/2006	111.75	5/17/2006	99.75
3/19/2006	635	4/22/2006	116	5/18/2006	102
3/20/2006	640	4/23/2006	115	5/20/2006	100.5
3/21/2006	630	4/24/2006	114	5/21/2006	98
3/22/2006	599	4/25/2006	122	5/22/2006	97.75
3/23/2006	570	4/26/2006	119.75	5/23/2006	93
3/25/2006	542	4/27/2006	116	5/24/2006	94
3/26/2006	544	4/29/2006	112	5/25/2006	94.25

3/27/2006 571	4/30/2006 112.5	5/27/2006 93.5
3/28/2006 599		5/28/2006 94
3/29/2006 628		5/29/2006 95.5
		5/30/2006 96.75
		5/31/2006 96.75



Close price for Riyadh Bank

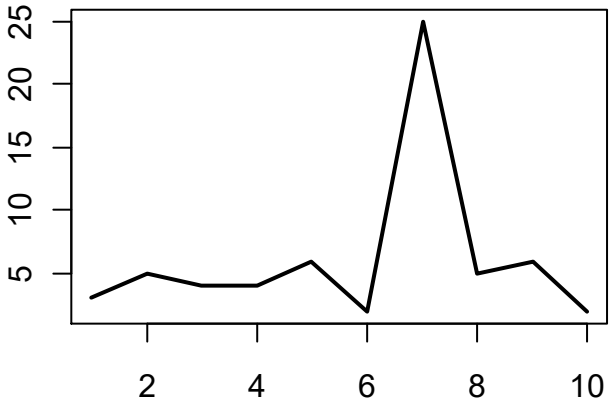
نلاحظ أن سعر الإقفال للسهم تغير بين اليومين 2006/5/4 و 2006/8/4 حيث تم تجزئة السهم إلى 5 أسهم كما تم إصدار بعض القواعد والنظم التي غيرت من إداء السوق.

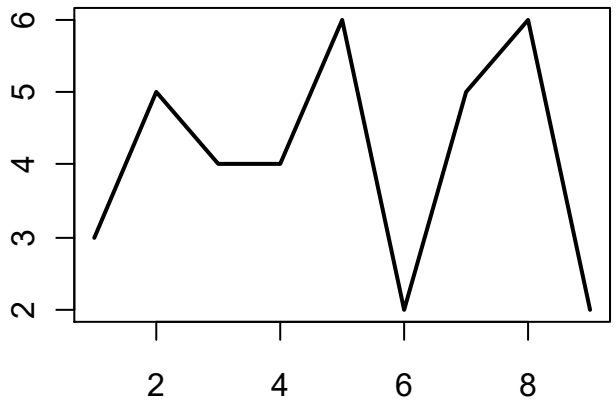
ألخوارج وأنواعها:

تعريف: القيمة الخارجة Outlier هي مشاهدة يكون إنحرافها عن متوسط المشاهدات بعد إستبعادها أكبر من 3 إنحرافات معيارية.

مثال: للمشاهدات 2, 6, 5, 2, 25, 6, 4, 4, 5, 3 المشاهدة 25 هي قيمة خارجة وذلك للأسباب الآتية:

1- رسم المشاهدات مع وبدون القيمة الخارجة





متوسط المشاهدات بدون المشاهدة 25 هو 4.1111 وإنحراف معياري 1.5366 وإنحراف المشاهدة الخارجة عن المتوسط هو 20.8889- وبالقيمة المطلقة 20.8889 وهو أكثر من 13 ضعف الإنحراف المعياري.

تعريف آخر: القيمة الخارجة هي مشاهدة تكون

$$x_{outlier} < Q_1 - 3IQ \quad or \quad x_{outlier} > Q_3 + 3IQ$$

حيث IQ مدي ما بين الربيعات Interquartile Range وو Q_1 الربيع الأول First Quartile و Q_3 الربيع الثالث Third Quartile.

مثال: المشاهدات 3, 5, 4, 4, 6, 2, 25, 5, 6, 2

الربيع الأول Q_1 =3.25 والربيع الثالث Q_3= 5.75 مدى مابين الربيعات $IQ = 2.5$ و $3IQ = 7.5$ وواضح أن

$$x_{outlier} > Q_3 + 3IQ$$
$$25 > 5.75 + 7.5 = 13.25$$

ملاحظة: القيمة الخارجة قدتكون أكبر أو أصغر حجما من بقية المشاهدات فمثلا في المثال السابق 25 اكبر حجما من جميع المشاهدات المتبقية وللمشاهدات

3, 5, 4, 4, 6, 2, 0.2, 5, 6, 2 المشاهدة 0.2 أصغر حجما من بقية المشاهدات.

أنواع الخوارج:

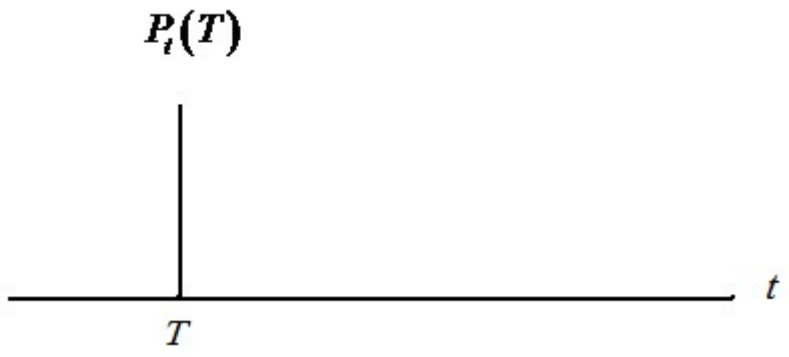
1- خوارج إضافية Additive Outliers (OA)

$$y_t = z_t + \omega_A P_t(T)$$

حيث

$$P_t(T) = \begin{cases} 1, & t = T \\ 0, & t \neq T \end{cases}$$

وتسمى $P_t(T)$ **دالة النبض Pluse Function** ولها الشكل التالي:



2- **خوارج الأخطاء Innovational Outliers (IO)**

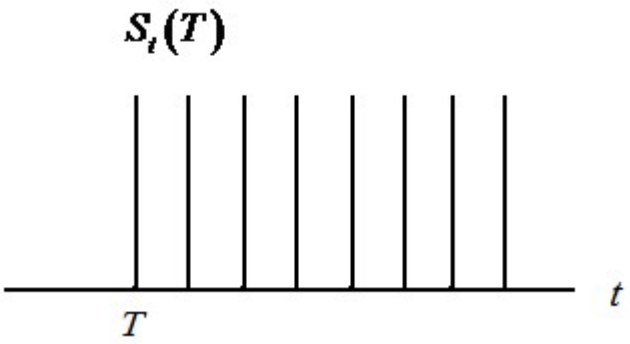
$$z_t = \frac{\theta_q(B)}{\phi_p(B)}\left(a_t + \omega_I P_t(T)\right)$$

3- **إزاحة مستوى Level Shift (LS)**

$$y_t = z_t + \omega_L S_t(T)$$

$$S_t(T) = \begin{cases} 0, & t < T \\ 1, & t \geq T \end{cases}$$

وتسمي $S_t(T)$ **دالة الدرج Step Function** ولها الشكل التالي:



4- **تغير مؤقت Temporary Change (TC)**

$$y_t = z_t + \frac{1}{1-\delta B}\omega_C P_t(T)$$

حيث $0 < \delta < 1$ يسمى عامل التخامد Dampening Factor.

لاحظ أن:

$$P_t(T) = S_t(T) - S_{t-1}(T) = (1-B)S_t(T)$$

وبالتالي نستطيع إستخدام أي من $P_t(T)$ أو $S_t(T)$ في نمذجة التدخل.

يوجد عدة ردود فعل أو تأثير Responses لدوال الدرج والنبض التدخلية فمثلا:

1- صدمة Impact ثابتة وغير معروفة للتدخل يبدأ بالتأثير بعد زمن b وهكذا إعتمادا على نوع التدخل فإن التأثير يكون $\omega B^b S_t(T)$ أو $\omega B^b P_t(T)$

2- يشعر بصدمة التدخل بعد زمن b بعد التدخل ولكن التأثير يكون تدريجي Gradual ويكون لتأثير درجي

$$\frac{\omega B^b}{(1-\delta B)} S_t(T)$$

أو تأثير نبضي

$$\frac{\omega B^b}{(1-\delta B)} P_t(T)$$

حيث $0 < \delta < 1$ كما عرفت سابقا.

بشكل عام يمكن تمثيل التأثر كدالة كسرية

$$\frac{\omega(B) B^b}{\delta(B)}$$

حيث

$$\omega(B) = \omega_0 - \omega_1 B - \cdots - \omega_s B^s$$
$$\delta(B) = 1 - \delta_1 B - \cdots - \delta_r B^r$$

كثيرات حدود في B و b زمن تأخر Time Delay حتى يأخذ التدخل بالتأثير والأوزان ω_jفي كثيرة الحدود $\omega(B)$ تمثل عادة التأثير المتوقع الأولي للتدخل وكثيرة الحدود $\delta(B)$ تقيس تصرف التأثير الدائم للتدخل. أصفار $\delta(B) = 0$ يفترض أن تكون على

أو خارج دائرة الوحدة. جذر الوحدة Unit Root يمثل صدمة تزداد خطيا و الجذر خارج دائرة الحدة يمثل ظاهرة لها تأثير تدريجي.

النموذج العام للتدخل:

يكتب على الشكل

$$z_t = \sum_{j=1}^{k} \frac{\omega_j(B)B^{b_j}}{\delta_j(B)} I_{jt} + \frac{\theta_q(B)}{\phi_p(B)} a_t$$

حيث $I_{jt}, j = 1,2,...,k$ متغيرات تدخل وقد تكون أي من دوال النبض و/أو الدرج وبشكل عام قد يكونوا متغيرات تأشير Indicator Variables.

النموذج السابق هو حالة خاصة من نموذج دالة التحويل العام الذي سنستعرضه لاحقا مع ملاحظة أن $\nu_j(B) = \omega_j(B)B^{b_j} / \delta_j(B)$ تفرض من الأحداث المؤدية للتداخل ولاتحدد بتحليل عكس دالة التحويل العامة.

ملاحظات:

1- يفترض في تحليل التدخل أن معالم النموذج هي نفسها قبل وبعد التدخل.

2- لاتوجد أحداث تدخل آنية (أكثر من حدث واحد) تؤثر على الظاهرة المدروسة.

منهجية النمذجة في تحليل التدخل:

سوف نستعرض خوارزم عام لتحليل التدخل كالتالي:

1- حدد نموذج لـ z_t مستخدما القيم $y_1,...,y_{T-1}$ أي قيم المتسلسلة قبل التدخل.

2- أستخدم النموذج الذي تم بنائه لـ z_t لكي تتنبؤ عن z_t لقيم $t \geq T$ ولتكن $z_T(\ell), \ell \geq T$.

3- أفحص البواقي $y_t - z_T(t), t \geq T$ لكي تحدد $\delta(B)$ و $\omega(B)$.

4- قم بتطبيق نموذج بكامل المشاهدات.

5- تحقق من مناسبة النموذج المرشح.

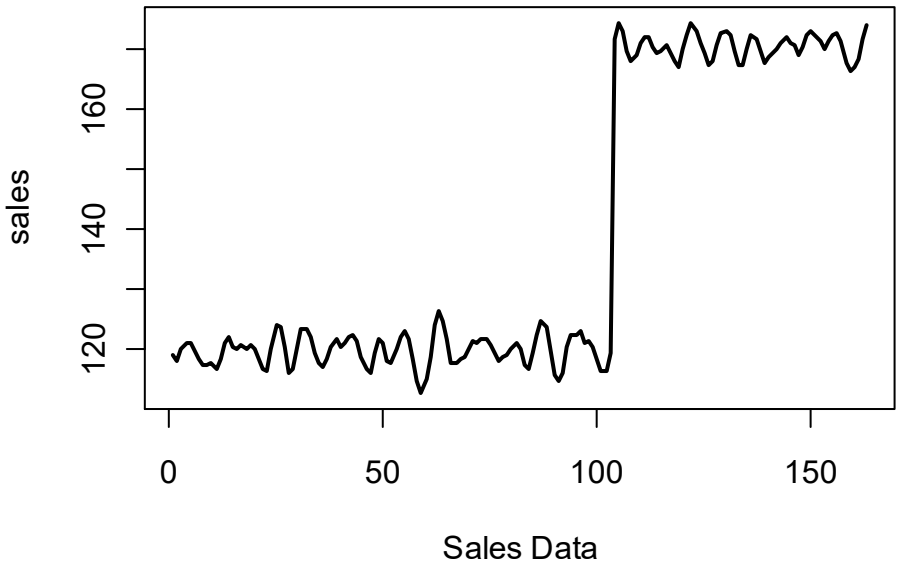
مثال:

رفعت أحد الشركات سعر بيع أحد منتجاتها نظرا لإرتفاع سعر المواد الخام. القيم التالية للأسعار اليومية للمنتج لعدد 163 يوما (حيث أن يوم رفع الأسعار معروف وهو اليوم 103 ترتيبا)

119.22	118.01	120.13	121.12	121.30	119.86	118.54	117.55
117.56	117.79	116.83	118.44	121.15	122.09	120.49	120.16
120.72	120.30	120.73	120.26	118.41	116.67	116.35	120.15
124.01	123.72	120.40	116.32	116.91	120.09	123.49	123.44
122.20	119.38	117.98	117.05	118.53	120.32	121.76	120.63
121.00	122.06	122.59	121.46	118.82	116.88	116.27	119.42
121.75	120.98	118.02	117.85	120.02	122.09	123.04	121.93
118.48	114.73	112.78	115.08	118.75	124.08	126.50	124.74
121.86	117.83	117.70	118.59	118.66	120.24	121.50	121.30
121.90	121.82	120.71	119.64	118.20	118.77	119.21	120.16
121.05	120.27	117.51	116.74	119.37	122.51	124.91	123.70
119.98	115.99	114.73	116.01	120.42	122.59	122.47	123.08
121.26	121.34	120.46	118.46	116.46	116.42	119.41	171.55
174.47	172.86	169.65	167.93	169.06	170.98	172.01	171.98
170.40	169.51	169.62	170.69	169.25	168.05	167.18	170.13
172.47	174.19	172.91	171.03	169.34	167.53	168.11	170.66
172.54	173.17	172.20	169.85	167.42	167.22	169.91	172.51
171.69	169.84	167.81	168.65	169.37	170.02	170.97	172.03
171.15	170.57	168.92	170.48	172.36	173.00	171.86	171.20
170.10	171.29	172.48	172.71	171.30	167.83	166.33	166.97
168.38	171.66	173.98					

ندخل البيانات في R كالتالي:

```
> library(stats)
> library(tseries)
> sales = read.ts("C:/Documents and Settings/amb/Desktop/ts
+ in R/sales.txt",header=TRUE)
> plot.ts(sales, type="l",lwd=2,xlab="Sales Data")
```

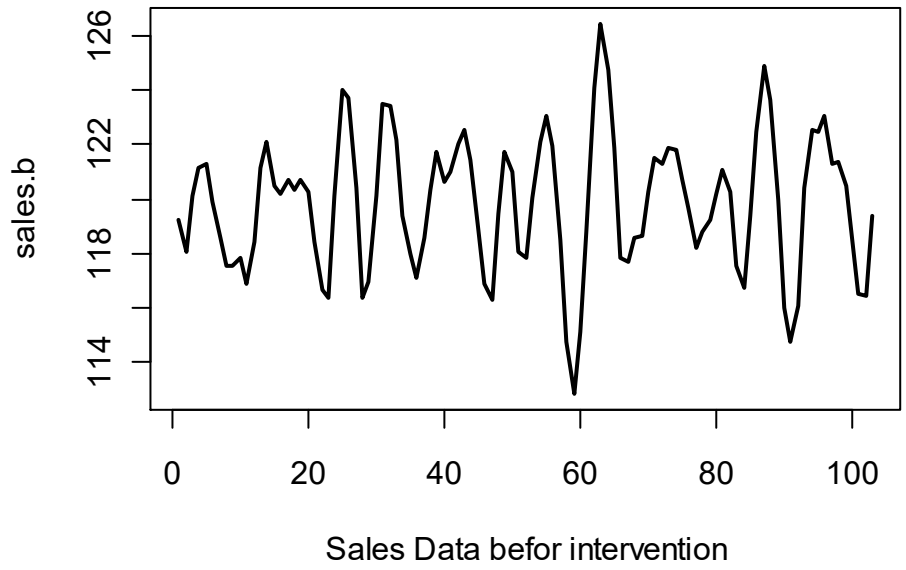


نأخذ جزء المتسلسلة قبل التدخل بإستخدام

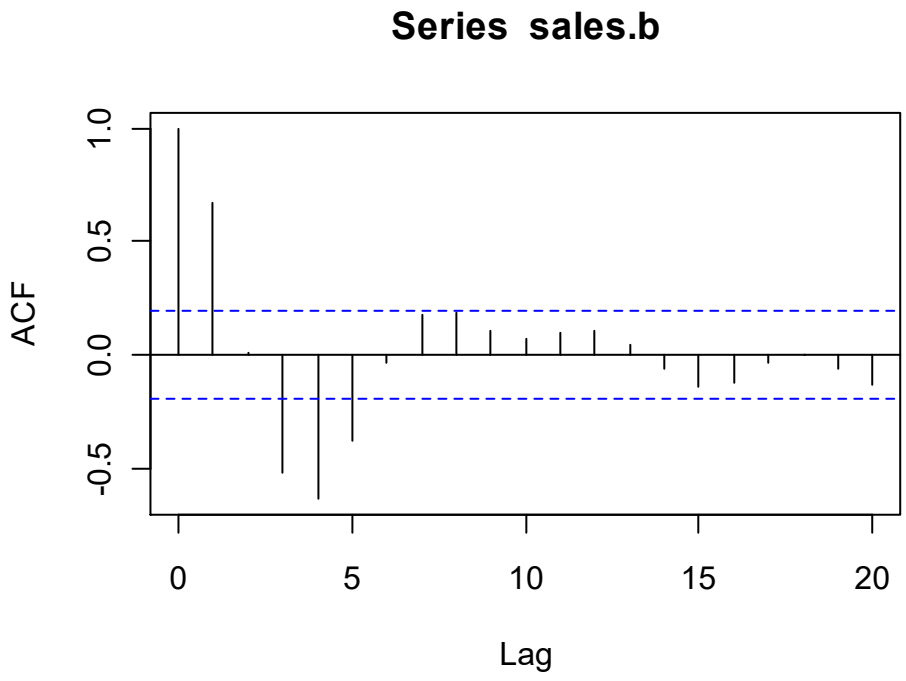
```
> sales.b = sales[1:103]
```

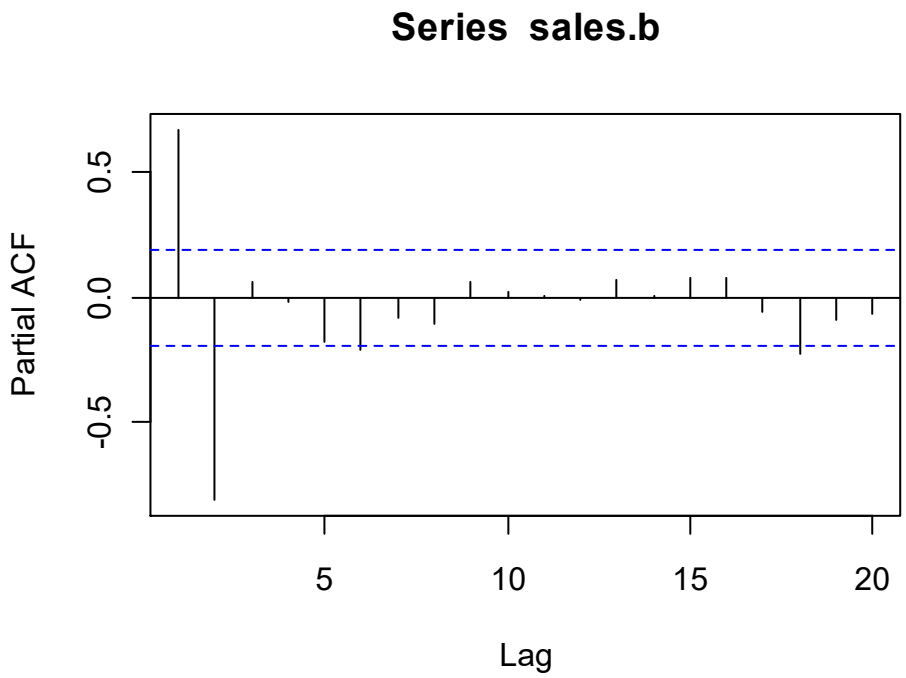
ثم نطبق نموذج $ARIMA(p,d,q)$ عليه

1- نرسم المتسلسلة



2- نرسم دوال ACF و $PACF$



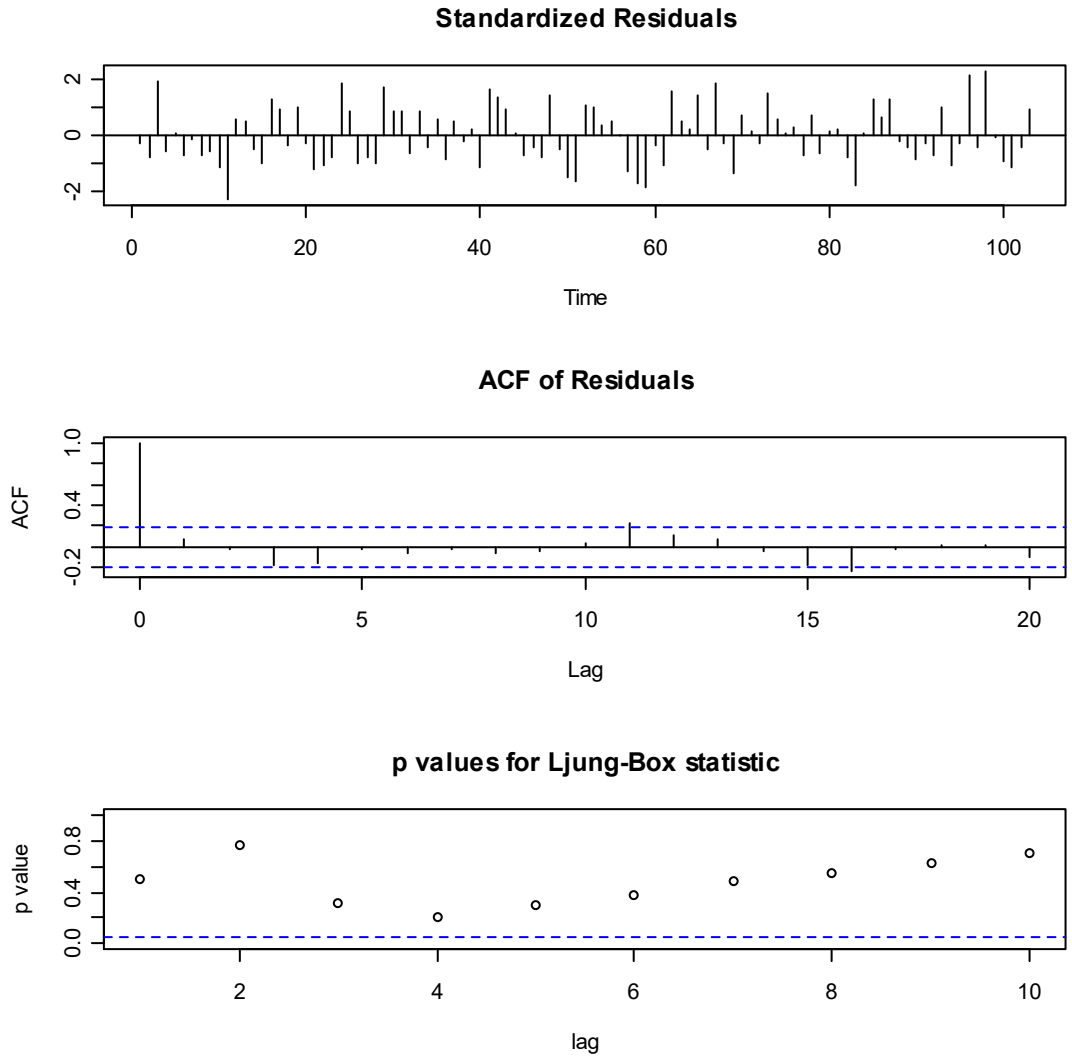


نلاحظ أن المتسلسلة مستقرة وقد تكون $ARIMA(2,0,0)$ نطبق هذا على المتسلسلة

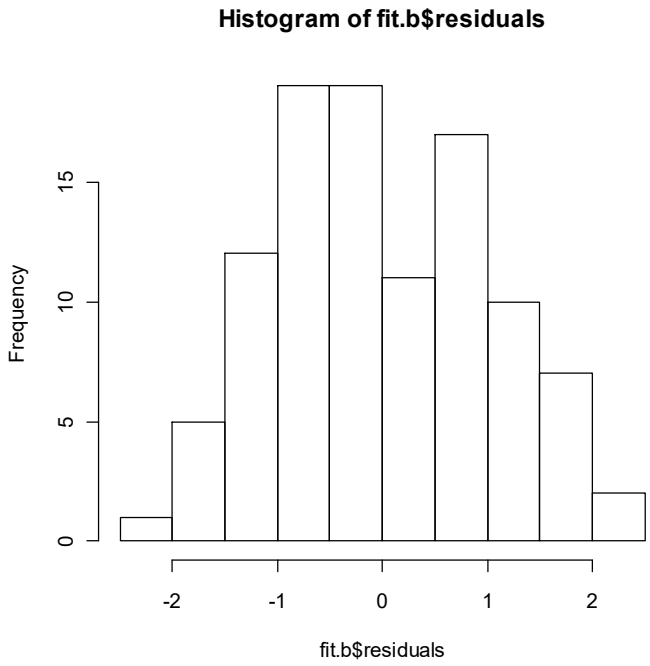
```
> fit.b = arima(sales.b,order=(c(2,0,0)))
> fit.b

Call:
arima(x = sales.b, order = (c(2, 0, 0)))

Coefficients:
         ar1      ar2  intercept
      1.2248  -0.8231   119.8952
s.e.  0.0549   0.0541     0.1730

sigma^2 estimated as 1.087:  log likelihood = -151.88,  aic
= 311.77
```

3- نتحقق من جودة التطبيق

```
> tsdiag(fit.b)
```

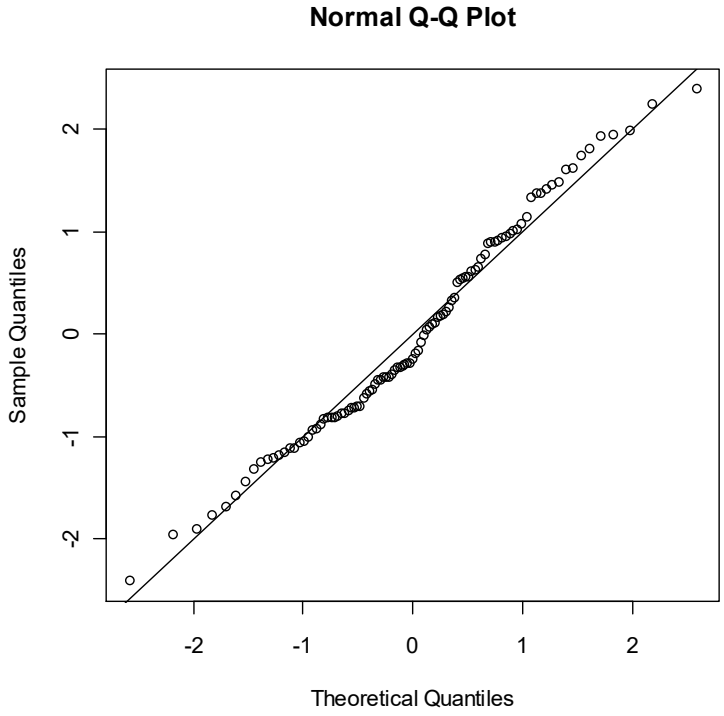



```
> hist(fit.b$residuals)
```



و

```
> qqnorm(fit.b$residuals)
> abline(0,1)
```



ونجري مجموعة الإختبارات المعتادة (أنظر الجزء الأول من الكتاب)

```
> white.test(fit.b$residuals)

        White Neural Network Test

data:  fit.b$residuals
X-squared = 0.1507, df = 2, p-value = 0.9274

> ks.test(fit.b$residuals,"pnorm")

        One-sample Kolmogorov-Smirnov test

data:  fit.b$residuals
D = 0.1042, p-value = 0.2130
alternative hypothesis: two-sided

> bds.test(residuals(fit.b))

         BDS Test

data:  residuals(fit.b)

Embedding dimension =  2 3

Epsilon for close points =  0.5239 1.0477 1.5716 2.0954

Standard Normal =
      [ 0.5239 ] [ 1.0477 ] [ 1.5716 ] [ 2.0954 ]
[ 2 ]     2.6926     0.9694     0.2411    -1.0522
[ 3 ]     3.3811     0.9964     0.0404    -0.5380
```

```
p-value =
      [ 0.5239 ] [ 1.0477 ] [ 1.5716 ] [ 2.0954 ]
[ 2 ]     0.0071     0.3323     0.8095     0.2927
[ 3 ]     0.0007     0.3191     0.9677     0.5906


> x <- factor(sign(residuals(fit.b)))
> runs.test(x)

        Runs Test

data:  x
Standard Normal = -0.4205, p-value = 0.6741
alternative hypothesis: two.sided

>
```

نلاحظ أن النموذج المطبق على المتسلسلة قبل التدخل جيد.

نكون الآن المتغير المؤشر I_t

```
> u = c(rep(0,103),rep(1,60))
```

نستخدم الأمر الذي سبق إستخدامه arima في R الذي له التركيب

```
arima(x, order = c(0, 0, 0),
      seasonal = list(order = c(0, 0, 0), period = NA),
      xreg = NULL, include.mean = TRUE,
      transform.pars = TRUE,
      fixed = NULL, init = NULL,
      method = c("CSS-ML", "ML", "CSS"),
```

```
      n.cond, optim.method = "BFGS",
      optim.control = list(), kappa = 1e6)
```

ولكن هنا نستخدم الخيار xreg لنقوم بتطبق معادلة الإنحدار

$$z_t = \frac{\omega(B)B^b}{\delta(B)} I_t + \frac{\theta_q(B)}{\phi_p(B)} a_t$$

```
> fit.inter = arima(sales2, order = c(2,0,0), xreg = u)
> fit.inter

Call:
arima(x = sales2, order = c(2, 0, 0), xreg = u)

Coefficients:
         ar1      ar2  intercept        u
      1.2051  -0.8168   119.9514  50.3751
s.e.  0.0450   0.0443     0.1608   0.2626

sigma^2 estimated as 1.001:  log likelihood = -232.78,  aic
= 475.57
>
```

النموذج المطبق هو

$$z_t = 119.95 + 50.38 I_t + \frac{1}{\left(1 - 1.21B + 0.817B^2\right)} a_t, \quad a_t \sim N(0, 1.001)$$

$$I_t = \begin{cases} 0, & t \le 103 \\ 1 & t > 103 \end{cases}$$

أو

$$z_t = 119.95 + 1.21 z_{t-1} - 0.817 z_{t-2} + 50.38 I_t + a_t, \quad a_t \sim N(0, 1.001)$$

$$I_t = \begin{cases} 0, & t \le 103 \\ 1 & t > 103 \end{cases}$$

أو

$$z_t = \begin{cases} 119.95 + 1.21 z_{t-1} - 0.817 z_{t-2} + a_t, & t \le 103 \\ 170.33 + 1.21 z_{t-1} - 0.817 z_{t-2} + a_t, & t > 103 \end{cases}, \quad a_t \sim N(0, 1.001)$$

الفصل الثاني

نماذج دالة التحويل أو ARMAX Models

Transfer Function Models (ARMAX Models)

نمذجة التدخل هي حالة خاصة من نماذج دالة التحويل حيث قمنا بإنحدار المتسلسلة على ذاتها وعلى متغير خارجي هو متغير المؤشر الذي يأخذ القيمة 0 قبل التدخل والقيمة 1 بعد التدخل.

في نمذجة دالة التحويل (ويسمى أحيانا نماذج ARMAX والإنحدار الحركي Dynamic Regression) يتم إنحدار المتسلسلة على ذاتها وعلى متسلسلة أو متسلسلات خارجية مرتبطة بها. وبالتالي يمكننا إستخدم المعلومات الإضافية في المتسلسلات الخارجية المرتبطة بالمتسلسلة ذات الإهتمام في التنبؤ عنها.

لكي نفهم طريقة دالة التحويل سوف نستعرض مثال يعطي إحساسا بما يجري في نمذجة دالة التحويل. هذا المثال مأخوذ من كتاب Makridakis وآخرين (أنظر المراجع).

مثال: لنفترض أننا على مدار20 يوم متتابعة نرسل مجموعة من الرسائل عن طريق مكتب بريد. النظام المتبع في مكتب البريد هو إرسال هذه الرسائل خلال الأيام القادمة. ليكن عدد الرسائل المرسلة في اليوم t هي x_t وعدد الرسائل الواصلة هو z_t الجدول التالي يعطي وصفا للبيانات.

أوزان دالة التحويل: $\nu_0 = 0, \nu_1 = .1, \nu_2 = .5, \nu_3 = .2, \nu_4 = .1, \nu_5 = .1$

العمود i يبين كيف أن الخطابات التي أرسلت في اليوم i تم تسليمها

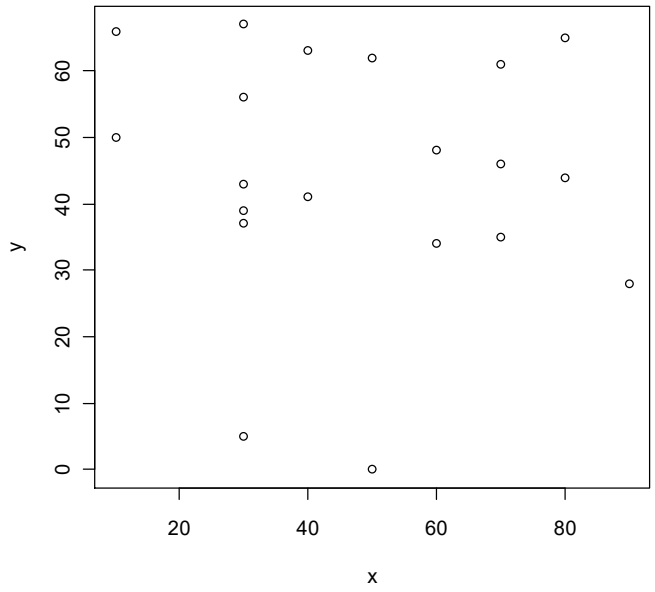
الخطابات المستقبلة

الخطابات المرسلة

Day	x_t	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	z_t
1	50																					0
2	30	5																				5
3	90	25	3																			28
4	60	10	15	9																		34
5	50	5	6	45	6																	62
6	70	5	3	18	30	5																61
7	30		3	9	12	25	7															56
8	40			9	6	10	35	3														63
9	80				6	5	14	15	4													44
10	70					5	7	6	20	8												46
11	80						7	3	8	40	7											65
12	10							3	4	16	35	8										66
13	30								4	8	14	40	1									67
14	30									8	7	16	5	3								39
15	70										7	8	2	15	3							35
16	30											8	1	6	15	7						37
17	60												1	3	6	35	3					48
18	40													3	3	14	15	6				41
19	10														3	7	6	30	4			50
20	30															7	3	1	2	1		43

لنركز الآن على العمودين "الخطابات المرسلة x_t " و "الخطابات المستقبلة z_t" . السؤال الآن هو: هل توجد علاقة بين z_t و x_t ؟ طبعا من الواضح انه يجب ان توجد علاقة ولكن لو اننا اجرينا التحليل المعتاد ورسمنا z ضد x وأجرينا تحليل الإنحدار البسيط فسنجد انه لاتوجد أي علاقة بينهما.

```
> plot(x,z)
> z.fit = lm(z~x)
```




```
> summary(z.fit)

Call:
lm(formula = z ~ x)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-44.350  -8.099   2.275  17.712  22.903

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  48.1046     9.8877   4.865 0.000124 ***
x            -0.0751     0.1859  -0.404 0.690952
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Residual standard error: 19.06 on 18 degrees of freedom
Multiple R-squared: 0.008987,   Adjusted R-squared: -
0.04607
F-statistic: 0.1632 on 1 and 18 DF,  p-value: 0.691

>
```

لاحظ أن الإنحدار غير معنوي p-value = 0.69 و $R^2 = 0.009$.

السبب في هذا أن قيم x_t موزعة حركيا (ديناميكيا) على فترات الأزمنة المستقبلية على حسب مايعرف بدالة التحويل. وهكذا فإن الـ 50 خطاب التي ارسلت في اليوم الأول تم تسليمها كالتالي:

0 (0%) تم تسليمها في نفس اليوم (يوم 1)

5 (10%) تم تسليما بعد يوم (يوم 2)

25 (50%) تم تسليمها بعد يومين (يوم 3)

10 (20%) تم تسليمها بعد ثلاثة أيام (يوم 4)

5 (10%) تم تسليمها بعد أربعة أيام (يوم 5)

5 (10%) تم تسليمها بعد خمسة أيام (يوم 6)

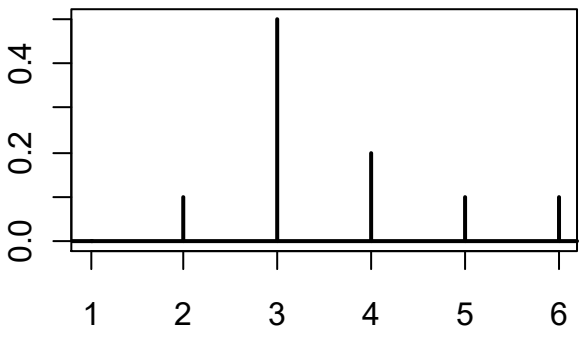
أنظر العمود المعنون "العمود i يبين كيف أن الخطابات التي أرسلت في اليوم i تم تسليمها" وهكذا لبقية الأيام في حال بقاء دالة التحويل كما هي فإن توزيع تسليم الخطابات سيكون هو نفسه أي بنفس النسب التي نسميها $\nu_0,\nu_1,\nu_2,\nu_3,\nu_4,\nu_5$ في الجدول السابق. العمود الأخير في الجدول هو عبارة عن مجموع الأعمدة ويمثل التأثير الكلي المجتمع لدالة التحويل العاملة على متسلسلة الإدخال x_t.

القيم ν_0 وحتى ν_5 تسمى أوزان الإستجابة اللحظية Impulse Response Weights أو أوزان دالة التحويل Transfer Function Weights ودالة التحويل نفسها تكتب كالتالي:

$$\begin{aligned} z_t &= \nu_0 x_t + \nu_1 x_{t-1} + \nu_2 x_{t-2} + \nu_3 x_{t-3} + \nu_4 x_{t-4} + \nu_5 x_{t-5} \\ &= \left(\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots + \nu_5 B^5\right) x_t \\ &= \nu(B) x_t \end{aligned}$$

يطلق على $\nu(B)$ دالة التحويل.

ولها شكل مثل التالي:



Impulse response for first table

أوزان الإستجابة اللحظية في هذا المثال جميعها موجبة ومجموعها واحد صحيح ولكن في الغالب يمكن أن تكون الأوزان موجبة أو سالبة ويمكن أن يكون مجموعها أكبر أو أصغر من الواحد الصحيح. في المصطلح الهندسي مجموع الأوزان يسمى " زيادة " Gain ويمكن توضيحه كالتالي:

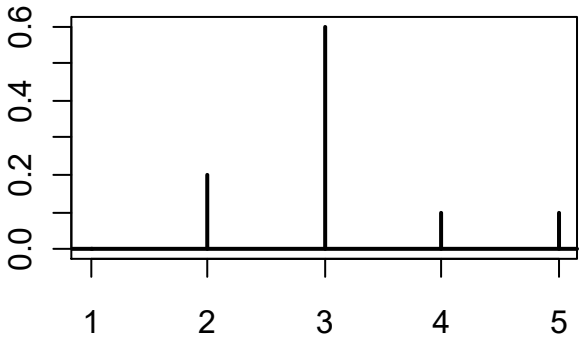
أوزان دالة التحويل: $\nu_0 = 0, \nu_1 = .2, \nu_2 = .6, \nu_3 = .1, \nu_4 = .1$											
دالة										دالة	
الإخراج										الإدخال	
z_t	10	9	8	7	6	5	4	3	2	1	الزمن
0										30	1
6									6	30	2
24								6	18	30	3
27							6	18	3	30	4
30						6	18	3	3	30	5
30					6	18	3	3		30	6
30				6	18	3	3			30	7
30			6	18	3	3				30	8
30		6	18	3	3					30	9
30	6	18	3	3						30	10

الجدول أعلاه يبين إذا ماكانت المتسلسلة المدخلة ثابتة (أي تاخذ قيمة واحدة ثابتة) ودالة التحويل لاتتغير فإن المتسلسلة المخرجة سوف تصل لحالة الإستقرار Steady State وبدون زيادة gain بعد وقت قصير. لاحظ أن مجموع الأوزان 1.

أوزان دالة التحويل: $v_0 = 0, v_1 = .2, v_2 = .6, v_3 = .2, v_4 = .1$											
دالة											دالة
الإخراج											الإدخال
z_t	10	9	8	7	6	5	4	3	2	1	x_t	الزمن
0											30	1
6										6	30	2
24									6	18	30	3
30								6	18	6	30	4
33							6	18	6	3	30	5
33						6	18	6	3		30	6
33					6	18	6	3			30	7
33				6	18	6	3				30	8
33			6	18	6	3					30	9
33		6	18	6	3						30	10

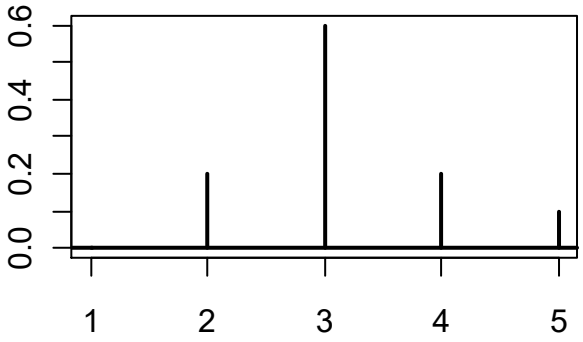
والجدول أعلاة يعطي إستجابة لدالة أوزان مجموعها أكبر من 1. المتسلسلة المدخلة ثابتة ودالة التحويل لاتتغير و المتسلسلة المخرجة سوف تصل لحالة الإستقرار Steady State و بزيادة gain بعد وقت قصير.

و لها الأشكال التالية:



Impulse response for second table

و



Impulse response for third table

الأشكال الأساسية لنموذج دالة التحويل:

نموذج دالة التحويل ثنائية المتغير Bivariate Transfer Function Model يمكن كتابتة في شكلين عامين.

الشكل الأول:

$$Y_t = \nu(B) X_t + N_t$$

حيث Y_t = متسلسة الإخراج output series

X_t = متسلسلة الإدخال input series

N_t = جميع التأثيرات الاخرى التي تؤثر على Y_t وتسمى متسلسلة الضجة noise series كما إن $Cov(X_t, N_t) = 0, \forall t$

$\nu(B) = (\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots + \nu_k B^k)$ حيث k درجة دالة التحويل.

وأحيانا $\nu(B) = (\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots) = \sum_{i=0}^{\infty} \nu_i B^i$ ولكي يكون نموذج دالة التحويل متوازن Stable يجب أن يتحقق الشرط $\sum_{i=0}^{\infty} |\nu_i| < \infty$.

الشكل الثاني:

$$z_t = \frac{\omega(B) B^b}{\delta(B)} x_t + n_t$$

أو

$$z_t = \frac{\omega_s(B) B^b}{\delta_r(B)} x_t + \frac{\theta_q(B)}{\phi_p(B)} a_t$$

حيث

$$\omega_s(B) = \omega_0 - \omega_1 B - \omega_2 B^2 - \cdots - \omega_s B^s$$

$$\delta_r(B) = 1 - \delta_1 B - \delta_2 B^2 - \cdots - \delta_r B^r$$

$$\theta_q(B) = 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q$$

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p B^p$$

و z_t هي Y_t بعد التحويل والتفريق

و x_t هي X_t بعد التحويل والتفريق

و a_t متسلسلة ضجة بيضاء ومستقلة عن x_t

و b و s و r و q و p أعداد صحيحة ثابتة موجبة

سوف نعيد كتابة الشكل الأخير كالتالي:

$$z_t = \frac{\omega_s(B)B^b}{\delta_r(B)} x_t + \frac{\theta_q(B)}{\phi_p(B)} a_t, \quad a_t \sim WN\left(0, \sigma_a^2\right)$$

العامل $\nu_k(B)$ استبدل بالعامل $\omega_s(B)B^b / \delta_r(B)$ وذلك لأن قيم k قد تمتد إلى مالانهاية بينما كل من s و r يأخذ قيم محدودة ولكي يكون النظام متوازن Stable يجب أن تكون أصفار $\delta_r(B) = 0$ خارج دائرة الوحدة.

لاحظ أن $B^b x_t = x_{t-b}$ متخلفة b وحدات زمنية عن z_t أي أنه يوجد b فترة زمنية قبل أن يبدأ تأثير x على z.

مجموعة المعالم (b,r,s) تسمى معالم دالة التحويل كما أن المجموعة (p,d,q) سميت في الجزء الأول من الكتاب معالم الإنحدار الذاتي التكاملي المتوسط المتحرك.

يقال أن نموذج دالة التحويل سببي Causal إذا كانت $\nu_i = 0, i < 0$ وهكذا في نموذج سببي لايتجاوب النظام مع متسلسلة الإدخال حتى يتم فعلا تطبيقها على النظام اي بمعنى ان المخرج الحالي يتأثر بمدخلات النظام فقط في كونها قيم حاضرة وماضية. النموذج السببي يسمى أيضا نموذج ممكن Realizable Model وفي الحقيقة جميع الأنظمة المادية هي سببية وسوف نتعامل فقط من النماذج السببية المتوازنة.

بعد إيجاد كل من $\omega_s(B)$ و $\delta_r(B)$ و b يمكن إيجاد أوزان دالة الإستجابة ν_i بمساوات معاملات B^i على طرفي العلاقة

$$\delta_r(B)\nu(B) = \omega_s(B)B^b$$

أو

$$\left(1-\delta_1 B-\cdots-\delta_r B^r\right)\left(\nu_0+\nu_1 B+\nu_2 B^2+\cdots\right)=\left(\omega_0-\omega_1 B-\cdots-\omega_s B^s\right)B^b$$

ومنها نجد

$$\begin{aligned}
&\nu_i = 0 && i < b\\
&\nu_i = \delta_1\nu_{i-1}+\delta_2\nu_{i-2}+\cdots+\delta_r\nu_{i-r}+\omega_0 && i = b\\
&\nu_i = \delta_1\nu_{i-1}+\delta_2\nu_{i-2}+\cdots+\delta_r\nu_{i-r}-\omega_{i-b} && i = b+1,b+2,...,b+s\\
&\nu_i = \delta_1\nu_{i-1}+\delta_2\nu_{i-2}+\cdots+\delta_r\nu_{i-r} && i > b+s
\end{aligned}$$

الأوزان $\nu_{b+s},\nu_{b+s-1},...,\nu_{b+s-r+1}$ (وعددها r) تستخدم كقيم أولية لحل المعادلة

$$\delta_r(B)\nu_i = 0, \quad i > b+s$$

وبالتالي فإن أوزان دالة التحويل تتكون من:

1) b من الأوزان الصفرية $\nu_0,\nu_1,...,\nu_{b-1}$.

2) $s-r+1$ من الأوزان $\nu_b,\nu_{b+1},...,\nu_{b+s-r}$ ليس لها نمط معين.

3) r من القيم الأولية $\nu_{b+s-r+1},\nu_{b+s-r+2},...,\nu_{b+s}$.

4) ν_i لقيم $i > b+s$ تتبع النمط المعطى بالمعادلة الفروقية

$$\delta_r(B)\nu_i = 0, \quad i > b+s$$

إذا في الملخص:

1) تحدد قيمة b من أن $\nu_i = 0$ لقيم $i < b$ و $\nu_b \neq 0$.

2) تحدد قيمة r من نمط أوزان دالة الإستجابة على نفس طريقة التعرف على درجة p في نموذج ARIMA من خلال أنماط دالة الترابط الذاتي.

3) لقيمة محددة b و $r = 0$ فإن قيمة s يمكن إيجادها من معرفة أن $\nu_i = 0$ لقيم $i > b + s$ وإذا كانت $r \neq 0$ فإن قيمة s يمكن إيجادها من فحص أين يبدأ نمط أوزان دالة الإستجابة في الإنخماد.

الجدول التالي يعطي دوال التحول والإستجابة لقيم $r = 0,1,2$ و $s = 0,1,2$ و b

(r,s,b)	TR	Impulse Response ν_i
$(0,0,b)$	$z_t = \omega_0 B^b x_t$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ 0, & j > b \end{cases}$
$(0,1,b)$	$z_t = (\omega_0 - \omega_1 B) B^b x_t$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ -\omega_1, & j = b+1 \\ 0, & j > b+1 \end{cases}$
$(0,2,b)$	$z_t = (\omega_0 - \omega_1 B - \omega_2 B^2) B^b x$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ -\omega_1, & j = b+1 \\ -\omega_2, & j = b+2 \\ 0, & j > b+2 \end{cases}$

$(1,0,b)$	$(1-\delta_1 B)z_t = \omega_0 B^b x_t$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ \delta_1 \nu_{j-1}, & j > b \end{cases}$
$(1,1,b)$	$(1-\delta_1 B)z_t = (\omega_0 - \omega_1 B)B^b x_t$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ \delta_1\omega_0 - \omega_1, & j = b+1 \\ \delta_1 \nu_{j-1}, & j > b+1 \end{cases}$
$(1,2,b)$	$(1-\delta_1 B)z_t = (\omega_0 - \omega_1 B - \omega_2 B^2)B^b x_t$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ \delta_1\omega_0 - \omega_1, & j = b+ \\ \delta_1^2\omega_0 - \delta_1\omega_1 - \omega_2, & j = b+ \\ \delta_1 \nu_{j-1}, & j > b+ \end{cases}$
$(2,0,b)$	$(1-\delta_1 B - \delta_2 B^2)z_t = \omega_0 B^b$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ \delta_1 \nu_{j-1} + \delta_2 \nu_{j-2}, & j > b \end{cases}$
$(2,1,b)$	$(1-\delta_1 B - \delta_2 B^2)z_t = (\omega_0 - \omega_1 B)B^b x_t$	$\nu_i = \begin{cases} 0, & j < b \\ \omega_0, & j = b \\ \delta_1\omega_0 - \omega_1, & j = b+ \\ \delta_1 \nu_{j-1} + \delta_2 \nu_{j-2}, & j > b+ \end{cases}$

$(2,2,b)$	$(1-\delta_1 B-\delta_2 B^2)z_t =$ $(\omega_0-\omega_1 B-\omega_2 B^2)B^b x_t$	$\nu_i=\begin{cases}0, & j<b\\ \omega_0, & j=b\\ \delta_1\omega_0-\omega_1, & j=b+\\ (\delta_1^2+\delta_2)\omega_0 & \\ -\delta_1\omega_1-\omega_2, & j=b+\\ \delta_1\nu_{j-1}+\delta_2\nu_{j-2}, & j>b+\end{cases}$

ملاحظة: الزيادة Gain في حالة وصول النظام Steady State تعطى بالعلاقة

$$Gain=\frac{\omega_0-\omega_1-\cdots-\omega_s}{1-\delta_1-\delta_2-\cdots-\delta_r}$$

<u>شكل دالة التنبؤ لدالة التحويل:</u>

تكتب على الشكل:

$$\delta_r(B)\phi_p(B)z_t=\phi_p(B)\omega_s(B)B^b x_t+$$
$$\delta_r(B)\theta_q(B)a_t,\quad a_t\sim WN(0,\sigma_a^2)$$

فمثلا لقيم $(b,r,s)=(b,1,1)$ و $(p,d,q)=(1,0,1)$ نجد

$$z_t=\frac{(\omega_0-\omega_1 B)}{(1-\delta_1 B)}x_{t-b}+\frac{(1-\theta_1 B)}{(1-\phi_1 B)}a_t,\quad a_t\sim WN(0,\sigma_a^2)$$

$$(1-\delta_1 B)(1-\phi_1 B)z_t=(1-\phi_1 B)(\omega_0-\omega_1 B)x_{t-b}$$
$$+(1-\delta_1 B)(1-\theta_1 B)a_t,\quad a_t\sim WN(0,\sigma_a^2)$$

$$z_t = (\delta_1 + \phi_1) z_{t-1} - \delta_1 \phi_1 z_{t-2} + \omega_0 x_{t-b} - (\omega_0 \phi_1 + \omega_1) x_{t-b-1} + \omega_1 \phi_1 x_{t-b-2} + a_t - (\delta_1 + \theta_1) a_{t-1} + \delta_1 \theta_1 a_{t-2}, \quad a_t \sim WN(0, \sigma_a^2)$$

والذي يستخدم للتنبؤ بعد تقدير المعالم.

دالة الترابط البيني Cross Correlation Function (CCF) :

تعتبر دالة الترابط البيني مقياس مفيد لقياس قوة وإتجاه العلاقة بين متغيرين عشوائيين.

إذا كان كل من x_t و z_t متسلسلتين زمنية (عمليتين عشوائيتين) مستقرة فإن دالة التغاير البيني Cross Covariance بينهما $\gamma_{xz}(t,s) = Cov(x_t, z_s)$ تعتمد على فرق الزمن $s-t$ فقط وتكتب

$$\gamma_{xz}(k) = E\left[(x_t - \mu_x)\right]\left[(z_{t+k} - \mu_z)\right], \quad k = 0, \pm 1, \pm 2, \ldots$$

ودالة الترابط البيني

$$\rho_{xz}(k) = \frac{\gamma_{xz}(k)}{\sigma_x \sigma_z}, \quad k = 0, \pm 1, \pm 2, \ldots$$

حيث σ_x و σ_z الإنحرافات المعيارية لكل من x_t و z_t.

لاحظ أن $\rho_{xx}(k) = \rho_k$ دالة الترابط الذاتي للمتسلسلة x_t.

لاحظ ايضا ان $\rho_k = \rho_{-k}$ بينما $\rho_{xz}(k) \neq \rho_{xz}(-k)$.

تقدر دالة الترابط البيني لعينة حجمها n للمتسلسلتين $(x_i, z_i), \quad i = 1, \ldots, n$ بدالة الترابط البيني العينية Sample Cross Correlation Function (SCCF) كالتالي:

$$r_{xz}(k)=\frac{c_{xz}(k)}{s_x s_z},\quad k=0,\pm1,\pm2,...$$

حيث

$$c_{xz}(k)=\begin{cases} n^{-1}\sum_{t=1}^{n-k}(x_t-\bar{x})(z_{t+k}-\bar{z}) & k\geq 0 \\ n^{-1}\sum_{t=1-k}^{n}(x_t-\bar{x})(z_{t+k}-\bar{z}) & k<0 \end{cases}$$

و

$$s_x^2=c_{xx}(0)=n^{-1}\sum_{t=1}^{n}(x_t-\bar{x})^2,\quad s_z^2=c_{zz}(0)=n^{-1}\sum_{t=1}^{n}(z_t-\bar{z})^2$$

$c_{xz}(k)$ قيمة مقدرة وتعتمد على عينة عشوائية ولهذا لها التغاير

$$\begin{aligned} Cov[c_{xz}(k),c_{xz}(k+j)] \simeq (n-k)^{-1}\{&\sum_{i=-\infty}^{\infty}\rho_{xx}(i)\rho_{zz}(i+j)+\rho_{xz}(i+k+j)\rho_{xz}(k-i) \\ &+\rho_{xz}(k)\rho_{xz}(k+j)\left[\rho_{xz}^2(i)+\frac{1}{2}\rho_{xx}^2(i)+\frac{1}{2}\rho_{zz}^2(i)\right] \\ &-\rho_{xz}(k)[\rho_{xx}(i)\rho_{xz}(i+k+j)+\rho_{xz}(-i)\rho_{zz}(i+k+j)] \\ &-\rho_{xz}(k+j)[\rho_{xx}(i)\rho_{xz}(i+k)+\rho_{xz}(-i)\rho_{zz}(i+k)]\} \end{aligned}$$

وتباين

$$\begin{aligned} Var[c_{xz}(k)] \simeq (n-k)^{-1}\{&\sum_{i=-\infty}^{\infty}\rho_{xx}(i)\rho_{zz}(i)+\rho_{xz}(k+i)\rho_{xz}(k-i) \\ &+\rho_{xz}^2(k)\left[\rho_{xz}^2(i)+\frac{1}{2}\rho_{xx}^2(i)+\frac{1}{2}\rho_{zz}^2(i)\right] \\ &-2\rho_{xz}(k)[\rho_{xx}(i)\rho_{xz}(i+k)+\rho_{xz}(-i)\rho_{zz}(i+k)]\} \end{aligned}$$

وتحت الفرضية أن المتسلسلتين x_t و z_t غير مترابطة و أن المتسلسلة x_t ضجة بيضاء تصبح العلاقتين السابقة

$$Cov[c_{xz}(k), c_{xz}(k+j)] \simeq (n-k)^{-1} \rho_{zz}(j)$$

و

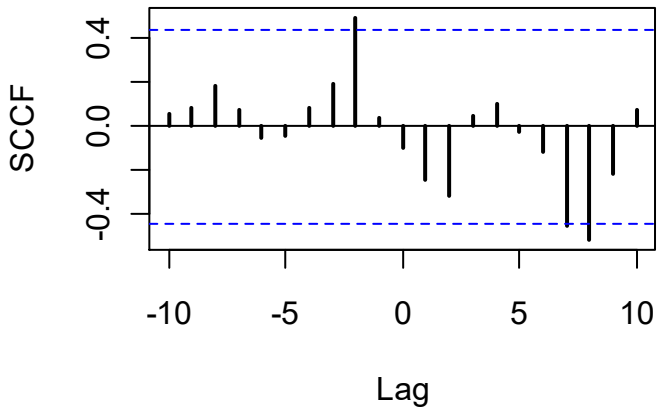
$$Var[c_{xz}(k)] \simeq (n-k)^{-1}$$

لإختبار الفرضية أن المتسلسلتين x_t و z_t غير مترابطة بينيا نقارن $r_{xz}(k)$ بأخطائها المعيارية التقريبية $1/\sqrt{(n-k)}$. تطبيقيا المتسلسلتين x_t و z_t تكونا ليست متسلسلات ضجة بيضاء لهذا يجب أن يتم تبييضهما Prewhiten كما يجب تصفية Filter المتسلسلة z_t كما سيأتي لاحقا.

<u>مثال:</u>

دالة الترابط البيني العينية Sample Cross Correlation Function (SCCF) للقيم في الجدول الأول

```
> ccf(x,z,zlab="SCCF",lwd=2)
```



العلاقة بين دالة التحويل ودالة الترابط البيني:

سبق أن عرفنا دالة التحويل بالعلاقة

$$z_t = \nu(B)x_t + n_t$$

للزمن $t+k$ نكتب

$$z_t = \nu_0 x_t + \nu_1 x_{t-1} + \nu_2 x_{t-2} + \cdots + n_t$$

سوف نفترض أن $\mu_x = 0$ و $\mu_z = 0$ (وذلك بالعمل على المتسلسلات بعد إزالة المتوسطات) بالضرب في x_{t+k} وأخذ التوقع نجد

$$\gamma_{xz}(k) = \nu_0\gamma_{xx}(k) + \nu_1\gamma_{xx}(k-1) + \nu_2\gamma_{xx}(k-2) + \cdots$$

حيث $\gamma_{xn}(k)=0$ لجميع قيم k ومن ثم

$$\rho_{xz}(k)=\frac{\sigma_x}{\sigma_z}\left[\nu_0\rho_x(k)+\nu_1\rho_x(k-1)+\nu_2\rho_x(k-2)+\cdots\right]$$

هذه العلاقة بين دالة التحويل ودالة الترابط البيني تتداخل فيها قيم الترابط الذاتي للمتسلسلة x_t مع أوزان دالة التحويل ويصبح من الصعب جدا إستخدامها لإيجاد قيم دالة التحويل ν_i حتى لو كانت $r=0$ في العلاقة $z_t=\left[\omega_s(B)B^b/\delta_r(B)\right]x_t+n_t$ أي للنموذج $z_t=\omega_s(B)B^b x_t+n_t$ وبهذا تكون $\nu(B)=\omega_s(B)B^b$ وبهذا تكون دالة التحويل تحتوي على عدد محدود من الأوزان بالرغم من ذلك يصعب حل العلاقة. للإتفاف حول هذه المشكلة لو كانت متسلسلة الإدخال متسلسلة ضجة بيضاء أي لوكان $\rho_x(k)=0,\quad k\neq 0$ فإن

$$\begin{aligned}\rho_{xz}(k)&=\frac{\sigma_x}{\sigma_z}\left[\nu_0\rho_x(k)+\nu_1\rho_x(k-1)+\nu_2\rho_x(k-2)+\cdots\right]\\&=\frac{\sigma_x}{\sigma_z}\nu_k\end{aligned}$$

ومنها نجد

$$\nu_k=\frac{\sigma_z}{\sigma_x}\rho_{xz}(k)$$

أي ان دالة الإستجابة اللحظية تتناسب طرديا مع دالة الترابط البيني.

ملاحظات هامة:

1- دالة الترابط البيني تكون معرفة وموجودة فقط إذا كانتا x_t و z_t تشكل معا عمليتين ثنائية مستقرة Jointly Bivariate Stationary Processes ولكي يتحقق الإستقرار قد نحتاج إلى التفريق Differencing وقد نحتاج أيضا عملية تثبيت تباين (أنظر الجزء الأول). وهكذا سنفترض دائما أن المتسلسلتين x_t و z_t مستقرتين معا Jointly Stationary.

2- في النموذج العام لدالة التحويل

$$z_t = \frac{\omega_s(B)B^b}{\delta_r(B)}x_t + n_t$$

سوف نفترض أن متسلسلة الإدخال x_t تتبع عملية ARMA

$$\phi_p^{(x)}(B)x_t = \theta_q^{(x)}(B)\alpha_t$$

حيث α_t (لاحظ أن α هي الحرف Alpha المقابل للحرف a) متسلسلة ضجة بيضاء و $\alpha_t = \left[\phi_p^{(x)}(B)/\theta_q^{(x)}(B)\right]x_t$ وتسمى أحيانا متسلسلة الإدخال المبيضةPrewhitened Input Series وبتطبيق نفس التبييض على متسلسلة الإخراج z_t نحصل على متسلسلة مصفاة Filtered output Series

$\beta_t = \left[\phi_p^{(x)}(B)/\theta_q^{(x)}(B)\right]z_t$ وبوضع $\varepsilon_t = \theta_x^{-1}(B)\phi_x(B)n_t$ يصبح نموذج دالة التحويل $\beta_t = \nu(B)\alpha_t + \varepsilon_t$ ويمكن إيجاد أوزان الإستجابة اللحظية ν_i من

$$\nu_k = \frac{\sigma_\beta}{\sigma_\alpha}\rho_{\alpha\beta}(k), \quad k = 0,1,2,...$$

وتقدر بالعلاقة

$$\hat{\nu}_k = \frac{s_\beta}{s_\alpha} r_{\alpha\beta}(k), \quad k = 0,1,2,...$$

وهذا يعطي طريقة للتعرف على وتقدير نموذج دالة التحويل.

التعرف على نماذج دالة التحويل Identification of TF Models:

للتعرف على نموذج دالة تحويل $\nu(B)$ لمتسلسلة إدخال x_t و متسلسلة إخراج z_t نتبع الخطوات التالية:

1- تبيض Prewhightenning متسلسلة الإدخال

$$\phi_p^{(x)}(B) x_t = \theta_q^{(x)}(B) \alpha_t$$

لنحصل على

$$\alpha_t = \frac{\phi_p^{(x)}(B)}{\theta_q^{(x)}(B)} x_t$$

حيث α_t متسلسلة ضجة بيضاء بمتوسط صفري وتباين σ_α^2. كما يلاحظ أن متسلسلتي الإدخال والإخراج قدتم تحويلهم بالشكل المناسب أي إزالة المتوسط وأخذ التفريق المناسب وتثبيت التباين إذا أحتاج الأمر لذلك.

2- أحسب متسلسلة الإخراج المصفاة Filtered output Series. أي متسلسلة الإخراج المحولة z_t مستخدمين نموذج التبييض أعلاه لتوليد المتسلسلة

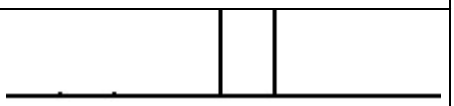
$$\beta_t = \frac{\phi_p^{(x)}(B)}{\theta_q^{(x)}(B)} z_t$$

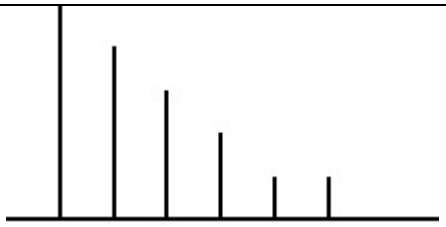
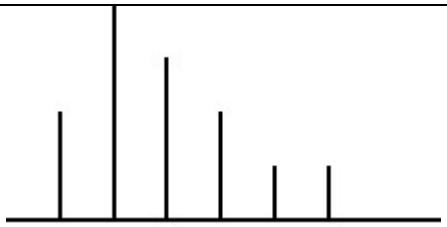
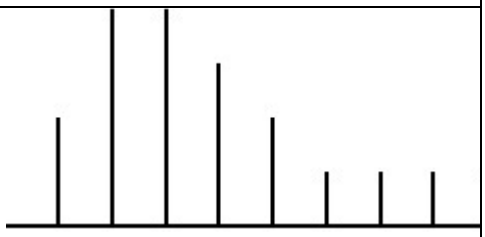
3- أحسب دالة الترابط البيني العيني SCCF $r_{\alpha\beta}(k)$ بين α_t و β_t لكي نقدر ν_k

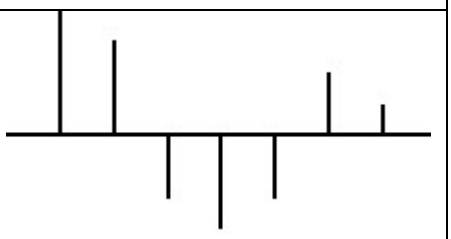
$$\hat{\nu}_k = \frac{\hat{\sigma}_\beta}{\hat{\sigma}_\alpha} r_{\alpha\beta}(k)$$

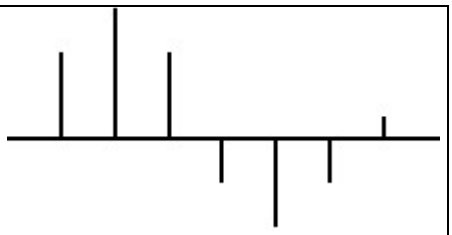
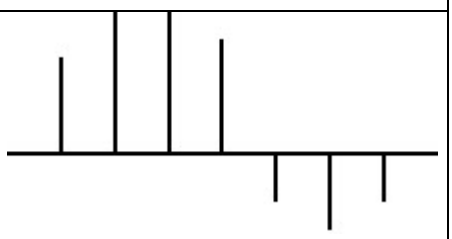
تختبر معنوية $r_{\alpha\beta}(k)$ أو مايكافئها $\hat{\nu}_k$ بمقارنتها مع خطأها المعياري $1/\sqrt{(n-k)}$.

4- تعرف على b و $\left(1-\delta_1 B-\delta_2 B^2-\cdots-\delta_r B^r\right)$ و $\delta_r(B)=$ و $\left(\omega_0-\omega_1 B-\omega_2 B^2-\cdots-\omega_s B^s\right)$ $\omega_s(B)=$ بتطبيق أنماط $\hat{\nu}_k$ بالأنماط النظرية كما في الجداول التالية:

TF for $r=0$		
(b,r,s)	TF	Impulse Weights
(2,0,0)	$\nu(B)x_t=\omega_0 x_{t-2}$	
(2,0,1)	$\nu(B)x_t=(\omega_0-\omega_1 B)x_{t-2}$	
(2,0,1)	$\nu(B)x_t=(\omega_0-\omega_1 B-\omega_2 B^2)x_{t-2}$	

TF for $r=1$		
(b,r,s)	TF	Impulse Weights
(2,1,0)	$\nu(B)x_t = \dfrac{\omega_0}{(1-\delta_1 B)}x_{t-2}$	
(2,1,1)	$\nu(B)x_t = \dfrac{(\omega_0-\omega_1 B)}{(1-\delta_1 B)}x_{t-2}$	
(2,1,2)	$\nu(B)x_t = \dfrac{(\omega_0-\omega_1 B-\omega_2 B^2)}{(1-\delta_1 B)}x_{t-}$	

TF for $r=2$		
(b,r,s)	TF	Impulse Weights
(2,2,0)	$\nu(B)x_t = \dfrac{\omega_0}{(1-\delta_1 B-\delta_2 B^2)}x_{t-2}$	

(2,2,1)	$\nu(B)x_t = \dfrac{(\omega_0 - \omega_1 B)}{(1-\delta_1 B - \delta_2 B^2)} x_{t-2}$	
(2,2,2)	$\nu(B)x_t = \dfrac{(\omega_0 - \omega_1 B - \omega_2 B^2)}{(1-\delta_1 B - \delta_2 B^2)} x_{t-2}$	

بعد إختيار قيم للمعالم b و r و s نوجد مقدرات أولية $\hat{\omega}_i$ و $\hat{\delta}_i$ من علاقاتهم مع ν_i أي من

$$\left(1-\delta_1 B-\cdots-\delta_r B^r\right)\left(\nu_0+\nu_1 B+\nu_2 B^2+\cdots\right)=\left(\omega_0-\omega_1 B-\cdots-\omega_s B^s\right)B^b$$

وهكذا يكون لدينا مقدر أولي لدالة التحويل $\nu(B)$ كالتالي:

$$\hat{\nu}(B)=\frac{\hat{\omega}_s(B)B^b}{\hat{\delta}_r(B)}$$

التعرف على نموذج الضجة:

بعد الحصول على دالة تحويل أولية نستطيع حساب مقدر لمتسلسلة الضجة

$$\hat{n}_t = z_t - \hat{\nu}(B)x_t$$

$$= z_t - \frac{\hat{\omega}_s(B)B^b}{\hat{\delta}_r(B)}x_t$$

ونستطيع تحديد النموذج المناسب لمتسلسلة الضجة بفحص دالتي الترابط والترابط الجزئي الذاتي العيني لها لنحصل على

$$\phi_p(B)n_t = \theta_q(B)a_t$$

<u>ملاحظات:</u>

للنموذج

$$z_t = \frac{\omega_s(B)B^b}{\delta_r(B)}x_t + \frac{\theta_q(B)}{\phi_p(B)}a_t, \quad a_t \sim WN\left(0,\sigma_a^2\right)$$

1- لبناء النموذج أفترضنا أن المتسلسلات الزمنية z_t و x_t و n_t جميعها مستقرة ولهذا إذا كانت أي منهم غير مستقر نجري عليها التحويل المناسب لجعلها مستقرة (أنظر الجزء الأول).

2- في عملية تحديد دالة التحويل $\nu(B)$ قمنا بتبييض دالة الإدخال. النموذج المبيض طبق على متسلسلة الإخراج لتصفيتها وليس بالضرورة لتبييضها وهذه الطريقة المعتادة والطبيعية لبناء نموذج سببي. ولبناء نموذج غير سببي والذي يحوي تغذية خلفية Feedback كل من x_t و z_t يؤثران على بعضهما البعض فإن كلا من متسلسلتي الإدخال والإخراج يجب أن يتم تبييضهما قبل فحص SCCF.

تقدير معالم نماذج دالة التحويل:

بعد تحديد نموذج أولي لدالة التحويل

$$z_t = \frac{\omega_s(B)B^b}{\delta_r(B)}x_t + \frac{\theta_q(B)}{\phi_p(B)}a_t, \quad a_t \sim WN\left(0,\sigma_a^2\right)$$

نحتاج لتقدير المعالم $\boldsymbol{\delta} = (\delta_1,...,\delta_r)'$ و $\boldsymbol{\omega} = (\omega_0,\omega_1,...,\omega_s)'$ و

$\boldsymbol{\varphi} = (\phi_1,...,\phi_p)'$ و $\boldsymbol{\theta} = (\theta_1,...,\theta_q)'$ و σ_a^2. نعيد كتابة النموذج السابق على الشكل

$$\delta_r(B)\phi_p(B)z_t = \phi_p(B)\omega_s(B)B^b x_t + \delta_r(B)\theta_q(B)a_t$$

أو على الشكل المختصر

$$c(B)z_t = d(B)x_{t-b} + e(B)a_t$$

حيث

$$\begin{aligned} c(B) = \delta_r(B)\phi_p(B) &= \left(1-\delta_1 B-\cdots-\delta_r B^r\right)\left(1-\phi_1 B-\cdots-\phi_p B^p\right) \\ &= \left(1-c_1 B-c_2 B^2-\cdots-c_{p+r}B^{p+r}\right) \end{aligned}$$

و

$$\begin{aligned} d(B) = \phi_p(B)\omega_r(B) &= \left(1-\phi_1 B-\cdots-\phi_p B^p\right)\left(\omega_0-\omega_1 B-\cdots-\omega_s B^s\right) \\ &= \left(d_0-d_1 B-d_2 B^2-\cdots-d_{p+s}B^{p+s}\right) \end{aligned}$$

و

$$e(B)=\delta_r(B)\theta_q(B)=\left(1-\delta_1 B-\cdots-\delta_r B^r\right)\left(1-\theta_1 B-\cdots-\theta_q B^q\right)$$
$$=\left(1-e_1 B-e_2 B^2-\cdots-e_{r+q}B^{r+q}\right)$$

وهكذا

$$a_t=\frac{c(B)}{e(B)}z_t-\frac{d(B)}{e(B)}x_{t-b}$$

وبفك العلاقة السابقة نجد

$$a_t=z_t-c_1 z_{t-1}-\cdots-c_{p+r}z_{t-p-r}-d_0 x_{t-b}+d_1 x_{t-b-1}$$
$$+\cdots+d_{p+s}x_{t-b-p-s}+e_1 a_{t-1}+\cdots+e_{r+q}a_{t-r-q}$$

حيث c_i و d_j و e_k دوال لكل من δ_i و ω_j و ϕ_k و θ_l. وتحت الفرضية أن متسلسلة الضجة $a_t\sim N\left(0,\sigma_a^2\right)$ نكتب دالة الأرجحية الشرطية

$$L\left(\delta,\omega,\phi,\theta,\sigma_a^2\middle|b,\mathbf{x},\mathbf{z},\mathbf{x}_0,\mathbf{z}_0,\mathbf{a}_0\right)=\left(2\pi\sigma_a^2\right)^{-n/2}\exp\left(-\frac{1}{2\sigma_a^2}\sum_{t=1}^{n}a_t^2\right)$$

حيث $\mathbf{x}_0,\mathbf{z}_0,\mathbf{a}_0$ قيم أولية مناسبة لحساب a_t نوجدها من العلاقة

$$a_t=z_t-c_1 z_{t-1}-\cdots-c_{p+r}z_{t-p-r}-d_0 x_{t-b}+d_1 x_{t-b-1}$$
$$+\cdots+d_{p+s}x_{t-b-p-s}+e_1 a_{t-1}+\cdots+e_{r+q}a_{t-r-q}$$

أيضا يمكن حساب المقدرات بطريقة المربعات الدنيا الشرطية غير الخطية وذلك بوضع قيم a_t الأولية غير المعروفة بتوقعاتها الصفرية وتصغير العلاقة

$$S(\delta,\omega,\phi,\theta|b)=\sum_{t=t_0}^{n}a_t^2$$

حيث $t_0=\max(p+r+1,b+p+s+1)$.

<u>ملاحظة:</u> الطرق السابقة في التقدير تفترض معرفة قيمة b لقيم معطاه r و s و p و q .
في حالة عدم معرفة b يمكن تقديرها بإيجاد أفضل الحلول للعلاقة $S(\delta,\omega,\phi,\theta|b)$ على مجال لقيم b الممكنة.

<u>الإختبارات التفحصية لنماذج دالة التحويل:</u>

لقد أفترضنا في بناء نموذج دالة التحويل أن a_t متسلسلة ضجة بيضاء وهي مستقلة عن متسلسلة الإدخال x_t وبالتالي هي مستقلة عن المتسلسلة المبيضة α_t.

ولهذا في الإختبارات التفحصية نفحص البواقي $\hat{a}_t$ من نموذج الضجة وكذلك نفحص البواقي α_t من نموذج الإدخال المبيض لكي نرى فيما إذا كانت الفرضيات صحيحة.

1- **<u>فحص الترابط البيني:</u>**

لكي نفحص فيما إذا كانت متسلسلة الضجة البيضاء a_t ومتسلسلة الإدخال x_t مستقلتين فللنموذج المناسب دالة الترابط البيني العيني SCCF $c_{\alpha\hat{a}}(k)$ بين a_t و α_t يجب الا تظهر أي أنماط وتقع وتقع بين $2(n-k)^{-1/2}$.

كما يمكن إستخدام إختبار البورتمانتو Portmanteau Test التالي

$$Q_0 = m(m+2)\sum_{j=0}^{K}(m-j)^{-1}c_{\alpha\hat{a}}^2$$

والذي يتبع تقريبا توزيع χ^2 بدرجات حرية $(K+1)-M$ حيث $m = n - t_0 + 1$ عدد البواقي المحسوبة $\hat{a}_t$ و M عدد المعالم المقدرة في دالة التحويل فقط حيث أثبت أن عدد درجات الحرية للإحصائة Q_0 لاتعتمد على عدد المعالم المقدرة لمتسلسلة الضجة.

2- **فحص الترابط الذاتي:**

لفحص مناسبة نموذج الضجة يجب فحص كل من SACF و SPACF للبواقي $\hat{a}_t$ وإجراء جميع الإختبارات المطلوبة (أنظر الجزء الأول).

ملاحظات:

1- قد يكون النموذج المقترح لدالة التحويل $\nu(B)$ غيرصحيح وبغض النظر عن كون نموذج الضجة صحيح او لا فإن قيم $\rho_{\alpha a}(k) \neq 0$ و $\rho_a(k) \neq 0$ لبعض قيم k . وهنا يجب إعادة تحديد شكل دالة التحويل و تعديل نموذج الضجة.

2- قد يكون النموذج المقترح لدالة التحويل $\nu(B)$ صحيح ونموذج الضجة غير مناسب ففي هذه الحالة $\rho_{\alpha a}(k) = 0$ لجميع قيم k ولكن $\rho_a(k) \neq 0$ لبعض قيم k . وهنا نستخدم الأنماط في $r_{\hat{a}}(k)$ لتعديل نموذج الضجة.

مثال على بناء نموذج دالة تحويل:

سوف نشرح بالتفصيل كيفية بناء نموذج دالة تحويل لمثال معطى في كتاب ماكريداكس Makridakis وآخرين (أنظر المراجع).

مسلسلة الإدخال x_t تعطي الإنفاق الشهري على الإعلانات بآلاف الدولارات ومتسلسلة الإخراج z_t تعطي المبيعات الكلية شهريا بآلاف القطع. سوف نتحرى مدى تأثر المبيعات بالإنفاق الإعلاني.

x_t

116.44 119.58 125.74 124.55 122.35 120.44 123.24 127.99
121.19 118.00 121.81 126.54 129.85 122.65 121.64 127.24
132.35 130.86 122.90 117.15 109.47 114.34 123.72 130.33
133.17 134.25 129.75 130.05 133.42 135.16 130.89 123.48
118.46 122.11 128.75 127.09 114.55 113.26 111.51 111.73
114.08 114.32 115.03 124.28 132.69 134.64 133.28 128.00
129.97 128.35 123.90 122.45 122.85 129.28 129.77 127.78
134.29 140.61 133.64 135.45 130.93 118.65 120.34 120.35
117.09 117.56 121.69 128.19 134.79 128.93 121.63 125.43
126.80 131.56 126.43 116.19 112.72 109.53 110.38 107.31
93.59 89.80 88.70 86.64 89.26 96.51 107.35 110.35
102.66 97.56 98.06 103.93 115.66 112.91 116.89 116.84
109.55 110.63 111.32 117.09

z_t

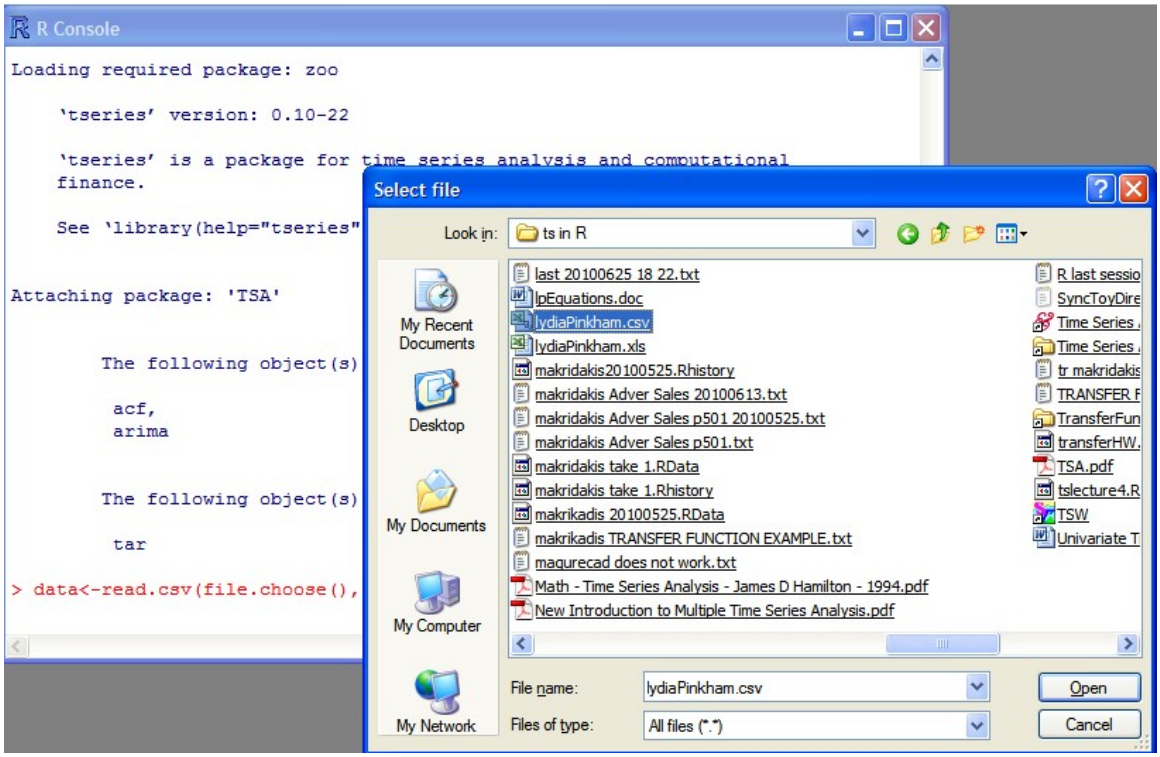
202.66 232.91 272.07 290.97 299.09 296.95 279.49 255.75
242.78 255.34 271.58 268.27 260.51 266.34 281.24 286.19
271.97 265.01 274.44 291.81 290.91 264.95 228.40 209.33
231.69 281.56 327.16 344.24 324.74 289.36 262.92 263.65
276.38 276.34 258.27 242.89 255.98 278.53 273.21 246.37
221.10 210.41 222.19 245.27 262.58 283.25 311.12 326.28
322.04 295.37 266.69 253.07 249.12 253.59 262.13 279.66
302.92 310.77 307.83 313.19 312.80 301.23 286.64 257.17
229.60 227.62 238.21 252.07 269.86 291.62 314.06 318.56
289.11 255.88 249.81 268.82 288.24 281.26 250.92 222.26

```
209.94 213.30 207.19 186.13 171.20 170.33 183.69 211.30
252.66 286.20 279.45 237.06 193.40 180.79 215.73 264.98
294.07 299.08 271.10 230.56
```

سوف نستخدم الأوامر التالية لإدخال المتسلسلات

```
> rm(list=ls())
> data<-read.csv(file.choose(), header=T)
```

لإختيار ملف البيانات تفاعليا فتظهر نافذة حوار إختيار الملف كالتالي:



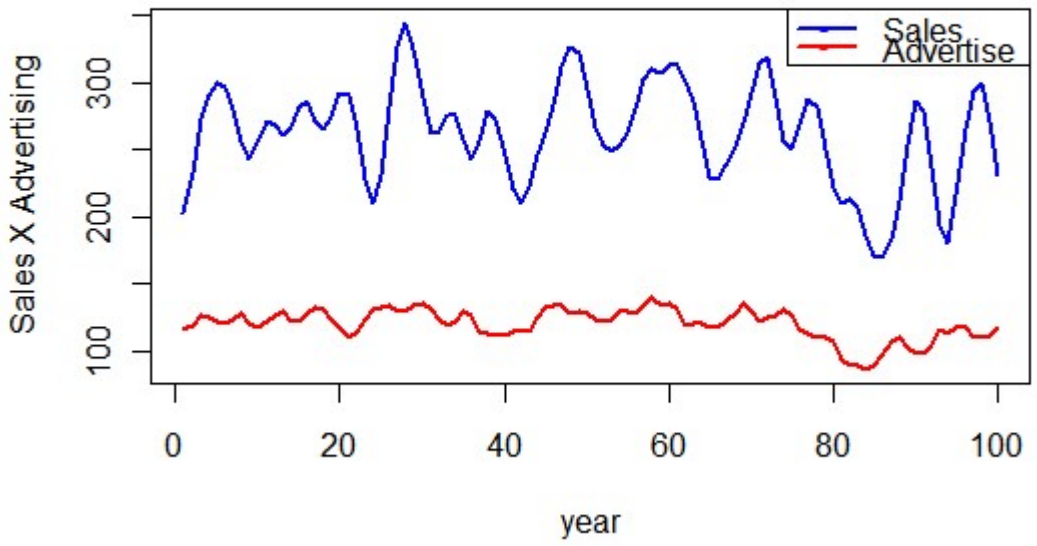
نعيد تسمية البيانات

```
> adver = ts(saladv$adver)
```

```
> sales = ts(saladv$sale)
```

نرسم المتسلسلات المدخلة كالتالي:

```
> ts.plot(sales, adver, gpars=list(xlab="year", ylab="Sales
X Advertising",col=c("blue","red"),lwd=2))
legend ("topright", pch = c("-","-") , col =c("blue","red")
, c("Sales", "Advertise"),lwd=2)
```

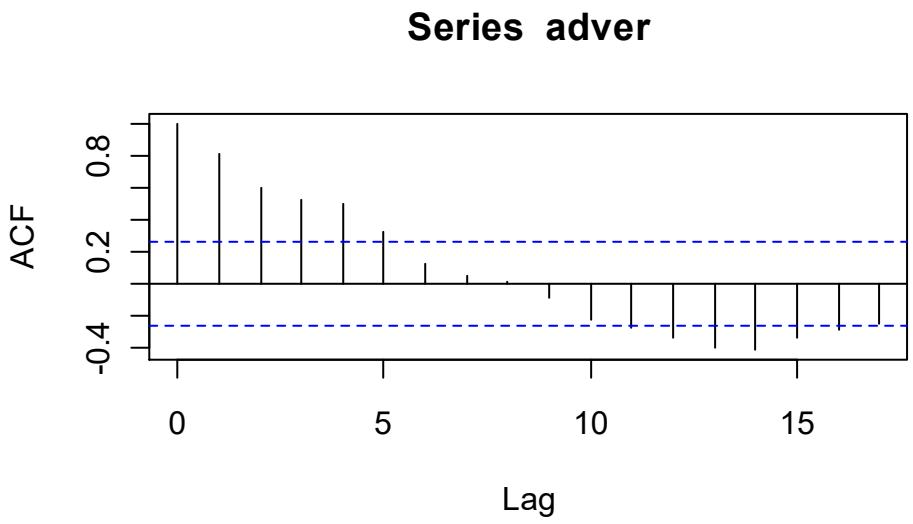


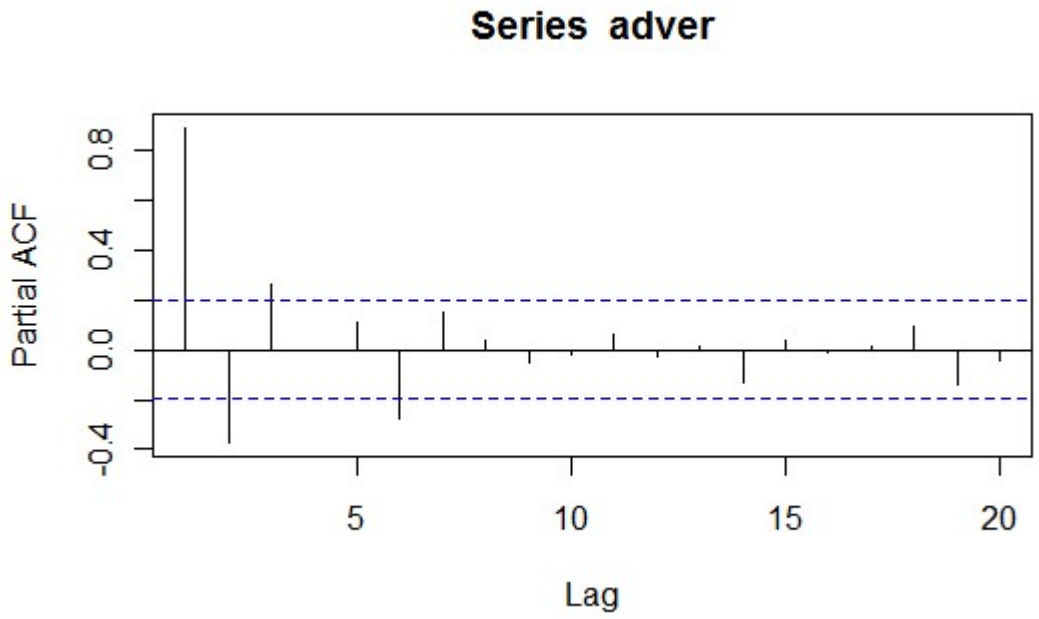
متسلسلة الإدخال x_t ومتسلسلة الإخراج z_t

1- الخطوة الاولى في بناء نموذج دالة تحويل هو تبييض متسلسلة الإدخال x_t :

أ- نفحص SACF و SPACF لمتسلسلة الإدخال x_t

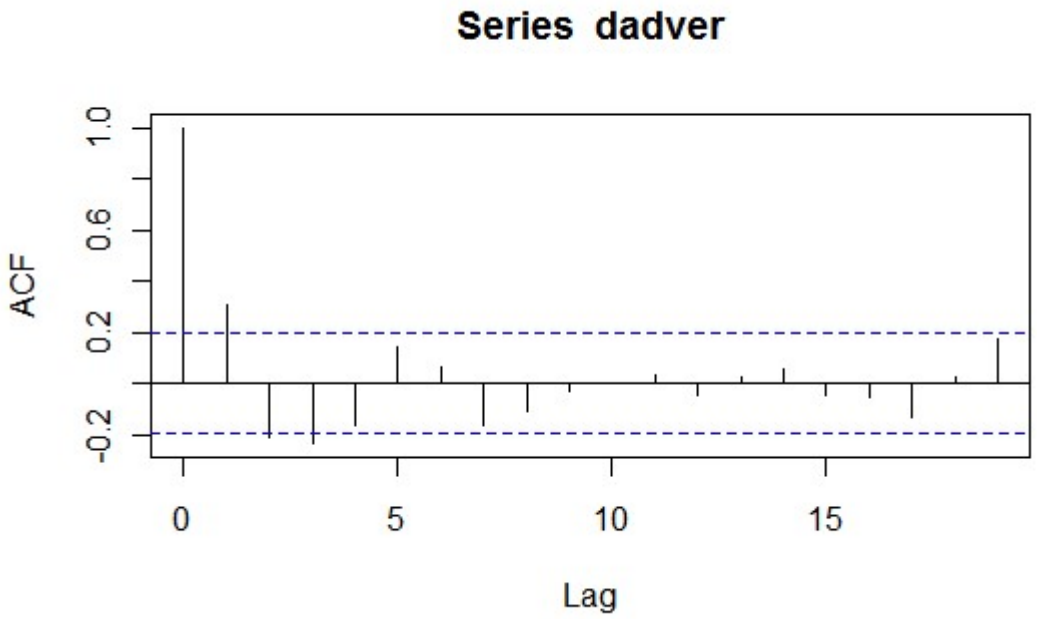
```
> acf(adver)
> pacf(adver)
```

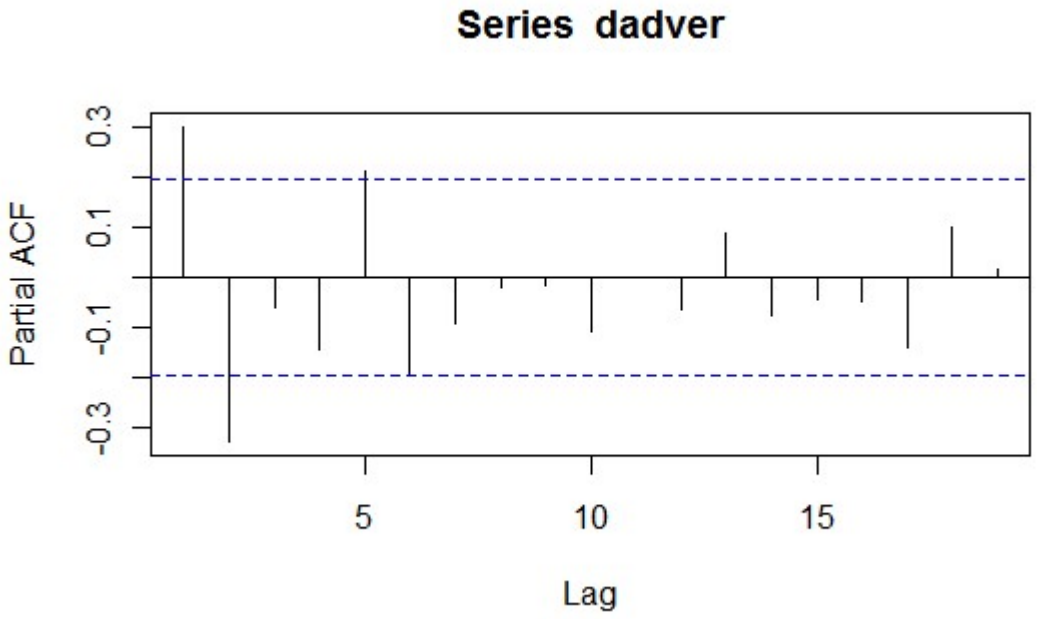




نلاحظ من أنماط SACF أن x_t غير مستقرة. نأخذ التفريق الأول ونعيد الفحص السابق

```
> dadver = diff(adver)
> acf(dadver)
> pacf(dadver)
```





نلاحظ أن متسلسلة الإدخال بعد تفريقها أصبحت مستقرة. وبفحص دالتي SACF و SPACF وبإستخدام AIC أن افضل نموذج يطبق على متسلسلة الإدخال هو AR(2) نطبق هذا النموذج

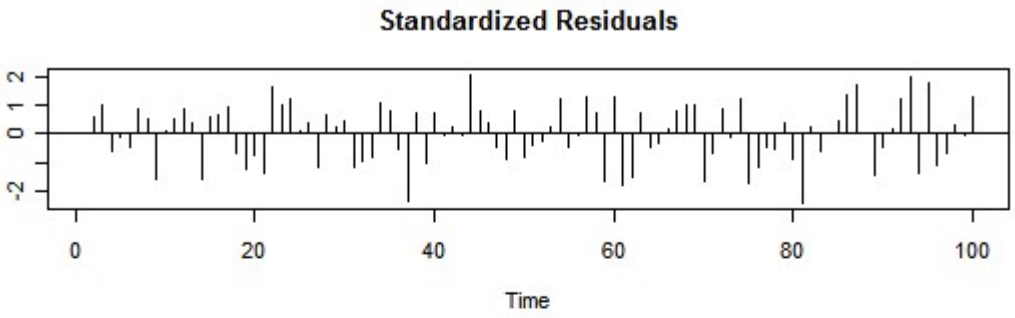
```
> fitdadver = arima(dadver, order = c(1,0,1),include.mean =
F)
> fitdadver
Call:
arima(x = dadver, order = c(1, 0, 1), include.mean = F)
Coefficients:
          ar1     ma1
      -0.2998  0.7822
s.e.   0.1438  0.0896

sigma^2 estimated as 21.91:  log likelihood = -293.52,  aic
= 593.05
>
```

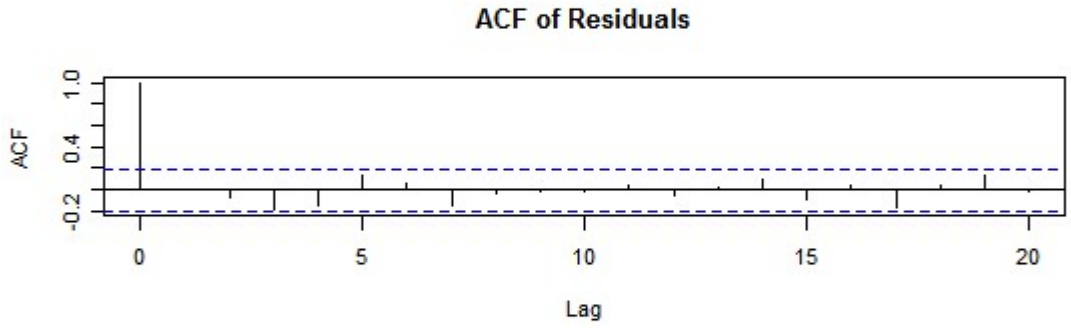
إذا النموذج المقترح هو

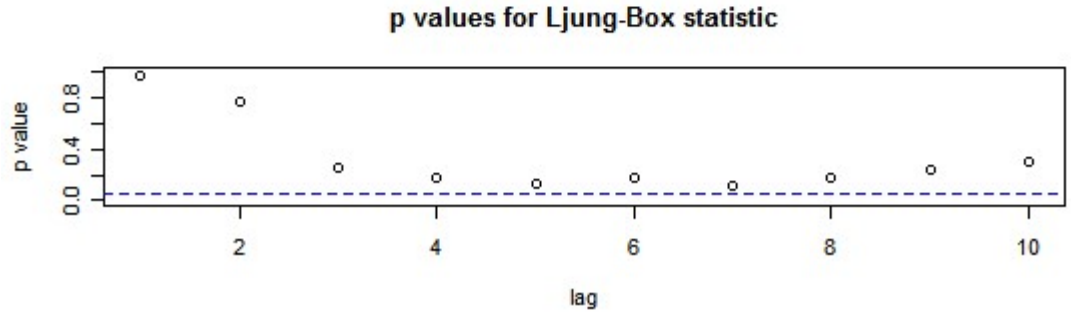
$$(1+0.2998)(1-B)x_t = (1+0.7822B)\alpha_t,\ \alpha_t \sim N(0,21.91)$$

نقوم بإجراء الإختبارات التفحصية المعتادة على البواقي

```
> tsdiag(fitdadver)
```





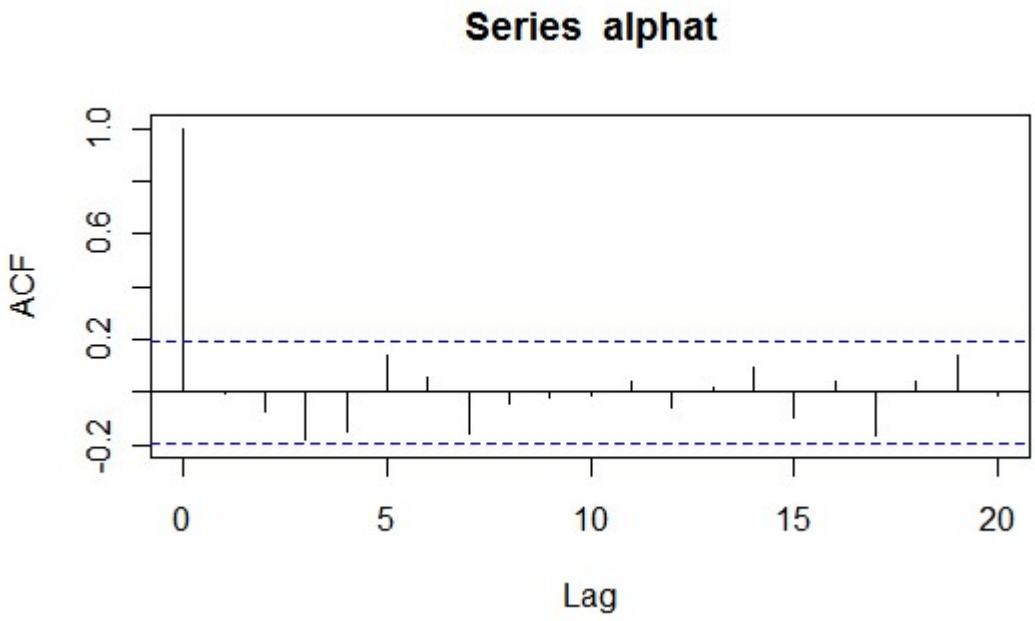


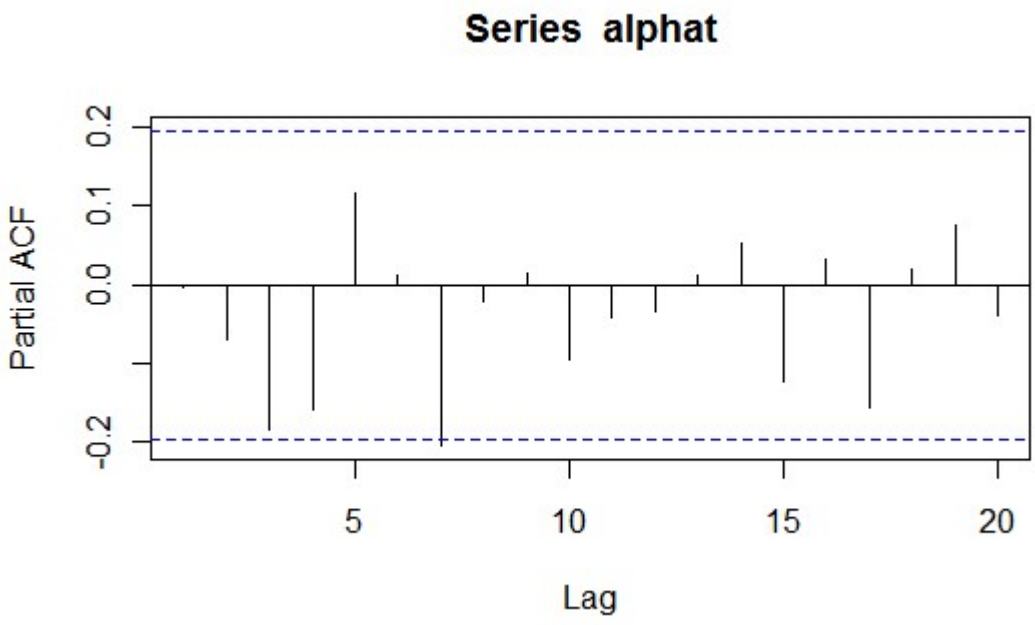
المتسلسلة المبيضة المقدرة

$$\hat{\alpha}_t = \left[\hat{\phi}_p^{(x)}(B)/\hat{\theta}_q^{(x)}(B)\right]x_t$$
$$= \frac{(1+0.2998B)}{(1+0.7822B)}(1-B)x_t$$

نسميها alphat

```
> alphat = fitdadver$residuals
> plot.ts(alphat)
> acf(alphat)
> pacf(alphat)
```



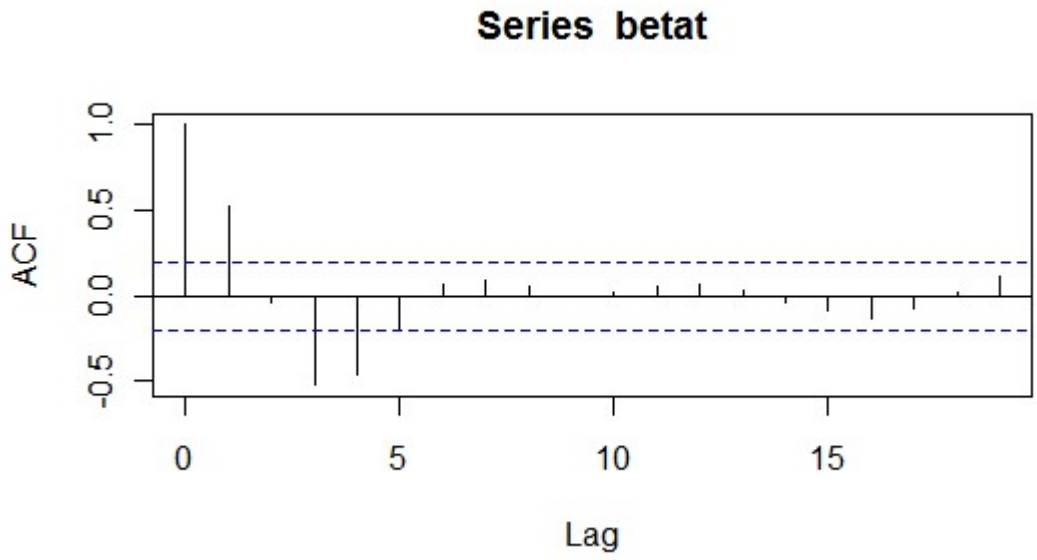


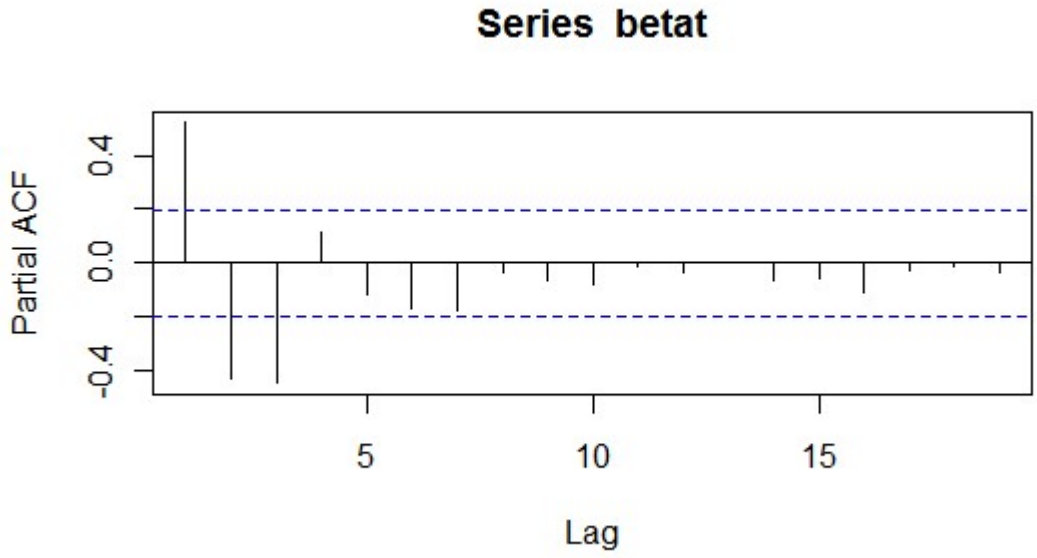
واضح جدا ان متسلسلة الإدخال تم تبييضها. سوف نطبق نفس النموذج على متسلسلة الإخراج بعد جعلها مستقرة وذلك بتصفيتها

$$a_t = \frac{(1+0.2998B)}{(1+0.7822B)}(1-B)z_t$$

ونسميها betat لهذا الغرض سوف نحتاج إلى كتابة دالة للقيام بهذا حيث أن الدالة filter الموجودة في R لاتقوم بهذا الغرض بشكل مرضي

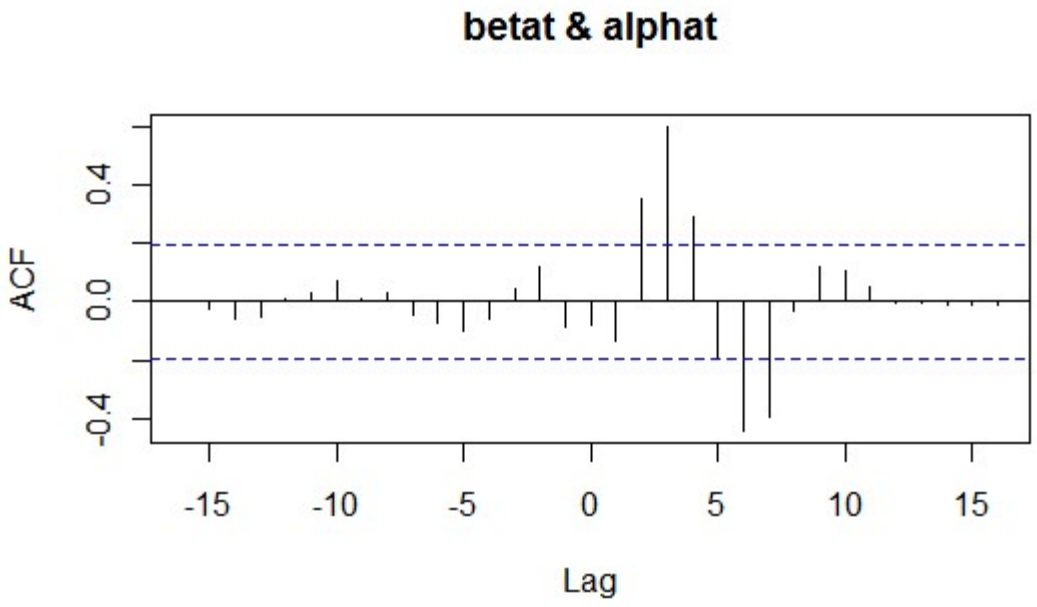
```
> c1 =fitdadver$coef[1]
> c2 =fitdadver$coef[2]
> myfilter = function(x,y,phi,theta) {y =
ts(rep(0,length(x)))
                if (length(theta) > 0 &&length(phi) > 0 &&
any(phi != 0))
                for ( i in 2:length(x)){
                y[i] = x[i]+ phi*x[i-1]+theta*y[i-1]}
                y}
> alphat = myfilter(x = dadver,y, phi = c1, theta = c2)
> betat = myfilter(x = dsale,y, phi = c1, theta = c2)

> acf(betat)
> pacf(betat)
```





بعد تطبيق التصفية على متسلسلة الأخراج نوجد دالة الترابط البيني العيني SCCF

```
> cd = ccf(betat,alphat)
```

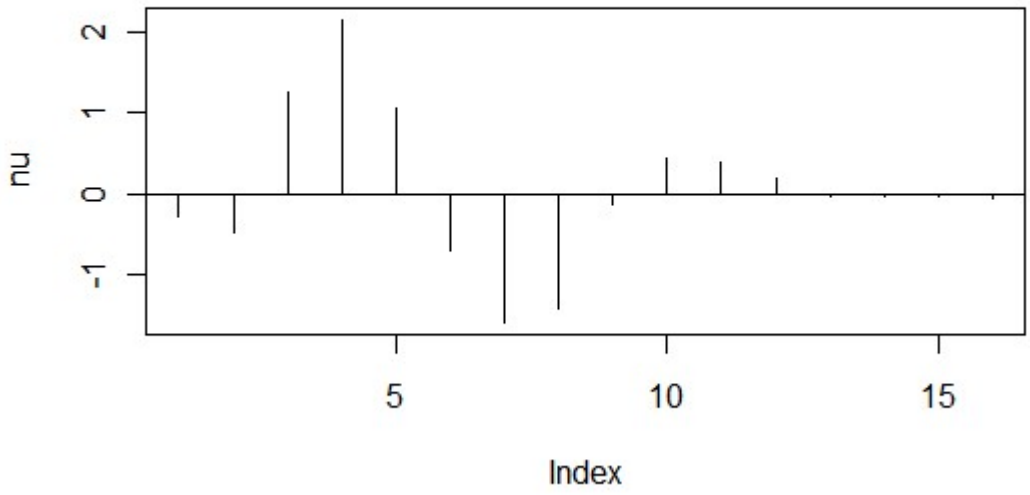


نحسب الآن الأوزان

$$\hat{\nu}_k = \frac{\hat{\sigma}_\beta}{\hat{\sigma}_\alpha} r_{\alpha\beta}(k)$$

```
> cd = ccf(betat,alphat)
> aa = cd[0:15]
> sdr = sd(betat)/sd(alphat)
> nu = sdr*aa$acf
> nu
, , 1

             [,1]
 [1,] -0.28784778
 [2,] -0.47181896
 [3,]  1.25608891
 [4,]  2.12998617
 [5,]  1.04804748
 [6,] -0.68933946
 [7,] -1.56934028
```

```
 [8,] -1.40711067
 [9,] -0.11707579
[10,]  0.43346998
[11,]  0.38775943
[12,]  0.18078461
[13,] -0.02584237
[14,] -0.02181128
[15,] -0.02838994
[16,] -0.04830890

> plot(nu,type="h")
> abline(0,0)
>
```



لاحظ أن $\hat{\sigma}_\beta, \hat{\sigma}_\alpha$ هما

```
> sd(alphat)
[1] 4.752239
> sd(betat)
[1] 17.00630
```

نلاحظ من أنماط $\hat{\nu}_k$ أن $b = 2$ و $s = 2$ و $r = 2$ مبدئيا.

لإيجاد مقدرات أولية للأوزان $\boldsymbol{\delta} = (\delta_1, ..., \delta_r)'$ نستخدم العلاقات

$$\hat{\boldsymbol{\delta}}_0 = \mathbf{A}^{-1}\mathbf{h}$$

$$A_{ij} = \begin{cases} \hat{\nu}_{b+s+i-j} & s+i \geq j \\ 0 & s+i < j \end{cases}$$

$$h_i = \hat{\nu}_{b+s+i}$$

$$i, j = 1, 2, ..., r$$

```
> b = 2
> s = 2
> r = 2
> a = matrix(c(rep(0,r*r)),r,r)
> for (i in 1:r) {for (j in 1:r) {if (s+i < j) a[i,j] = 0
else a[i,j] = bb[b+s+i-j+1]}}
> h = c(rep(0,r))
> for (i in 1:r) { h[i] = bb[b+s+i+1]}
> deltah = solve(a)%*% h
```

فينتج مقدرات $\boldsymbol{\delta} = (\delta_1, \delta_2)'$ الأولية

```
> deltah
          [,1]
```

```
[1,]  1.0075392
[2,] -0.8249332
```

نوجد مقدرات قيم ω الأولية من العلاقات

$$\hat{\omega}_{00} = \hat{\nu}_b$$

$$\hat{\omega}_{j0} = \sum_{i=1}^{j} \hat{\delta}_i \hat{\nu}_{b+j-i} - \hat{\nu}_{b+j}, \quad r \geq s$$

$$\hat{\omega}_{j0} = \left\{ \begin{array}{ll} \sum_{i=1}^{j} \hat{\delta}_i \hat{\nu}_{b+j-i} - \hat{\nu}_{b+j}, & j = 1,2,...,r \\ \sum_{i=1}^{r} \hat{\delta}_i \hat{\nu}_{b+j-i} - \hat{\nu}_{b+j}, & j = r+1,...,s \end{array} \right\} \quad r < s$$

كالتالي:

```
> if (r>=s) { omega = c(rep(0,s+1))
 omega[1] = bb[b+1]
 for(j in 1:r){ttemp = 0
 for (i in 1:j) {ttemp = ttemp + deltah[i]*bb[b+1+j-i]}
 omega[j+1] = ttemp - bb[b+1+j]} } else { {omega =
c(rep(0,s+1))
 omega[1] = bb[b+1]
 for(j in r+1:s){ttemp = 0
 for (i in 1:r) {ttemp = ttemp + deltah[i]*bb[b+1+j-i]}
 omega[j+1] = ttemp - bb[b+1+j]}
 for(j in 1:s){ttemp = 0
 for (i in 1:r) {ttemp = ttemp + deltah[i]*bb[b+1+j-i]}
 omega[j+1] = ttemp - bb[b+1+j]}} }
```

فتنتج مقدرات الأولية $\boldsymbol{\omega} = (\omega_0, \omega_1, \omega_2)'$

```
> omega
[1]  1.25350909 -0.84863376  0.04794291
```

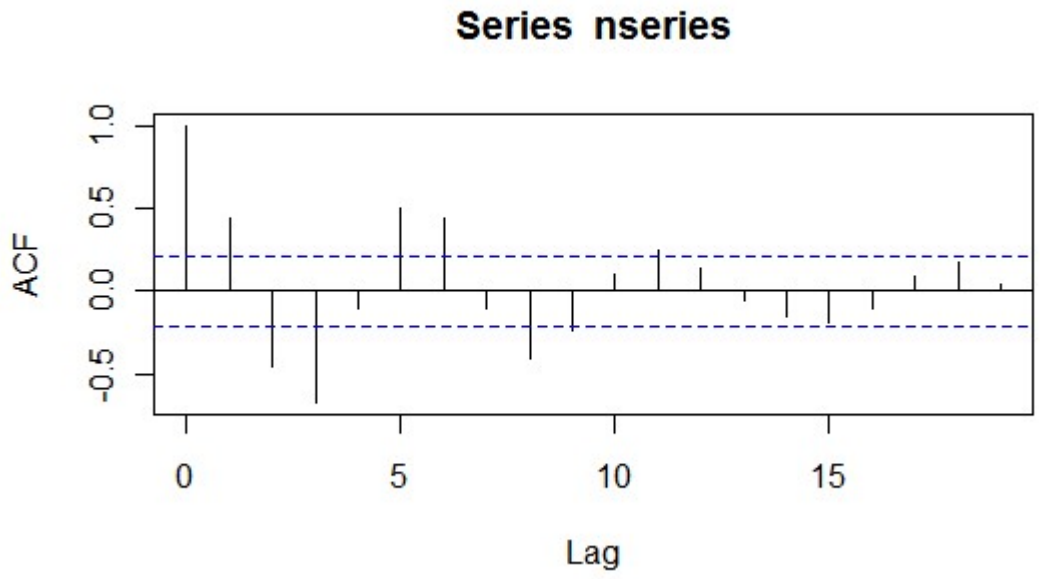
حيث $\omega_0 = 1.25350909, \omega_1 = -0.84863376, \omega_2 = 0.04794291$

الخطوة التالية هي تقدير متسلسلة الضجة n_t كالتالي

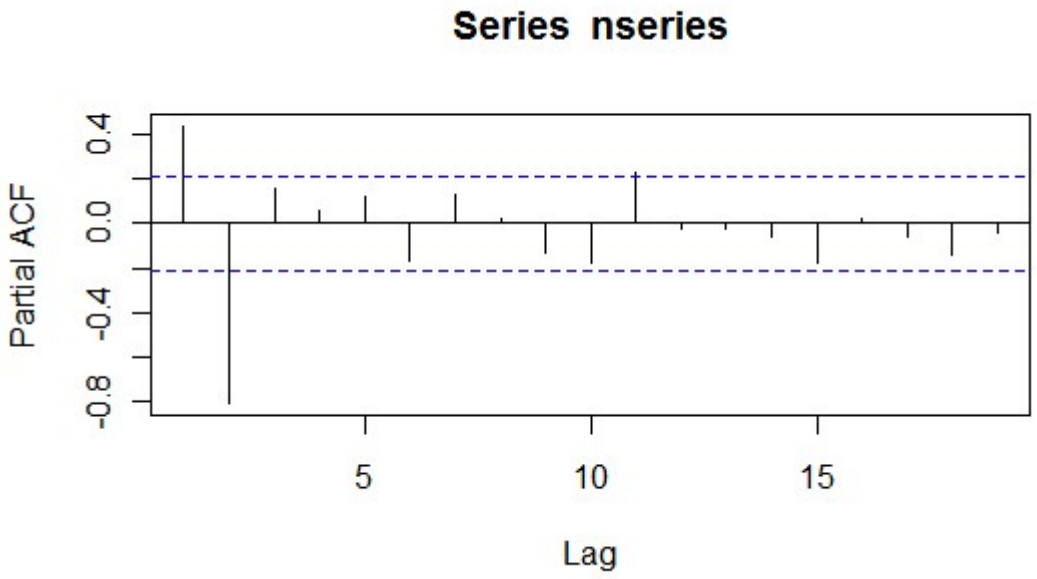
```
> tt =filter(dadver, nu,method ="convolution",sides =
1,circular = F)
> ns = dsale - tt
> nseries = ts(ns[length(nu):length(dsale)])
```

ونتعرف على نموذج لها

```
> acf(nseries)
```




```
> pacf(nseries)
```



نلاحظ من الأنماط (أنظر الجزء الأول من الكتاب) أنها قد ينطبق عليها النموذج ARIMA(2,0,1) كنموذج أولي

```
> nfit = arima(nseries, order=c(2,0,1),include.mean = F)
> nfit

Call:
arima(x = nseries, order = c(2, 0, 1), include.mean = F)

Coefficients:
         ar1      ar2     ma1
      0.7348  -0.8066  0.2376
s.e.  0.0793   0.0681  0.1383

sigma^2 estimated as 16.09:  log likelihood = -237.26,  aic
= 482.51
```

أي $\boldsymbol{\varphi} = (\phi_1, \phi_2)' = \begin{pmatrix} 0.7348 & -0.8066 \\ (0.0793) & (0.1383) \end{pmatrix}'$ و $\boldsymbol{\theta} = (\theta_1)' = \begin{pmatrix} 0.2376 \\ (0.1383) \end{pmatrix}'$ و

$\sigma_a^2 = 16.09$.

النموذج المقترح الأولي:

$$y_t = \frac{(1.25 + 0.85B - 0.05B^2)}{(1 - 1.01B + 0.083B^2)} x_{t-2} + \frac{(1 - 0.03B)}{(1 - 0.79B + 0.81B^2)} a_t,$$

$$a_t \sim WN(0, 16.1)$$

لتقدير المعالم النهائية نكون مجموع مربعات الأخطاء للنموذج المقترح كالتالي:

1- نوجد

$$z_t = \delta_1 z_{t-1} + \cdots + \delta_r z_{t-r} + \omega_0 x_{t-b} - \omega_1 x_{t-b-1} - \cdots \omega_s x_{t-b-s}, \; t \geq b + s + 1$$

2- ثم منها نوجد $n_t = y_t - z_t$

3- ومن ثم

$$a_t = n_t - \phi_1 n_{t-1} + \cdots + \phi_p n_{t-p} + \theta_1 a_{t-1} + \cdots \theta_q a_{t-q}, \; t \geq b + s + p + 1$$

```
phi = c(c3,c4)
theta = c(c5)
r=length(deltah)
s=length(omega)-1
p=length(phi)
q=length(theta)
yit = c(rep(0,length(dadver)))
for (i in b+s+1:(length(dadver)-b)) { for(j in 1:r) { ttd =
deltah[j]*yit[i-j]}
  for(j in 1:s) {tto = omega[j]*dadver[i-b+1-j]}
```

```
  yit[i]=ttd+tto}
yits = ts(yit, start=2, end=100)
nt = dsale - yits
ait = c(rep(0,length(dadver)))
for (i in b+s+p+1:(length(dadver)-p)) { for(j in 1:p) { ttd
= phi[j]*nt[i-j]}
  for(j in 1:q) {tto = theta[j]*ait[i-j]}
  ait[i]=ttd+tto}
aits = ts(ait, start=2, end=100)
aitsf <- function(dadver,dsale,b,delta,omega,phi,theta){
 r=length(deltah)
 s=length(omega)-1
 p=length(phi)
 q=length(theta)
 yit = c(rep(0,length(dadver)))
 for (i in b+s+1:(length(dadver)-b)) { for(j in 1:r) { ttd
= deltah[j]*yit[i-j]}
  for(j in 1:s) {tto = omega[j]*dadver[i-b+1-j]}
  yit[i]=ttd+tto}
 yits = ts(yit, start=2, end=100)
 nt = dsale - yits
 ait = c(rep(0,length(dadver)))
 for (i in b+s+p+1:(length(dadver)-p)) { for(j in 1:p) {
ttd = phi[j]*nt[i-j]}
  for(j in 1:q) {tto = theta[j]*ait[i-j]}
  ait[i]=ttd+tto}
 aitsf = ts(ait, start=2, end=100)
 aitsf}
```

4- ثم نستخدم مكتبة minpack.lm لتقدير المعالم

```
library(minpack.lm)
parStart <- c(deltah,omega,phi,theta)
```

```
nls.out <- nls.lm(par=parStart, fn = aitsf)
```

إستخدام مكتبة TSA لتطبيق نموذج دالة التحويل:

للمثال السابق سوف نستخدم مكتبة TSA لتطبيق دالة تحويل كالتالي:

```
> rm(list=ls())
> library(TSA)
> data =read.csv(file.choose(), header=T)
# C:\Documents and Settings\amb\Desktop\ts in
#R\xls\makridakis Adver Sales p501.csv
> data
     t  adver   sale
1    0 116.44 202.66
.....................
100 99 117.09 230.56
> adver = ts(data$adver)
> sale = ts(data$sale)
>
makri1=arimax(diff(sale),order=c(2,0,1),xtransf=lag(diff(ad
ver),-2),transfer=list(c(2,2)),method='ML', include.mean=F)
> makri1

Call:
arimax(x = diff(sale), order = c(2, 0, 1), include.mean =
F, method = "ML",
    xtransf = lag(diff(adver),-2), transfer = list(c(2,
2)))

Coefficients:
```

```
ar1      ar2      ma1     T1-AR1   T1-AR2   T1-MA0  T1-MA1  T1-MA2
1.5087 -0.9439 -0.9089  1.1305  -0.7551   0.082  -0.2070  1.6624
s.e.
0.0667   0.0441   0.0696  0.0257   0.0235   0.084   0.1051  0.0838

sigma^2 estimated as 13.87:  log likelihood = -273.83,  aic = 563.66
```

حيث:

$$ar1 = \phi_1 = 1.5087,$$

$$ar2 = \phi_2 = -0.9439,$$

$$ma1 = \theta_1 = -0.9089,$$

$$T1-AR1 = \delta_1 = 1.1305,$$

$$T1-AR2 = \delta_2 = -0.7551$$

$$T1-MA0 = \omega_0 = 0.082,$$

$$T1-MA1 = \omega_1 = -0.2070,$$

$$T1-MA2 = \omega_2 = 1.6624$$

ويكون النموذج المطبق هو

$$z_t = \frac{\left(0.082 - 0.207B + 1.6624B^2\right)}{\left(1 + 1.1305B - 0.7551B^2\right)} x_{t-2}$$

$$+ \frac{\left(1 - 0.9089B\right)}{\left(1 - 1.5087B - 0.9439B^2\right)} a_t, \quad a_t \sim WN\left(0, 13.87\right)$$

التنبؤ بإستخدام نموذج دالة التحويل:

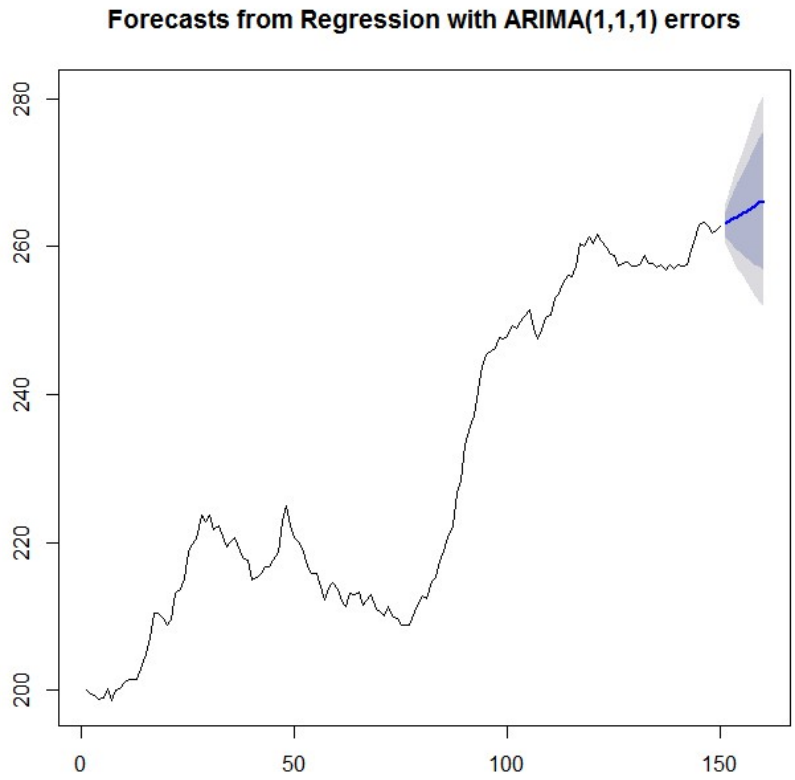
سوف نستخدم مكتبة forecast لتطبيق دالة تحويل على المتسلسلة M في كتاب Box-Jenkins (انظر المراجع).

```
> library(forecast)
> BJsales
Time Series:
Start = 1
End = 150
Frequency = 1
  [1] 200.1 199.5 199.4 198.9 199.0 200.2 198.6 200.0 200.3
201.2 201.6 201.5
      ........
> BJsales.lead
Time Series:
Start = 1
End = 150
Frequency = 1
  [1] 10.01 10.07 10.32  9.75 10.33 10.13 10.36 10.32 10.13
10.16 10.58 10.62
      .......
> yvar <-BJsales
> xvar <-BJsales.lead
> yxvar <- auto.arima(yvar, xreg=xvar)
> summary(yxvar)
Series: yvar
Regression with ARIMA(1,1,1) errors

Coefficients:
```

```
         ar1      ma1   drift     xvar
      0.8429  -0.6152  0.4030  -0.1718
s.e.  0.0812   0.1155  0.2593   0.3127

sigma^2 estimated as 1.799:  log likelihood=-253.24
AIC=516.48   AICc=516.9   BIC=531.5

Training set error measures:
                     ME     RMSE      MAE          MPE
MAPE      MASE       ACF1
Training set 0.01044087 1.318572 1.033739 0.005077292
0.4555999 0.8903301 -0.0123175
```

سوف ندخل قيم جديدة للمتغير BJsales.lead لتوليد تنبؤات لـ 10 قيم للأمام

```
> newx <- ts(c(12.92,12.64,12.79,13.05,12.69,13.01,12.90,13.12,12.47,13.5),frequency = 1)
> forecast(yxvar, xreg=newx,h=10)
    Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
151       263.0806 261.3618 264.7993 260.4520 265.7091
152       263.4432 260.7219 266.1646 259.2813 267.6052
153       263.7459 260.0910 267.4008 258.1563 269.3356
154       264.0414 259.4869 268.5960 257.0758 271.0070
155       264.4533 259.0258 269.8809 256.1526 272.7540
156       264.7567 258.4816 271.0319 255.1597 274.3537
157       265.1410 258.0436 272.2384 254.2865 275.9955
158       265.4745 257.5803 273.3687 253.4014 277.5476
159       265.9625 257.2967 274.6282 252.7094 279.2156
160       266.1660 256.7535 275.5784 251.7708 280.5611
> ff <-forecast(yxvar, xreg=newx,h=10)
> plot(ff)
>
```



إستخدام مكتبة astsa

سوف نستخدم بيانات عن الوفاة بالسكتة القلبية Cardiovascular Mortality من دراسة تمت عن التلوث Pollution في مدينة Los Angles من السنة 1970 و حتى 1979

المتغيرات هي

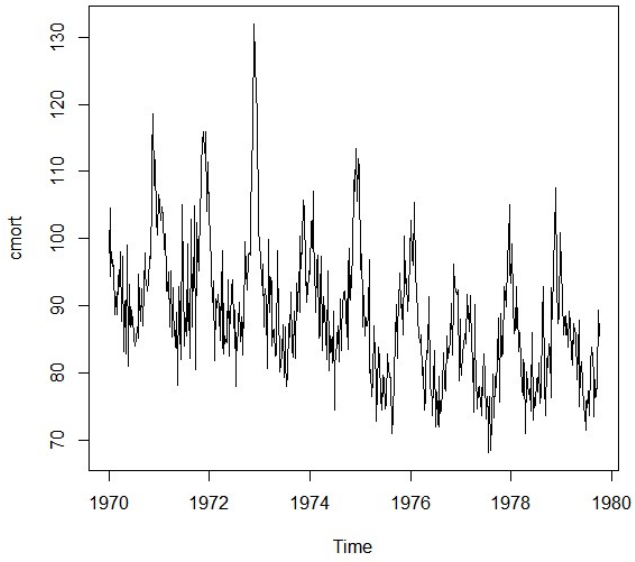
متوسط عدد الوفيات اسبوعيا (Cardiovascular Mortality)	cmort
متوسط درجة الحرارة (Temperatures)	tempr
متوسط عدد الملوثات في الجو اسبوعيا (Particulate levels)	part

تثبت مكتبة astsa (إذا لم تكن مثبتة سابقا)

```
> install.packages("astsa")
> library(astsa)
```

نوسط الزمن

```
> trend  = time(cmort) - mean(time(cmort))
> plot.ts(cmort)
```



نطبق النموذج

$$M_t = \beta_0 + \phi_1 M_{t-1} + \phi_2 M_{t-2} + \beta_1 t + \beta_2 T_{t-1} + \beta_3 P_t + \beta_4 P_{t-4} + a_t$$

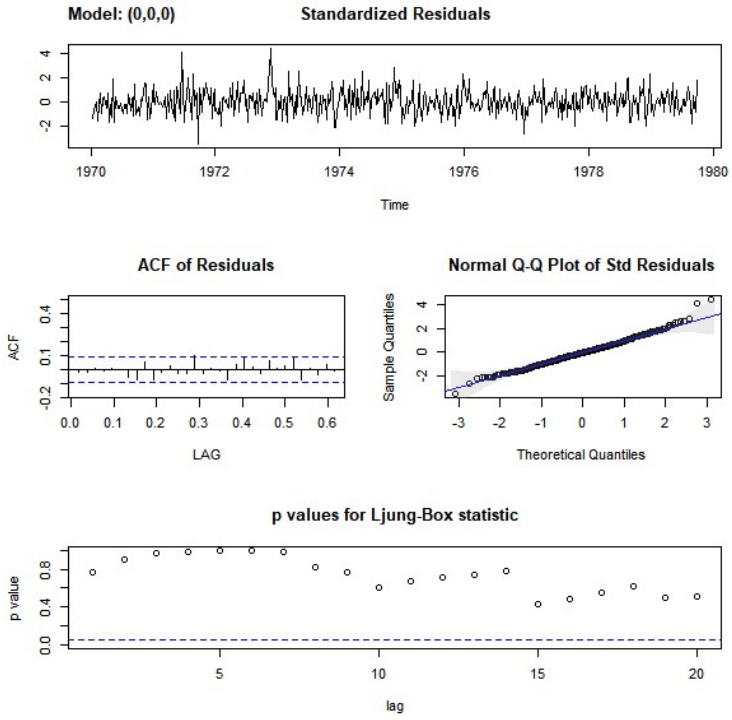
حيث M = cmort و T = tempr و P = part

اولا نطبق النموذج $ARIMAX(p = 2, q = 0)$ و نكون متجه من المتغيرات السابقة

```
> u = ts.intersect(M=cmort,M1=lag(cmort,-1),M2=lag(cmort,-
2), T1=lag(tempr,-1),P=part,P4=lag(part-4),trend)
> sarima(u[,1], 0,0,0, xreg=u[,2:7])
initial  value 1.622759
iter   1 value 1.622759
final  value 1.622759
converged
initial  value 1.622759
iter   1 value 1.622759
final  value 1.622759
```

```
converged
$fit

Call:
stats::arima(x = xdata, order = c(p, d, q), seasonal =
list(order = c(P, D,
    Q), period = S), xreg = xreg, optim.control =
list(trace = trc, REPORT = 1,
    reltol = tol))

Coefficients:
      intercept      M1      M2       T1       P     P4
trend
        40.2932  0.3190  0.3157  -0.1961  0.1210  0.021  -
0.4837
s.e.     4.6285  0.0372  0.0391   0.0307  0.0186  0.018
0.0955

sigma^2 estimated as 25.67:  log likelihood = -1536.06,
aic = 3088.11

$degrees_of_freedom
[1] 498

$ttable
          Estimate     SE t.value p.value
intercept  40.2932 4.6285  8.7055  0.0000
M1          0.3190 0.0372  8.5826  0.0000
M2          0.3157 0.0391  8.0846  0.0000
T1         -0.1961 0.0307 -6.3828  0.0000
P           0.1210 0.0186  6.5102  0.0000
P4          0.0210 0.0180  1.1633  0.2453
```

```
trend       -0.4837 0.0955 -5.0642   0.0000

$AIC
[1] 4.27324

$AICc
[1] 4.277776

$BIC
[1] 3.331798
```



نلاحظ ان P_4 غير معنوية فتخرج من النموذج

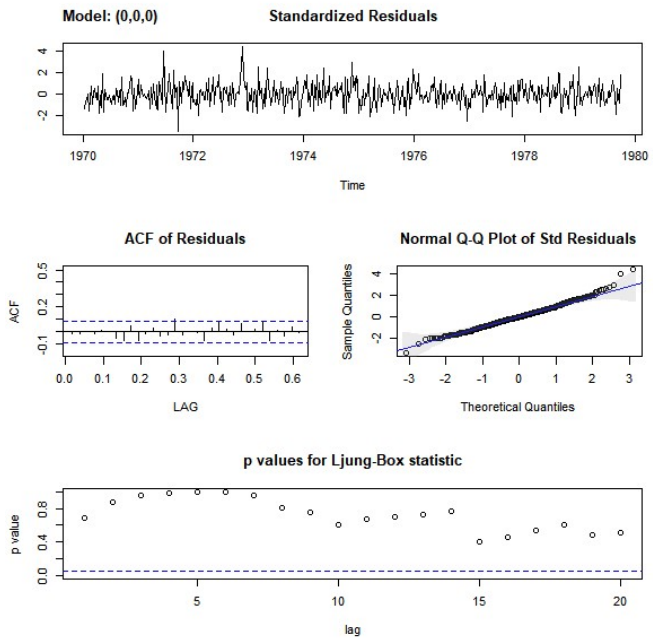
نطبق النموذج

$$M_t = \beta_0 + \phi_1 M_{t-1} + \phi_2 M_{t-2} + \beta_1 t + \beta_2 T_{t-1} + \beta_3 P_t + a_t$$

```
> sarima(u[,1], 0,0,0, xreg=u[,2:7][,-5])
initial  value 1.624097
iter   1 value 1.624097
```

```
final  value 1.624097
converged
initial  value 1.624097
iter   1 value 1.624097
final  value 1.624097
converged
$fit

Call:
stats::arima(x = xdata, order = c(p, d, q), seasonal =
list(order = c(P, D,
    Q), period = S), xreg = xreg, optim.control =
list(trace = trc, REPORT = 1,
    reltol = tol))

Coefficients:
      intercept      M1      M2       T1       P    trend
        40.2077  0.3222  0.3166  -0.1947  0.1321  -0.4806
s.e.     4.6341  0.0371  0.0391   0.0307  0.0160   0.0956

sigma^2 estimated as 25.74:  log likelihood = -1536.73,
aic = 3087.47

$degrees_of_freedom
[1] 499

$ttable
          Estimate     SE t.value p.value
intercept  40.2077 4.6341  8.6765       0
M1          0.3222 0.0371  8.6806       0
M2          0.3166 0.0391  8.0985       0
T1         -0.1947 0.0307 -6.3325       0
```

```
P            0.1321 0.0160  8.2666       0
trend       -0.4806 0.0956 -5.0275       0

$AIC
[1] 4.271956

$AICc
[1] 4.276362

$BIC
[1] 3.322149
```

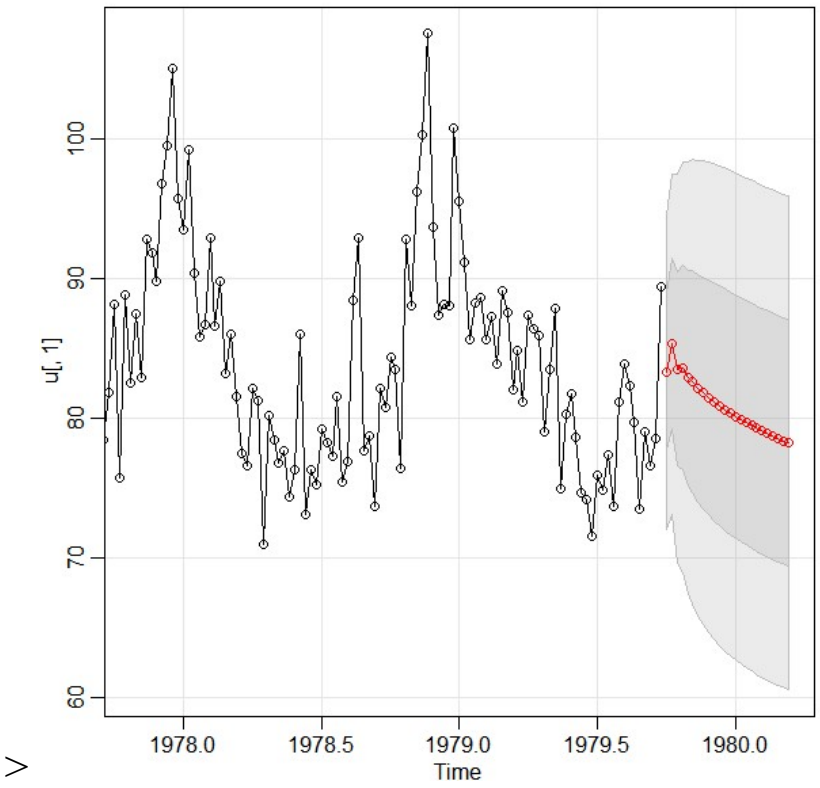


ونلاحظ ان النموذج مناسب.

التنبؤ بالنموذج

```
> sarima.for(u[,1],n.ahead,2,0,0,0,0,0,0, xreg=time(u[,1]),
newxreg=nureg)
$pred
Time Series:
```

```
Start = c(1979, 40)
End = c(1980, 11)
Frequency = 52
 [1] 83.28959 85.30583 83.52185 83.61793 82.88304 82.59994
82.15185 81.82657 81.48211 81.18158 80.89099 80.62301
80.36851 80.12907 79.90149
[16] 79.68514 79.47841 79.28035 79.08991 78.90625 78.72860
78.55627 78.38865 78.22522

$se
Time Series:
Start = c(1979, 40)
End = c(1980, 11)
Frequency = 52
 [1] 5.652002 6.109661 6.961128 7.346426 7.719485 7.965258
8.167390 8.316790 8.434760 8.525260 8.596115 8.651190
8.694286 8.727960 8.754336
[16] 8.774995 8.791193 8.803895 8.813861 8.821682 8.827821
8.832641 8.836425 8.839397
```



>

إستخدام مكتبة dynlm

```
> install.packages("dynlm")
> library(dynlm)
> u2 = ts.intersect(T1=lag(tempr,-1),P=part,P4=lag(part-
4),trend)
> fit = dynlm(cmort~u2)
> summary(fit)

Time series regression with "ts" data:
Start = 1970(2), End = 1979(39)

Call:
dynlm(formula = cmort ~ u2)

Residuals:
    Min      1Q  Median      3Q     Max
```

```
-14.580  -4.170  -0.629   3.464  32.001

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 114.74251    2.95843  38.785  < 2e-16 ***
u2T1         -0.48975    0.03386 -14.466  < 2e-16 ***
u2P           0.16561    0.02402   6.895 1.63e-11 ***
u2P4          0.05719    0.02345   2.439   0.0151 *
u2trend      -1.44344    0.10532 -13.705  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Residual standard error: 6.626 on 501 degrees of freedom
Multiple R-squared:  0.5653,    Adjusted R-squared:  0.5618
F-statistic: 162.8 on 4 and 501 DF,  p-value: < 2.2e-16
```

الفصل الثالث

نماذج المتسلسلات الزمنية المتجهيه للمتسلسلات المتعددة المتغيرات

Series Models Vector Multivariate Time

سوف نعمم طرق التنبؤ لمتسلسلة واحدة *Univariate* و التي تم تغطيتها في الجزء الأول من الكتاب للتنبؤ بعدة متسلسلات *Multivariate* حيث ان النظرية هي نفسها و التعميم مباشر.

لنفترض انه لدينا m متسلسلة زمنية $\{Z_{ti}, t = 0, \pm 1, ...\}, i = 1, 2, ..., m$ مستقرة وطبيعية بمتوسطات وتغايرات

$$\mu_{ti} = E(Z_{ti})$$

$$\gamma_{ij}(t+\tau, t) = E\left[\left(Z_{t+\tau,i} - \mu_{ti}\right)\left(Z_{t+\tau,j} - \mu_{tj}\right)\right]$$

لسهولة التعامل مع المتسلسلات الزمنية المتعددة المتغيرات سوف نستخدم المتجهات للتعبير عنها كالتالي:

$$\mathbf{Z}_t = \begin{bmatrix} Z_{t1} \\ \vdots \\ Z_{tm} \end{bmatrix}, t = 0, \pm 1, ...$$

ومتجه متوسطات

$$\boldsymbol{\mu}_t = E(\mathbf{Z}_t) = \begin{bmatrix} \mu_{t1} \\ \vdots \\ \mu_{tm} \end{bmatrix}$$

ومصفوفة تغاير

$$\boldsymbol{\Gamma}(t+\tau,t)=\begin{bmatrix}\gamma_{11}(t+\tau,t) & \cdots & \gamma_{1m}(t+\tau,t)\\ \vdots & \ddots & \vdots\\ \gamma_{m1}(t+\tau,t) & \cdots & \gamma_{mm}(t+\tau,t)\end{bmatrix}$$

او

$$\boldsymbol{\Gamma}(t+\tau,t)=E\left[(\mathbf{Z}_{t+\tau}-\boldsymbol{\mu}_{t+\tau})(\mathbf{Z}_t-\boldsymbol{\mu}_t)'\right]$$

حيث

$$\gamma_{ij}(t+\tau,t)=\operatorname{cov}\left(Z_{t+\tau,i},Z_{t+\tau,j}\right)$$

تعريف:

المتسلسلة الزمنية ذات m متغير $\{\mathbf{Z}_t\}$ تعتبر مستقرة إذا تحققت الشروط التالية:

$(1)\quad \boldsymbol{\mu}_{\mathbf{Z}}(t)$

لاتعتمد على t

و

$(2)\quad \boldsymbol{\Gamma}_{\mathbf{Z}}(t+\tau,t)$

لاتعتمد على t لكل τ.

للمتسلسلات المستقرة سوف نرمز لمتجه المتوسطات بالرمز

$$\boldsymbol{\mu}=E(\mathbf{Z}_t)=\begin{bmatrix}\mu_1\\ \vdots\\ \mu_m\end{bmatrix}$$

ومصفوفة التغاير بالرمز

$$\boldsymbol{\Gamma}(\tau)=E\left[(\mathbf{Z}_{t+\tau}-\boldsymbol{\mu})(\mathbf{Z}_t-\boldsymbol{\mu})'\right]=\begin{bmatrix}\gamma_{11}(\tau) & \cdots & \gamma_{1m}(\tau)\\ \vdots & \ddots & \vdots\\ \gamma_{m1}(\tau) & \cdots & \gamma_{mm}(\tau)\end{bmatrix}$$

العنصر $\gamma_{ii}(.)$ هي دالة التغاير للمتسلسلة المستقرة $\{Z_{t,i}\}$ و $\gamma_{ij}(.), i \neq j$ هي دالة التغير البيني cross-covariance بين المتسلسلتين المستقرة $\{Z_{t,i}\}$ و $\{Z_{t,j}\}$.

ملاحظة:

عموما قد لاتكون مصفوفة $\boldsymbol{\Gamma}(\tau)$ متناظرة أي $\gamma_{ij}(.) \neq \gamma_{ji}(.), i \neq j$.

مصفوفة الترابط $\mathbf{R}(.)$ تعرف بالتالي

$$\mathbf{R}(\tau) = \begin{bmatrix} \rho_{11}(\tau) & \cdots & \rho_{1m}(\tau) \\ \vdots & \ddots & \vdots \\ \rho_{m1}(\tau) & \cdots & \rho_{mm}(\tau) \end{bmatrix}$$

حيث

$$\rho_{ij}(\tau) = \gamma_{ij}(\tau) \Big/ \left[\gamma_{ii}(0)\gamma_{jj}(0)\right]^{1/2}$$

الخواص الأساسية للدالة $\boldsymbol{\Gamma}(.)$

1. $\boldsymbol{\Gamma}(\tau) = \boldsymbol{\Gamma}'(-\tau),$
2. $\left|\gamma_{ij}(\tau)\right| \leq \left[\gamma_{ii}(0)\gamma_{jj}(0)\right]^{1/2}, i, j = 1, 2, ..., m$
3. $\gamma_{ii}(0)$

هي دالة تغاير ذاتي لكل $i = 1, 2, ..., m$

4. $\sum_{j,k=1}^{n} \mathbf{a}_j' \boldsymbol{\Gamma}(j-k)\mathbf{a}_k \geq 0 \quad \forall n \in \{1, 2, ...\}, \mathbf{a}_1, ..., \mathbf{a}_n \in \mathbb{R}^m$

تعريف:

المتسلسلة الزمنية ذات m متغير $\{\mathbf{a}_t\}$ تسمى **متسلسلة ضجة بيضاء** بمتوسط $\mathbf{0}$ ومصفوفة تغاير $\boldsymbol{\Sigma}_a$ وتكتب

$$\{\mathbf{a}_t\} \sim WN(\mathbf{0}, \boldsymbol{\Sigma}_a)$$

إذا كانت $\{\mathbf{a}_t\}$ مستقرة بمتوسط صفري و مصفوفة تغاير

$$\mathbf{\Gamma}_a(\tau) = \begin{cases} \mathbf{\Sigma}_a, & if\ \tau = 0 \\ 0 & if\ \tau \neq 0 \end{cases}$$

تعريف:

المتسلسلة الزمنية ذات m متغير $\{\mathbf{a}_t\}$ تسمى **متسلسلة ضجة *iid*** بمتوسط **0** ومصفوفة تغاير $\mathbf{\Sigma}_a$ وتكتب

$$\{\mathbf{a}_t\} \sim iid(\mathbf{0}, \mathbf{\Sigma}_a)$$

إذا كانت المتجهات العشوائية $\{\mathbf{a}_t\}$ مستقلة و موزعة توزيع متساوي بمتوسط صفري و مصفوفة تغاير $\mathbf{\Sigma}_a$.

كما في حالة متسلسلات المتغير الواحد فإن متسلسلة الضجة البيضاء المتعددة تستخدم لبناء عدد كبير من المتسلسلات الزمنية متعددة المتغيرات.

تعريف:

المتسلسلة ذات m متغير $\{\mathbf{Z}_t\}$ تسمى عملية خطية linear process إذا كانت على الشكل التالي:

$$\mathbf{Z}_t = \sum_{j=-\infty}^{\infty} C_j \mathbf{a}_{t-j}, \quad \{\mathbf{a}_t\} \sim WN(\mathbf{0}, \mathbf{\Sigma}_a)$$

حيث $\{C_j\}$ متتابعة من المصفوفات ذات الأبعاد $m \times m$ وبعناصر قابلة الجمع بقيم مطلقة.

من السهل إثبات ان العملية الخطية مستقرة بمتوسط **0** ودالة تغاير

$$\mathbf{\Gamma}(\tau) = \sum_{j=-\infty}^{\infty} C_{j+\tau} \mathbf{\Sigma}_a C_j', \ \tau = 0, \pm 1, \ldots$$

تقدير المتوسط ودالة التغاير:

1. تقدير المتوسط $\boldsymbol{\mu}$:

لنفترض ان لدينا المشاهدات $\mathbf{z}_1, \mathbf{z}_2, ... \mathbf{z}_n$ من متسلسلات زمنية مستقرة متعددة المتغيرات. مقدر غير منحاز لمتجة المتوسطات $\boldsymbol{\mu} = (\mu_1, \mu_2, ..., \mu_m)'$ يعطى بالعلاقة:

$$\overline{\mathbf{z}}_n = \frac{1}{n}\sum_{t=1}^{n} \mathbf{z}_t$$

حيث

$$\mu_i = \frac{1}{n}\sum_{t=1}^{n} z_{ti}, \; i = 1, 2, ..., m$$

2. تقدير دالة التغاير $\boldsymbol{\Gamma}(\tau)$:

لنفترض ان لدينا المشاهدات $\mathbf{z}_1, \mathbf{z}_2, ... \mathbf{z}_n$ من متسلسلات زمنية مستقرة متعددة المتغيرات. مقدر لدالة التغاير $\boldsymbol{\Gamma}(\tau) = E\left[(\mathbf{Z}_{t+\tau} - \boldsymbol{\mu})(\mathbf{Z}_t - \boldsymbol{\mu})' \right]$ يعطى بالعلاقة:

$$\hat{\boldsymbol{\Gamma}}(\tau) = \begin{cases} \frac{1}{n}\sum_{t=1}^{n-\tau} (\mathbf{z}_{t+\tau} - \overline{\mathbf{z}}_n)(\mathbf{z}_t - \overline{\mathbf{z}}_n)' & , 0 \le \tau \le -1 \\ \hat{\boldsymbol{\Gamma}}'(-\tau) & , -n+1 \le \tau < 0 \end{cases}$$

$$\hat{\boldsymbol{\Gamma}}(\tau) = \begin{bmatrix} \hat{\gamma}_{11}(\tau) & \cdots & \hat{\gamma}_{1m}(\tau) \\ \vdots & \ddots & \vdots \\ \hat{\gamma}_{m1}(\tau) & \cdots & \hat{\gamma}_{mm}(\tau) \end{bmatrix}$$

حيث

$$\hat{\gamma}_{ij} = \frac{1}{n}\sum_{t=1}^{n-\tau}\left(z_{t+\tau,i} - \overline{z}_{n,i}\right)\left(z_{t,j} - \overline{z}_{n,j}\right)$$

وتقدر دالة الترابط بالعلاقة

$$\hat{\mathbf{R}}(\tau) = \begin{bmatrix} \hat{\rho}_{11}(\tau) & \cdots & \hat{\rho}_{1m}(\tau) \\ \vdots & \ddots & \vdots \\ \hat{\rho}_{m1}(\tau) & \cdots & \hat{\rho}_{mm}(\tau) \end{bmatrix}$$

حيث

$$\hat{\rho}_{ij}(\tau) = \hat{\gamma}_{ij}(\tau) \Big/ \left[\hat{\gamma}_{ii}(0)\hat{\gamma}_{jj}(0)\right]^{1/2}$$

عندما $i = j$ تصبح $\hat{\rho}_{ij}(\tau)$ دالة العينة للترابط الذاتى للمتسلسلة الزمنية i .

عمليات VARMA متعددة المتغيرات:

تعريف:

العملية $\{\mathbf{Z}_t\}$ يقال انها عملية $\mathbf{VARMA}(p,q)$ إذا كانت مستقرة وتحقق العلاقة:

$$\mathbf{Z}_t - \mathbf{\Phi}_1\mathbf{Z}_{t-1} - \cdots - \mathbf{\Phi}_p\mathbf{Z}_{t-p} = \mathbf{a}_t + \mathbf{\Theta}_1\mathbf{a}_{t-1} + \cdots + \mathbf{\Theta}_q\mathbf{a}_{t-q}$$

حيث

$$\{\mathbf{a}_t\} \sim WN(\mathbf{0}, \mathbf{\Sigma}_a)$$

او نكتب على الشكل

$$\mathbf{\Phi}(B)\mathbf{Z}_t = \mathbf{\Theta}(B)\mathbf{a}_t, \quad \{\mathbf{a}_t\} \sim WN(\mathbf{0}, \mathbf{\Sigma}_a)$$

حيث

$$\mathbf{\Phi}(B) = \mathbf{I} - \mathbf{\Phi}_1 B - \cdots - \mathbf{\Phi}_p B^p$$

و

$$\boldsymbol{\Theta}(B) = \mathbf{I} + \boldsymbol{\Theta}_1 B - \cdots + \boldsymbol{\Theta}_q B^q$$

مصفوفات كثيرات الحدود و $\mathbf{I}$ مصفوفة الوحدة ذات الأبعاد $m \times m$ و B عامل الإزاحة الخلفي (أنظر الجزء الأول) و $\boldsymbol{\Phi}(B)$ يسمى عامل الإنحدار الذاتي و $\boldsymbol{\Theta}(B)$ عامل المتوسط المتحرك (أنظر الجزء الأول).

عملية $\mathbf{VAR}(1)$ متعددة المتغيرات:

بوضع $p = 1$ و $q = 0$ في تعريف العملية $\mathbf{VARMA}(p,q)$ نجد

$$\mathbf{Z}_t = \boldsymbol{\Phi}\mathbf{Z}_{t-1} + \mathbf{a}_t, \quad \{\mathbf{a}_t\} \sim WN(\mathbf{0}, \boldsymbol{\Sigma}_a)$$

و تسمى عملية عملية $\mathbf{VAR}(1)$ متعددة المتغيرات.

تعريف:

دالة الأوزان لنموذج $\mathbf{VARMA}(p,q)$ هي

$$\boldsymbol{\Psi}(B) = \frac{\boldsymbol{\Theta}_q(B)}{\boldsymbol{\Phi}_p(B)} = \boldsymbol{\Psi}_0 B^0 + \boldsymbol{\Psi}_1 B^1 + \boldsymbol{\Psi}_2 B^2 + \cdots, \quad \boldsymbol{\Psi}_0 = \mathbf{I}$$

او

$$\boldsymbol{\Psi}(B) = \frac{\boldsymbol{\Theta}_q(B)}{\boldsymbol{\Phi}_p(B)} = \sum_{j=0}^{\infty} \boldsymbol{\Psi}_j B^j, \quad \boldsymbol{\Psi}_0 = \mathbf{I}$$

على شرط ان كثيرة الحدود $\boldsymbol{\Phi}_p(B)$ تكون كل أصفارها داخل كرة الوحدة أو أن الأوزان تتقارب أي $\sum_{j=0}^{\infty} \left\| \boldsymbol{\Psi}_j \right\| < \infty$.

مثال:

دالة الأوزان لعملية متعددة المتغيرات:

$$\boldsymbol{\Psi}(B)=\frac{\mathbf{I}}{\boldsymbol{\Phi}_p(B)}$$

$$\boldsymbol{\Psi}(B)\boldsymbol{\Phi}_1(B)=\mathbf{I}$$

ومنها نجد

$$\boldsymbol{\Psi}_0=\mathbf{I}$$

$$\boldsymbol{\Psi}_1=\boldsymbol{\Phi}\boldsymbol{\Psi}_0=\boldsymbol{\Phi},$$

$$\boldsymbol{\Psi}_2=\boldsymbol{\Phi}\boldsymbol{\Psi}_1=\boldsymbol{\Phi}^2,$$

$$\vdots$$

$$\boldsymbol{\Psi}_j=\boldsymbol{\Phi}\boldsymbol{\Psi}_{j-1}=\boldsymbol{\Phi}^j,\quad j\geq 3$$

مصفوفة دالة التغاير لعملية $\mathbf{VARMA}(p,q)$ المستقرة بدلالة دالة الأوزان

$$\boldsymbol{\Gamma}(\tau)=\sum_{j=0}^{\infty}\boldsymbol{\Psi}_{\tau+j}\boldsymbol{\Sigma}_a\boldsymbol{\Psi}'_j,\quad \tau=0,\pm1,...$$

التنبؤات ذات متوسط مربع الخطأ الأدنى لعملية $\mathbf{VARMA}(p,q)$ متعددة المتغيرات:

كما في حالة العمليات نماذج $ARMA(p,q)$ أحادة المتغير يمكن إثبات أن دالة التنبؤ ℓ خطوة للأمام تعطى بالعلاقة:

$$\mathbf{Z}_n(\ell)=\boldsymbol{\Psi}_\ell\mathbf{a}_n+\boldsymbol{\Psi}_{\ell+1}\mathbf{a}_{n-1}+\boldsymbol{\Psi}_{\ell+2}\mathbf{a}_{n-2}+\cdots,\quad \ell\geq 1$$

و أخطاء التنبؤ

$$\mathbf{e}_n(\ell)=\mathbf{Z}_{n+\ell}-\mathbf{Z}_n(\ell)=\mathbf{a}_{n+\ell}+\boldsymbol{\Psi}_1\mathbf{a}_{n+\ell-1}+\boldsymbol{\Psi}_2\mathbf{a}_{n+\ell-2}+\cdots+\boldsymbol{\Psi}_{\ell-1}\mathbf{a}_{n+1},\quad \ell\geq 1$$

و تباينات أخطاء التنبؤ

$$\mathbf{V}\left[\mathbf{e}_n(\ell)\right]=\boldsymbol{\Sigma}_a\left(\mathbf{I}+\boldsymbol{\Psi}_1^2+\boldsymbol{\Psi}_2^2+\cdots+\boldsymbol{\Psi}_{\ell-1}^2\right),\quad \ell\geq 1$$

نظرية:

المتنبىء ذا متوسط مربع الخطأ الأدنى لعملية $\mathbf{VARMA}(p,q)$ يعطى بالعلاقة:

$$\mathbf{Z}_n(\ell) = E(\mathbf{Z}_{n+\ell} \mid \mathbf{Z}_n, \mathbf{Z}_{n-1}, \cdots), \quad \ell \geq 1$$

أي هو التوقع الشرطي للقيمة المستقبلية $\mathbf{Z}_{n+\ell}$ لقيم $\ell \geq 1$ معطى

$\{\mathbf{Z}_1, \mathbf{Z}_2, ..., \mathbf{Z}_{n-1}, \mathbf{Z}_n\}$.

كالسابق (انظر الجزء الأول) نستخدم القاعدة التالية في إيجاد التنبؤات في النظرية السابقة

$$1 - E(\mathbf{Z}_{n+j} \mid \mathbf{Z}_n, \mathbf{Z}_{n-1}, \cdots) = \begin{cases} \mathbf{Z}_{n+j} & j \leq 0 \\ \mathbf{Z}_n(j) & j > 0 \end{cases}$$

$$2 - E(\mathbf{a}_{n+j} \mid \mathbf{a}_n, \mathbf{a}_{n-1}, \cdots) = \begin{cases} \mathbf{a}_{n+j} & j \leq 0 \\ \mathbf{0} & j > 0 \end{cases}$$

مثال عددي:

سوف نقوم بنمذجة المتسلسلة الزمنية متعددة المتغيرات للنسبة المئوية لمعدل النمو الربع سنوي للناتج المحلى الكلي للمملكة المتحدة و كندا والولايات المتحدة من الربع الأول من عام 1980 و حتى الربع الأخير من عام 2010 . سوف نستخدم مكتبة MTS (**M**ultivariate **T**ime **S**eries) في R

```
> library(MTS)
> qda=read.table("q-gdp-ukcaus.txt",header=T)
```

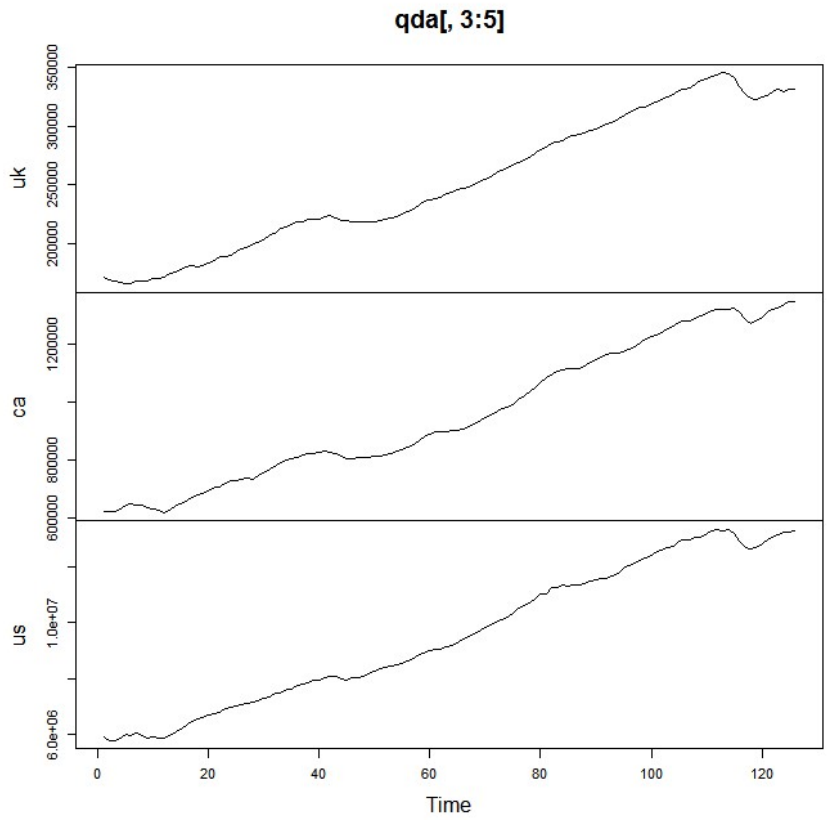
أدخلنا البيانات. ننظر للـ 6 مشاهدات الأولى و الأخيرة

```
> head(qda)
  year mon     uk     ca      us
1 1980   1 172436 624794 5908500
2 1980   4 169359 623433 5787400
```

```
3 1980   7 169038 623215 5776600
4 1980  10 167180 630215 5883500
5 1981   1 166052 645957 6005700
6 1981   4 166393 651954 5957800
> tail(qda)
    year mon     uk      ca       us
121 2010   1 325360 1312845 12937700
122 2010   4 328836 1320294 13058500
123 2010   7 330860 1328350 13139600
124 2010  10 329189 1338481 13216100
125 2011   1 330724 1350078 13227900
126 2011   4 331263 1348342 13271800
>
```

نرسم المتسلسلات

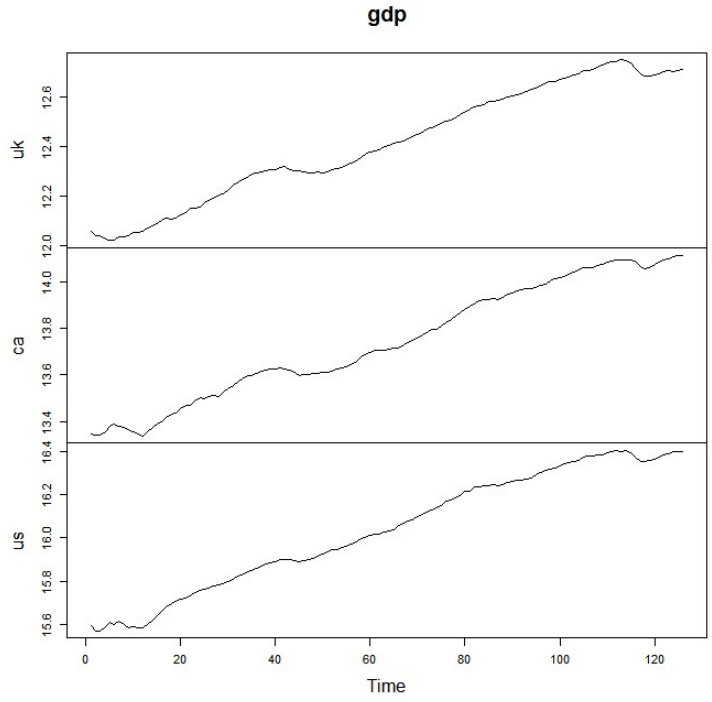
```
> plot.ts(qda[,3:5])
```



نستخدم التحويل اللوغارثمى لإيجاد المعدلات

```
> gdp=log(qda[,3:5])
```

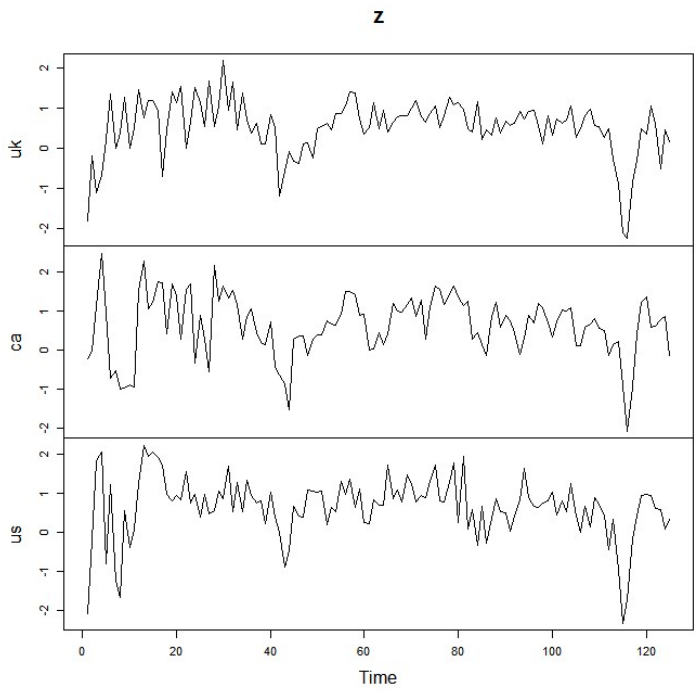
ونرسم الناتج



المتسلسلات غير مستقرة لذلك نأخذ التفريق الأول

```
> z=diffM(gdp)
> z=z*100
```

ضربنا في 100 لضمان عدم وجود تفرد *singularity* في مصفوفات التقدير



نلاحظ ان المتسلسلات تبدو مستقرة. سوف نستخدم الدالة VARorder في MTS للتعرف على درجة النموذج

```
> VARorder(z)
selected order: aic =  2
selected order: bic =  1
selected order: hq =  1
Summary table:
       p     AIC     BIC      HQ     M(p) p-value
 [1,]  0 -3.3250 -3.3250 -3.3250   0.0000  0.0000
 [2,]  1 -4.2520 -4.0484 -4.1693 115.1329  0.0000
 [3,]  2 -4.3333 -3.9260 -4.1678  23.5389  0.0051
 [4,]  3 -4.2926 -3.6817 -4.0444  10.4864  0.3126
 [5,]  4 -4.2661 -3.4516 -3.9352  11.5767  0.2382
 [6,]  5 -4.1508 -3.1326 -3.7372   2.7406  0.9737
 [7,]  6 -4.0801 -2.8583 -3.5838   6.7822  0.6598
 [8,]  7 -3.9869 -2.5615 -3.4079   4.5469  0.8719
 [9,]  8 -4.1260 -2.4969 -3.4642  24.4833  0.0036
```

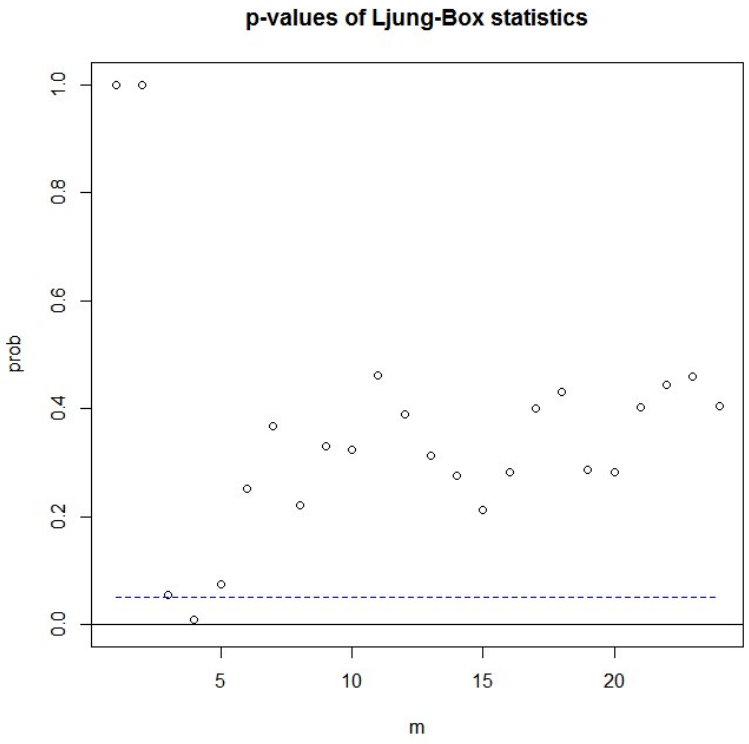
```
[10,]  9 -4.0586 -2.2259 -3.3141   6.4007  0.6992
[11,] 10 -3.9683 -1.9320 -3.1411   4.3226  0.8889
[12,] 11 -3.9726 -1.7326 -3.0626  11.4922  0.2435
[13,] 12 -3.9872 -1.5436 -2.9945  11.8168  0.2238
[14,] 13 -4.0408 -1.3935 -2.9654  14.1266  0.1179
>
```

نلاحظ ان $p = 2$ و $q = 0$ اي $\mathbf{VARMA}(1,0)$ أو $\mathbf{VAR}(1)$ نقدر الآن معالم النموذج

```
> m1=VAR(z,2)
Constant term:
Estimates:  0.1258163 0.1231581 0.2895581
Std.Error:  0.07266338 0.07382941 0.0816888

AR coefficient matrix
AR( 1 )-matrix
      [,1]  [,2]   [,3]
[1,] 0.393 0.103 0.0521
[2,] 0.351 0.338 0.4691
[3,] 0.491 0.240 0.2356

standard error
       [,1]   [,2]   [,3]
[1,] 0.0934 0.0984 0.0911
[2,] 0.0949 0.1000 0.0926
[3,] 0.1050 0.1106 0.1024

AR( 2 )-matrix
        [,1]   [,2]     [,3]
[1,]  0.0566  0.106  0.01889
[2,] -0.1914 -0.175 -0.00868
[3,] -0.3120 -0.131  0.08531
```

```
standard error
       [,1]   [,2]   [,3]
[1,] 0.0924 0.0876 0.0938
[2,] 0.0939 0.0890 0.0953
[3,] 0.1038 0.0984 0.1055

Residuals cov-mtx:
           [,1]       [,2]       [,3]
[1,] 0.28244420 0.02654091 0.07435286
[2,] 0.02654091 0.29158166 0.13948786
[3,] 0.07435286 0.13948786 0.35696571

det(SSE) =  0.02258974
AIC =  -3.502259
BIC =  -3.094982
HQ  =  -3.336804

> names(m1)
 [1] "data"      "cnst"      "order"     "coef"      "aic"
"bic"
 [7] "hq"        "residuals" "secoef"    "Sigma"     "Phi"
"Ph0"
> resi=m1$residuals
Ljung-Box Statistics:
          m       Q(m)     df    p-value
 [1,]   1.000     0.816  -9.000     1.00
 [2,]   2.000     3.978   0.000     1.00
 [3,]   3.000    16.665   9.000     0.05
 [4,]   4.000    35.122  18.000     0.01
 [5,]   5.000    38.189  27.000     0.07
 [6,]   6.000    41.239  36.000     0.25
```

```
 [7,]   7.000    47.621  45.000     0.37
 [8,]   8.000    61.677  54.000     0.22
 [9,]   9.000    67.366  63.000     0.33
[10,]  10.000    76.930  72.000     0.32
[11,]  11.000    81.567  81.000     0.46
[12,]  12.000    93.112  90.000     0.39
[13,]  13.000   105.327  99.000     0.31
[14,]  14.000   116.279 108.000     0.28
[15,]  15.000   128.974 117.000     0.21
[16,]  16.000   134.704 126.000     0.28
[17,]  17.000   138.552 135.000     0.40
[18,]  18.000   146.256 144.000     0.43
[19,]  19.000   162.418 153.000     0.29
[20,]  20.000   171.948 162.000     0.28
[21,]  21.000   174.913 171.000     0.40
[22,]  22.000   182.056 180.000     0.44
[23,]  23.000   190.276 189.000     0.46
[24,]  24.000   202.141 198.000     0.41
> mq(resi,adj=18)
```



>

ويكون النموذج المبدئي

$$\begin{bmatrix} \hat{\mathbf{Z}}_{t,uk} \\ \hat{\mathbf{Z}}_{t,ca} \\ \hat{\mathbf{Z}}_{t,us} \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\mu}}_{uk} \\ \hat{\boldsymbol{\mu}}_{ca} \\ \hat{\boldsymbol{\mu}}_{us} \end{bmatrix} + \begin{bmatrix} 0.393 & 0.103 & 0.0521 \\ 0.351 & 0.338 & 0.4691 \\ 0.491 & 0.240 & 0.2356 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{Z}}_{t-1,uk} \\ \hat{\mathbf{Z}}_{t-1,ca} \\ \hat{\mathbf{Z}}_{t-1,us} \end{bmatrix} + \begin{bmatrix} 0.000 & 0 & 0 \\ -0.197 & 0 & 0 \\ -0.301 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{Z}}_{t-2,uk} \\ \hat{\mathbf{Z}}_{t-2,ca} \\ \hat{\mathbf{Z}}_{t-2,us} \end{bmatrix} + \hat{\mathbf{a}}_t,$$

$$\{\hat{\mathbf{a}}_t\} \sim WN\left(\mathbf{0}, \hat{\boldsymbol{\Sigma}}_a\right)$$

حيث

$$\hat{\boldsymbol{\mu}} = \begin{bmatrix} \hat{\boldsymbol{\mu}}_{uk} \\ \hat{\boldsymbol{\mu}}_{ca} \\ \hat{\boldsymbol{\mu}}_{us} \end{bmatrix} = \begin{bmatrix} 0.1258163 & 0.1231581 & 0.2895581 \end{bmatrix}'$$

$$s.e\left(\hat{\boldsymbol{\mu}}\right) = \begin{bmatrix} 0.0726633 & 0.0738294 & 0.0816888 \end{bmatrix}$$

$$\hat{\mathbf{\Phi}}_1 = \begin{bmatrix} 0.393 & 0.103 & 0.0521 \\ 0.351 & 0.338 & 0.4691 \\ 0.491 & 0.240 & 0.2356 \end{bmatrix}$$

$$s.e\left(\hat{\mathbf{\Phi}}_1\right) = \begin{bmatrix} 0.0790 & 0.0686 & 0.0000 \\ 0.0921 & 0.0875 & 0.0913 \\ 0.1027 & 0.0963 & 0.1023 \end{bmatrix}$$

$$\hat{\mathbf{\Phi}}_2 = \begin{bmatrix} 0.000 & 0 & 0 \\ -0.197 & 0 & 0 \\ -0.301 & 0 & 0 \end{bmatrix}$$

$$s.e\left(\hat{\mathbf{\Phi}}_2\right) = \begin{bmatrix} 0.0000 & 0 & 0 \\ 0.0921 & 0 & 0 \\ 0.1008 & 0 & 0 \end{bmatrix}$$

$$\hat{\Sigma}_a = \begin{bmatrix} 0.29003669 & 0.01803456 & 0.07055856 \\ 0.01803456 & 0.30802503 & 0.14598345 \\ 0.07055856 & 0.14598345 & 0.36268779 \end{bmatrix}$$

$$\hat{\mathbf{\Gamma}}(0) = \begin{array}{c} \\ uk \\ ca \\ us \end{array} \begin{array}{c} \begin{array}{ccc} uk & ca & us \end{array} \\ \begin{bmatrix} 0.2924 & 0.0182 & 0.0711 \\ 0.0182 & 0.3084 & 0.1472 \\ 0.0711 & 0.1472 & 0.3657 \end{bmatrix} \end{array}$$

الخطوة التالية نقوم بها لتبسيط refine النموذج وذلك بإزاحة المعالم الغير معنوية

```
> m2=refVAR(m1,thres=1.96)
Constant term:
```

```
Estimates:  0.1628247 0 0.2827525
Std.Error:  0.06814101 0 0.07972864

AR coefficient matrix
AR( 1 )-matrix
      [,1]  [,2]  [,3]
[1,] 0.467 0.207 0.000
[2,] 0.334 0.270 0.496
[3,] 0.468 0.225 0.232

standard error
       [,1]   [,2]   [,3]
[1,] 0.0790 0.0686 0.0000
[2,] 0.0921 0.0875 0.0913
[3,] 0.1027 0.0963 0.1023


AR( 2 )-matrix
       [,1] [,2] [,3]
[1,]  0.000    0    0
[2,] -0.197    0    0
[3,] -0.301    0    0


standard error
       [,1] [,2] [,3]
[1,] 0.0000    0    0
[2,] 0.0921    0    0
[3,] 0.1008    0    0
```

```
Residuals cov-mtx:
           [,1]       [,2]       [,3]
[1,] 0.29003669 0.01803456 0.07055856
[2,] 0.01803456 0.30802503 0.14598345
[3,] 0.07055856 0.14598345 0.36268779


det(SSE) =  0.02494104
AIC =  -3.531241
BIC =  -3.304976
HQ  =  -3.439321
>
```

الإختبارات التفحصية للنموذج المقترح:

نستخدم الدالة **MTSdiag** لتفحص و إختبار فيما إذا كان النموذج مناسب

```
> MTSdiag(m2,adj=12)
[1] "Covariance matrix:"
       uk     ca     us
uk 0.2924 0.0182 0.0711
ca 0.0182 0.3084 0.1472
us 0.0711 0.1472 0.3657

CCM at lag:  0
       [,1]   [,2]  [,3]
[1,] 1.0000 0.0605 0.218
[2,] 0.0605 1.0000 0.438
[3,] 0.2175 0.4382 1.000
Simplified matrix:
CCM at lag:  1
. . .
. . .
```

. . .

CCM at lag: 2

. . .

. . .

. . .

.

.

CCM at lag: 23

. . .

. . .

. . .

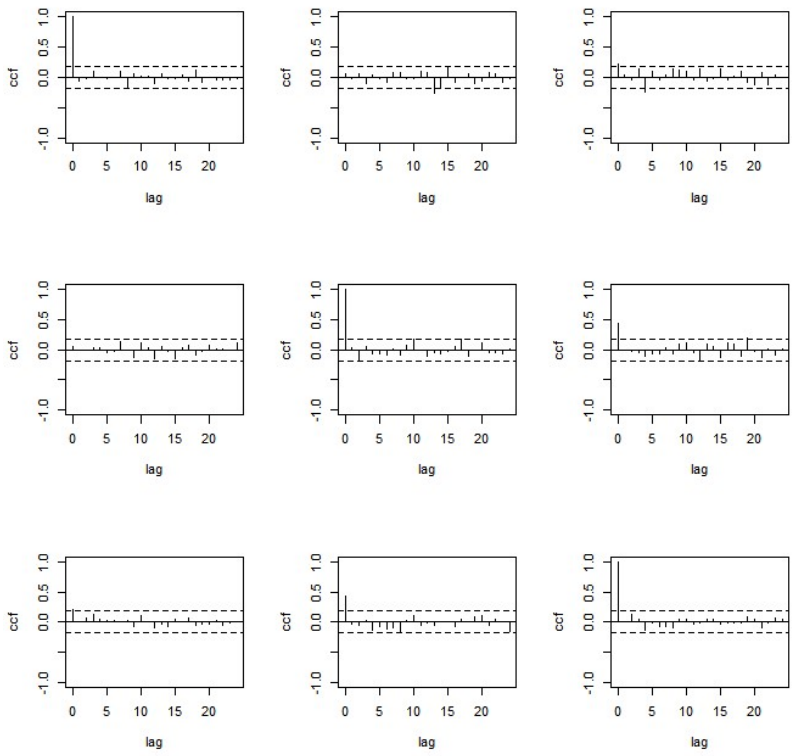
CCM at lag: 24

. . .

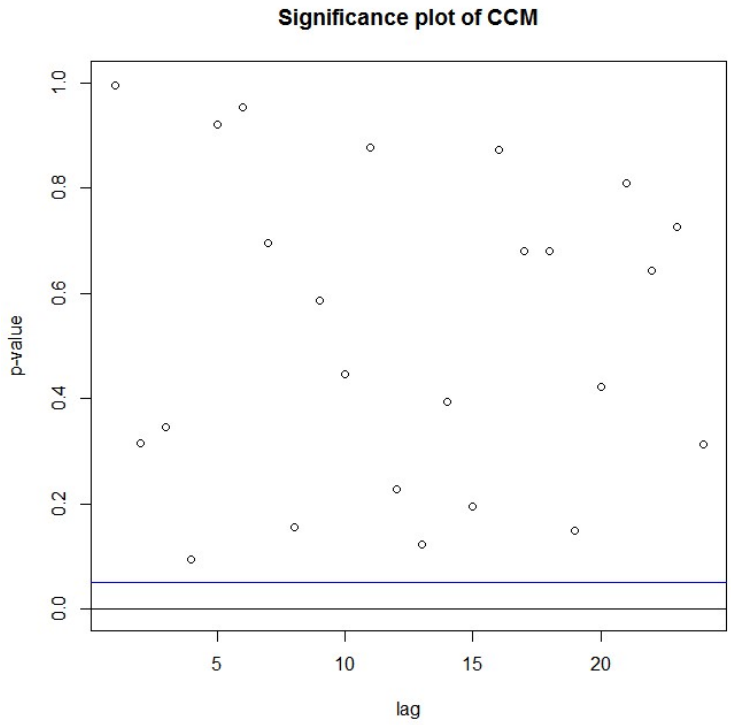
. . .

. . .

Hit Enter for p-value plot of individual ccm:



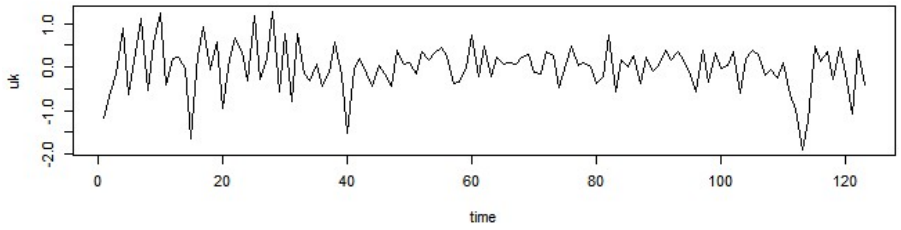
Hit Enter to compute MQ-statistics:

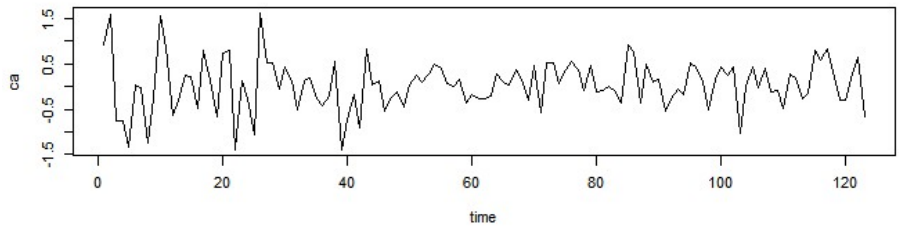


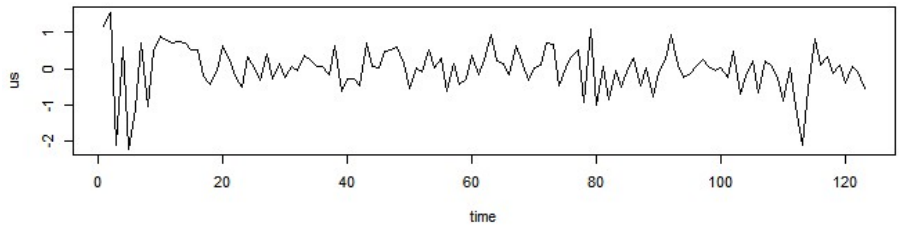
Ljung-Box Statistics:

	m	Q(m)	df	p-value
[1,]	1.00	1.78	-3.00	1.00
[2,]	2.00	12.41	6.00	1.00
[3,]	3.00	22.60	15.00	0.09
[4,]	4.00	37.71	24.00	0.04
[5,]	5.00	41.65	33.00	0.14
[6,]	6.00	44.95	42.00	0.35
[7,]	7.00	51.50	51.00	0.45
[8,]	8.00	64.87	60.00	0.31
[9,]	9.00	72.50	69.00	0.36
[10,]	10.00	81.58	78.00	0.37
[11,]	11.00	86.12	87.00	0.51
[12,]	12.00	98.08	96.00	0.42

```
[13,]  13.00    112.31  105.00     0.30
[14,]  14.00    121.89  114.00     0.29
[15,]  15.00    134.58  123.00     0.22
[16,]  16.00    139.16  132.00     0.32
[17,]  17.00    145.85  141.00     0.37
[18,]  18.00    152.56  150.00     0.43
[19,]  19.00    165.91  159.00     0.34
[20,]  20.00    175.22  168.00     0.34
[21,]  21.00    180.56  177.00     0.41
[22,]  22.00    187.40  186.00     0.46
[23,]  23.00    193.78  195.00     0.51
[24,]  24.00    204.65  204.00     0.47
Hit Enter to obtain residual plots:
```







تمرين: أكتب شكل النموذج بعد التبسيط.

إيجاد التنبؤات للنموذج

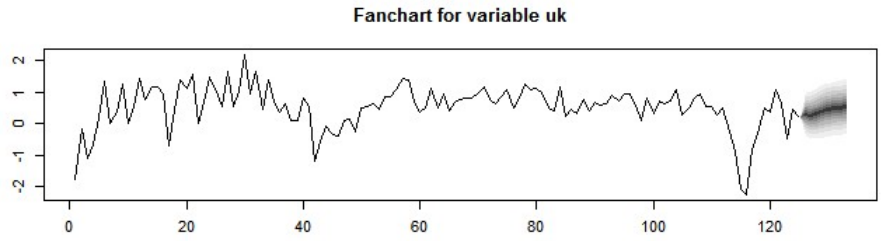
سوف نستخدم الدالة **VARpred** لإيجاد تنبؤات لـ $\ell = 8$ خطوات للأمام

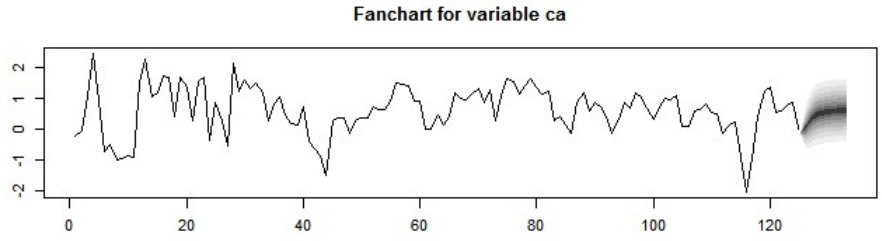
```
> VARpred(m1,8)
orig  125
Forecasts at origin:  125
         uk      ca     us
[1,] 0.3129 0.05166 0.1660
[2,] 0.2647 0.31687 0.4889
[3,] 0.3143 0.48231 0.5205
[4,] 0.3839 0.53053 0.5998
[5,] 0.4412 0.56978 0.6297
[6,] 0.4799 0.59478 0.6530
[7,] 0.5068 0.60967 0.6630
[8,] 0.5247 0.61689 0.6688
Standard Errors of predictions:
       [,1]   [,2]   [,3]
[1,] 0.5315 0.5400 0.5975
[2,] 0.5804 0.7165 0.7077
[3,] 0.6202 0.7672 0.7345
[4,] 0.6484 0.7785 0.7442
[5,] 0.6629 0.7824 0.7475
[6,] 0.6692 0.7838 0.7484
[7,] 0.6719 0.7842 0.7486
[8,] 0.6729 0.7843 0.7487
Root mean square errors of predictions:
       [,1]   [,2]   [,3]
[1,] 0.5461 0.5549 0.6140
```

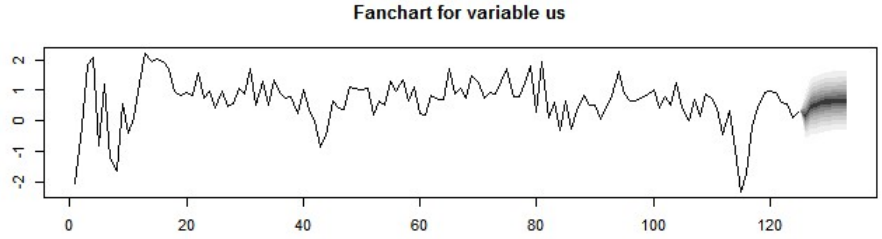
```
[2,] 0.6001 0.7799 0.7499
[3,] 0.6365 0.7879 0.7456
[4,] 0.6601 0.7832 0.7484
[5,] 0.6689 0.7841 0.7488
[6,] 0.6719 0.7844 0.7488
[7,] 0.6730 0.7844 0.7487
[8,] 0.6734 0.7844 0.7487
>
```

سوف نستخدم مكتبة **forecast** للنمذجة و التنبؤ للبيانات السابقة ويترك للقارئ للمقارنة بينهما

```
> library(forecast)
> library(ggplot2)
> library(ggfortify)
> library(vars)
> m3 <- VAR(z, p=2, type="const")
> m3pre <- predict(m3, n.ahead=8,ci=0.95)
> fanchart(m3pre)
>
```








```
> m3 <- VAR(z, p=2, type="const")
> m3pre <- predict(m3, n.ahead=8,ci=0.95)
> fanchart(m3pre)
> summary(m3)

VAR Estimation Results:
=========================
Endogenous variables: uk, ca, us
Deterministic variables: const
Sample size: 123
Log Likelihood: -290.487
Roots of the characteristic polynomial:
0.6136 0.6136 0.356 0.3321 0.3321 0.03676
Call:
```

```
VAR(y = z, p = 2, type = "const")

Estimation results for equation uk:
===================================
uk = uk.l1 + ca.l1 + us.l1 + uk.l2 + ca.l2 + us.l2 + const

      Estimate Std. Error t value Pr(>|t|)
uk.l1  0.39307    0.09342   4.208 5.11e-05 ***
ca.l1  0.10311    0.09838   1.048    0.297
us.l1  0.05214    0.09113   0.572    0.568
uk.l2  0.05660    0.09237   0.613    0.541
ca.l2  0.10552    0.08756   1.205    0.231
us.l2  0.01889    0.09382   0.201    0.841
const  0.12582    0.07266   1.731    0.086 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1


Residual standard error: 0.5473 on 116 degrees of freedom
Multiple R-Squared: 0.3829,     Adjusted R-squared: 0.3509
F-statistic: 11.99 on 6 and 116 DF,  p-value: 1.907e-10


Estimation results for equation ca:
===================================
ca = uk.l1 + ca.l1 + us.l1 + uk.l2 + ca.l2 + us.l2 + const

       Estimate Std. Error t value Pr(>|t|)
uk.l1  0.351314   0.094917   3.701 0.000330 ***
ca.l1  0.338142   0.099963   3.383 0.000979 ***
us.l1  0.469094   0.092589   5.066 1.55e-06 ***
```

```
uk.l2 -0.191350   0.093856  -2.039 0.043747 *
ca.l2 -0.174833   0.088964  -1.965 0.051780 .
us.l2 -0.008678   0.095326  -0.091 0.927624
const  0.123158   0.073829   1.668 0.097984 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1


Residual standard error: 0.556 on 116 degrees of freedom
Multiple R-Squared: 0.5238,     Adjusted R-squared: 0.4992
F-statistic: 21.27 on 6 and 116 DF,  p-value: < 2.2e-16


Estimation results for equation us:
===================================
us = uk.l1 + ca.l1 + us.l1 + uk.l2 + ca.l2 + us.l2 + const

      Estimate Std. Error t value Pr(>|t|)
uk.l1  0.49070    0.10502   4.672 8.08e-06 ***
ca.l1  0.24000    0.11060   2.170 0.032053 *
us.l1  0.23564    0.10245   2.300 0.023226 *
uk.l2 -0.31196    0.10385  -3.004 0.003265 **
ca.l2 -0.13118    0.09843  -1.333 0.185259
us.l2  0.08531    0.10547   0.809 0.420253
const  0.28956    0.08169   3.545 0.000568 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1


Residual standard error: 0.6152 on 116 degrees of freedom
```

```
Multiple R-Squared: 0.3606,     Adjusted R-squared: 0.3276
F-statistic:  10.9 on 6 and 116 DF,  p-value: 1.325e-09



Covariance matrix of residuals:
        uk      ca      us
uk 0.29949 0.02814 0.07884
ca 0.02814 0.30918 0.14791
us 0.07884 0.14791 0.37851

Correlation matrix of residuals:
        uk      ca     us
uk 1.00000 0.09248 0.2342
ca 0.09248 1.00000 0.4324
us 0.23416 0.43236 1.0000


> fitted(m3)
               uk          ca          us
1   -0.126152688  0.35769170  0.55767599
2   -0.115813466  1.01674969  0.49520689
3    0.311217253  1.68419755  1.38859806
.
.
.
120  0.832302327  0.80898826  0.95763976
121  0.599422267  0.52452712  0.55610531
122  0.146020537  0.24536897  0.14184938
123  0.464762464  0.57918518  0.85372651
> plot(m3)
```

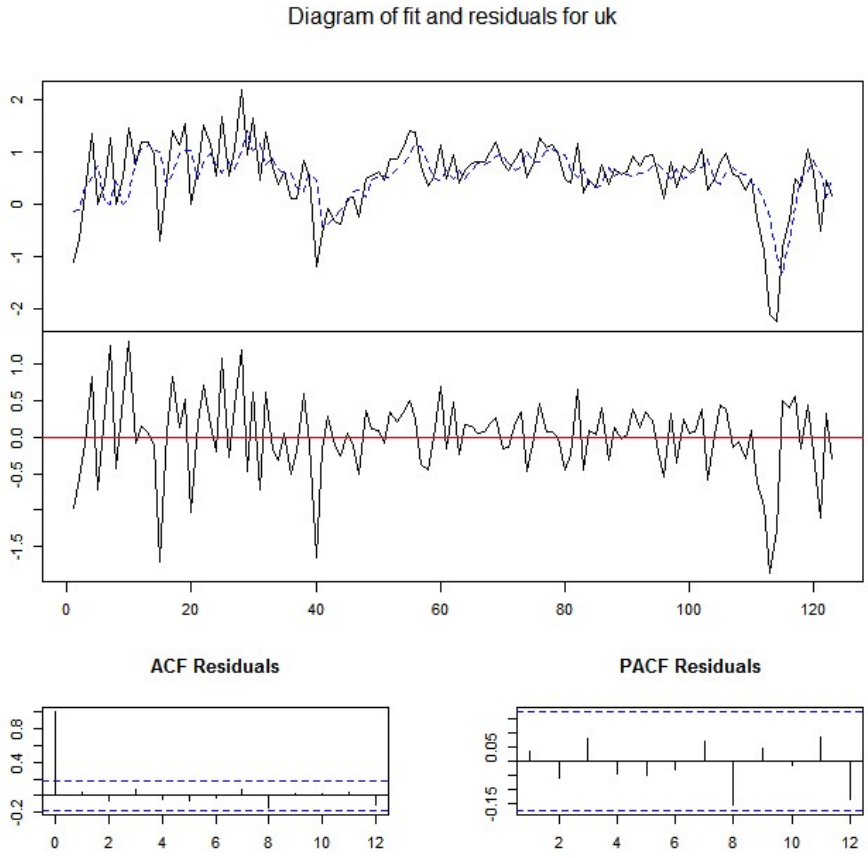


Diagram of fit and residuals for uk
ACF Residuals
PACF Residuals

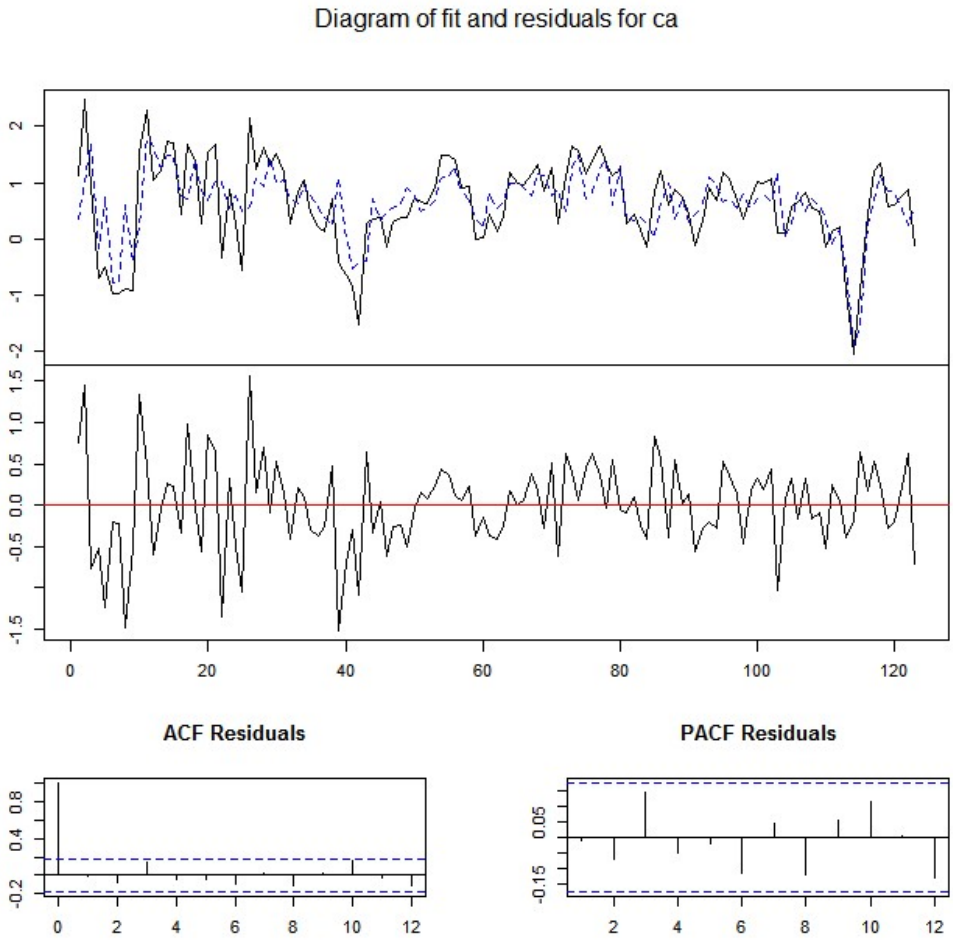
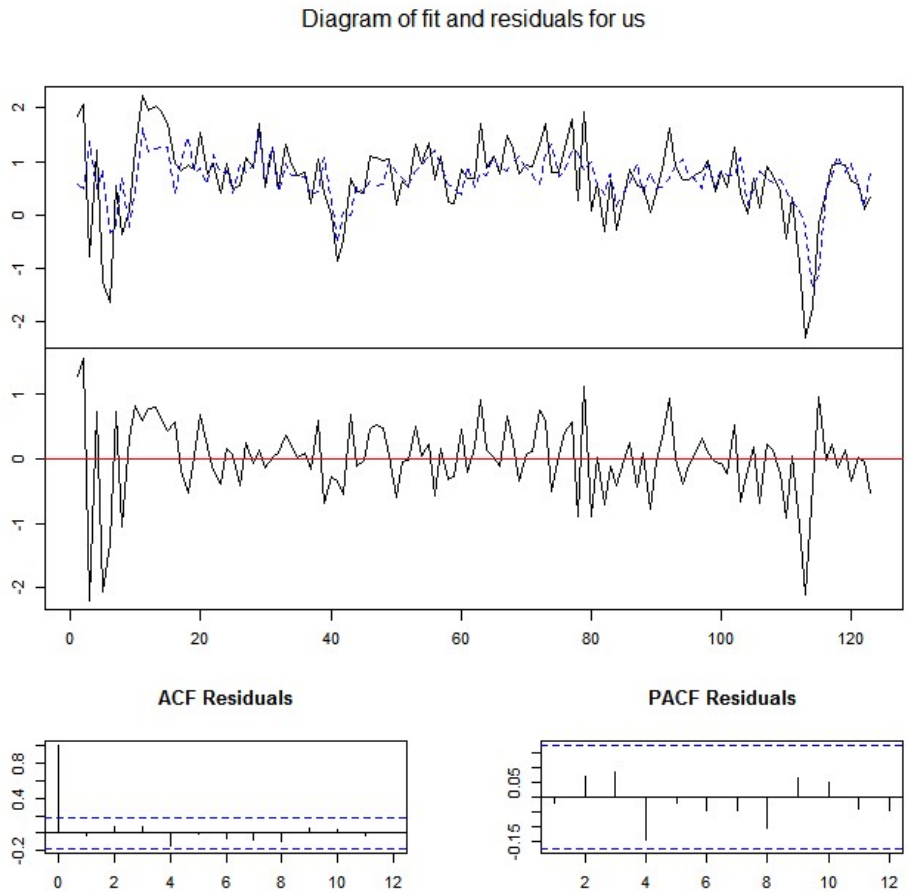
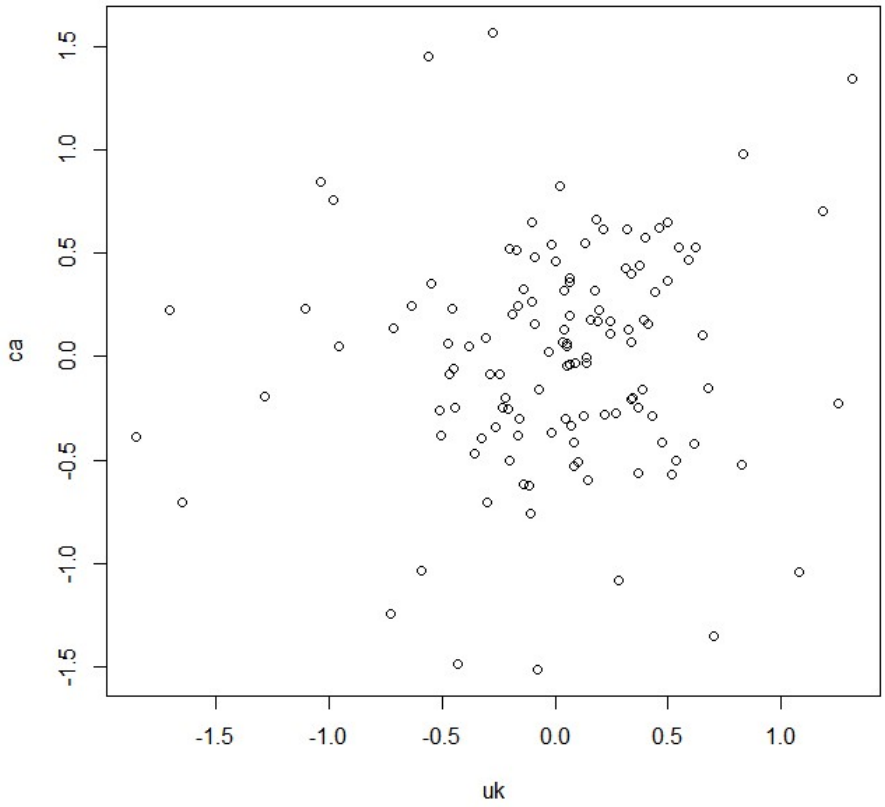


Diagram of fit and residuals for ca
2
1
0
-1
-2
1.5
1.0
0.5
0.0
-0.5
-1.5
0
20
40
60
80
100
120

ACF Residuals
PACF Residuals
0.8
0.4
-0.2
0.05
-0.15
0
2
4
6
8
10
12




```
> plot(residuals(mmm))
>
```


ca
uk
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-1.5
-1.0
-0.5
0.0
0.5
1.0

الفصل الرابع:

نماذج فضاء الحالة ومرشح كالمن

State Space Models and Kalman Filtering

في نماذج فضاء الحالة يكون لدينا عملية عشوائية $\{X_t\}$ لايمكن مشاهدتها أو قياسها وعملية عشوائية يمكن مشاهدتها وقياسها $\{Y_t\}$ (أنظر الجزء الأول من الكتاب حيث $\{Z_t\}$ يمكن مشاهدتها وقياسها ولكن $\{a_t\}$ لايمكن مشاهدتها أو قياسها). نفترض أن العملية $\{Y_t\}$ مرتبطة بعلاقة خطية أو دالة خطية بالعملية $\{X_t\}$ مع ضجة مضافة ويمكن إعتبار أن الدلة الخطية تتغير مع الزمن. هدفنا هنا هو إيجاد أفضل مقدر خطي لـ X_t بدلالة $\{Y_s, s \le t\}$.

العديد من نماذج المتسلسلات الزمنية مثل عمليات $ARMA(p,q)$ يمكن نمذجتها على شكل فضاء الحالة إذا مثلت على شكل متجهات عشوائية $\mathbf{X}_t \in \mathbb{R}^k$ و $\mathbf{Y}_t \in \mathbb{R}^m$.

نموذج فضاء الحالة متعدد المتغيرات يعرف بمعادلة الحالة *State Equation*

$$\mathbf{X}_{t+1} = \mathbf{A}_t \mathbf{X}_t + \mathbf{B}_t \boldsymbol{\varepsilon}_{t+1} \in \mathbb{R}^k \qquad (1)$$

والذي يصف الإعتماد الزمني للحالة $\mathbf{X}_t \in \mathbb{R}^k$ و معادلة المشاهدة *Observation Equation*

$$\mathbf{Y}_t = \mathbf{C}_t \mathbf{X}_t + \boldsymbol{\eta}_t \in \mathbb{R}^m \qquad (2)$$

سوف نفترض أن $(\mathbf{A}_t)$ و $(\mathbf{B}_t)$ و $(\mathbf{C}_t)$ متتابعات من المتجهات المعروفة و $\{\boldsymbol{\varepsilon}_t\}$ و $\{\boldsymbol{\eta}_t\}$ عمليات ضجة بيضاء لايوجد بينهما ترابط بحيث $\boldsymbol{\varepsilon}_t \sim WN(\mathbf{0},\mathbf{Q}_t)$ و $\boldsymbol{\eta}_t \sim WN(\mathbf{0},\mathbf{R}_t)$ (لاحظ أن $\mathbf{Q}_t = \mathrm{cov}(\boldsymbol{\varepsilon}_t) = E(\boldsymbol{\varepsilon}_t\boldsymbol{\varepsilon}_t')$ وكذلك $\mathbf{R}_t = \mathrm{cov}(\boldsymbol{\eta}_t) = E(\boldsymbol{\eta}_t\boldsymbol{\eta}_t')$ وسوف نفترض أيضا أن $\mathbf{X}_0$ و $\boldsymbol{\varepsilon}_t$ و $\boldsymbol{\eta}_t$ لقيم $t \geq 1$ غير مترابطة (يقال أن المتجهان العشوائيان $\mathbf{W} \in \mathbb{R}^p$ و $\mathbf{V} \in \mathbb{R}^q$ غيرمترابطة إذا حققا العلاقة $E\left(\left(\mathbf{W}-E(\mathbf{W})\right)\left(\mathbf{V}-E(\mathbf{V})\right)'\right) = \mathbf{0}$). نقول أن المتسلسلة الزمنية $\{Y_t\}$ لها تمثيل فضاء حالة إذا حققت العلاقات (1) و (2).

<u>مثال 1:</u>

لتكن $\{\eta_t\} \in \mathbb{R}$ متسلسلة ضجة بيضاء ولتكن

$$Y_t = \mu_t + \eta_t$$

مع إنجراف خطي

$$\mu_t = a + bt$$

هذا النموذج البسيط يمكن تمثيلة في فضاء الحالة كالتالي:

عرف متجه الحالة $\mathbf{X}_t$ كالتالي

$$\mathbf{X}_t = \begin{pmatrix} \mu_t \\ 1 \end{pmatrix}$$

وضع

$$\mathbf{A} = \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix}$$

ومن العلاقة $\mu_{t+1} = \mu_t + b$ نحصل على معادلة الحالة

$$\mathbf{X}_{t+1} = \begin{pmatrix} \mu_{t+1} \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & b \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mu_t \\ 1 \end{pmatrix} = \mathbf{A}\mathbf{X}_t$$

وبوضع $\mathbf{C} = \begin{pmatrix} 1 & 0 \end{pmatrix}$ تكون معادلة المشاهدات

$$Y_t = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} \mu_t \\ 1 \end{pmatrix} + \eta_t = \mathbf{C}\mathbf{X}_t + \eta_t$$

لاحظ أن الحالة $\mathbf{X}_t$ غير عشوائية لأن $\mathbf{B}_t = \mathbf{0}$ كما أن هذا النموذج لايتغير مع الزمن لأن المصفوفات $\mathbf{A}$ و $\mathbf{B} = \mathbf{B}_t$ و $\mathbf{C}$ لاتعتمد على الزمن.

<u>مثال 2:</u>

تمثيل عملية $AR(p)$ على شكل فضاء الحالة

$$z_t = \phi_1 z_{t-1} + \cdots + \phi_p z_{t-p} + a_t, \quad a_t \sim WN\left(0, \sigma^2\right)$$

متجه الحالة

$$\mathbf{X}_t = \left(z_t, z_{t-1}, ..., z_{t-p+1}\right)'$$

وبتعريف المصفوفة ذات الأبعاد $p \times p$

$$\mathbf{A} = \begin{pmatrix} \phi_1 & \phi_2 & \cdots & \phi_{p-1} & \phi_p \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}$$

والمصفوفات ذات الأبعاد $p \times 1$

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}' = \mathbf{C}'$$

نحصل على معادلة الحالة

$$\mathbf{X}_{t+1} = \mathbf{A}\mathbf{X}_t + \mathbf{B}\boldsymbol{\varepsilon}_{t+1}$$

ومعادلة المشاهدات

$$Y_t = \mathbf{C}\mathbf{X}_t$$

تمرين: تحقق أن تمثيل فضاء الحالة يؤدي للنموذج الأصلي.

<u>مثال 3:</u>

تمثيل عملية $MA(q)$ على شكل فضاء الحالة

$$z_t = a_t + \theta_1 a_{t-1} + \cdots + \theta_q a_{t-q}, \quad a_t \sim WN\left(0, \sigma^2\right)$$

نعرف متجه الحالة غير المشاهد *non observable*

$$\mathbf{X}_t = \left(\varepsilon_t, \varepsilon_{t-1}, ..., \varepsilon_{t-q}\right)' \in \mathbb{R}^{q+1}$$

والمصفوفة ذات الأبعاد $(q+1)\times(q+1)$

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix}$$

والمصفوفة ذات الأبعاد $1\times(q+1)$

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}'$$

والمصفوفة ذات الأبعاد $(q+1)\times 1$

$$\mathbf{C} = \begin{pmatrix} 1 & b_1 & \cdots & b_q \end{pmatrix}$$

فنحصل على معادلة الحالة

$$\mathbf{X}_{t+1} = \mathbf{A}\mathbf{X}_t + \mathbf{B}\boldsymbol{\varepsilon}_{t+1}$$

ومعادلة المشاهدات

$$Y_t = \mathbf{C}\mathbf{X}_t$$

تمرين: تحقق أن تمثيل فضاء الحالة يؤدي للنموذج الأصلي.

مثال 4:

تمثيل فضاء الحالة لعملية $ARMA(p,q)$

$$z_t = \phi_1 z_{t-1} + \cdots + \phi_p z_{t-p} + a_t + \theta_1 a_{t-1} + \cdots + \theta_q a_{t-q}, \quad a_t \sim WN\left(0,\sigma^2\right)$$

متجه الحالة

$$\mathbf{X}_t = \left(z_t, z_{t-1}, ..., z_{t-p+1}, \varepsilon_t, \varepsilon_{t-1}, ..., \varepsilon_{t-q+1}\right)' \in \mathbb{R}^{p+q}$$

والمصفوفة ذات الأبعاد $(p+q)\times(p+q)$

$$\mathbf{A} = \begin{pmatrix} \phi_1 & \phi_2 & \cdots & \phi_{p-1} & \phi_p & \theta_1 & \theta_2 & \cdots & \theta_{q-1} & \theta_q \\ 1 & 0 & \cdots & 0 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 & \vdots & & & & \vdots \\ \vdots & & \ddots & & \vdots & \vdots & & & & \vdots \\ 0 & \cdots & 0 & 1 & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & \vdots & 1 & 0 & \cdots & \cdots & 0 \\ \vdots & & & & \vdots & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & \vdots & \vdots & & \ddots & & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 \end{pmatrix}$$

والمصفوفة ذات الأبعاد $(p+q)\times 1$

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{pmatrix}'$$

حيث تأخذ القيمة 1 عند المدخل الأول و المدخل ذا الرتبة $(p+1)$

والمصفوفة ذات الأبعاد $1\times(p+q)$

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}$$

فنحصل على معادلة الحالة

$$\mathbf{X}_{t+1} = \mathbf{A}\mathbf{X}_t + \mathbf{B}\boldsymbol{\varepsilon}_{t+1}$$

ومعادلة المشاهدات

$$Y_t = \mathbf{C}\mathbf{X}_t$$

تمرين: تحقق أن تمثيل فضاء الحالة يؤدي للنموذج الأصلي.

مرشح كالمان The Kalman-Filter

المشكلة الأساسية في نماذج فضاء الحالة التي تنمذج على الشكل *State Equation*

$$\mathbf{X}_{t+1} = \mathbf{A}_t\mathbf{X}_t + \mathbf{B}_t\boldsymbol{\varepsilon}_{t+1} \in \mathbb{R}^k$$

Observation Equation

$$\mathbf{Y}_t = \mathbf{C}_t\mathbf{X}_t + \boldsymbol{\eta}_t \in \mathbb{R}^m$$

بحيث $\boldsymbol{\varepsilon}_t \sim WN(\mathbf{0},\mathbf{Q}_t)$ و $\boldsymbol{\eta}_t \sim WN(\mathbf{0},\mathbf{R}_t)$ و $\operatorname{cov}(\boldsymbol{\varepsilon}_t,\eta_t) = E(\boldsymbol{\varepsilon}_t\boldsymbol{\eta}_t') = \mathbf{0}$

هي في تقدير متجه الحالة غير الممكن مشاهدته $\mathbf{X}_t$.

لحسن الحظ يمكن إشتقاق مقدر لمتجه الحالة $\mathbf{X}_t$ بشكل تكراري من تقدير $\mathbf{X}_{t-1}$ مع آخر مشاهدة $\mathbf{Y}_t$ وتسمى هذه تكرارات كالمان ***Kalman Recursions*** . وفي هذه الحالة نحصل على طريقة تنبؤ موحدة لكل نموذج متسلسلة زمنية يمكن وضعه على شكل نموذج فضاء الحالة.

نرغب في حساب أفضل متنبئ خطي *Best Linear Prediction*

$$\hat{\mathbf{X}}_t = \mathbf{D}_1\mathbf{Y}_1 + \cdots + \mathbf{D}_t\mathbf{Y}_t \qquad (3)$$

لمتجه الحالة $\mathbf{X}_t$ بدلالة $\mathbf{Y}_1,\cdots,\mathbf{Y}_t$ بحيث أن المصفوفات ذات الأبعاد $k \times m$ تصغر متوسط مربع الخطأ

$$E\left(\left(\mathbf{X}_t - \hat{\mathbf{X}}_t\right)'\left(\mathbf{X}_t - \hat{\mathbf{X}}_t\right)\right) = E\left(\left(\mathbf{X}_t - \sum_{j=1}^{t}\mathbf{D}_j\mathbf{Y}_j\right)'\left(\mathbf{X}_t - \sum_{j=1}^{t}\mathbf{D}_j\mathbf{Y}_j\right)\right)$$

$$= \min_{\mathbf{D}_1',\ldots,\mathbf{D}_t'} E\left(\left(\mathbf{X}_t - \sum_{j=1}^{t}\mathbf{D}_j'\mathbf{Y}_j\right)'\left(\mathbf{X}_t - \sum_{j=1}^{t}\mathbf{D}_j'\mathbf{Y}_j\right)\right) \qquad (4)$$

<u>فرضية:</u>

إذا حقق المقدر $\hat{\mathbf{X}}_t$ لمتجه الحالة المعرف في العلاقة $\mathbf{Y}_t = \mathbf{C}_t\mathbf{X}_t + \boldsymbol{\eta}_t$ التالي:

$$E\left(\left(\mathbf{X}_t - \hat{\mathbf{X}}_t\right)\mathbf{Y}_s'\right) = \mathbf{0}, \qquad 1 \le s \le t$$

فإنه يصغر متوسط مربع الخطأ المعطى بالعلاقة (4).

لاحظ أن $E\left(\left(\mathbf{X}_t-\hat{\mathbf{X}}_t\right)\mathbf{Y}_s'\right)$ هي مصفوفة ذات أبعاد $k\times m$ نحصل عليها بضرب كل حد من $\mathbf{X}_t-\hat{\mathbf{X}}_t\in\mathbb{R}^k$ بكل حد من $\mathbf{Y}_s\in\mathbb{R}^m$.

مكسب أو زيادة كالمان Kalman Gain

ليكن $\hat{\mathbf{X}}_{t-1}$ أفضل متنبئ خطي للمتجه $\mathbf{X}_{t-1}$ بدلالة $\mathbf{Y}_1,\cdots,\mathbf{Y}_{t-1}$ عندئذ

$$\tilde{\mathbf{X}}_t=\mathbf{A}_{t-1}\hat{\mathbf{X}}_{t-1}$$

هو أفضل متنبئ خطي للمتجه $\mathbf{X}_t$ بدلالة $\mathbf{Y}_1,\cdots,\mathbf{Y}_{t-1}$ والذي يلاحظ من إبدال $\boldsymbol{\varepsilon}_t$ في معادلة الحالة بقيمتها المتوقعة $\mathbf{0}$ وكذلك فإن $\boldsymbol{\varepsilon}_t$ و $\mathbf{Y}_s$ غير مترابطة إذا كانت $s<t$ أي أن $E\left(\left(\mathbf{X}_t-\tilde{\mathbf{X}}_t\right)\mathbf{Y}_s'\right)=\mathbf{0},\quad 1\le s\le t-1$ والتي منها نحصل على

$$\tilde{\mathbf{Y}}_t=\mathbf{C}_t\tilde{\mathbf{X}}_t$$

كأفضل متنبئ خطي للمتجه $\mathbf{Y}_t$ معتمدا على $\mathbf{Y}_1,\cdots,\mathbf{Y}_{t-1}$ لأن

$$E\left(\left(\mathbf{Y}_t-\tilde{\mathbf{Y}}_t\right)\mathbf{Y}_s'\right)=E\left(\left(\mathbf{C}_t\left(\mathbf{X}_t-\tilde{\mathbf{X}}_t\right)+\boldsymbol{\eta}_t\right)\mathbf{Y}_s'\right)=\mathbf{0},\quad 1\le s\le t-1$$

لاحظ أن $\boldsymbol{\eta}_t$ و $\mathbf{Y}_s$ إذا كانت $s<t$.

لنعرف

$$\boldsymbol{\Delta}_t=E\left(\left(\mathbf{X}_t-\hat{\mathbf{X}}_t\right)\left(\mathbf{X}_t-\hat{\mathbf{X}}_t\right)'\right)$$

و

$$\tilde{\boldsymbol{\Delta}}_t=E\left(\left(\mathbf{X}_t-\tilde{\mathbf{X}}_t\right)\left(\mathbf{X}_t-\tilde{\mathbf{X}}_t\right)'\right)$$

مصفوفات التغاير *Covariance Matrices* للخطأ التقريبي فيكون

$$\tilde{\boldsymbol{\Delta}}_t = E\left(\left(\mathbf{A}_{t-1}\left(\mathbf{X}_{t-1}-\hat{\mathbf{X}}_{t-1}\right)+\mathbf{B}_{t-1}\boldsymbol{\varepsilon}_t\right)\left(\mathbf{A}_{t-1}\left(\mathbf{X}_{t-1}-\hat{\mathbf{X}}_{t-1}\right)+\mathbf{B}_{t-1}\boldsymbol{\varepsilon}_t\right)'\right)$$

$$= E\left(\mathbf{A}_{t-1}\left(\mathbf{X}_{t-1}-\hat{\mathbf{X}}_{t-1}\right)\right)\left(\mathbf{A}_{t-1}\left(\mathbf{X}_{t-1}-\hat{\mathbf{X}}_{t-1}\right)'\right)$$

$$+ E\left(\left(\mathbf{B}_{t-1}\boldsymbol{\varepsilon}_t\right)\left(\mathbf{B}_{t-1}\boldsymbol{\varepsilon}_t\right)'\right)$$

$$= \mathbf{A}_{t-1}\boldsymbol{\Delta}_{t-1}\mathbf{A}'_{t-1} + \mathbf{B}_{t-1}\mathbf{Q}_t\mathbf{B}'_{t-1}$$

لأن كل من $\boldsymbol{\varepsilon}_t$ و $\mathbf{X}_{t-1}-\hat{\mathbf{X}}_{t-1}$ غير مترابطة بالطبع. وبنفس الطريقة يمكن إثبات أن

$$E\left(\left(\mathbf{Y}_t-\tilde{\mathbf{Y}}_t\right)\left(\mathbf{Y}_t-\tilde{\mathbf{Y}}_t\right)'\right) = \mathbf{C}_t\tilde{\boldsymbol{\Delta}}_t\mathbf{C}'_t + \mathbf{R}_t$$

لنفترض أننا شاهدنا $\mathbf{Y}_1,\cdots,\mathbf{Y}_{t-1}$ وأننا تنبأنا المتجه $\mathbf{X}_t$ بواسطة $\tilde{\mathbf{X}}_t = \mathbf{A}_{t-1}\hat{\mathbf{X}}_{t-1}$.

لنفترض أننا نشاهد الآن $\mathbf{Y}_t$ فكيف يمكننا إستخدام هذه المعلومة الإضافية لتحسين التنبؤ $\tilde{\mathbf{X}}_t$ للمتجه $\mathbf{X}_t$ لهذا سوف نضيف المصفوفة $\mathbf{K}_t$ بحيث نحصل على أفضل تنبؤ $\hat{\mathbf{X}}_t$ مبني على $\mathbf{Y}_1,\cdots,\mathbf{Y}_t$ بحيث

$$\tilde{\mathbf{X}}_t + \mathbf{K}_t\left(\mathbf{Y}_t-\tilde{\mathbf{Y}}_t\right) = \hat{\mathbf{X}}_t$$

علينا أن نختار $\mathbf{K}_t$ بحيث $E\left(\left(\mathbf{X}_t-\hat{\mathbf{X}}_t\right)\mathbf{Y}'_s\right)=\mathbf{0},\quad 1\le s\le t$ وفي هذه الحالة تسمى $\mathbf{K}_t$ مكسب أو زيادة كالمان.

<u>فرضية:</u>

المصفوفة $\mathbf{K}_t$ هي حل للمعادلة

$$\mathbf{K}_t\left(\mathbf{C}_t\tilde{\boldsymbol{\Delta}}_t\mathbf{C}'_t + \mathbf{R}_t\right) = \tilde{\boldsymbol{\Delta}}_t\mathbf{C}'_t$$

وفي حالة كون المصفوفة $\mathbf{C}_t\tilde{\mathbf{\Delta}}_t\mathbf{C}_t' + \mathbf{R}_t$ قابلة للإنعكاس يكون

$$\mathbf{K}_t = \tilde{\mathbf{\Delta}}_t\mathbf{C}_t'\left(\mathbf{C}_t\tilde{\mathbf{\Delta}}_t\mathbf{C}_t' + \mathbf{R}_t\right)^{-1}$$

و أيضا يمكن إثبات أن

$$\mathbf{\Delta}_t = \tilde{\mathbf{\Delta}}_t - \mathbf{K}_t\mathbf{C}_t\tilde{\mathbf{\Delta}}_t$$

التكرار في مرشح كالمان يتم على خطوتين:

من $\mathbf{X}_{t-1}$ و $\mathbf{\Delta}_{t-1}$ نحسب خطوة التنبؤ *Prediction Step* أولا

$$\tilde{\mathbf{X}}_t = \mathbf{A}_{t-1}\hat{\mathbf{X}}_{t-1}$$

$$\tilde{\mathbf{Y}}_t = \mathbf{C}_t\tilde{\mathbf{X}}_t \qquad (5)$$

$$\tilde{\mathbf{\Delta}}_t = \mathbf{A}_{t-1}\mathbf{\Delta}_{t-1}\mathbf{A}_{t-1}' + \mathbf{B}_{t-1}\mathbf{Q}_t\mathbf{B}_{t-1}'$$

وفي خطوة التحديث *Updating Step* نحسب $\mathbf{K}_t$ والقيم المحدثة $\hat{\mathbf{X}}_t$ و $\mathbf{\Delta}_t$

$$\mathbf{K}_t = \tilde{\mathbf{\Delta}}_t\mathbf{C}_t'\left(\mathbf{C}_t\tilde{\mathbf{\Delta}}_t\mathbf{C}_t' + \mathbf{R}_t\right)^{-1}$$

$$\hat{\mathbf{X}}_t = \tilde{\mathbf{X}}_t + \mathbf{K}_t\left(\mathbf{Y}_t - \tilde{\mathbf{Y}}_t\right) \qquad (6)$$

$$\mathbf{\Delta}_t = \tilde{\mathbf{\Delta}}_t - \mathbf{K}_t\mathbf{C}_t\tilde{\mathbf{\Delta}}_t$$

طبعا المشكلة الآن هي في إختيار القيم الأولية $\tilde{\mathbf{X}}_1$ و $\tilde{\mathbf{\Delta}}_1$. في معظم التطبيقات توضع $\tilde{\mathbf{X}}_1 = \mathbf{0}$ و $\tilde{\mathbf{\Delta}}_1 = diag\left(\sigma^2\right)$ (المصفوفة القطرية بعناصر $\sigma^2 > 0$ والعدد σ^2 يعكس درجة عدم التأكد حول النموذج المقترح). الدراسات السابقة وكذلك المحاكاة أثبتت أن المقدرات $\hat{\mathbf{X}}_t$ غالبا لا تتأثر بالقيم الأولية $\tilde{\mathbf{X}}_1$ و $\tilde{\mathbf{\Delta}}_1$ إذا كانت t كبيرة.

بتكرار خطوة التنبؤ $\tilde{\mathbf{X}}_t = \mathbf{A}_{t-1}\hat{\mathbf{X}}_{t-1}$ للمقدر $\hat{\mathbf{X}}_t$ h مرة نحصل على الخطوة h لتنبؤ مرشح كالمان

$$\tilde{\mathbf{X}}_{t+h} = \mathbf{A}_{t+h-1}\hat{\mathbf{X}}_{t+h-1}, \quad h \geq 1$$

مع القيم الأولية $\tilde{\mathbf{X}}_{t+0} = \hat{\mathbf{X}}_t$ و تكون الخطوة h للتنبؤ بالمتجه $\mathbf{Y}_{t+h}$ هي

$$\tilde{\mathbf{Y}}_{t+h} = \mathbf{C}_{t+h}\tilde{\mathbf{X}}_{t+h}, \quad h \geq 1$$

<u>مثال 5:</u>

نموذج المشي العشوائي

$$Y_t = \mu + \varepsilon_t, \quad \varepsilon_t \sim WN\left(0, \sigma^2\right)$$

والذي يوضع على شكل فضاء الحالة بوضع $X_t = \mu$

معادلة الحالة

$$X_{t+1} = X_t$$

معادلة المشاهدة

$$Y_t = X_t + \varepsilon_t$$

هنا $A_t = C_t = 1$ و $B_t = 0$.

خطوة التنبؤ لمرشح كالمان

$$\tilde{X}_t = \hat{X}_{t-1}$$

$$\tilde{Y}_t = \tilde{X}_t$$

$$\tilde{\Delta}_t = \Delta_{t-1}$$

خطوة التحديث لمرشح كالمان

$$K_t = \frac{\Delta_{t-1}}{\Delta_{t-1} + \sigma^2}$$

$$\hat{X}_t = \hat{X}_{t-1} + K_t\left(Y_t - \hat{X}_{t-1}\right)$$

$$\Delta_t = \Delta_{t-1} - K_t \Delta_{t-1} = \Delta_{t-1} \frac{\sigma^2}{\Delta_{t-1} + \sigma^2}$$

لاحظ أن $\Delta_t = E\left(\left(X_t - \hat{X}_t\right)^2\right) \geq 0$ وهكذا

$$0 \leq \Delta_t = \Delta_{t-1} \frac{\sigma^2}{\Delta_{t-1} + \sigma^2} \leq \Delta_{t-1}$$

متتابعة محدودة ومتناقصة *Decreasing and Bounded Sequence* وبالتالي يكون

$$\Delta = \lim_{t\to\infty} \Delta_t = \Delta \frac{\sigma^2}{\Delta + \sigma^2} \Rightarrow \Delta = 0$$

هذا يعني أن متوسط مربع الخطأ $E\left(\left(X_t - \hat{X}_t\right)^2\right) = E\left(\left(\mu - \hat{X}_t\right)^2\right) \to 0$ أي

يؤول إلى الصفر مهما كان إختيار القيم الأولية $\tilde{X}_1$ و $\tilde{\Delta}_1$. أيضا لدينا $\lim_{t\to\infty} K_t = 0$

وهذا يعني أن المشاهدات الإضافية Y_t لاتضيف شيئا للمقدر $\hat{X}_t$ عندما تكون t كبيرة.

أخيرا نتحصل على متوسط مربع الخطأ للتنبؤ h خطوة $\tilde{Y}_{t+h}$ لـ Y_{t+h}

$$E\left(\left(Y_{t+h} - \tilde{Y}_{t+h}\right)^2\right) = E\left(\left(\mu + \varepsilon_{t+h} - \hat{X}_t\right)^2\right)$$
$$= E\left(\left(\mu - \hat{X}_t\right)^2\right) + E\left(\varepsilon_{t+h}^2\right) \overset{t\to\infty}{\to} 0$$

الحزمة dlm في R:

في كثير من المراجع وفي الحزمة R يمثل نموذج فضاء الحالة على الشكل:

معادلة الحالة

$$\mathbf{X}_t = \mathbf{G}_t \mathbf{X}_{t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim N_p(\mathbf{0}, \mathbf{W}_t), \quad \mathbf{X}_0 \sim N_p(\mathbf{m}_0, \mathbf{C}_0)$$

و معادلة المشاهدة

$$\mathbf{Y}_t = \mathbf{F}_t \mathbf{X}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim N_m(\mathbf{0}, \mathbf{V}_t)$$

حيث المصفوفات $\mathbf{G}_t$ و $\mathbf{F}_t$ غير معروفة وذات أبعاد $p \times p$ و $m \times p$ على التوالي و

$$E(\mathbf{v}_t \mathbf{w}_t') = \mathbf{0}, \forall t$$

<u>مقارنة بين التمثيلين:</u>

معادلة الحالة *State Equation*

$$\mathbf{X}_t = \mathbf{G}_t \mathbf{X}_{t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim N_p(\mathbf{0}, \mathbf{W}_t), \quad \mathbf{X}_0 \sim N_p(\mathbf{m}_0, \mathbf{C}_0)$$

$$\left(\mathbf{X}_{t+1} = \mathbf{A}_t \mathbf{X}_t + \mathbf{B}_t \boldsymbol{\varepsilon}_{t+1} \in \mathbb{R}^k \quad \boldsymbol{\varepsilon}_t \sim WN(\mathbf{0}, \mathbf{Q}_t) \right)$$

و معادلة المشاهدة *Observation Equation*

$$\mathbf{Y}_t = \mathbf{F}_t \mathbf{X}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim N_m(\mathbf{0}, \mathbf{V}_t)$$

$$\left(\mathbf{Y}_t = \mathbf{C}_t \mathbf{X}_t + \boldsymbol{\eta}_t \in \mathbb{R}^m \quad \boldsymbol{\eta}_t \sim WN(\mathbf{0}, \mathbf{R}_t) \right)$$

نلاحظ أن

$$\mathbf{C}_t = \mathbf{F}_t, \quad \mathbf{R}_t = \mathbf{V}_t, \quad \mathbf{A}_t = \mathbf{G}_t, \quad \mathbf{Q}_t = \mathbf{W}_t, \quad \mathbf{B}_t = \mathbf{I}$$

في الحزمة dlm تعين المصفوفات $\mathbf{C}_0, \mathbf{W}, \mathbf{G}, \mathbf{V}, \mathbf{F}$ والمتجه $\mathbf{m}_0$ وتخزن في dlm Object كالعناصر $C0, W, GG, V, FF$ و $m0$

<u>مثال لتوليد dlm Object:</u>

سوف نولد dlm Object لنموذج المشي العشوائي مع ضجة مضافة التالي

$$Y_t = \mu_t + v_t, \quad v_t \sim N(0,V)$$

$$\mu_t = \mu_{t-1} + w_t, \quad w_t \sim N(0,W)$$

سوف نفترض أن $V = 1.4$ و $W = 0.2$ وواضح أن $FF = GG = 1$ ونأخذ

$C0 = 10$ و $m0 = 0$

```
> rw = dlm(m0 = 0, C0 = 10, FF =1, V = 1.4, GG = 1, W =
0.2)
> unlist(rw)
  m0   C0   FF    V   GG    W
 0.0 10.0  1.0  1.4  1.0  0.2
> rw
$FF
     [,1]
[1,]    1

$V
     [,1]
[1,]  1.4

$GG
     [,1]
[1,]    1

$W
     [,1]
[1,]  0.2

$m0
[1] 0

$C0
```

```
     [,1]
[1,]   10

> is.dlm(rw)
[1] TRUE
>
```

نولد البيانات

```
> y = rnorm(100)
>
```

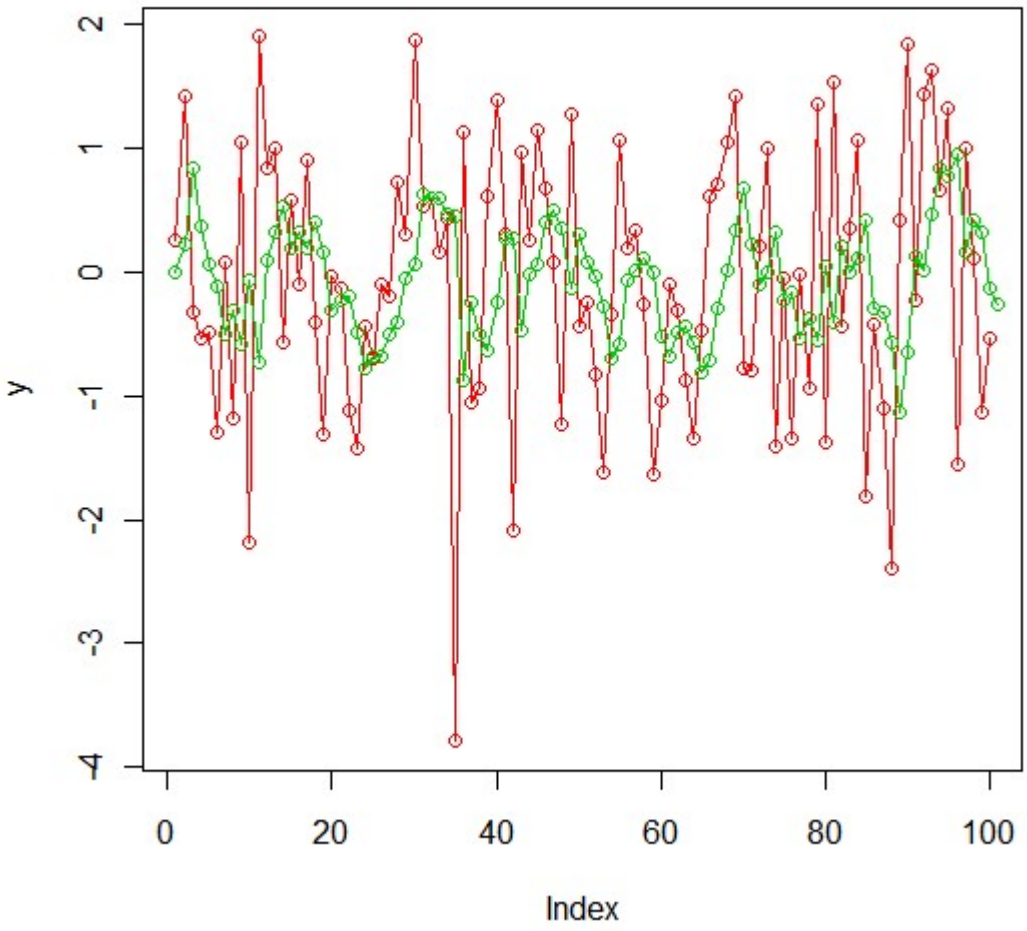
نطبق مرشح كالمان

```
> yfiltred = dlmFilter(y,rw)
> yfiltred
$y
[1]  0.26423211  1.42962164 -0.32765109 -0.53537696
….
[97]  0.99822551  0.11139582 -1.12928248 -0.53325193

$mod
$FF
     [,1]
[1,]    1

$V
     [,1]
[1,]  1.4

$GG
     [,1]
[1,]    1

$W
```

```
     [,1]
[1,]  0.2

$m0
[1] 0

$C0
     [,1]
[1,]   10


$m
[1]  0.000000000  0.232342029  0.837544270  0.379239833
…
[96]  0.951045739  0.164082162  0.425357595  0.327016585 -
0.129134229
[101] -0.255714422

$U.C
$U.C[[1]]
     [,1]
[1,]    1

$U.C[[2]]
     [,1]
[1,]    1
…..

$U.C[[99]]
     [,1]
[1,]    1
```

```
$U.C[[100]]
     [,1]
[1,]    1

$U.C[[101]]
     [,1]
[1,]    1


$D.C
            [,1]
[1,] 3.1622777
[2,] 1.1095199
…..
[99,] 0.6622058
[100,] 0.6622058
[101,] 0.6622058

$a
[1]  0.000000000  0.232342029  0.837544270  0.379239833
…
[96]  0.951045739  0.164082162  0.425357595  0.327016585 -
0.129134229

$U.R
$U.R[[1]]
     [,1]
[1,]    1

$U.R[[2]]
     [,1]
[1,]    1
```

```
…..

$U.R[[99]]
     [,1]
[1,]    1

$U.R[[100]]
     [,1]
[1,]    1



$D.R
            [,1]
[1,] 3.1937439
[2,] 1.1962585
…..
[99,] 0.7990723
[100,] 0.7990723

$f
[1]  0.000000000  0.232342029  0.837544270  0.379239833
…
[96]  0.951045739  0.164082162  0.425357595  0.327016585 -
0.129134229

attr(,"class")
[1] "dlmFiltered"
> plot(y, type='o', col= 2)
> lines(yfiltred$m, type='o', col= 3)
```

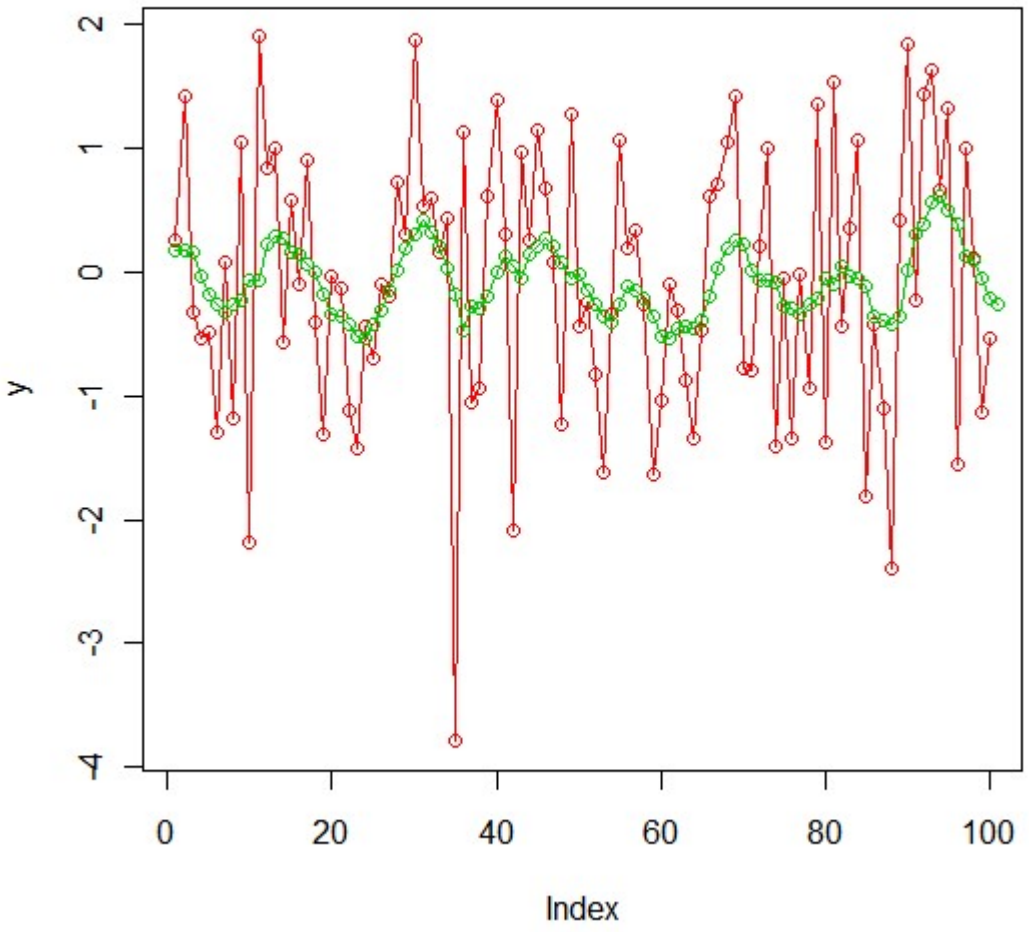


المتسلسلة الأصلية باللون الأحمر والمتسلسلة المرشحة باللون الأخضر

```
> ysmooth = dlmSmooth(yfiltred)
> ysmooth
$s
[1]  0.1710690921  0.1744904740  0.1650916219 -0.0249543761
…
[99] -0.0459585254 -0.2160662077 -0.2557144225

$U.S
```

```
$U.S[[1]]
     [,1]
[1,]    1

$U.S[[2]]
     [,1]
[1,]    1
…..

$U.S[[99]]
     [,1]
[1,]    1

$U.S[[100]]
     [,1]
[1,]    1

$U.S[[101]]
     [,1]
[1,]    1


$D.S
            [,1]
[1,] 0.7747213
[2,] 0.6484142
…..
[99,] 0.5474417
[100,] 0.5866726
[101,] 0.6622058

> plot(y, type='o', col= 2)
```

```
> lines(ysmooth$s, type='o', col =3)
>
```



المتسلسلة الأصلية باللون الأحمر والمتسلسلة الممهدة باللون الأخضر

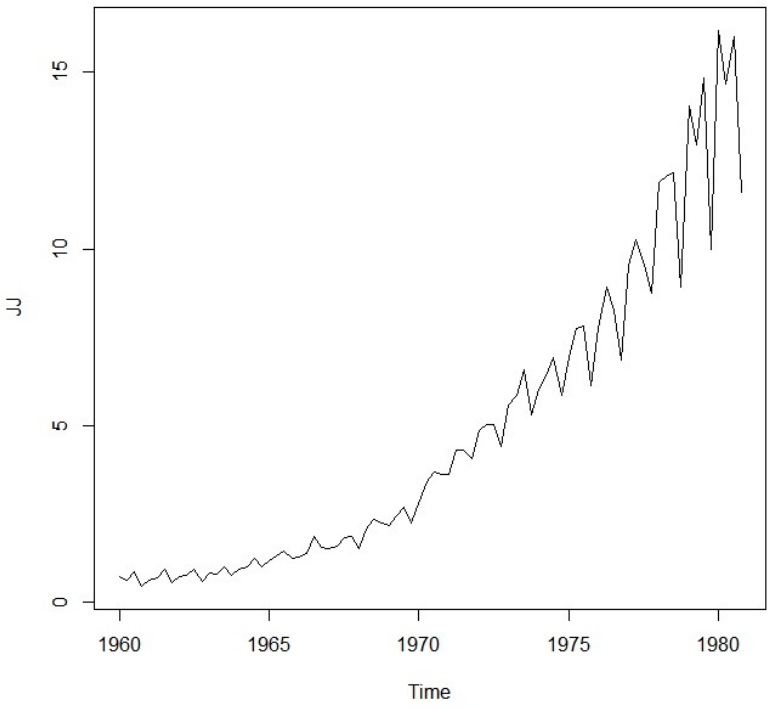
```
> attach(yforecast)
> invisible(lapply(newObs, function(x)
+ lines(x, col = 4, type ='o', pch = 4)))
> lines(f ,type = 'o', lwd = 2, pch = 16)
>
```

مثال آخر:

سوف نطبق نموذج فضاء حالة على المتسلسلة الزمنية للأرباح الربع سنوية لشركة Johnson & Johonson و نولد تنبؤات بإستخدام مرشح كالمان.

سوف نستخدم مكتبة TSA في R

```
> library(TSA)
> data(JJ)
> head(JJ)
[1] 0.71 0.63 0.85 0.44 0.61 0.69
> tail(JJ)
[1] 14.85  9.99 16.20 14.67 16.02 11.61
> plot.ts(JJ)
```



نلاحظ أن المتسلسلة غير مستقرة كما يوجد إنجراف و موسمية تتكرر كل ربع سنة والتي تزداد حجما مع الزمن. إجراء اي تحويل (لوغارثمي أو جزر تربيعي) لن يفيد

في هذه الحالة. لنمذجتها عن طريق فضاء الحالة لنفترض اننا نستطيع تمثيلها (انظر الجزء الأول) على الشكل

$$y_t = T_t + S_t + v_t$$

حيث T مركبة الإنجراف و S_t مركبة الموسمية و v_t مركبة الخطأ (ضجة بيضاء).

لنفترض أن مركبة الإنجراف تزداد بشكل اسي

$$T_t = \phi T_{t-1} + w_{t1}$$

حيث $\phi > 1$ تسبب الزيادة الاسية. و المركبة الموسمية تنمذج على الشكل

$$S_t + S_{t-1} + S_{t-2} + S_{t-3} = w_{t2}$$

و التي تعني ان المركبة الوسمية على مدى كل اربعة اشهر تجمع لصفر (او خطأ بسيط) . للتعبير عن هذا النموذج عن طريق فضاء الحالة (سوف نستخدم ترميز المكتبة dlm) معادلة الحالة هي

$$\mathbf{x}_t = \mathbf{\Phi}\mathbf{x}_{t-1} + \mathbf{w}_t, \quad \{\mathbf{w}_t\} \sim iid\ N_p(\mathbf{0}, \mathbf{Q})$$

و معادلة المشاهدات

$$\mathbf{y}_t = \mathbf{A}_t\mathbf{x}_t + \mathbf{v}_t, \quad \{\mathbf{v}_t\} \sim iid\ N_q(\mathbf{0}, \mathbf{R})$$

حيث المصفوفات $\mathbf{\Phi}$ مصفوفة التركيب و $\mathbf{A}_t$ مصفوفة المشاهدات ذات الأبعاد $q \times p$

ونفترض ان العملية بدأت بالمتجه

$$\mathbf{x}_0 \sim N_p(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$$

الآن لنفترض ان

$$\mathbf{x}_t = (T_t, S_t, S_{t-1}, S_{t-2})'$$

متجه الحالة فتكون معادلة المشاهدات

$$y_t = \begin{pmatrix} 1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} T_t \\ S_t \\ S_{t-1} \\ S_{t-2} \end{pmatrix} + v_t$$

و معادلة الحالة

$$\begin{pmatrix} T_t \\ S_t \\ S_{t-1} \\ S_{t-2} \end{pmatrix} = \begin{pmatrix} \phi & 0 & 0 & 0 \\ 0 & -1 & -1 & -1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} T_{t-1} \\ S_{t-1} \\ S_{t-2} \\ S_{t-3} \end{pmatrix} + \begin{pmatrix} w_{t1} \\ w_{t2} \\ 0 \\ 0 \end{pmatrix}$$

حيث $\mathbf{R} = r_{11}$ و

$$\mathbf{Q} = \begin{pmatrix} q_{11} & 0 & 0 & 0 \\ 0 & q_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

(انظر الجزء الأول) هنا $p = 4$ و $q = 1$ و المعالم التي يجب تقديرها هي r_{11} و q_{11} و q_{22} و ϕ بالاضافة تباينات الإنجراف و الموسمية. النمو هو حوالي 3% في السنة فنقدر $\phi = 1.03$ و المتوسط الأولى $\boldsymbol{\mu}_0 = \left(0.7, 0, 0, 0\right)'$ و

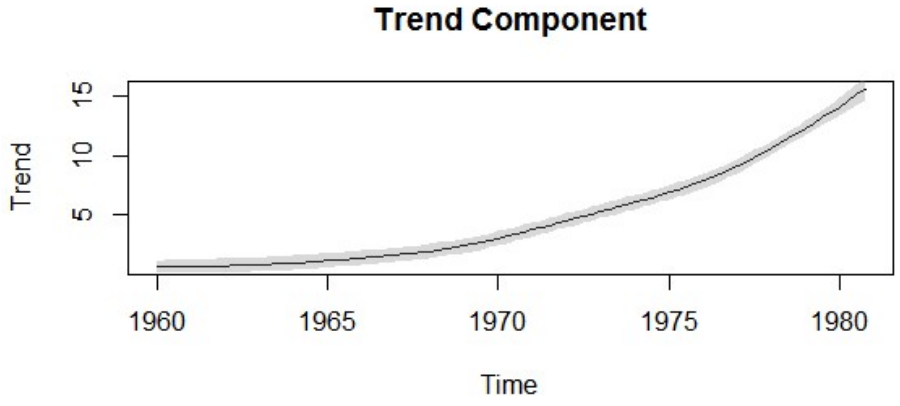
$$\boldsymbol{\Sigma}_0 = \begin{pmatrix} 0.4 & 0 & 0 & 0 \\ 0 & 0.4 & 0 & 0 \\ 0 & 0 & 0.4 & 0 \\ 0 & 0 & 0 & 0.4 \end{pmatrix}$$

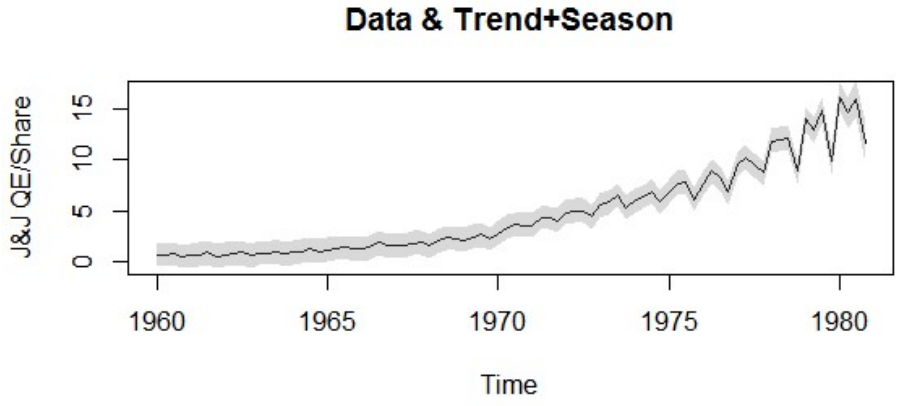
و $q_{11} = 0.01$ و $q_{22} = 0.01$ $r_{11} = 0.25$

```
> library(dlm)
> library(TSA)
> library(astsa)
> data(JJ)
> num = length(JJ)
> A = cbind(1,1,0,0)
> Linn =function(para){
+ Phi = diag(0,4); Phi[1,1] = para[1]
+ Phi[2,]=c(0,-1,-1,-1); Phi[3,]=c(0,1,0,0);
Phi[4,]=c(0,0,1,0)
+ cQ1 = para[2]; cQ2 = para[3] # sqrt q11 and q22
+ cQ = diag(0,4); cQ[1,1]=cQ1; cQ[2,2]=cQ2
+ cR = para[4] # sqrt r11
+ kf = Kfilter0(num, jj, A, mu0, Sigma0, Phi, cQ, cR)
+ return(kf$like) }
> # Initial Parameters
> mu0 = c(.7,0,0,0); Sigma0 = diag(.04,4)
> init.par = c(1.03,.1,.1,.5) # Phi[1,1], the 2 cQs and cR
> #Estimation and Results
> est = optim(init.par, Linn,NULL, method='BFGS',
hessian=TRUE,
+ control=list(trace=1,REPORT=1))
initial  value 2.693644
iter   2 value -0.853526
iter   3 value -9.416505
iter   4 value -10.241752
iter   5 value -19.419809
iter   6 value -30.441188
iter   7 value -31.825543
iter   8 value -32.248413
iter   9 value -32.839918
iter  10 value -33.019870
```

```
iter  11 value -33.041749
iter  12 value -33.050583
iter  13 value -33.055492
iter  14 value -33.078152
iter  15 value -33.096870
iter  16 value -33.098405
iter  17 value -33.099018
iter  18 value -33.099385
iter  19 value -33.099498
iter  19 value -33.099498
final  value -33.099498
converged
> SE = sqrt(diag(solve(est$hessian)))
> u = cbind(estimate=est$par, SE)
> rownames(u)=c('Phi11','sigw1','sigw2','sigv'); u
          estimate         SE
Phi11 1.0350847657 0.00253645
sigw1 0.1397255477 0.02155155
sigw2 0.2208782663 0.02376430
sigv  0.0004655672 0.24174702
> # Smooth
> Phi = diag(0,4); Phi[1,1] = est$par[1]
> Phi[2,]=c(0,-1,-1,-1); Phi[3,]=c(0,1,0,0);
Phi[4,]=c(0,0,1,0)
> cQ1 = est$par[2]; cQ2 = est$par[3]
> cQ = diag(1,4); cQ[1,1]=cQ1; cQ[2,2]=cQ2
> cR = est$par[4]
> ks = Ksmooth0(num,jj,A,mu0,Sigma0,Phi,cQ,cR)
> # Plots
> Tsm = ts(ks$xs[1,,], start=1960, freq=4)
> Ssm = ts(ks$xs[2,,], start=1960, freq=4)
> p1 = 3*sqrt(ks$Ps[1,1,]); p2 = 3*sqrt(ks$Ps[2,2,])
```

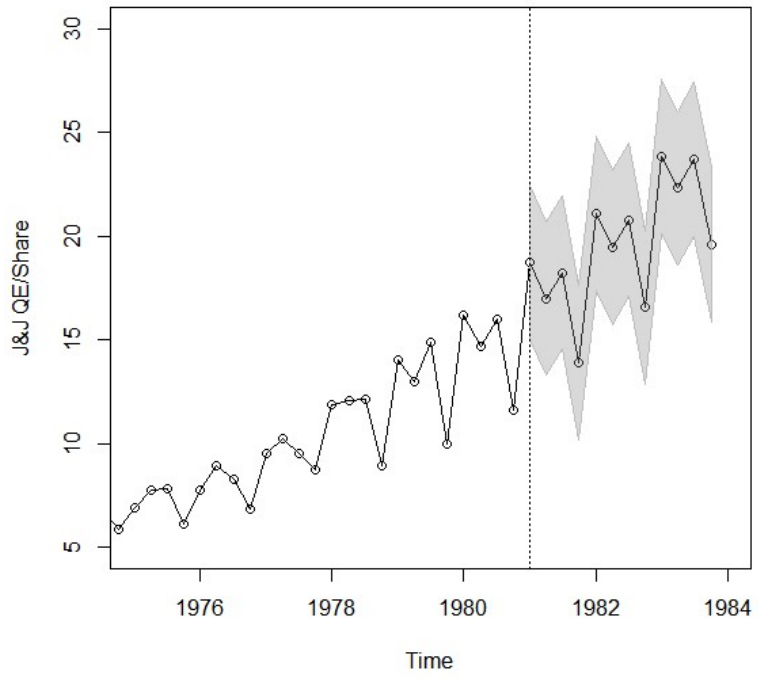
```
> par(mfrow=c(2,1))
> plot(Tsm, main='Trend Component', ylab='Trend')
> xx = c(time(jj), rev(time(jj)))
> yy = c(Tsm-p1, rev(Tsm+p1))
> polygon(xx, yy, border=NA, col=gray(.5, alpha = .3))
> plot(jj, main='Data & Trend+Season', ylab='J&J QE/Share',
ylim=c(-.5,17))
> xx = c(time(jj), rev(time(jj)) )
> yy = c((Tsm+Ssm)-(p1+p2), rev((Tsm+Ssm)+(p1+p2)) )
> polygon(xx, yy, border=NA, col=gray(.5, alpha = .3))
```






```
> # Forecast
> n.ahead = 12;
> y = ts(append(jj, rep(0,n.ahead)), start=1960, freq=4)
```

```
> rmspe = rep(0,n.ahead); x00 = ks$xf[,,num]; P00 =
ks$Pf[,,num]
> Q = t(cQ)%*%cQ; R = t(cR)%*%(cR)
> for (m in 1:n.ahead){
+ xp = Phi%*%x00; Pp = Phi%*%P00%*%t(Phi)+Q
+ sig = A%*%Pp%*%t(A)+R; K = Pp%*%t(A)%*%(1/sig)
+ x00 = xp; P00 = Pp-K%*%A%*%Pp
+ y[num+m] = A%*%xp; rmspe[m] = sqrt(sig) }
> plot(y, type='o', main='', ylab='J&J QE/Share',
ylim=c(5,30),
+ xlim=c(1975,1984))
> upp = ts(y[(num+1):(num+n.ahead)]+2*rmspe, start=1981,
freq=4)
> low = ts(y[(num+1):(num+n.ahead)]-2*rmspe, start=1981,
freq=4)
> xx = c(time(low), rev(time(upp)))
> yy = c(low, rev(upp))
> polygon(xx, yy, border=8, col=gray(.5, alpha = .3))
> abline(v=1981, lty=3)
```



مثال آخر بإستخدام مكتبة TSA

تمرين: يترك للقارئ تفسير ترميز R لهذا المثال

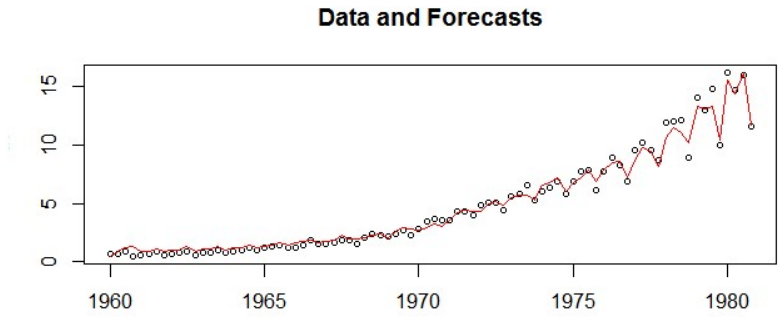
```
> # This example fits a structural State-space model to the
> # Johnson & Johnson quarterly earnings data
> # INITIALIZE: Set up and define parameters, matrices,
> # etc. Parameter estimates are given in book.
> A = t(c(1, 1, 0, 0))
> A
     [,1] [,2] [,3] [,4]
[1,]    1    1    0    0
> R = .0086
> theta = 1.035
> Phi = c(theta, 0 , 0, 0, 0, -1, -1, -1, 0, 1, 0, 0, 0, 0,
1, 0)
> Phi
```

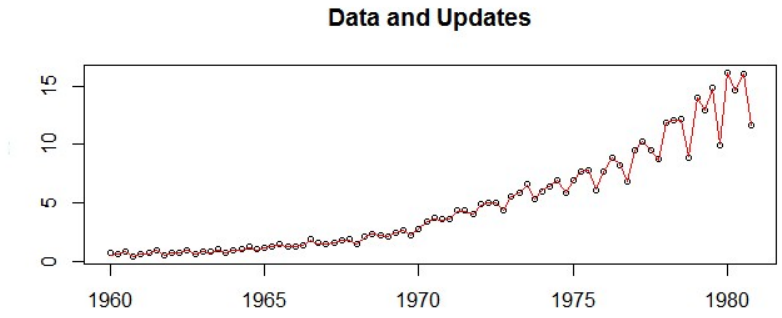
```
 [1]  1.035  0.000  0.000  0.000  0.000 -1.000 -1.000 -
1.000  0.000  1.000  0.000  0.000  0.000  0.000  1.000
0.000
> Phi = matrix(data=Phi, nrow=4, ncol=4, byrow=TRUE)
> Phi
      [,1] [,2] [,3] [,4]
[1,] 1.035    0    0    0
[2,] 0.000   -1   -1   -1
[3,] 0.000    1    0    0
[4,] 0.000    0    1    0
> Q = c(.0169, 0 , 0, 0, 0, .0497, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0)
> Q
 [1] 0.0169 0.0000 0.0000 0.0000 0.0000 0.0497 0.0000
0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
0.0000
> Q = matrix(data=Q, nrow=4, ncol=4, byrow=TRUE)
> Q
       [,1]   [,2] [,3] [,4]
[1,] 0.0169 0.0000    0    0
[2,] 0.0000 0.0497    0    0
[3,] 0.0000 0.0000    0    0
[4,] 0.0000 0.0000    0    0
> mu = c(.55, .21, .15, .06)
> mu
[1] 0.55 0.21 0.15 0.06
> eye = c(1, 0 , 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1)
> eye
 [1] 1 0 0 0 0 1 0 0 0 0 1 0 0 0 0 1
> eye = matrix(data=eye, nrow=4, ncol=4, byrow=TRUE)
> eye
     [,1] [,2] [,3] [,4]
```

```
[1,]    1    0    0    0
[2,]    0    1    0    0
[3,]    0    0    1    0
[4,]    0    0    0    1
> Sigma = eye*.01
> Sigma
     [,1] [,2] [,3] [,4]
[1,] 0.01 0.00 0.00 0.00
[2,] 0.00 0.01 0.00 0.00
[3,] 0.00 0.00 0.01 0.00
[4,] 0.00 0.00 0.00 0.01
> N = length(JJ)
> N
[1] 84
> xf = matrix(0,nrow=4,ncol=N)

> xf = matrix(0,nrow=4,ncol=N)
> xu = matrix(0,nrow=4,ncol=N)
> low.bd.fore = vector("numeric",length=N)
# To draw upper and lower bounds; not part of the
Kalmanfilter.
> low.bd.fore
> upp.bd.fore = vector("numeric",length=N)
> low.bd.data = vector("numeric",length=N)
> upp.bd.data = vector("numeric",length=N)
> low.bd.upd = vector("numeric",length=N)
> upp.bd.upd = vector("numeric",length=N)
> # The Kalman filter recursions:
> # The forcast and update cycle for t=1
> xf[,1] = Phi%*%mu # The forecast
> K = Sigma%*%t(A)%*%solve(A%*%Sigma%*%t(A) + R) # The
Kalman Gain
```

```
> xu[,1] = xf[,1] + K%*%(JJ[1] - A%*%xf[,1]) # The update
> Pu = (eye - K%*%A)%*%Sigma # The posterior error
covariance
> for(j in 2: N)
+  {
+  # The forecast
+  xf[,j] = Phi%*% xu[,j-1]
+  Pf = Phi%*%Pu%*%t(Phi) + Q
+  low.bd.fore[j] = A%*%xf[,j] - 1.96*sqrt(A%*%Pf%*%t(A) )
# State forecast bounds
+  upp.bd.fore[j] = A%*%xf[,j] + 1.96*sqrt(A%*%Pf%*%t(A) )
+  low.bd.data[j] = A%*%xf[,j] - 1.96*sqrt(A%*%Pf%*%t(A) +
R) # Data forecast bounds
+  upp.bd.data[j] = A%*%xf[,j] + 1.96*sqrt(A%*%Pf%*%t(A) +
R)
+  # The update
+  K = Pf%*%t(A)%*%solve(A%*%Pf%*%t(A) + R)
+  xu[,j] = xf[,j] + K%*%(JJ[j] - A%*%xf[,j])
+  Pu = (eye - K%*%A)%*%Pf
+  low.bd.upd[j] = A%*%xu[,j] - 1.96*sqrt(A%*%Pu%*%t(A))
#State filter bounds
+  upp.bd.upd[j] = A%*%xu[,j] + 1.96*sqrt(A%*%Pu%*%t(A))
+  }
> # Plotting
> par(mfrow=c(3,1))
> forecast = A%*%xf
> plot(JJ, type="l", cex = .75, main="Data",
xlab="Quarter", ylab="JJ")
> plot(t(forecast), type="l", cex = .75, main="Forecasts",
xlab="Quarter", ylab="JJ")
> update = A%*%xu
```

```
> plot(t(update), type="l", cex = .75, main="Updates",
xlab="Quarter", ylab="JJ")
>
```

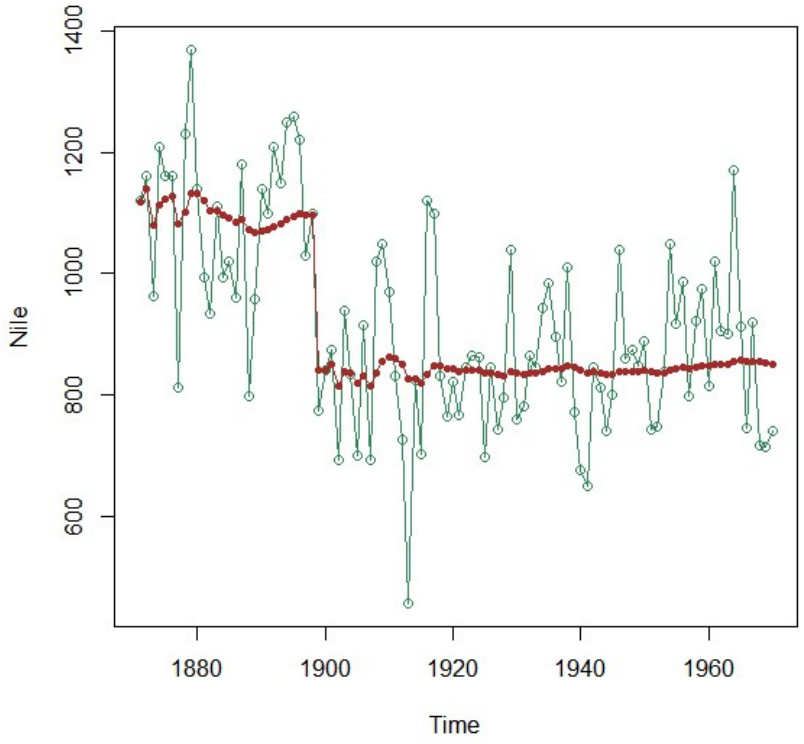




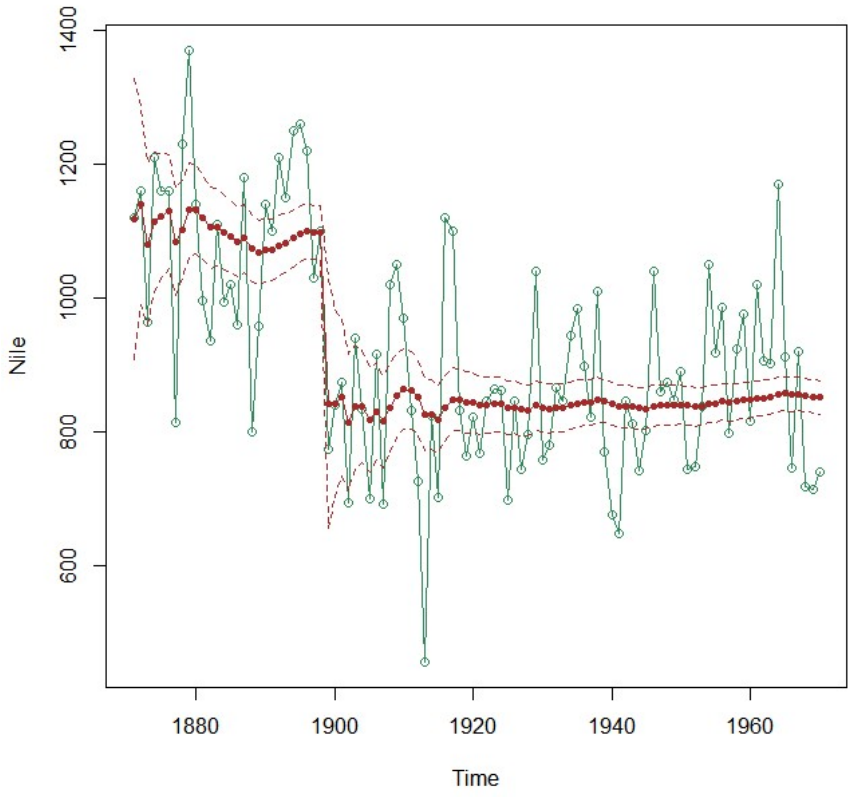
مثال آخر

```
data(Nile)
library(dlm)
buildFun <- function(x) {
 dlmModPoly(1, dV = exp(x[1]), dW = exp(x[2]))
 }
fit <- dlmMLE(Nile, parm = c(0,0), build = buildFun)
fit$conv
dlmNile <- buildFun(fit$par)
V(dlmNile)
```

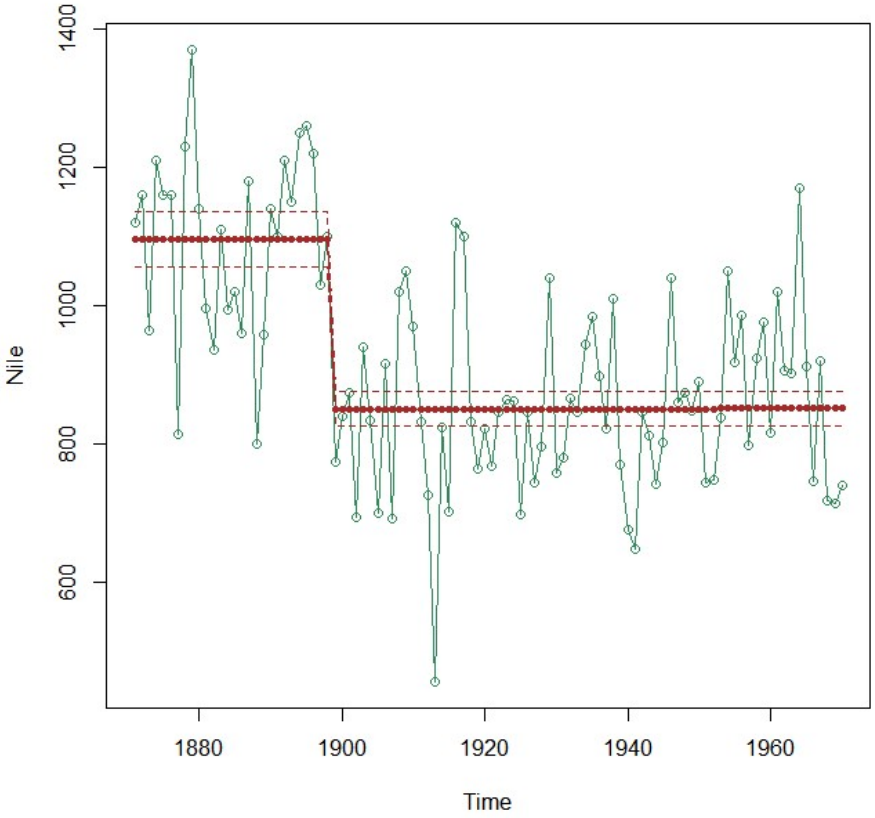
```
W(dlmNile)
StructTS(Nile, "level")
buildFun <- function(x) {
 m <- dlmModPoly(1, dV = exp(x[1]))
 m$JW <- matrix(1)
 m$X <- matrix(exp(x[2]), nc = 1, nr = length(Nile))
 j <- which(time(Nile) == 1899)
 m$X[j,1] <- m$X[j,1] * (1 + exp(x[3]))
 return(m)
 }
fit <- dlmMLE(Nile, parm = c(0,0,0), build = buildFun)
fit$conv
dlmNileJump <- buildFun(fit$par)
V(dlmNileJump)
dlmNileJump$X[c(1, which(time(Nile) == 1899)), 1]
nileJumpFilt <- dlmFilter(Nile, dlmNileJump)
plot(Nile, type = 'o', col = "seagreen")
lines(dropFirst(nileJumpFilt$m), type = 'o',
 pch = 20, col = "brown")
```




```
attach(nileJumpFilt)
v <- unlist(dlmSvd2var(U.C, D.C))
pl <- dropFirst(m) + qnorm(0.05, sd = sqrt(v[-1]))
pu <- dropFirst(m) + qnorm(0.95, sd = sqrt(v[-1]))
detach()
lines(pl, lty = 2, col = "brown")
lines(pu, lty = 2, col = "brown")
```




```
nileJumpSmooth <- dlmSmooth(nileJumpFilt)
plot(Nile, type = 'o', col = "seagreen")
attach(nileJumpSmooth)
lines(dropFirst(s), type = 'o', pch = 20, col = "brown")
v <- unlist(dlmSvd2var(U.S, D.S))
pl <- dropFirst(s) + qnorm(0.05, sd = sqrt(v[-1]))
pu <- dropFirst(s) + qnorm(0.95, sd = sqrt(v[-1]))
detach()
lines(pl, lty = 2, col = "brown")
lines(pu, lty = 2, col = "brown")
```


Nile
1400
1200
1000
800
600
1880
1900
1920
1940
1960
Time

الفصل الخامس:

نماذج ARCH ونماذج GARCH

وتطبيقاتها في التنبؤ المالي Finantial Time Series Forecasting

نماذج Autocorrelated Conditional Heteroscedastic (ARCH) ونماذج Generalized Autocorrelated Conditional Heteroscedastic (GARCH) تطبق غالبا في نمذجة الخطورة المالية Financial Risk Modelling و التي هي إحتمال الخسارة المالية الكلية والتي تقاس بالهشاشة Volatility (أنظر الملحق) .

أول معادلة في نماذج ARCH هي المعادلة

$$y_t = \mathbf{x}_t'\boldsymbol{\beta} + \varepsilon_t$$

و التي تختلف عن الإنحدار التقليدي في فرضية إستقلال ووحدة التوزيع الطبيعي للأخطاء *iid errors*. حيث y_t متجه المشاهدات و $\mathbf{x}_t'$ مصفوفة التصميم desighn matrix و ε_t الأخطاء (أنظر اي كتاب عن الإنحدار)

عملية الأخطاء ε_t يفترض انها موزعة طبيعيا ولكن **التباين غير ثابت ويتغير مع الزمن**. هذا التغير للتباين يعبر عنه كإعتماد على المعلومات المتوفرة عند الزمن t - 1 و الذي نرمز له بـ $\boldsymbol{\Psi}_{t-1}$ أي

$$\varepsilon_t \mid \boldsymbol{\Psi}_{t-1} \sim N(0, h_t)$$

حيث

$$\boldsymbol{\Psi}_{t-1} = \{y_{t-1}, x_{t-1}, y_{t-2}, x_{t-2}, \ldots\}$$

ملاحظة: هناك ترميز آخر يقابلنا في كتب اخرى هو

$$\varepsilon_t = \eta_t \sqrt{h_t},$$
$$\eta_t \sim D_\nu(0,1)$$

حيث η_t ترمز لمتغير عشوائي له توزيع D بتوقع صفري و تباين 1 . هناك معالم إضافية اخرى للتوزيع توجد ضمنيا في ν.

النقطة الثانية في بناء نماذج ARCH هي معادلة التباين ففي نموذج $ARCH(q)$ التباين الشرطي يعبر عنه بتاريخ مربع الخطأ حتى التخلف الزمن q أي

$$h_t = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \cdots + \alpha_q \varepsilon_{t-q}^2$$

حيث $\alpha_0 > 0$ و $\alpha_i \geq 0, i = 1,\cdots,q$. حصر قيم المعالم هذا يضمن كون التباين الشرطي موجب. إضافة المعلومات المتوفرة حتى الزمن t - 1 ظاهر من المعادلة

$$\varepsilon_{t-i} = y_{t-i} - \mathbf{x}'_{t-i}\boldsymbol{\beta}, \ i = 1,\cdots,q$$

أي أن التباين الشرطي قد تم وصفه بألأخطاء في الفترات السابقة.

التعميم أي نموذج $GARCH(p,q)$ هو

$$h_t = \alpha_0 + \beta_1 h_{t-1} + \cdots + \beta_p h_{t-p} + \alpha_1 \varepsilon_{t-1}^2 + \cdots + \alpha_q \varepsilon_{t-q}^2$$

حيث $\beta_i \geq 0, i = 1,\cdots,p$.

العملية $GARCH(p,q)$ تكون مستقرة (إستقرار ضعيف) إذا تحقق الشرط

$$\sum_{j=1}^{q} \alpha_j + \sum_{j=1}^{p} \beta_j < 1$$

مثال:

سوف نستخدم مكتبة tseries لتطبيق ARCH(1) و ARCH(2) و ARCH(3) و GARCH(1,1) و GARCH(2,1) و GARCH(2,2) لمتسلسلة العائدات اليومية لشركة سابك.

ملاحظات:

1) الدالة `garch(x,c(q,p))` تنمذج المتلسلة x كـ GARCH(p,q) لاحظ في R الترميز معكوس.

2) الدالة `garch(x,c(q,p))` لا تقبل قيم مفقودة missing values لهذا يستخدم الإمداد الخطي linear interpolation لإستبدال القيم المفقودة بقيم مقدرة خطيا بإستخدام الدالة interpNA لتخليص البيانات من القيم المفقودة.

```
> rm(list=ls())
> library(xlsx)
> sabicRet <- read.xlsx("C:/Users/ibin/Desktop/GARCH/sabic
return.xlsx",1, header=T)
> head(sabicRet)
        Date Return.Sabic
1 2016-04-12   -0.7741766
2 2016-04-11    1.9703782
3 2016-04-10    0.7132668
4 2016-04-07   -1.5580737
5 2016-04-06   -1.0725965
6 2016-04-05   -0.5685575
> library(tseries)
> rs <- sabicRet[,2]
> head(rs)
[1] -0.7741766  1.9703782  0.7132668 -1.5580737 -1.0725965
-0.5685575
> rs<- as.ts(rs)
> arch1 <- garch(rs, order=c(0,1),trace=F )
> summary(arch1)

Call:
garch(x = rs, order = c(0, 1),trace=F)
```

```
Model:
GARCH(0,1)

Residuals:
     Min       1Q   Median       3Q      Max
-4.86273 -0.55311  0.01341  0.57947  3.55667

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0   3.37171     0.22149   15.223  < 2e-16 ***
a1   0.35467     0.08565    4.141 3.46e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1
```

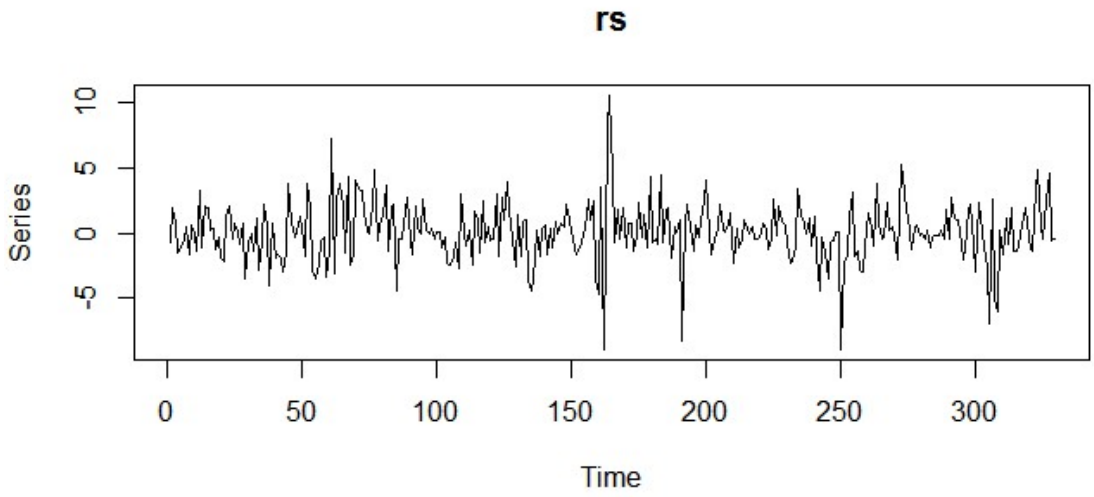
$$\hat{\alpha}_0 = 3.37171, \quad s.e\left(\hat{\alpha}_0\right) = 0.22149$$

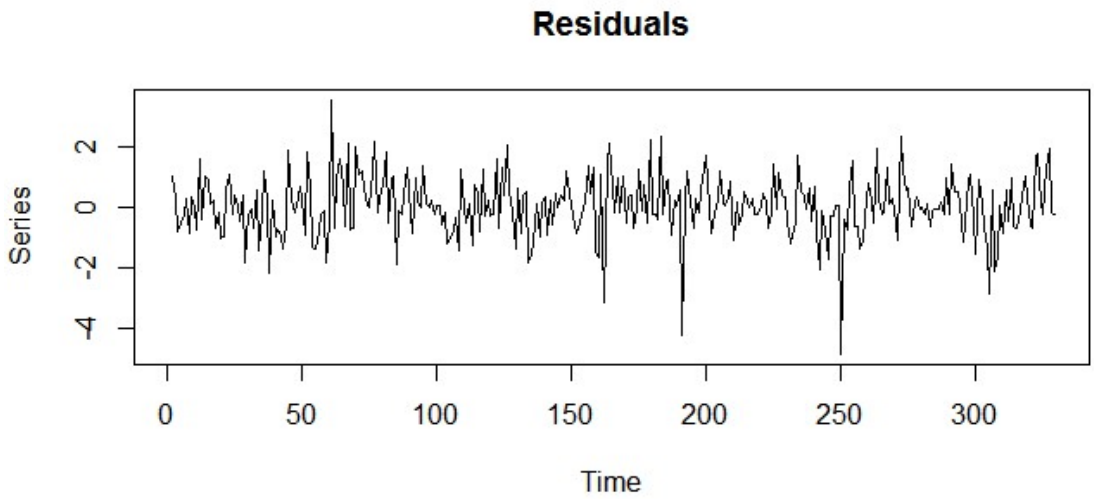
$$\hat{\alpha}_1 = 0.35467, \quad s.e\left(\hat{\alpha}_1\right) = 0.08565$$

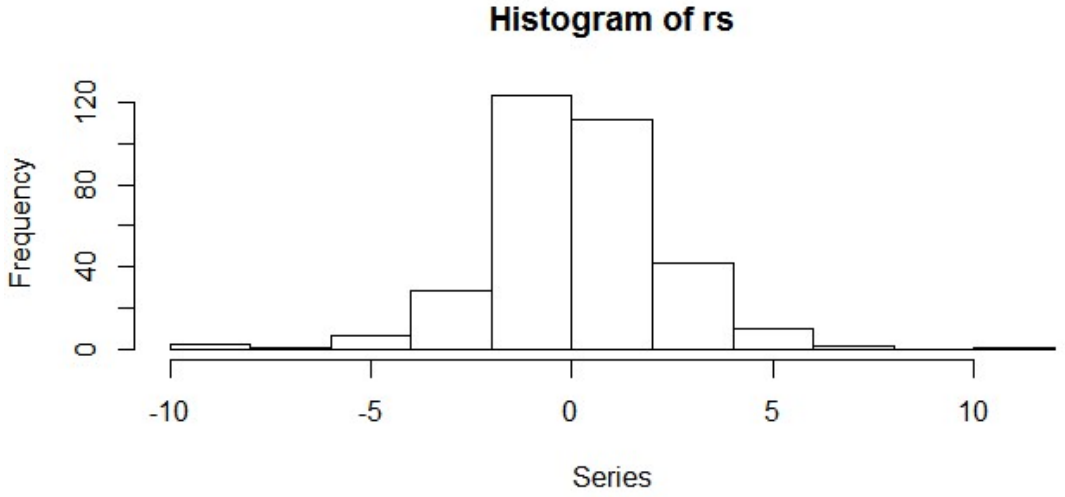
و كلاهما عالي المعنوية highly significant

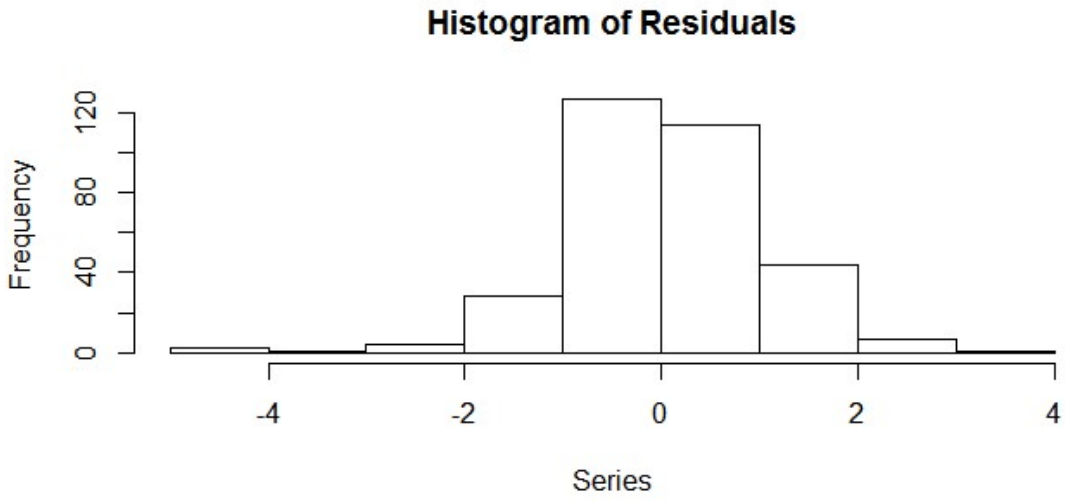
```
Diagnostic Tests:
        Jarque Bera Test

data:  Residuals
X-squared = 101.6, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  Squared.Residuals
X-squared = 0.68165, df = 1, p-value = 0.409
```

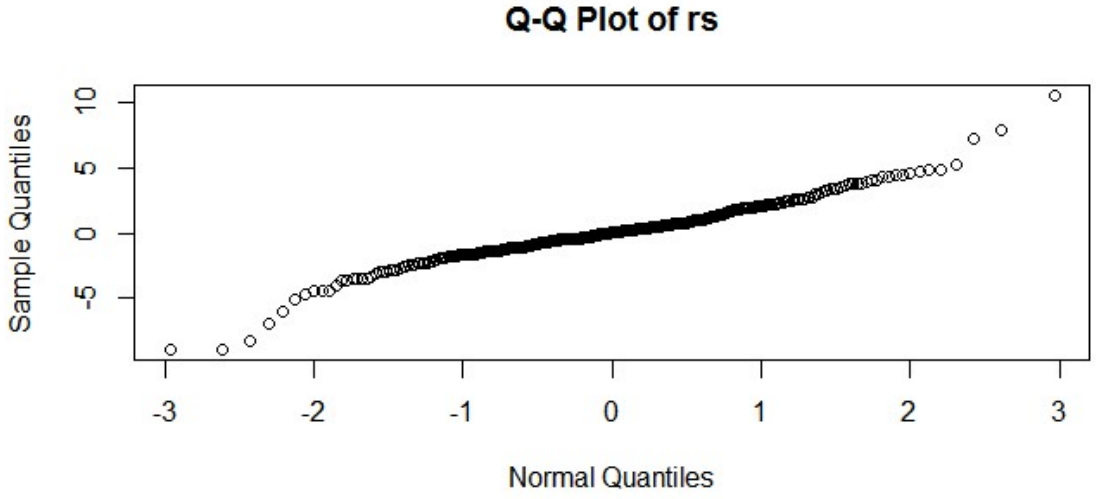
أى أن $ARCH(1)$ يعطي افضل تطبيق (أنظر التالي)

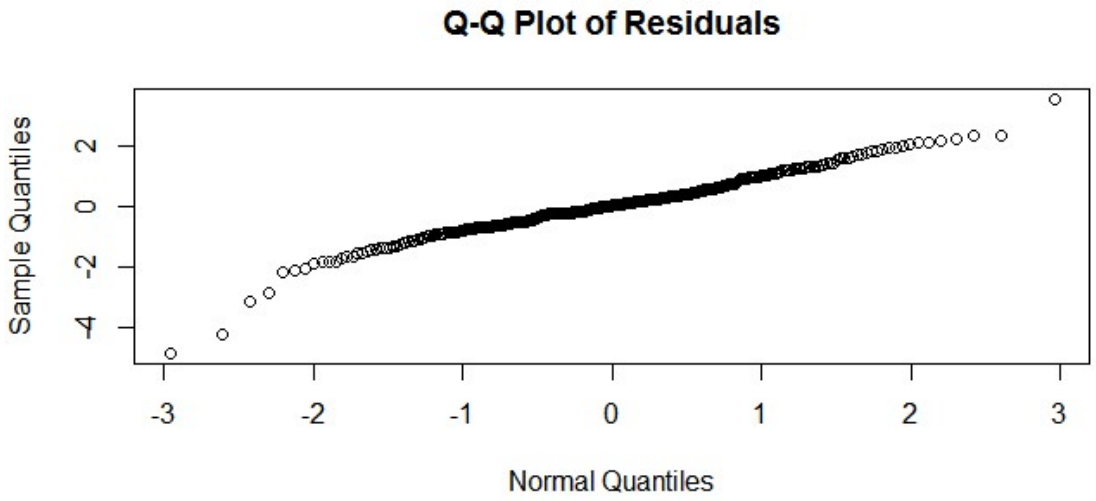


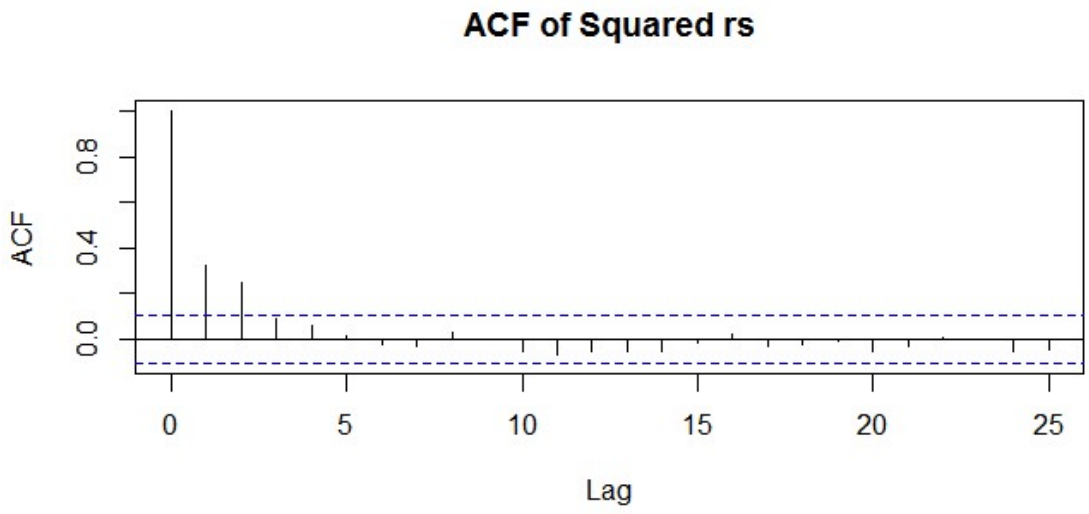


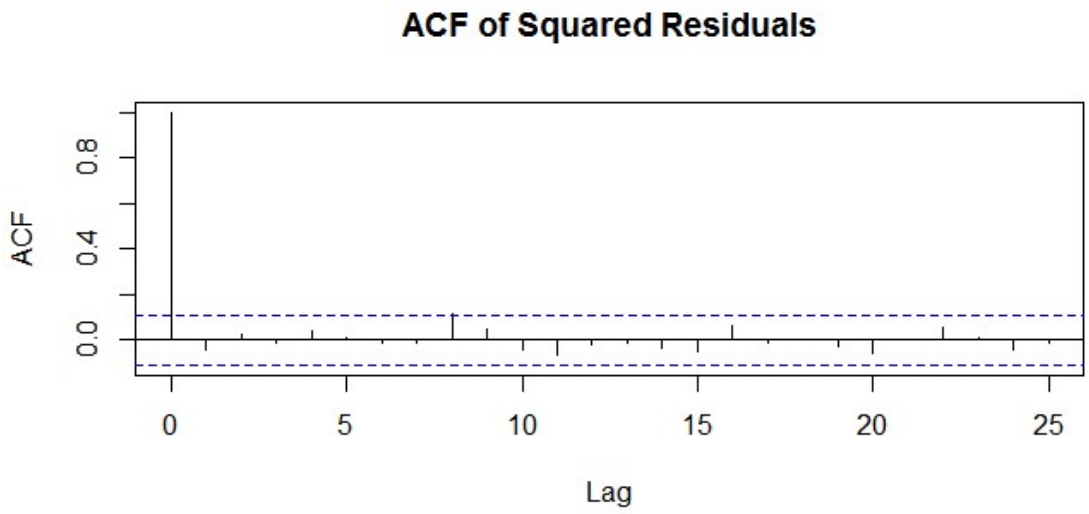

Histogram of rs
Frequency
0
40
80
120
-10
-5
0
5
10
Series


Histogram of Residuals
Frequency
0
40
80
120
-4
-2
0
2
4
Series


Q-Q Plot of rs
Sample Quantiles
-5
0
5
10
-3
-2
-1
0
1
2
3
Normal Quantiles


Q-Q Plot of Residuals
Sample Quantiles
-4
-2
0
2
-3
-2
-1
0
1
2
3
Normal Quantiles






```
> arch2 <- garch(rs, order=c(0,2),trace=F)

> summary(arch2)

Call:
garch(x = rs, order = c(0, 2),trace=F)

Model:
GARCH(0,2)
```

```
Residuals:
      Min        1Q    Median        3Q       Max
-4.876144 -0.556238  0.009877  0.581949  3.511916

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0   3.35301     0.22161   15.130  < 2e-16 ***
a1   0.33916     0.10203    3.324 0.000887 ***
a2   0.01359     0.04737    0.287 0.774273
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1
```

نلاحظ أن $ARCH(2)$ غير معنوي. لاحظ أن

$$\hat{\alpha}_2 = 0.01359, \quad s.e(\hat{\alpha}_2) = 0.04737, \; p-value = 0.774273$$

```
Diagnostic Tests:
        Jarque Bera Test

data:  Residuals
X-squared = 103.26, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  Squared.Residuals
X-squared = 0.60612, df = 1, p-value = 0.4363

> arch3 <- garch(rs, order=c(0,3),trace=F)

> summary(arch3)
```

```
Call:
garch(x = rs, order = c(0, 3),trace=F)

Model:
GARCH(0,3)

Residuals:
      Min         1Q    Median         3Q       Max
-4.283332 -0.523156  0.007952  0.558971  3.055953

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 4.345e+00   3.330e-01   13.048   <2e-16 ***
a1 1.406e-01   8.230e-02    1.708   0.0876 .
a2 7.989e-02   6.350e-02    1.258   0.2083
a3 1.031e-13   6.038e-02    0.000   1.0000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Diagnostic Tests:
        Jarque Bera Test

data:  Residuals
X-squared = 77.124, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  Squared.Residuals
X-squared = 0.2911, df = 1, p-value = 0.5895
```

```
> garch11 <- garch(rs, order=c(1,1),trace=F)
> summary(garch11)

Call:
garch(x = rs, order = c(1, 1),trace=F)

Model:
GARCH(1,1)

Residuals:
     Min       1Q   Median       3Q      Max
-4.16467 -0.51949  0.01273  0.55640  3.23545

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 4.597e+00   1.349e+00    3.407 0.000656 ***
a1 1.712e-01   7.540e-02    2.270 0.023178 *
b1 4.321e-14   2.713e-01    0.000 1.000000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1
```

لاحظ $\hat{\beta}_1 = 4.32 \times 10^{-14} \simeq 0$

```
Diagnostic Tests:
        Jarque Bera Test

data:  Residuals
X-squared = 76.091, df = 2, p-value < 2.2e-16


        Box-Ljung test
```

```
data:  Squared.Residuals
X-squared = 0.40425, df = 1, p-value = 0.5249

> garch12 <- garch(rs, order=c(1,2),trace=F)
> summary(garch12)

Call:
garch(x = rs, order = c(1, 2),trace=F)

Model:
GARCH(1,2)

Residuals:
     Min       1Q   Median       3Q      Max
-4.28458 -0.52361  0.01105  0.55187  3.07546

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 4.342e+00   3.599e+00    1.207   0.2276
a1 1.432e-01   8.217e-02    1.742   0.0814 .
a2 7.351e-02   1.421e-01    0.517   0.6049
b1 5.985e-14   8.166e-01    0.000   1.0000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Diagnostic Tests:
        Jarque Bera Test

data:  Residuals
X-squared = 78.169, df = 2, p-value < 2.2e-16
```

```
        Box-Ljung test

data:  Squared.Residuals
X-squared = 0.29194, df = 1, p-value = 0.589

> garch21 <- garch(rs, order=c(2,1),trace=F)
> summary(garch21)

Call:
garch(x = rs, order = c(2, 1),trace=F)

Model:
GARCH(2,1)

Residuals:
     Min       1Q   Median       3Q      Max
-4.26062 -0.53052  0.01101  0.55881  3.29839

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 4.343e+00   1.785e+00    2.433    0.015 *
a1 1.693e-01   7.392e-02    2.291    0.022 *
b1 1.119e-02   2.673e-01    0.042    0.967
b2 2.009e-14   3.318e-01    0.000    1.000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1
```

لايمكن أن يكون $GARCH(2,1)$

```
Diagnostic Tests:
        Jarque Bera Test
```

```
data:  Residuals
X-squared = 76.358, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  Squared.Residuals
X-squared = 0.32001, df = 1, p-value = 0.5716

> garch22 <- garch(rs, order=c(2,2),trace=F)
> summary(garch22)

Call:
garch(x = rs, order = c(2, 2),trace=F)

Model:
GARCH(2,2)

Residuals:
     Min       1Q   Median       3Q      Max
-4.41057 -0.53848  0.01123  0.56482  3.14114

Coefficient(s):
    Estimate  Std. Error  t value Pr(>|t|)
a0 4.087e+00   4.134e+00    0.989   0.3229
a1 1.442e-01   7.891e-02    1.827   0.0677 .
a2 7.453e-02   2.332e-01    0.320   0.7493
b1 2.687e-03   1.520e+00    0.002   0.9986
b2 1.402e-13   6.341e-01    0.000   1.0000
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1
```

لايمكن أن يكون $GARCH(2,2)$

```
Diagnostic Tests:
        Jarque Bera Test

data:  Residuals
X-squared = 79.759, df = 2, p-value < 2.2e-16


        Box-Ljung test

data:  Squared.Residuals
X-squared = 0.1869, df = 1, p-value = 0.6655

>
```

سوف نستخدم مكتبة fGarch لنمذجة و تحليل نموذج GARCH للهشاشة في عائدات شركة شابك

```
> rm(list=ls())
> library(xlsx)
> sabicRet <- read.xlsx("C:/Users/ibin/Desktop/GARCH/sabic
return.xlsx",1, header=T)
> head(sabicRet)
        Date Return.Sabic
1 2016-04-12   -0.7741766
2 2016-04-11    1.9703782
3 2016-04-10    0.7132668
4 2016-04-07   -1.5580737
5 2016-04-06   -1.0725965
6 2016-04-05   -0.5685575
```

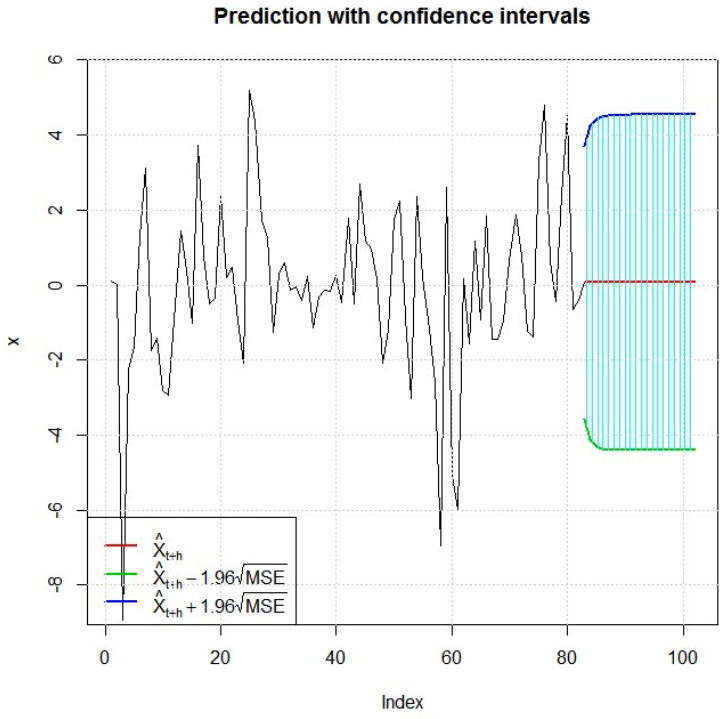
```
> sabicReturn <- sabicRet[,2]
> head(sabicReturn)
[1] -0.7741766  1.9703782  0.7132668 -1.5580737 -1.0725965
-0.5685575
> sabicReturn<- as.ts(sabicReturn)
> library(fGarch)
> fit = garchFit(~ garch(1, 0), data = sabicReturn, trace =
FALSE)
> fit

Title:
 GARCH Modelling

Call:
 garchFit(formula = ~garch(1, 0), data = sabicReturn, trace
= FALSE)

Mean and Variance Equation:
 data ~ garch(1, 0)
<environment: 0x0000000004c2a028>
 [data = sabicReturn]

Conditional Distribution:
 norm

Coefficient(s):
      mu     omega    alpha1
0.064918  3.349636  0.356988

Std. Errors:
 based on Hessian
```

```
Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu       0.06492     0.10726    0.605  0.54501
omega    3.34964     0.38298    8.746  < 2e-16 ***
alpha1   0.35699     0.11164    3.198  0.00139 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Log Likelihood:
 -716.5673    normalized:  -2.178016

Description:
 Sun May 21 12:32:51 2017 by user: ibin

> fit2 = garchFit(~ garch(1, 1), data = sabicReturn, trace
= FALSE)
> fit2

Title:
 GARCH Modelling

Call:
 garchFit(formula = ~garch(1, 1), data = sabicReturn, trace
= FALSE)

Mean and Variance Equation:
 data ~ garch(1, 1)
<environment: 0x0000000004c24270>
 [data = sabicReturn]
```

```
Conditional Distribution:
 norm

Coefficient(s):
      mu     omega    alpha1     beta1
0.069969  3.197655  0.341206  0.039785

Std. Errors:
 based on Hessian

Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu       0.06997     0.10889    0.643  0.52051
omega    3.19766     0.59651    5.361 8.29e-08 ***
alpha1   0.34121     0.11991    2.845  0.00443 **
beta1    0.03978     0.13017    0.306  0.75988
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Log Likelihood:
 -716.5108    normalized:  -2.177844

Description:
 Sun May 21 12:34:05 2017 by user: ibin

> fit3 = garchFit(~ garch(2, 1), data = sabicReturn, trace
= FALSE)
Warning message:
In sqrt(diag(fit$cvar)) : NaNs produced
> fit3 = garchFit(~ garch(1,2), data = sabicReturn, trace =
FALSE)
```

```
> fit3

Title:
 GARCH Modelling

Call:
 garchFit(formula = ~garch(1, 2), data = sabicReturn, trace
= FALSE)

Mean and Variance Equation:
 data ~ garch(1, 2)
<environment: 0x0000000006208e68>
 [data = sabicReturn]

Conditional Distribution:
 norm

Coefficient(s):
        mu       omega      alpha1       beta1       beta2
0.06788703  3.20606677  0.34093545  0.03745789  0.00000001

Std. Errors:
 based on Hessian

Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu     6.789e-02   1.096e-01    0.620 0.535488
omega  3.206e+00   9.490e-01    3.378 0.000729 ***
alpha1 3.409e-01   1.589e-01    2.146 0.031873 *
beta1  3.746e-02   1.314e-01    0.285 0.775591
beta2  1.000e-08   1.896e-01    0.000 1.000000
---
```

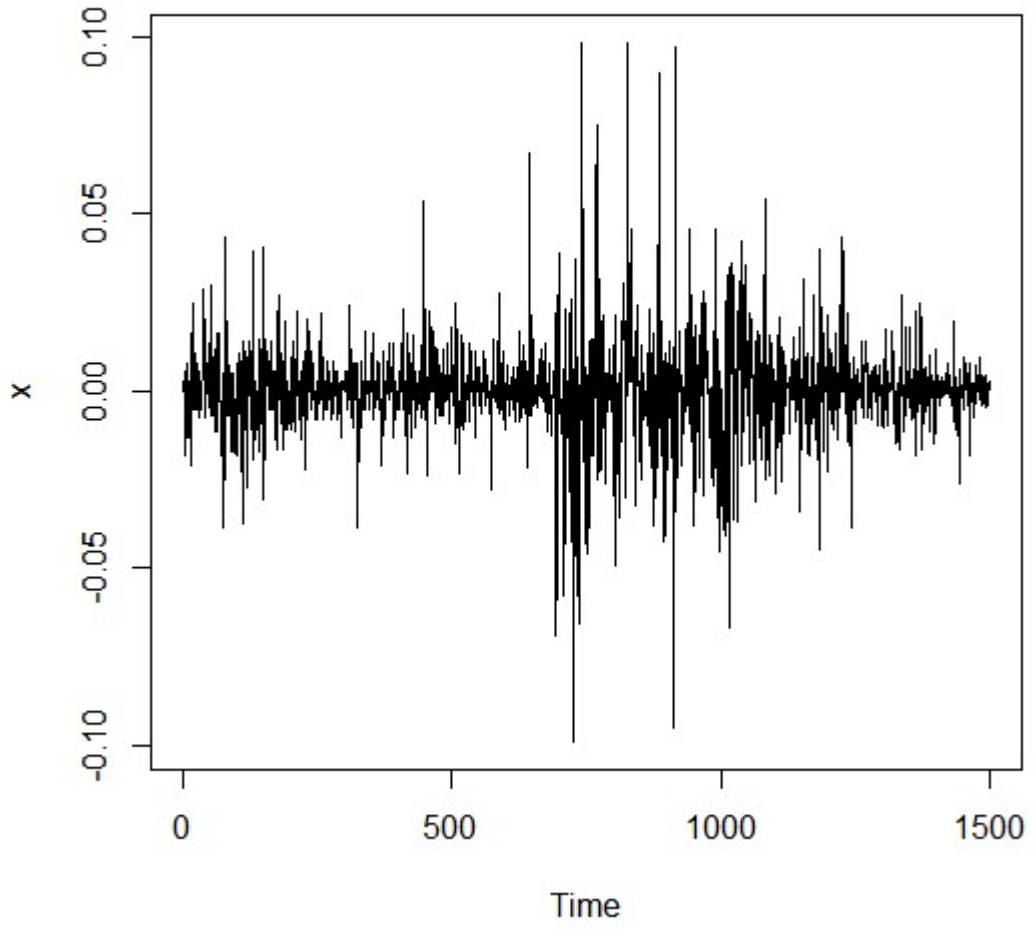
```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Log Likelihood:
 -716.5442    normalized:  -2.177946

Description:
 Sun May 21 12:35:12 2017 by user: ibin

> predict(fit,n.ahead=20,plot=TRUE)
   meanForecast meanError standardDeviation lowerInterval upperInterval
1    0.06491827  1.851645          1.851645     -3.564239      3.694075
2    0.06491827  2.138598          2.138598     -4.126657      4.256494
3    0.06491827  2.232120          2.232120     -4.309957      4.439793
4    0.06491827  2.264571          2.264571     -4.373559      4.503396
5    0.06491827  2.276043          2.276043     -4.396045      4.525881
6    0.06491827  2.280125          2.280125     -4.404045      4.533881
7    0.06491827  2.281580          2.281580     -4.406897      4.536734
8    0.06491827  2.282100          2.282100     -4.407915      4.537751
9    0.06491827  2.282285          2.282285     -4.408278      4.538115
10   0.06491827  2.282351          2.282351     -4.408408      4.538244
11   0.06491827  2.282375          2.282375     -4.408454      4.538291
12   0.06491827  2.282383          2.282383     -4.408471      4.538307
13   0.06491827  2.282386          2.282386     -4.408477      4.538313
14   0.06491827  2.282387          2.282387     -4.408479      4.538315
15   0.06491827  2.282388          2.282388     -4.408479      4.538316
16   0.06491827  2.282388          2.282388     -4.408480      4.538316
17   0.06491827  2.282388          2.282388     -4.408480      4.538316
18   0.06491827  2.282388          2.282388     -4.408480      4.538316
19   0.06491827  2.282388          2.282388     -4.408480      4.538316
20   0.06491827  2.282388          2.282388     -4.408480      4.538316
>
```



مثال آخر:

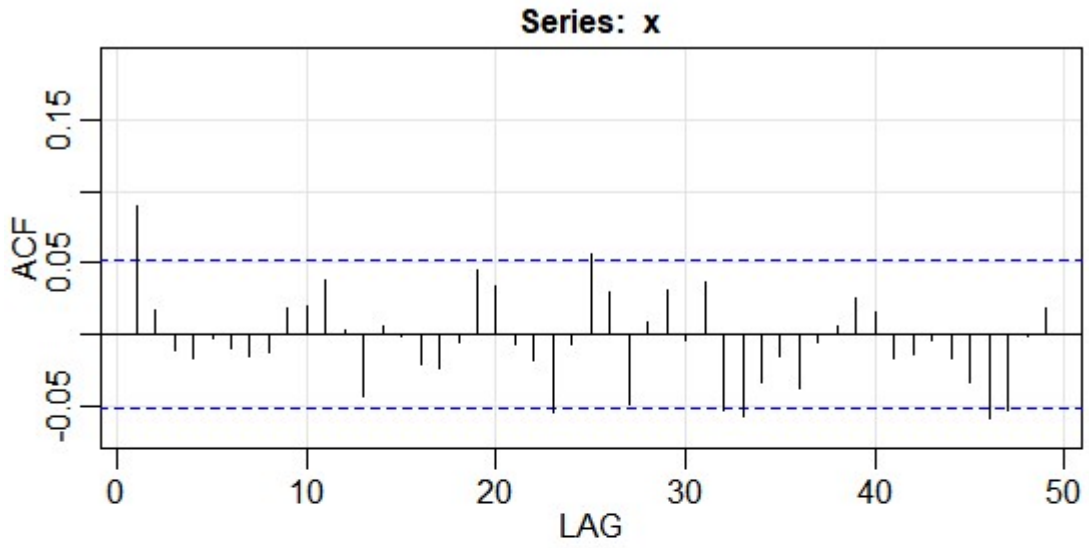
سوف نستخدم مكتباتxts و fGarch و astsa لتطبيق نموذج ARMA(0,1) و GARCH(1,1) لبيانات العائدات لشركة سابك من 1/1/2012 و حتى 31/12/2017 و التي يمكن الحصول عليها من https://www.tadawul.com.sa

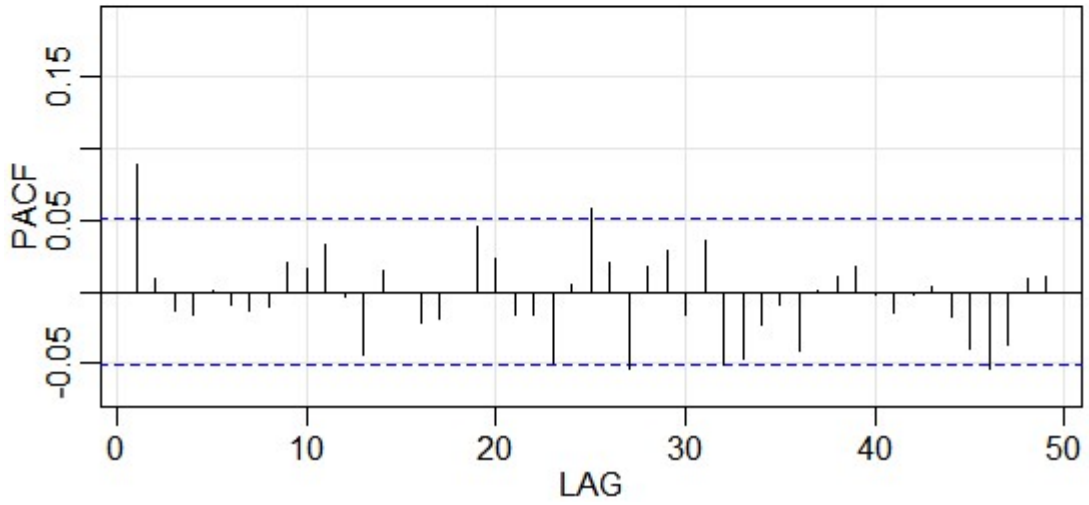
```
> lapply( c('xts', 'fGarch','astsa'), library, char=TRUE)
> sabicr <- read.csv(file.choose())
> sbr <- sabicr[,3]
> sbrt = as.ts(sbr)
> x = sbrt
> head(x)
   ï..date close       return
1 1/1/2012 95.75  0.000000000
2 1/2/2012 96.25  0.005221932
3 1/3/2012 96.75  0.005194805
4 1/4/2012 96.25 -0.005167959
5 1/7/2012 94.50 -0.018181818
6 1/8/2012 95.25  0.007936508
> dx = diff(x)
> plot.ts(dx)
> acf2(dx)
> plot.ts(x)
```




```
> acf2(x)
         ACF  PACF
 [1,]  0.09  0.09
 [2,]  0.02  0.01
 [3,] -0.01 -0.01
 [4,] -0.02 -0.02
 [5,]  0.00  0.00
 [6,] -0.01 -0.01
 [7,] -0.01 -0.01
 [8,] -0.01 -0.01
 [9,]  0.02  0.02
[10,]  0.02  0.02
[11,]  0.04  0.03
[12,]  0.00  0.00
[13,] -0.04 -0.04
[14,]  0.01  0.01
[15,]  0.00  0.00
```

```
[16,] -0.02 -0.02
[17,] -0.02 -0.02
[18,] -0.01  0.00
[19,]  0.04  0.05
[20,]  0.03  0.02
[21,] -0.01 -0.02
[22,] -0.02 -0.02
[23,] -0.05 -0.05
[24,] -0.01  0.01
[25,]  0.06  0.06
[26,]  0.03  0.02
[27,] -0.05 -0.05
[28,]  0.01  0.02
[29,]  0.03  0.03
[30,]  0.00 -0.02
[31,]  0.04  0.04
[32,] -0.05 -0.05
[33,] -0.06 -0.05
[34,] -0.03 -0.02
[35,] -0.01 -0.01
[36,] -0.04 -0.04
[37,] -0.01  0.00
[38,]  0.01  0.01
[39,]  0.03  0.02
[40,]  0.02  0.00
[41,] -0.02 -0.01
[42,] -0.01  0.00
[43,]  0.00  0.00
[44,] -0.02 -0.02
[45,] -0.03 -0.04
[46,] -0.06 -0.05
[47,] -0.05 -0.04
[48,]  0.00  0.01
[49,]  0.02  0.01
```





اولا نطبق ARMA(1,1) و GARCH(1,1) على بيانات العائد لسهم سابك

```
> summary(x.g <- garchFit(~arma(1,1)+garch(1,1),
data=x,cond.dist='std'))

Series Initialization:
 ARMA Model:                 arma
 Formula Mean:               ~ arma(1, 1)
 GARCH Model:                garch
 Formula Variance:           ~ garch(1, 1)
 ARMA Order:                 1 1
 Max ARMA Order:             1
 GARCH Order:                1 1
 Max GARCH Order:            1
 Maximum Order:              1
 Conditional Dist:           std
 h.start:                    2
```

```
 llh.start:                1
 Length of Series:         1499
 Recursion Init:           mci
 Series Scale:             0.01528974

Parameter Initialization:
 Initial Parameters:          $params
 Limits of Transformations:   $U, $V
 Which Parameters are Fixed?  $includes
 Parameter Matrix:
                       U           V      params includes
    mu      -0.10367074   0.1036707  0.01036543     TRUE
    ar1     -0.99999999   1.0000000  0.16222075     TRUE
    ma1     -0.99999999   1.0000000 -0.07347188     TRUE
    omega    0.00000100 100.0000000  0.10000000     TRUE
    alpha1   0.00000001   1.0000000  0.10000000     TRUE
    gamma1  -0.99999999   1.0000000  0.10000000    FALSE
    beta1    0.00000001   1.0000000  0.80000000     TRUE
    delta    0.00000000   2.0000000  2.00000000    FALSE
    skew     0.10000000  10.0000000  1.00000000    FALSE
    shape    1.00000000  10.0000000  4.00000000     TRUE
 Index List of Parameters to be Optimized:
    mu    ar1    ma1  omega alpha1  beta1  shape
     1      2      3      4      5      7     10
 Persistence:                  0.9


--- START OF TRACE ---
Selected Algorithm: nlminb

R coded nlminb Solver:

  0:     1737.2297: 0.0103654 0.162221 -0.0734719 0.100000
0.100000 0.800000  4.00000
  1:     1709.4794: 0.0103655 0.158897 -0.0767846 0.0721444
0.100744 0.785211  3.99939
  2:     1693.7152: 0.0103656 0.152116 -0.0835431 0.0474851
0.118564 0.784780  3.99903
  3:     1686.3205: 0.0103657 0.145525 -0.0901354 0.0489988
0.141566 0.799177  3.99905
  4:     1684.3202: 0.0103663 0.136821 -0.0989758 0.0231014
0.142114 0.799107  3.99824
  5:     1676.9314: 0.0103679 0.121111 -0.115019 0.0353113
0.144743 0.811834  3.99641
  6:     1672.6731: 0.0103716 0.118519 -0.118236 0.0209918
0.139368 0.835518  3.99274
```

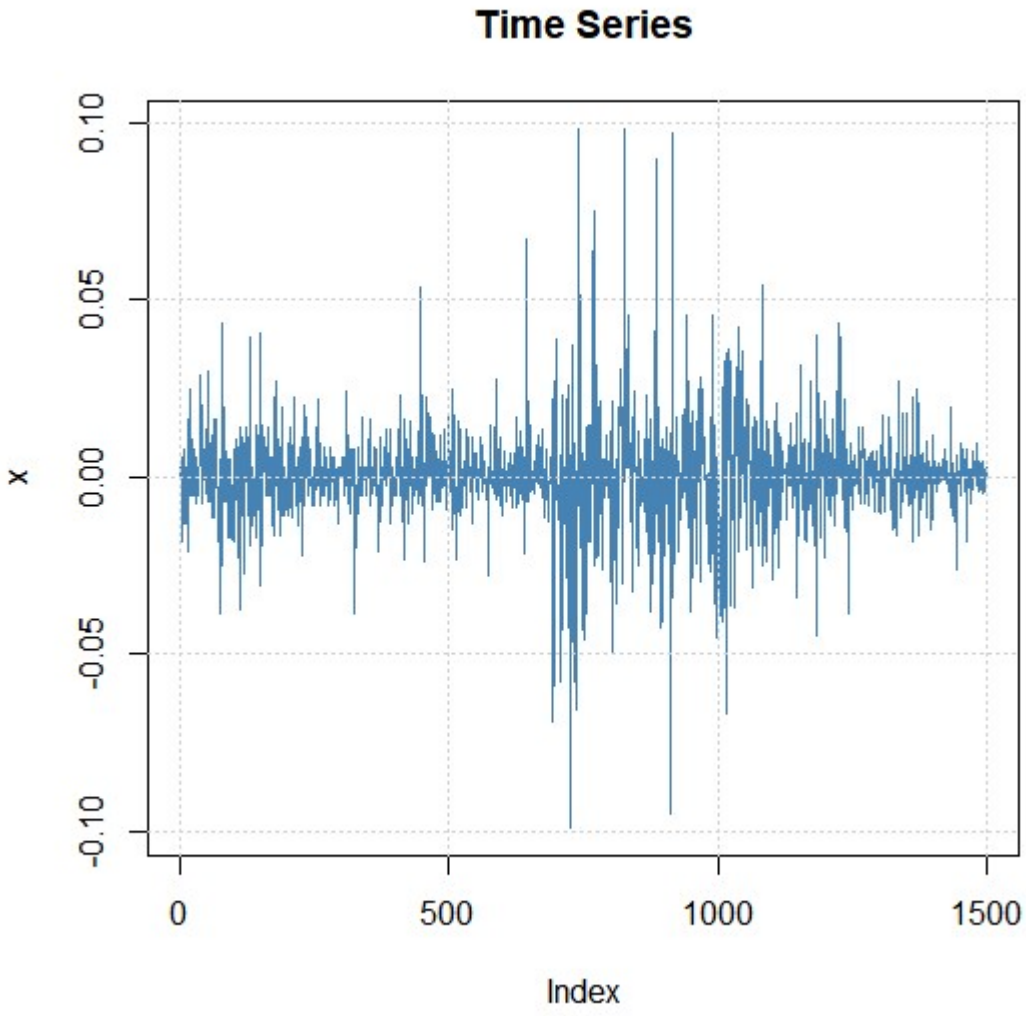
```
  7:     1672.5651: 0.0103729 0.118361 -0.118540 0.0128174
0.141335 0.845571  3.99155
  8:     1672.2967: 0.0103754 0.118819 -0.118383 0.0155316
0.137552 0.857585  3.98890
  9:     1671.4091: 0.0103798 0.111459 -0.126181 0.0131596
0.132117 0.859102  3.98436
 10:     1671.3364: 0.0103864 0.115478 -0.122698 0.0102110
0.127024 0.867089  3.97756
 11:     1671.0455: 0.0103950 0.113570 -0.125338 0.0109042
0.121011 0.873757  3.96857
 12:     1670.8252: 0.0104044 0.111996 -0.127694 0.00963238
0.114722 0.878927  3.95874
 13:     1670.5342: 0.0104305 0.113574 -0.128300 0.0104547
0.108134 0.882753  3.93059
 14:     1670.2578: 0.0104537 0.111527 -0.132628 0.00809538
0.100508 0.892629  3.90468
 15:     1670.0605: 0.0104791 0.112811 -0.133941 0.00830881
0.0993227 0.895727  3.87570
 16:     1669.7734: 0.0105039 0.116926 -0.132408 0.00815764
0.102884 0.892394  3.84722
 17:     1668.1689: 0.0108186 0.145659 -0.137739 0.00958739
0.118326 0.887967  3.49309
 18:     1666.7377: 0.0111398 0.155036 -0.160775 0.00575791
0.115036 0.901951  3.13862
 19:     1666.5159: 0.0111833 0.156242 -0.163954 0.00662426
0.106892 0.912354  3.12598
 20:     1666.1720: 0.0112615 0.161237 -0.167897 0.00781201
0.114052 0.902914  3.11480
 21:     1666.0883: 0.0113354 0.164617 -0.173610 0.00630957
0.107893 0.908580  3.10076
 22:     1665.9784: 0.0114288 0.170467 -0.180142 0.00704338
0.107305 0.909244  3.08644
 23:     1665.8391: 0.0118784 0.204002 -0.216268 0.00726013
0.104932 0.911055  3.03105
 24:     1665.6024: 0.0131149 0.316585 -0.333049 0.00427751
0.0923633 0.928090  2.93132
 25:     1665.4337: 0.0138980 0.463577 -0.484924 0.00482091
0.0970101 0.925033  2.90828
 26:     1665.3500: 0.0135027 0.512019 -0.535890 0.00513468
0.0978826 0.924534  2.89992
 27:     1665.1667: 0.0123381 0.588117 -0.617202 0.00580616
0.102410 0.921671  2.87121
 28:     1665.0581: 0.0115494 0.622870 -0.656233 0.00644891
0.103915 0.920370  2.86360
 29:     1664.8900: 0.00994495 0.691813 -0.727861
0.00710583 0.108554 0.917013  2.84235
```

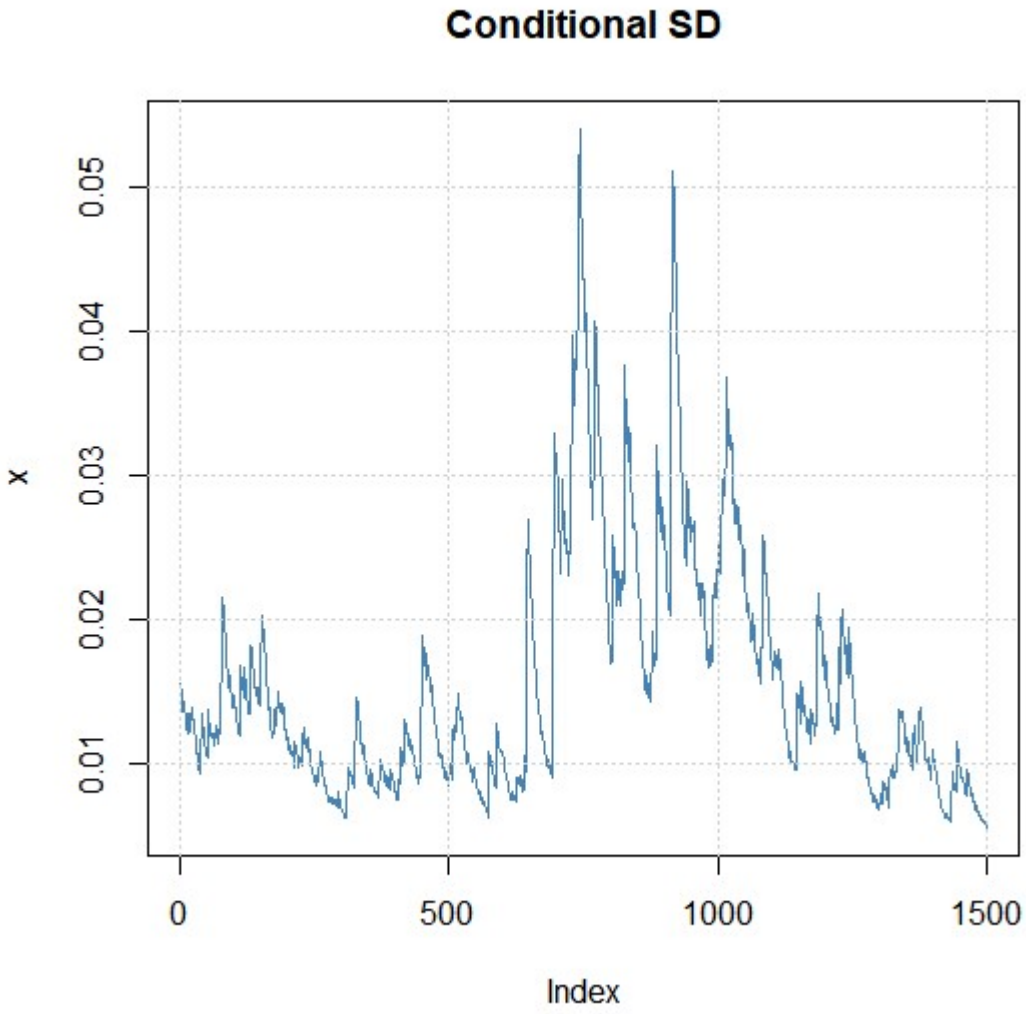
```
 30:     1664.6775: 0.00818608 0.715655 -0.755722
0.00817033 0.117106 0.910844  2.82086
 31:     1664.4274: 0.00652580 0.764326 -0.805038
0.00878412 0.120094 0.907895  2.83625
 32:     1664.1782: 0.00571635 0.785287 -0.822403
0.00926662 0.122376 0.907152  2.84156
 33:     1663.8692: 0.00411728 0.829131 -0.863849
0.00965410 0.126311 0.904373  2.83598
 34:     1663.7047: 0.00356755 0.848602 -0.880996
0.00896157 0.126988 0.903718  2.83424
 35:     1663.4570: 0.00246939 0.887158 -0.916235
0.00930922 0.128254 0.903326  2.83174
 36:     1663.2762: 0.00219012 0.903740 -0.924043
0.00900345 0.128122 0.904893  2.83521
 37:     1663.1478: 0.00191092 0.917235 -0.936874
0.00829966 0.127381 0.905077  2.83694
 38:     1663.0351: 0.00163531 0.931131 -0.949923
0.00840351 0.126661 0.905802  2.83917
 39:     1662.9658: 0.00140159 0.940196 -0.959012
0.00772368 0.123279 0.907665  2.85052
 40:     1662.8560: 0.00166455 0.929975 -0.951621
0.00593554 0.107188 0.916971  2.90174
 41:     1662.7965: 0.00120703 0.945284 -0.962976
0.00481387 0.0943480 0.924902  2.93688
 42:     1662.7930: 0.00123459 0.947171 -0.964412
0.00530045 0.0951023 0.924879  2.93508
 43:     1662.7699: 0.00124741 0.947370 -0.964562
0.00499458 0.0952252 0.924532  2.93662
 44:     1662.7646: 0.00127309 0.944848 -0.962429
0.00511230 0.0963204 0.923600  2.93906
 45:     1662.7585: 0.00133744 0.943413 -0.961247
0.00531151 0.0977715 0.922239  2.94643
 46:     1662.7580: 0.00136045 0.942612 -0.960760
0.00529358 0.0980230 0.922075  2.94772
 47:     1662.7580: 0.00135866 0.942700 -0.960771
0.00531245 0.0980352 0.922002  2.94853
 48:     1662.7580: 0.00135813 0.942719 -0.960787
0.00530898 0.0980245 0.922017  2.94850
 49:     1662.7580: 0.00135817 0.942717 -0.960786
0.00530943 0.0980189 0.922018  2.94852

Final Estimate of the Negative LLH:
 LLH:  -4603.921    norm LLH:  -3.071329
           mu            ar1            ma1          omega
alpha1         beta1          shape
 2.076599e-05  9.427166e-01 -9.607860e-01  1.241217e-06
9.801889e-02  9.220182e-01  2.948525e+00
```

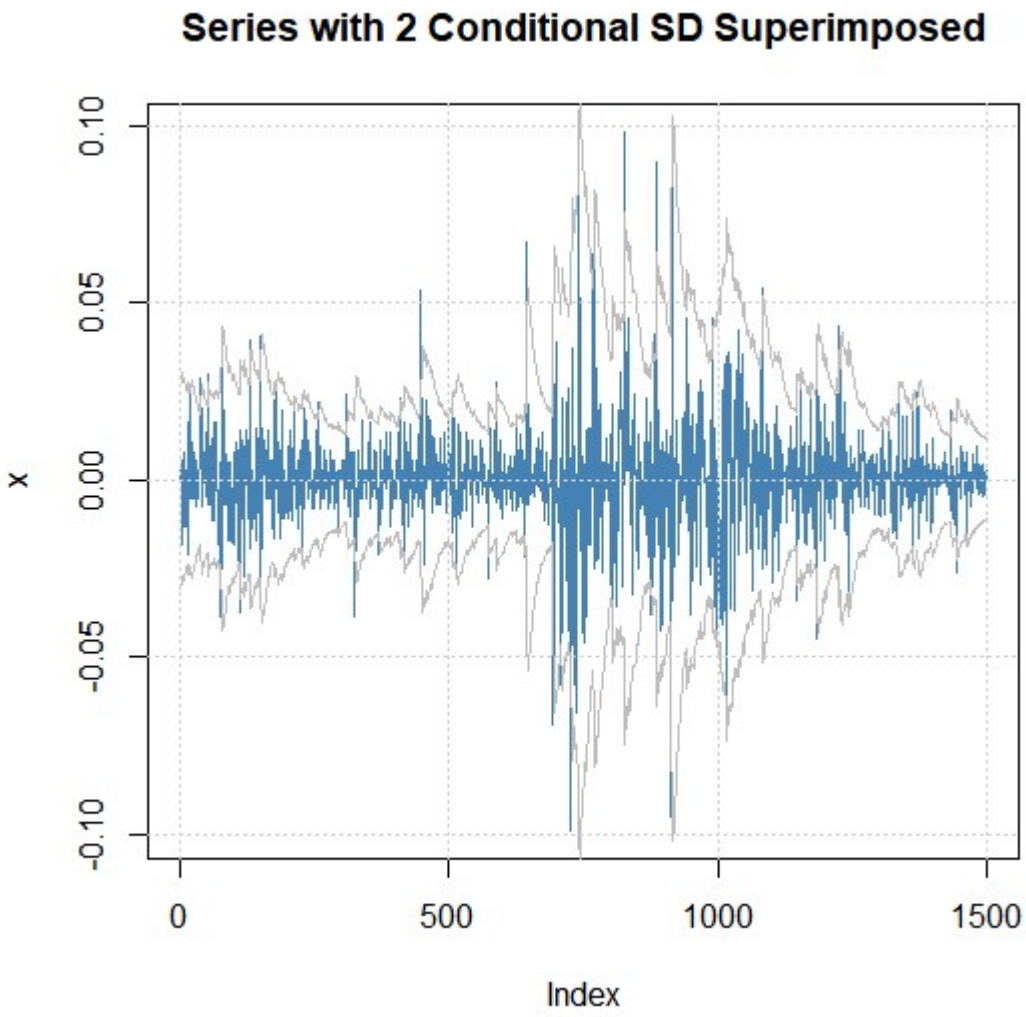
```
R-optimhess Difference Approximated Hessian Matrix:
                  mu           ar1           ma1
omega        alpha1         beta1
mu     -1.556818e+10 -8041396.8524 -3.493776e+06
1.226140e+09  5.869674e+05  4.015076e+05
ar1    -8.041397e+06   -24351.2565 -2.571409e+04
1.576032e+07  1.086196e+03  1.626774e+03
ma1    -3.493776e+06   -25714.0856 -2.982219e+04
1.728899e+07  7.455998e+02  1.288392e+03
omega   1.226140e+09 15760317.3932  1.728899e+07 -
5.589552e+12 -2.872150e+08 -5.557153e+08
alpha1  5.869674e+05     1086.1959  7.455998e+02 -
2.872150e+08 -2.948495e+04 -4.681338e+04
beta1   4.015076e+05     1626.7744  1.288392e+03 -
5.557153e+08 -4.681338e+04 -7.938210e+04
shape   4.979607e+04      102.4048  7.588068e+01 -
1.865740e+07 -1.751802e+03 -2.854076e+03
               shape
mu      4.979607e+04
ar1     1.024048e+02
ma1     7.588068e+01
omega  -1.865740e+07
alpha1 -1.751802e+03
beta1  -2.854076e+03
shape  -1.198930e+02
attr(,"time")
Time difference of 0.09374809 secs

--- END OF TRACE ---


Time to Estimate Parameters:
 Time difference of 0.3642411 secs

Title:
 GARCH Modelling

Call:
 garchFit(formula = ~arma(1, 1) + garch(1, 1), data = x,
cond.dist = "std")

Mean and Variance Equation:
 data ~ arma(1, 1) + garch(1, 1)
<environment: 0x0000000007a8b6b0>
 [data = x]
```

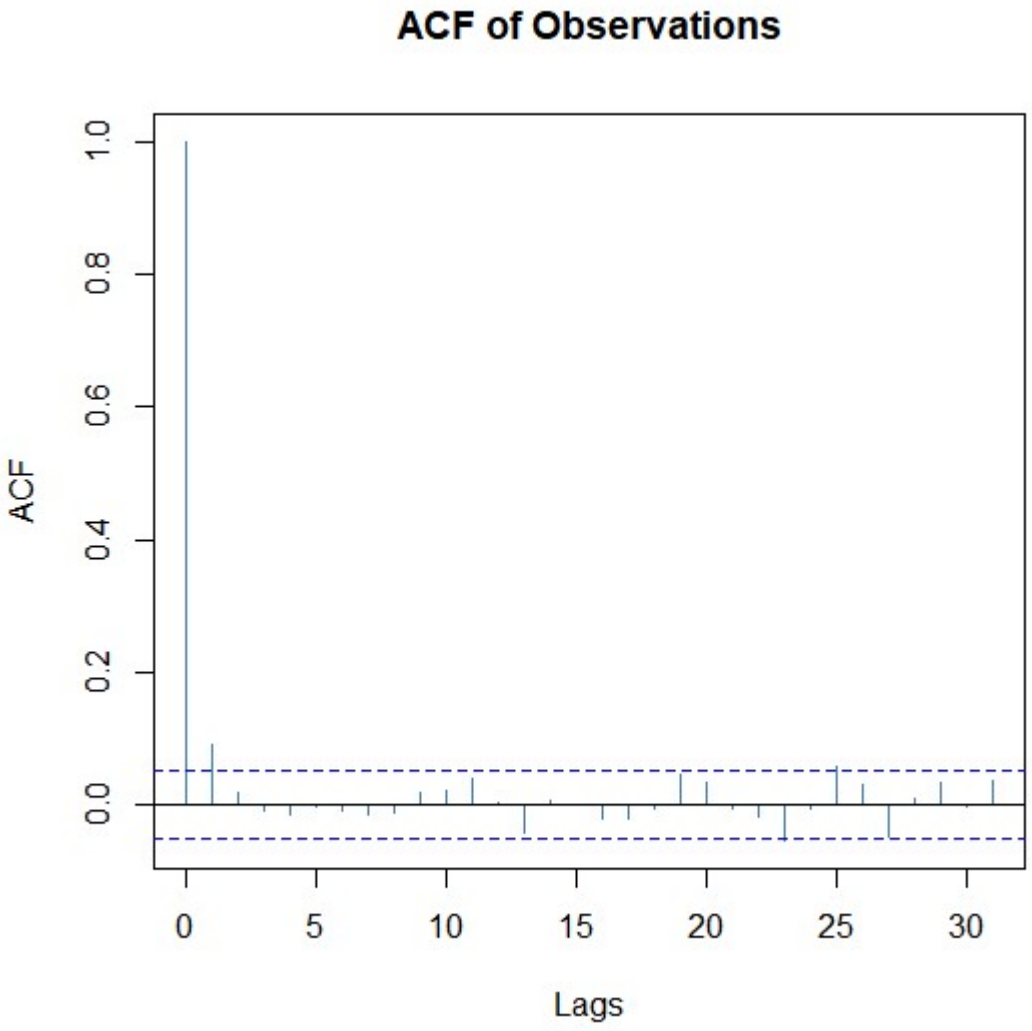
```
Conditional Distribution:
 std

Coefficient(s):
         mu           ar1           ma1         omega
alpha1         beta1         shape
 2.0766e-05    9.4272e-01  -9.6079e-01    1.2412e-06
9.8019e-02    9.2202e-01    2.9485e+00

Std. Errors:
 based on Hessian

Error Analysis:
         Estimate  Std. Error  t value Pr(>|t|)
mu      2.077e-05   1.688e-05    1.230  0.21873
ar1     9.427e-01   4.462e-02   21.129  < 2e-16 ***
ma1    -9.608e-01   3.720e-02  -25.830  < 2e-16 ***
omega   1.241e-06   1.107e-06    1.121  0.26215
alpha1  9.802e-02   3.489e-02    2.809  0.00497 **
beta1   9.220e-01   2.613e-02   35.286  < 2e-16 ***
shape   2.949e+00   2.619e-01   11.259  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Log Likelihood:
 4603.921    normalized:  3.071329

Description:
 Mon Jan 01 13:02:33 2018 by user: ibin


Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  5346.221  0
 Shapiro-Wilk Test  R    W      0.9110185 0
 Ljung-Box Test     R    Q(10)  9.861711  0.4527086
 Ljung-Box Test     R    Q(15)  13.64625  0.5525097
 Ljung-Box Test     R    Q(20)  19.5071   0.4891169
 Ljung-Box Test     R^2  Q(10)  2.355941  0.9928129
 Ljung-Box Test     R^2  Q(15)  3.559716  0.9988601
 Ljung-Box Test     R^2  Q(20)  6.469548  0.9980963
 LM Arch Test       R    TR^2   3.115992  0.9946682

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-6.133318 -6.108509 -6.133361 -6.124075
```

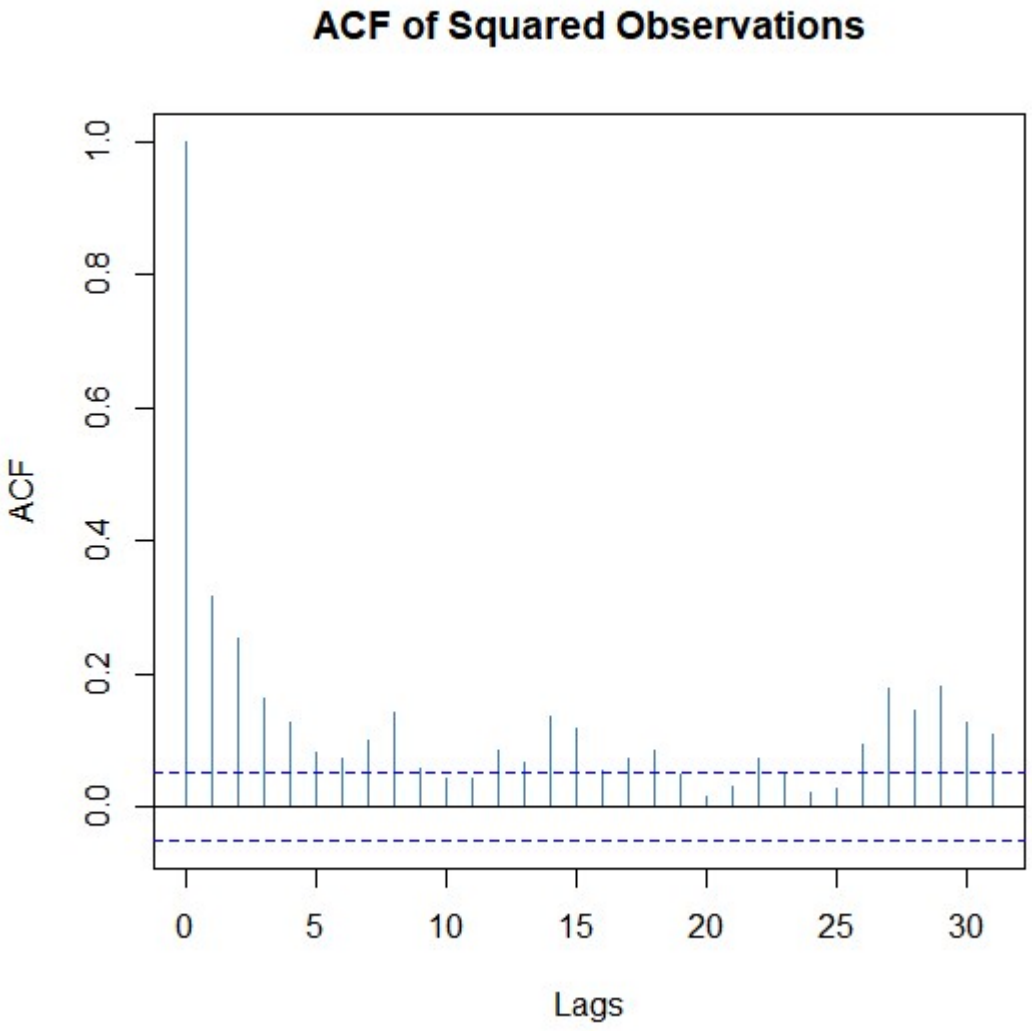
```
> plot(x.g )

Make a plot selection (or 0 to exit):

 1:   Time Series                                  2:
Conditional SD
 3:   Series with 2 Conditional SD Superimposed   4:   ACF
of Observations
 5:   ACF of Squared Observations                  6:
Cross Correlation
 7:   Residuals                                   8:
Conditional SDs
 9:   Standardized Residuals                      10:   ACF
of Standardized Residuals
11:   ACF of Squared Standardized Residuals       12:
Cross Correlation between r^2 and r
13:   QQ-Plot of Standardized Residuals

Selection: 1
```

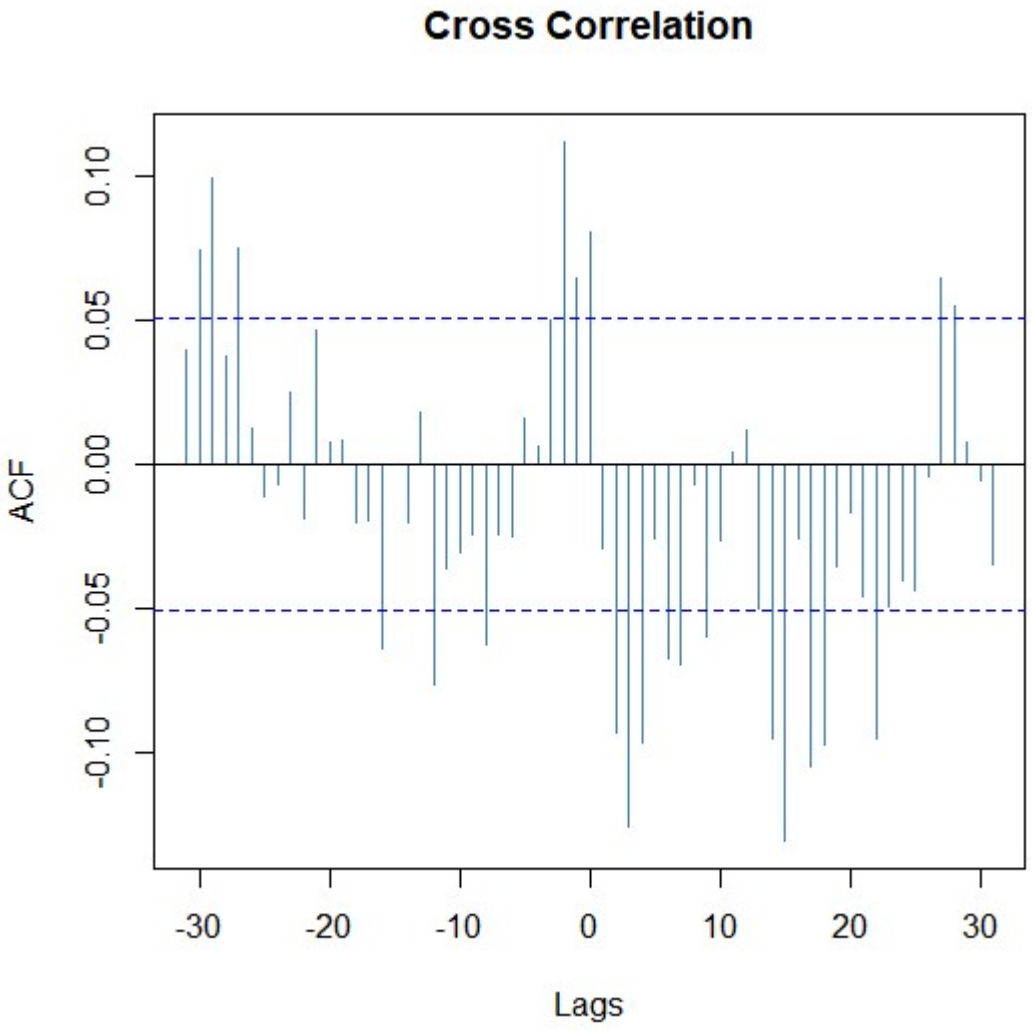

Time Series
x
0.10
0.05
0.00
-0.05
-0.10
0
500
1000
1500
Index

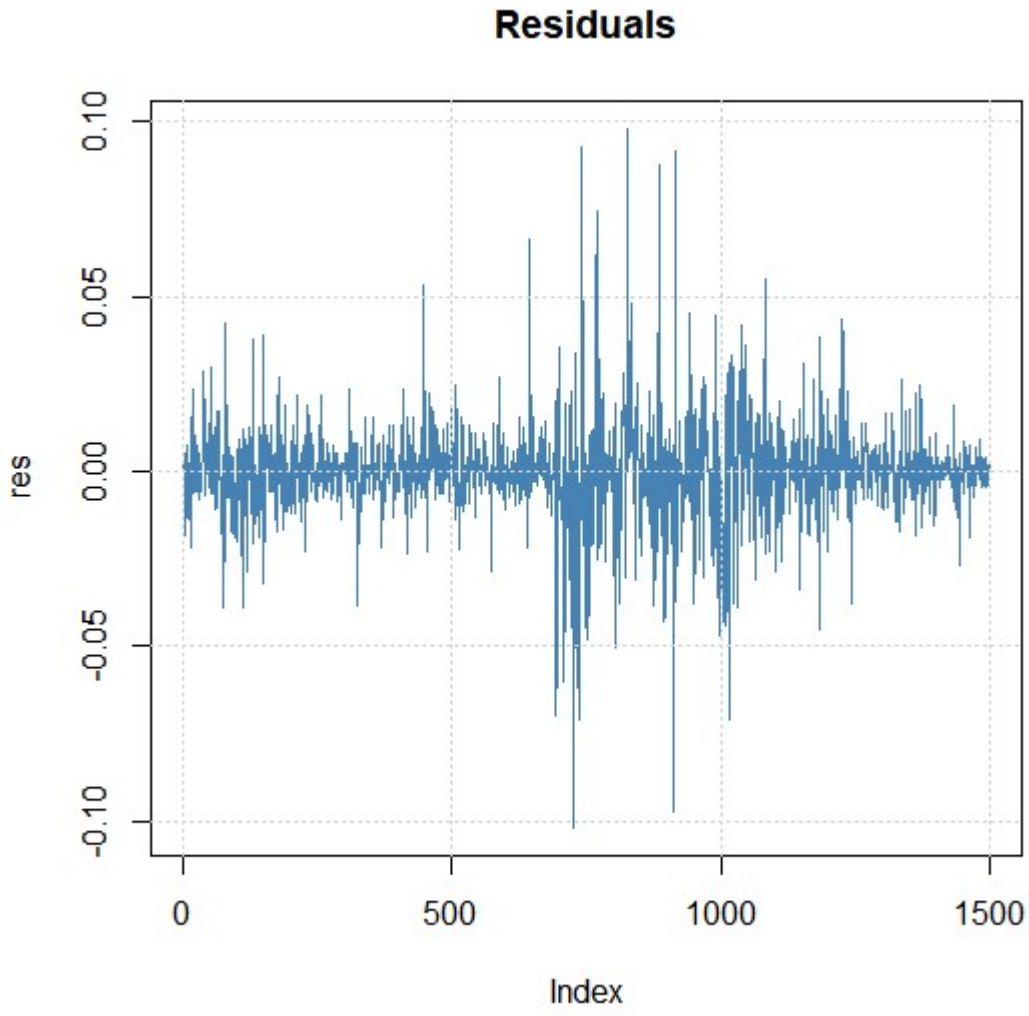

Conditional SD
x
0.01
0.02
0.03
0.04
0.05
0
500
1000
1500
Index

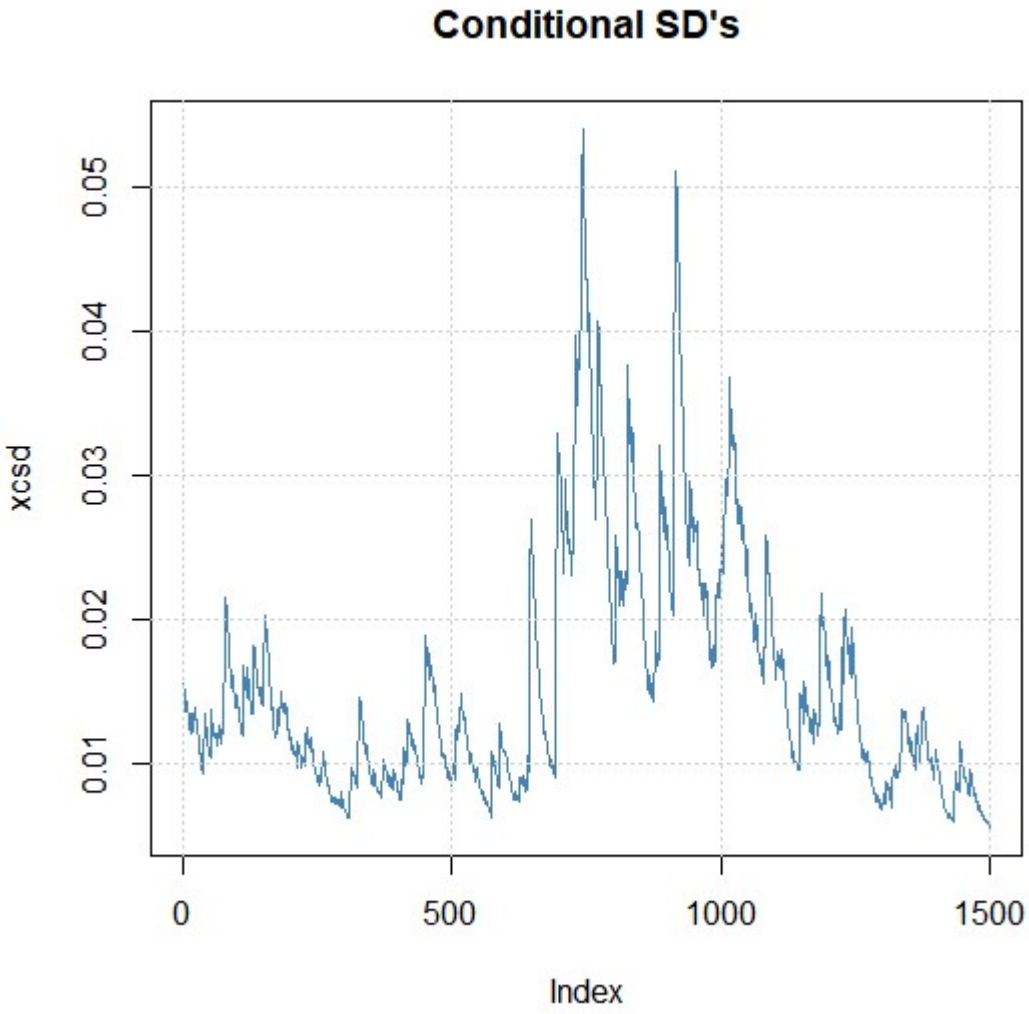

Series with 2 Conditional SD Superimposed
x
0.10
0.05
0.00
-0.05
-0.10
0
500
1000
1500
Index

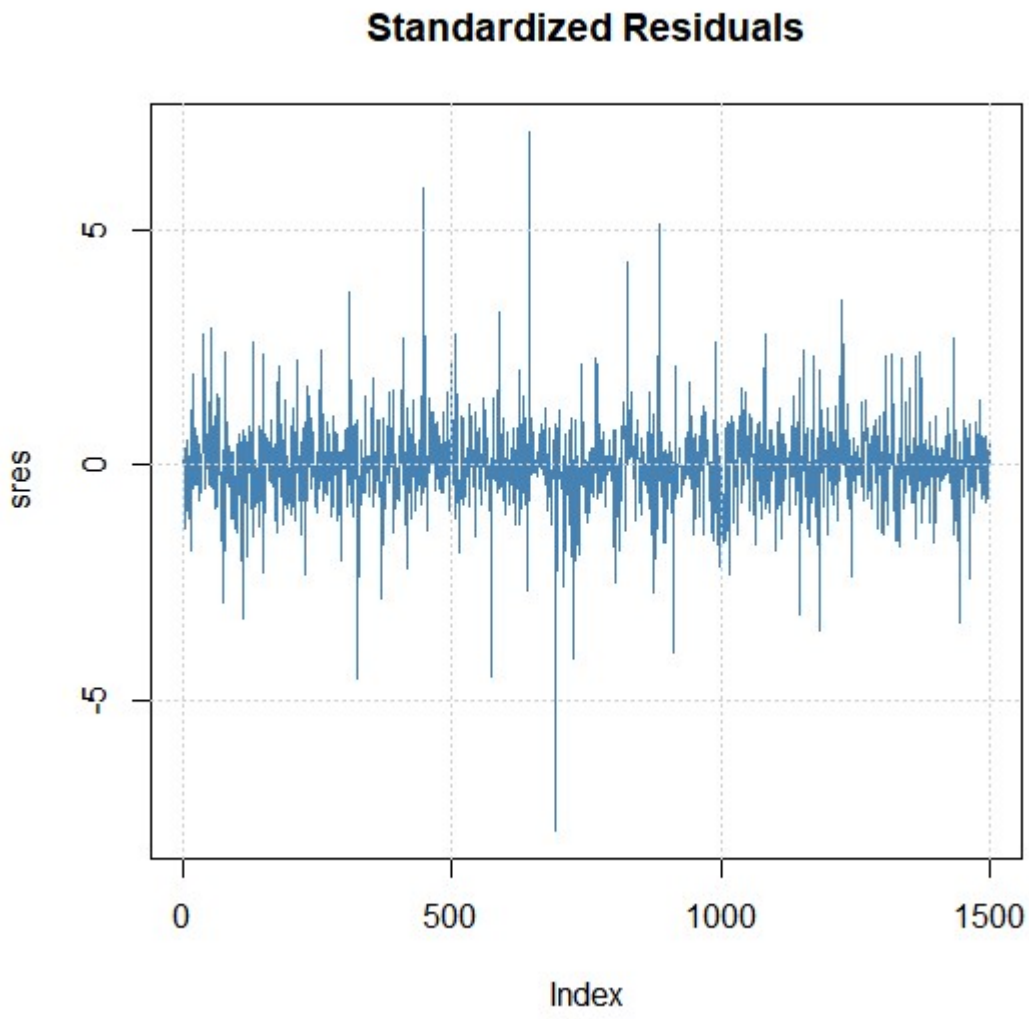

ACF of Observations
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
Lags

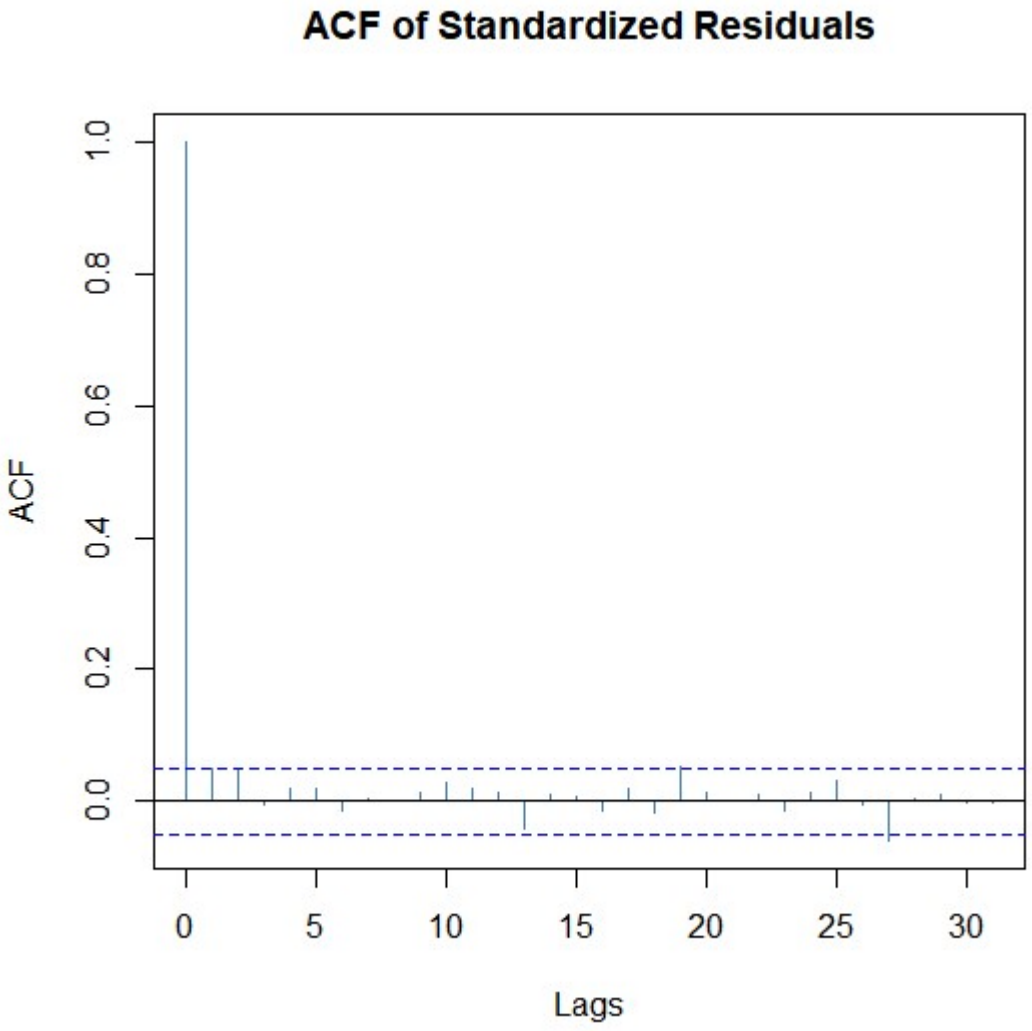

ACF of Squared Observations
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
Lags

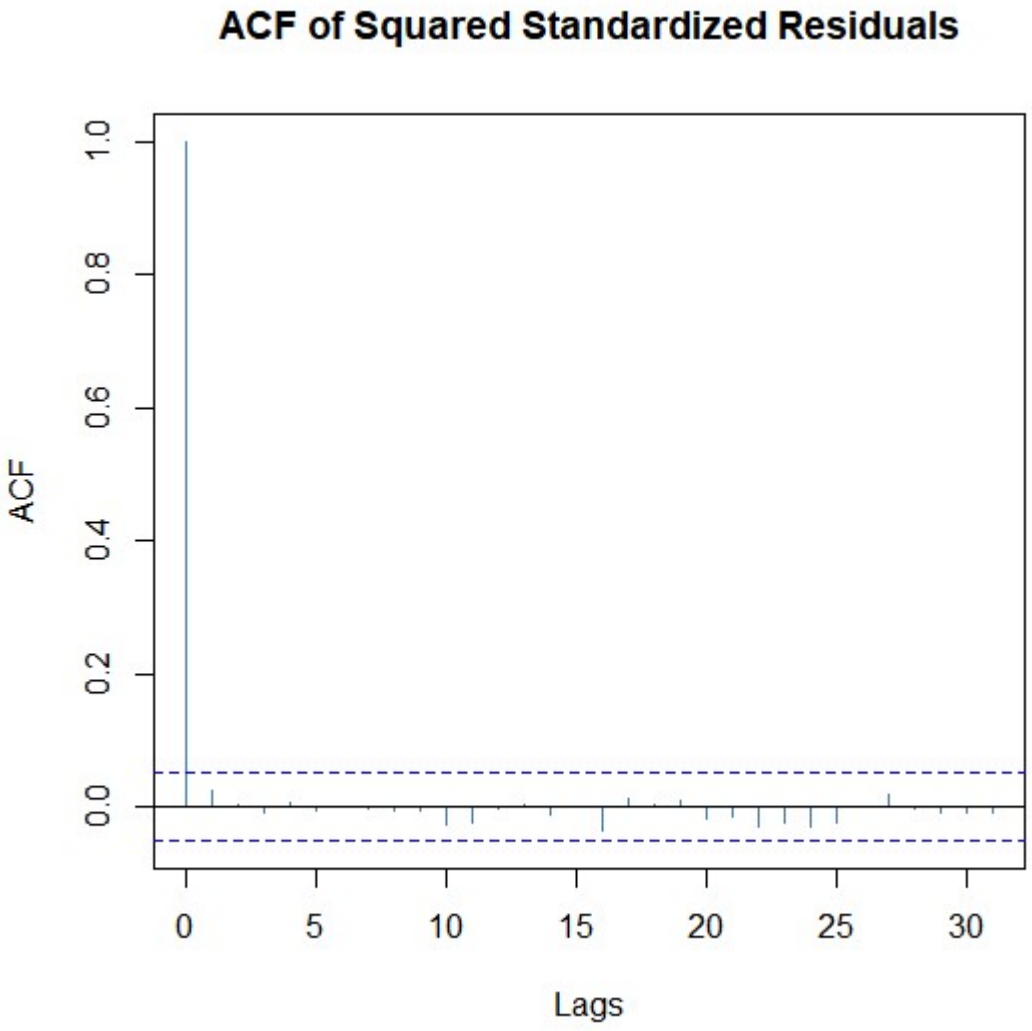

Cross Correlation
ACF
0.10
0.05
0.00
-0.05
-0.10
-30
-20
-10
0
10
20
30
Lags

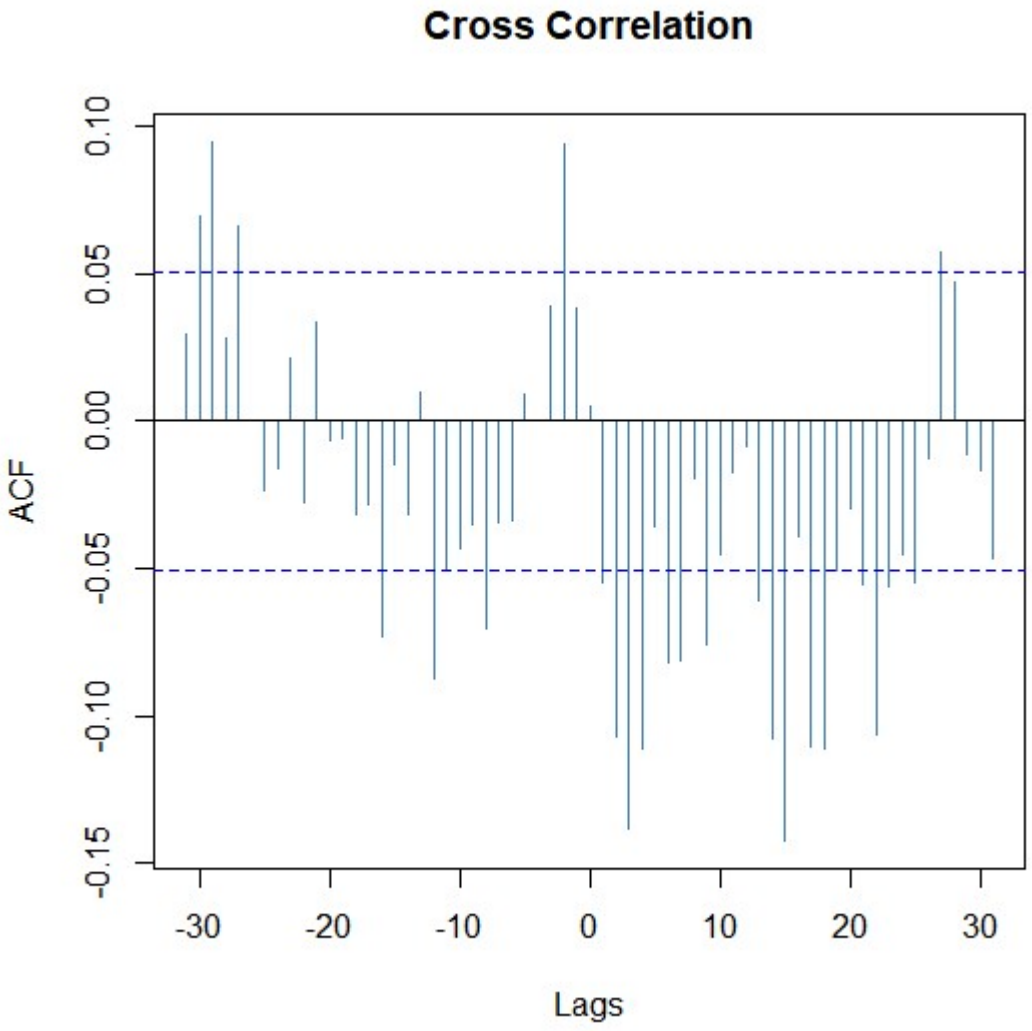

Residuals
res
0.10
0.05
0.00
-0.05
-0.10
0
500
1000
1500
Index

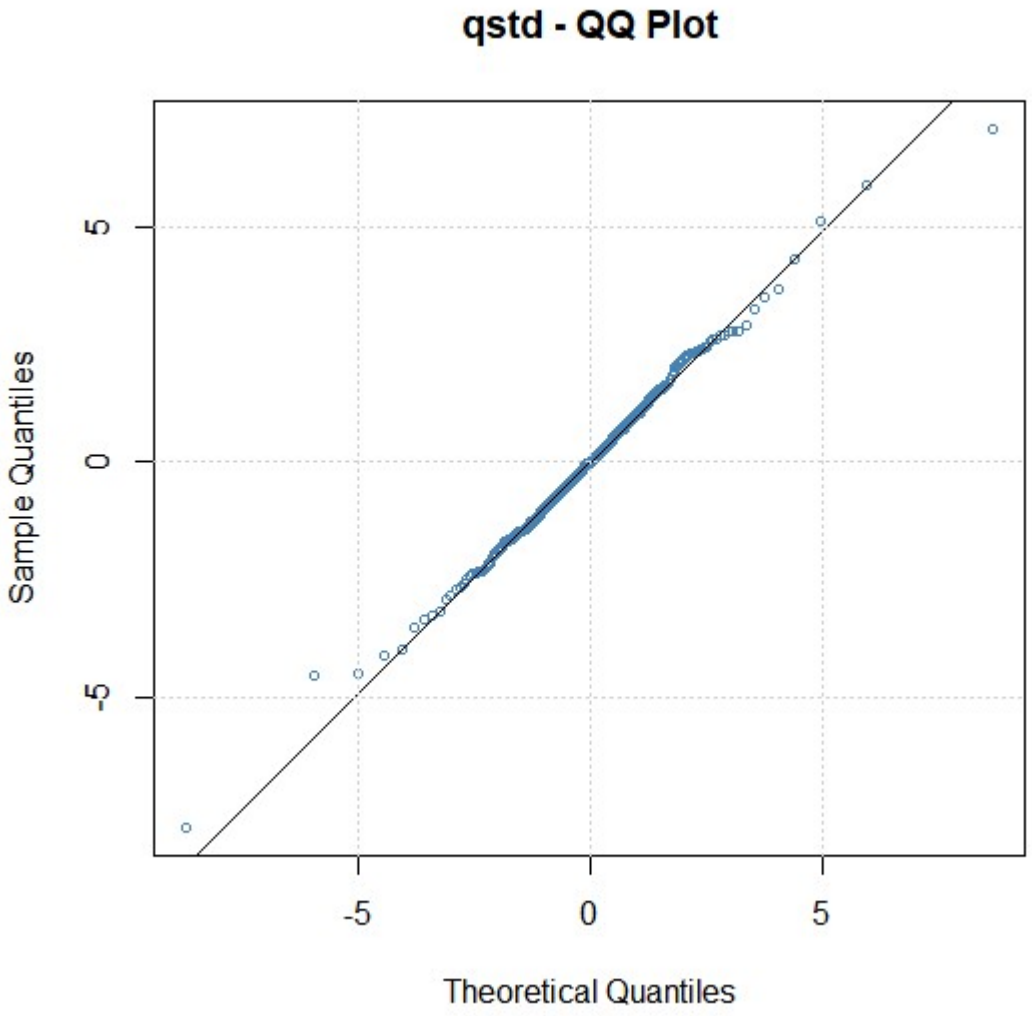

Conditional SD's
xcsd
0.01
0.02
0.03
0.04
0.05
0
500
1000
1500
Index


Standardized Residuals
sres
5
0
-5
0
500
1000
1500
Index


ACF of Standardized Residuals
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
30
Lags


ACF of Squared Standardized Residuals
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
Lags


Cross Correlation
ACF
0.10
0.05
0.00
-0.05
-0.10
-0.15
-30
-20
-10
0
10
20
30
Lags



ثانيا نطبق ARMA(0,1) و GARCH(1,1)

```
> summary(x2.g <- garchFit(~arma(0,1)+garch(1,1),
data=x,cond.dist='std'))

Series Initialization:
 ARMA Model:                 arma
 Formula Mean:               ~ arma(0, 1)
 GARCH Model:                garch
 Formula Variance:           ~ garch(1, 1)
 ARMA Order:                 0 1
 Max ARMA Order:             1
 GARCH Order:                1 1
 Max GARCH Order:            1
 Maximum Order:              1
 Conditional Dist:           std
 h.start:                    2
```

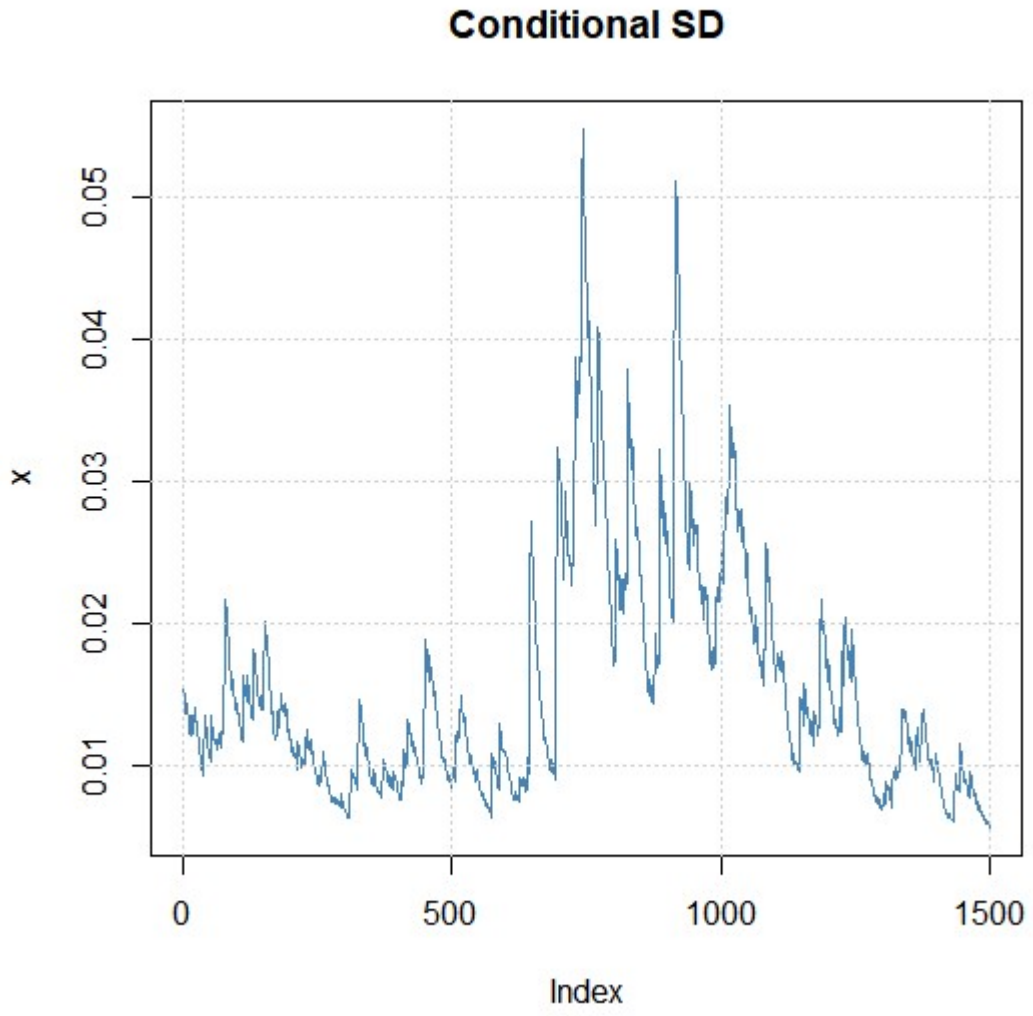
```
 llh.start:                1
 Length of Series:         1499
 Recursion Init:           mci
 Series Scale:             0.01528974

Parameter Initialization:
 Initial Parameters:          $params
 Limits of Transformations:   $U, $V
 Which Parameters are Fixed?  $includes
 Parameter Matrix:
                      U           V     params includes
    mu      -0.10367074   0.1036707 0.01036675     TRUE
    ma1     -0.99999999   1.0000000 0.08681115     TRUE
    omega    0.00000100 100.0000000 0.10000000     TRUE
    alpha1   0.00000001   1.0000000 0.10000000     TRUE
    gamma1  -0.99999999   1.0000000 0.10000000    FALSE
    beta1    0.00000001   1.0000000 0.80000000     TRUE
    delta    0.00000000   2.0000000 2.00000000    FALSE
    skew     0.10000000  10.0000000 1.00000000    FALSE
    shape    1.00000000  10.0000000 4.00000000     TRUE
 Index List of Parameters to be Optimized:
    mu    ma1  omega alpha1  beta1  shape
     1      2      3      4      6      9
 Persistence:                  0.9


--- START OF TRACE ---
Selected Algorithm: nlminb

R coded nlminb Solver:

  0:     1737.2366: 0.0103668 0.0868112 0.100000 0.100000
0.800000  4.00000
  1:     1709.7726: 0.0103668 0.0833656 0.0718051 0.100670
0.784993  3.99938
  2:     1694.5612: 0.0103669 0.0758967 0.0468389 0.119475
0.784965  3.99902
  3:     1688.6457: 0.0103671 0.0674235 0.0496831 0.145458
0.801633  3.99905
  4:     1682.6523: 0.0103681 0.0486483 0.0236293 0.144995
0.801594  3.99790
  5:     1676.7422: 0.0103699 0.0204845 0.0337235 0.148688
0.812520  3.99573
  6:     1672.7849: 0.0103732 0.00304183 0.0201319 0.140740
0.834189  3.99240
  7:     1672.5771: 0.0103734 0.00243673 0.0206556 0.143024
0.837014  3.99226
```

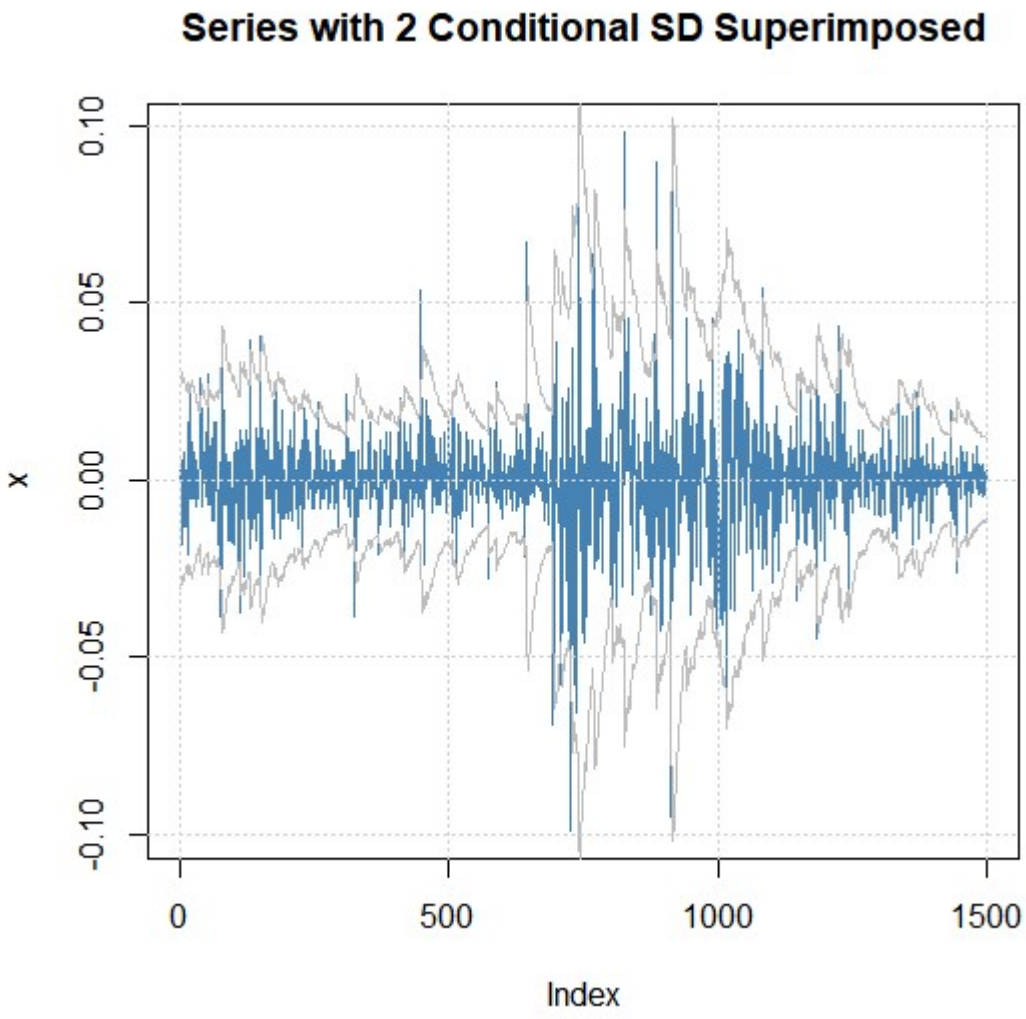
```
  8:     1672.3250: 0.0103738 0.00251038 0.0175163 0.142242
0.838806  3.99183
  9:     1672.1203: 0.0103745 0.00232845 0.0178213 0.141607
0.842397  3.99117
 10:     1671.8757: 0.0103762 0.00129327 0.0150680 0.138474
0.848238  3.98949
 11:     1671.6680: 0.0103787 -0.00219471 0.0151461
0.135272 0.853354  3.98688
 12:     1671.4927: 0.0103822 -0.00613631 0.0137544
0.131762 0.856766  3.98315
 13:     1671.1734: 0.0103955 -0.00228403 0.0111333
0.124175 0.869623  3.96904
 14:     1670.6272: 0.0104289 -0.0165732 0.0105362 0.106869
0.884169  3.93288
 15:     1670.6247: 0.0104290 -0.0165060 0.00931774
0.106714 0.883960  3.93282
 16:     1670.5646: 0.0104290 -0.0164794 0.00983980
0.106889 0.884253  3.93280
 17:     1670.5370: 0.0104293 -0.0161476 0.00944868
0.107339 0.885261  3.93253
 18:     1670.5349: 0.0104293 -0.0161010 0.00931974
0.107291 0.885290  3.93249
 19:     1670.5329: 0.0104294 -0.0160439 0.00938524
0.107281 0.885406  3.93244
 20:     1670.5292: 0.0104296 -0.0159619 0.00925299
0.107199 0.885479  3.93219
 21:     1668.9559: 0.0107507 0.0178591 0.00444697
0.0728189 0.926368  3.58174
 22:     1667.6114: 0.0110964 0.0124793 0.00597563
0.0952057 0.910779  3.22690
 23:     1666.4610: 0.0112942 -0.000760589 0.00445058
0.0809205 0.930090  3.13828
 24:     1666.3632: 0.0114823 -0.00585499 0.00258241
0.0761488 0.935492  3.10396
 25:     1666.1624: 0.0118552 -0.00760651 0.00339598
0.0772800 0.934572  3.08696
 26:     1666.0854: 0.0122630 -0.00856104 0.00385565
0.0796293 0.931561  3.08198
 27:     1665.8884: 0.0137489 -0.0177237 0.00325812
0.0787754 0.934209  3.03460
 28:     1665.6826: 0.0169712 -0.0216138 0.00459687
0.0916952 0.923634  3.04667
 29:     1665.6126: 0.0185106 -0.0230683 0.00497450
0.0926427 0.924249  2.97973
 30:     1665.5904: 0.0195867 -0.0239203 0.00520031
0.0933542 0.924095  2.97723
```

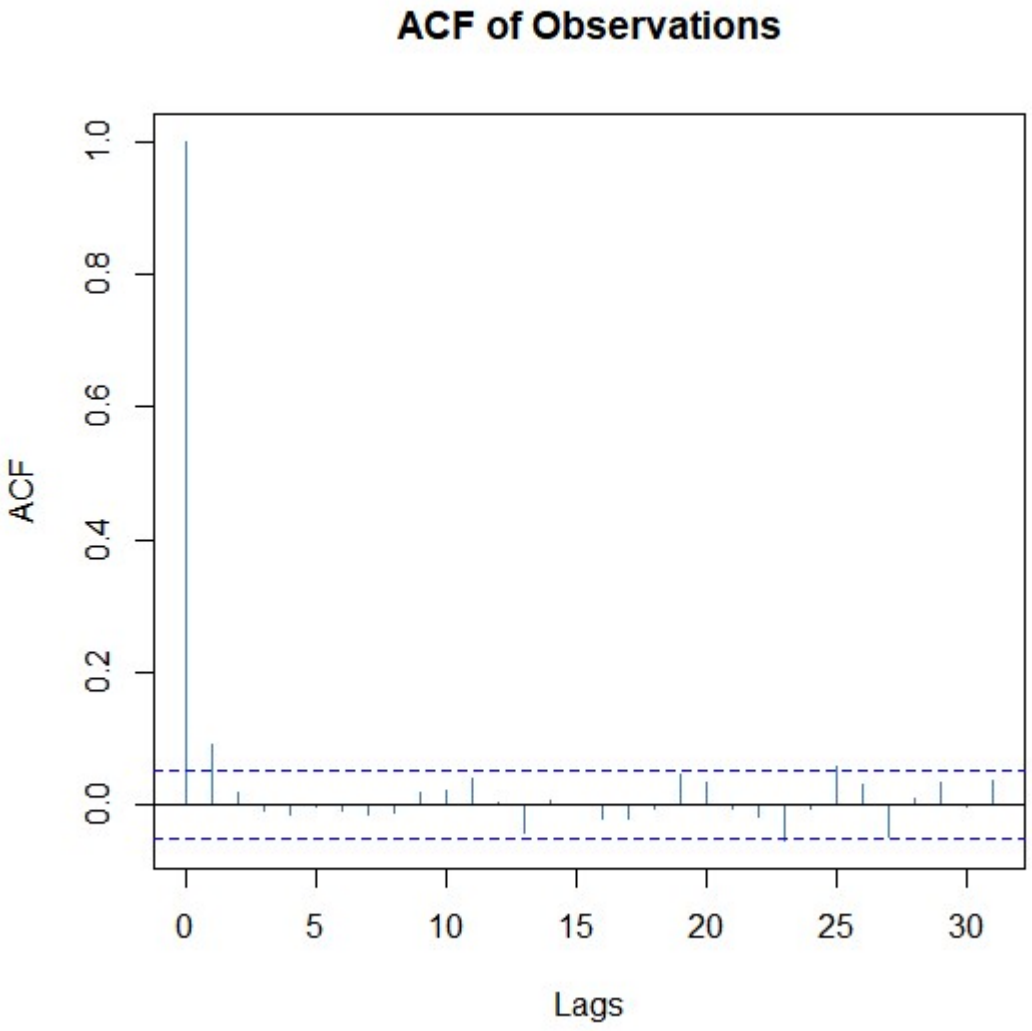
```
 31:     1665.5838: 0.0197572 -0.0232652 0.00536802
0.0957587 0.922832  2.96429
 32:     1665.5818: 0.0194966 -0.0232065 0.00548221
0.0971229 0.922127  2.95750
 33:     1665.5816: 0.0193427 -0.0233547 0.00552527
0.0975070 0.921928  2.95540
 34:     1665.5816: 0.0193238 -0.0233849 0.00553160
0.0975310 0.921908  2.95530
 35:     1665.5816: 0.0193224 -0.0233871 0.00553289
0.0975315 0.921903  2.95535

Final Estimate of the Negative LLH:
 LLH:  -4601.098    norm LLH:  -3.069445
           mu            ma1          omega         alpha1
beta1          shape
 2.954346e-04 -2.338708e-02  1.293458e-06  9.753150e-02
9.219035e-01  2.955347e+00

R-optimhess Difference Approximated Hessian Matrix:
                  mu            ma1          omega
alpha1          beta1          shape
mu     -25628786.922   -8967.32046  1.597822e+07
2.564225e+04  1.995432e+04  2.125748e+03
ma1         -8967.320   -1726.37078  4.977395e+06
6.331666e+02  8.843456e+02  5.960177e+01
omega   15978224.097 4977395.42171 -5.484305e+12 -
2.853894e+08 -5.514558e+08 -1.839913e+07
alpha1     25642.248     633.16660 -2.853894e+08 -
2.956196e+04 -4.677487e+04 -1.734731e+03
beta1      19954.323     884.34556 -5.514558e+08 -
4.677487e+04 -7.890739e+04 -2.822389e+03
shape       2125.748      59.60177 -1.839913e+07 -
1.734731e+03 -2.822389e+03 -1.174696e+02
attr(,"time")
Time difference of 0.07811594 secs

--- END OF TRACE ---


Time to Estimate Parameters:
 Time difference of 0.2500031 secs

Title:
 GARCH Modelling

Call:
```

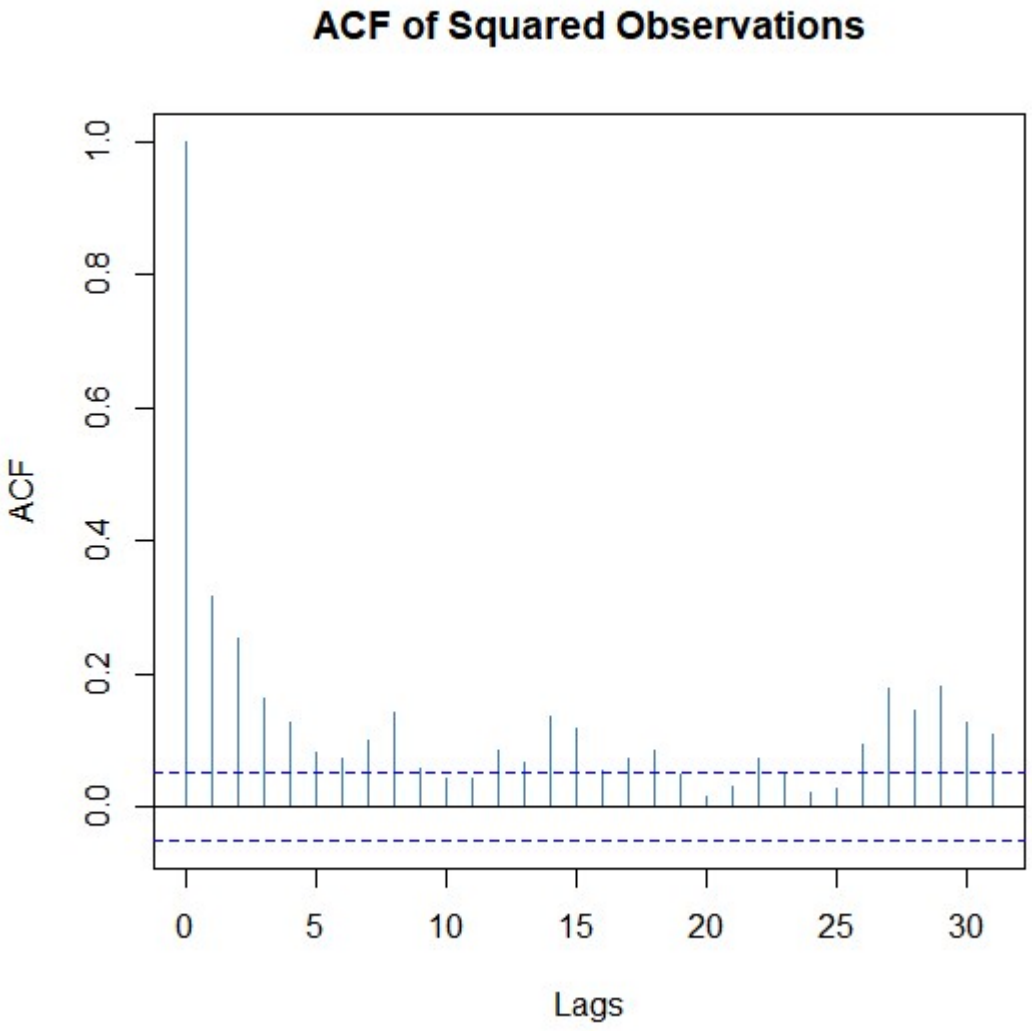
```
 garchFit(formula = ~arma(0, 1) + garch(1, 1), data = x,
cond.dist = "std")

Mean and Variance Equation:
 data ~ arma(0, 1) + garch(1, 1)
<environment: 0x0000000024807960>
 [data = x]

Conditional Distribution:
 std

Coefficient(s):
         mu          ma1        omega       alpha1
beta1        shape
 2.9543e-04  -2.3387e-02   1.2935e-06   9.7532e-02
9.2190e-01   2.9553e+00

Std. Errors:
 based on Hessian

Error Analysis:
         Estimate  Std. Error  t value Pr(>|t|)
mu      2.954e-04   1.985e-04    1.489  0.13661
ma1    -2.339e-02   2.450e-02   -0.955  0.33982
omega   1.293e-06   1.205e-06    1.074  0.28295
alpha1  9.753e-02   3.715e-02    2.625  0.00866 **
beta1   9.219e-01   2.877e-02   32.046  < 2e-16 ***
shape   2.955e+00   2.666e-01   11.085  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘
’ 1

Log Likelihood:
 4601.098    normalized:  3.069445

Description:
 Mon Jan 01 13:13:37 2018 by user: ibin


Standardised Residuals Tests:
                                Statistic p-Value
 Jarque-Bera Test   R    Chi^2  5248.865  0
 Shapiro-Wilk Test  R    W      0.9114866 0
 Ljung-Box Test     R    Q(10)  8.606014  0.5698569
 Ljung-Box Test     R    Q(15)  12.8662   0.6126315
 Ljung-Box Test     R    Q(20)  18.02749  0.5855973
 Ljung-Box Test     R^2  Q(10)  2.526332  0.9904864
```

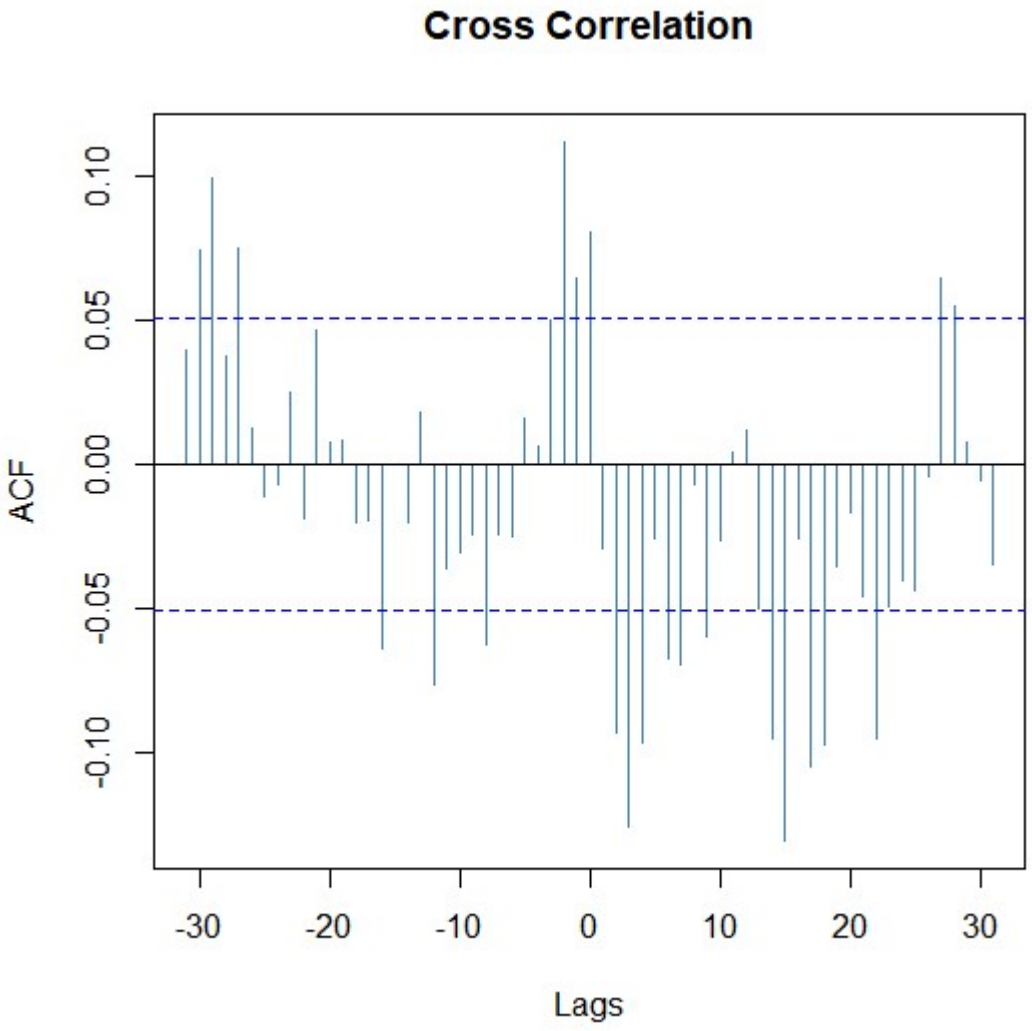
```
 Ljung-Box Test      R^2  Q(15)  3.755772  0.9984336
 Ljung-Box Test      R^2  Q(20)  6.647152  0.997692
 LM Arch Test        R    TR^2   3.346481  0.9925641

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-6.130884 -6.109620 -6.130916 -6.122962

> plot(x2.g)

Make a plot selection (or 0 to exit):

 1:   Time Series                                    2:
Conditional SD
 3:   Series with 2 Conditional SD Superimposed      4:   ACF
of Observations
 5:   ACF of Squared Observations                    6:
Cross Correlation
 7:   Residuals                                     8:
Conditional SDs
 9:   Standardized Residuals                       10:   ACF
of Standardized Residuals
11:   ACF of Squared Standardized Residuals        12:
Cross Correlation between r^2 and r
13:   QQ-Plot of Standardized Residuals

Selection: 1
```

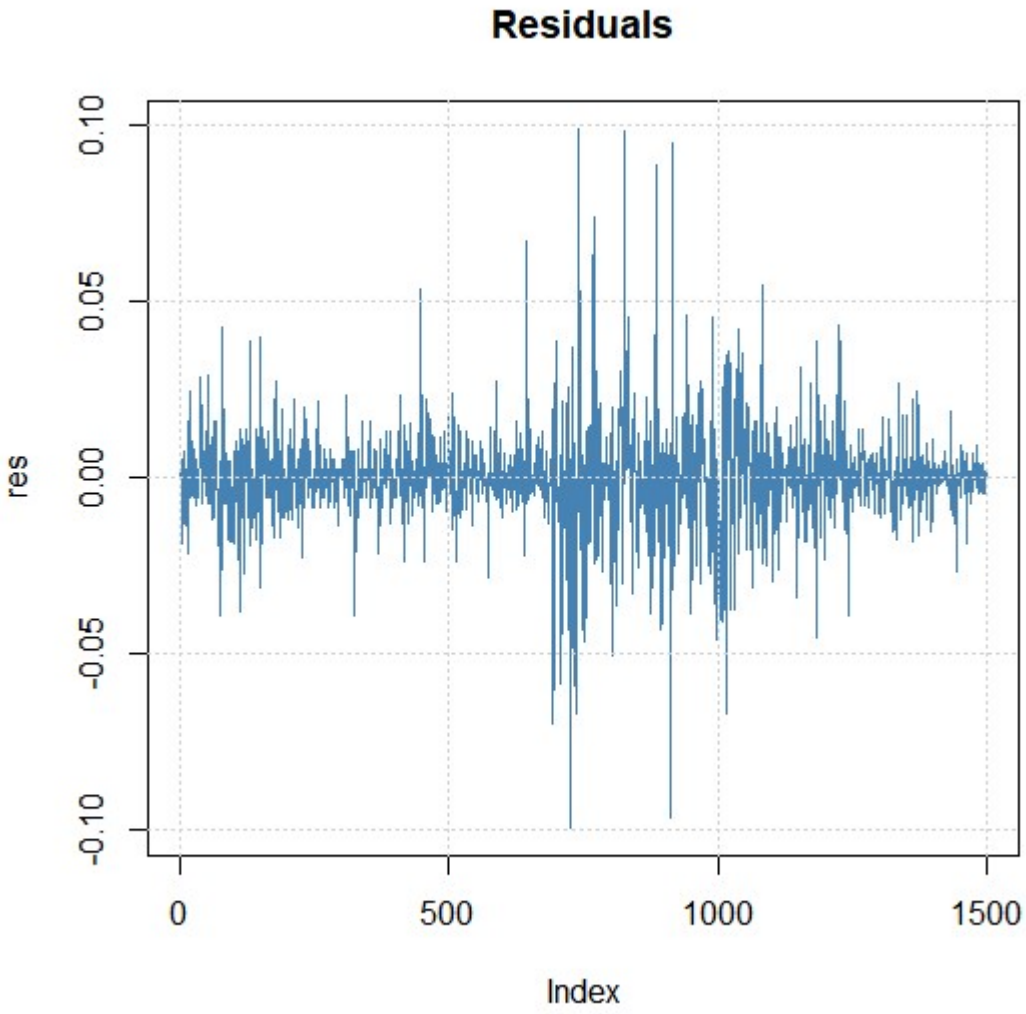

Conditional SD
x
0.01
0.02
0.03
0.04
0.05
0
500
1000
1500
Index

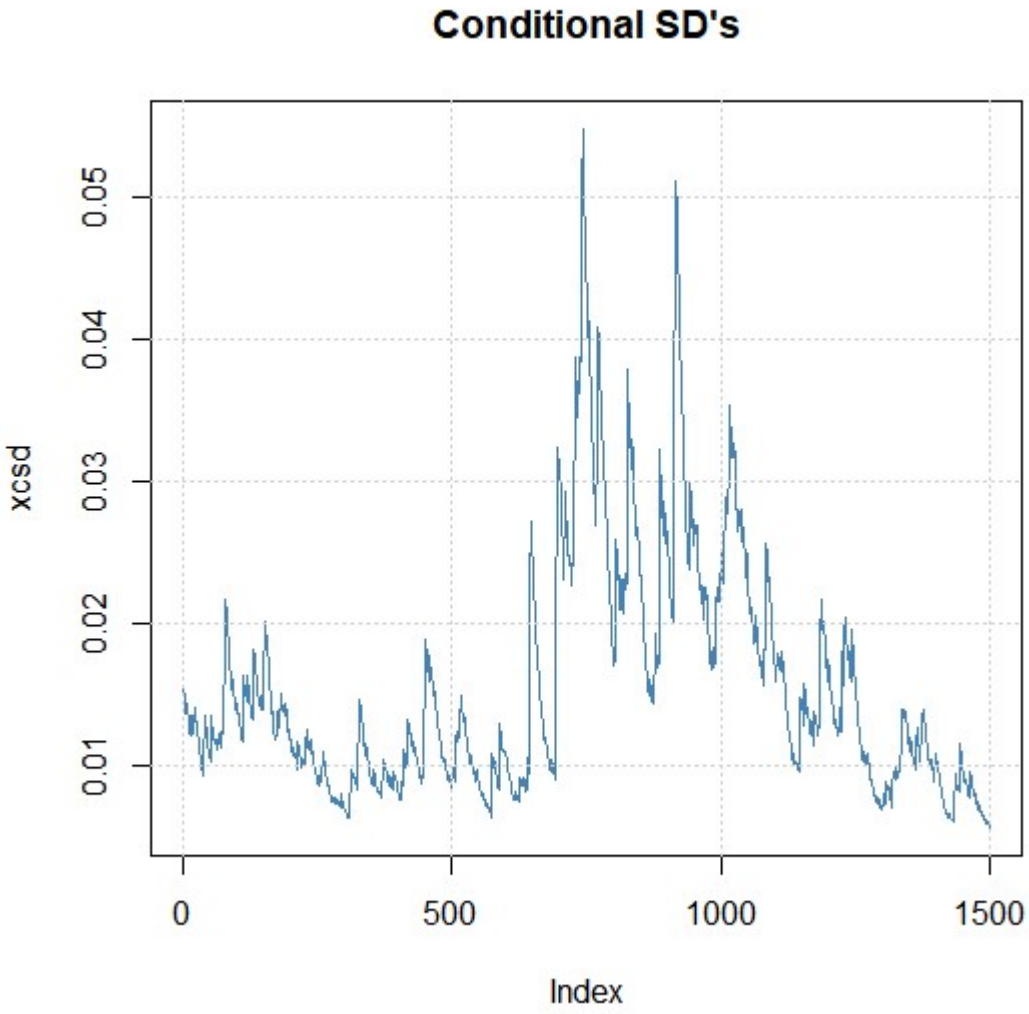

Series with 2 Conditional SD Superimposed
x
0.10
0.05
0.00
-0.05
-0.10
0
500
1000
1500
Index

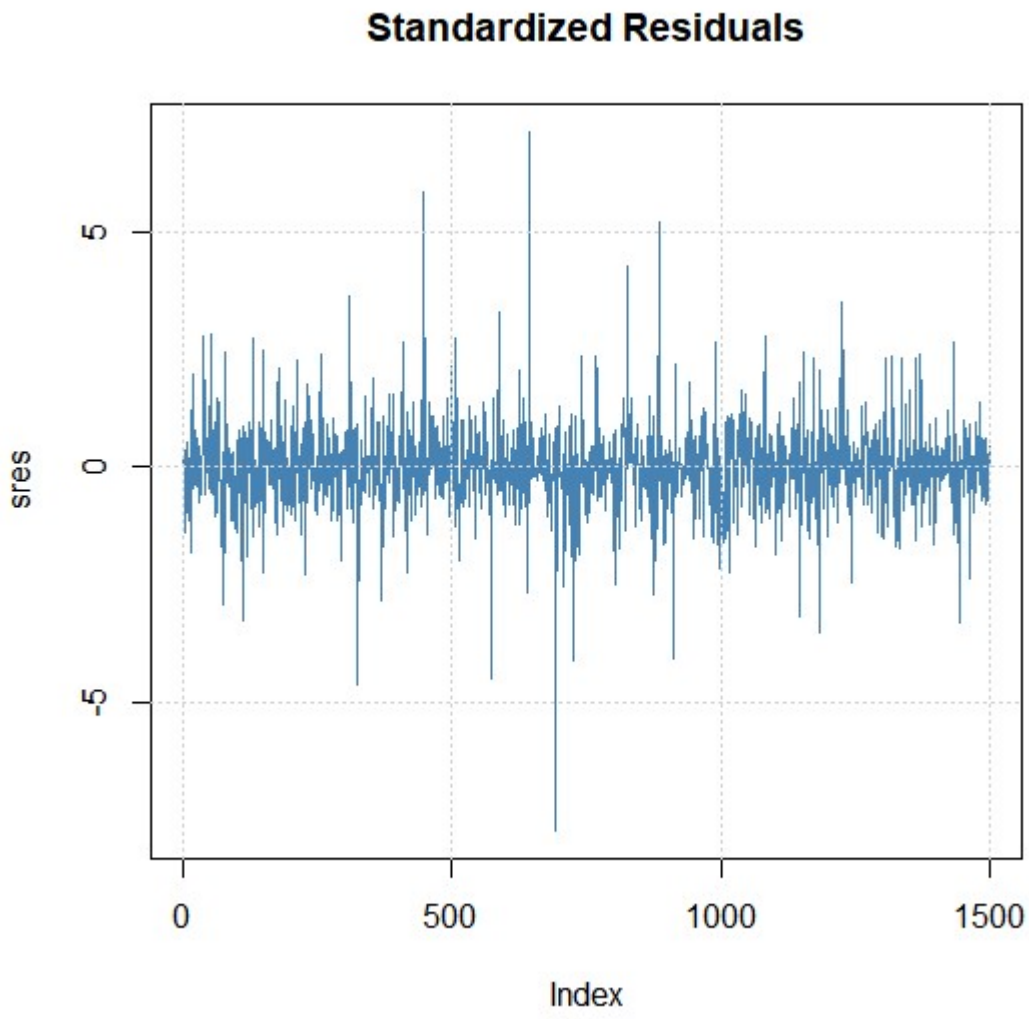

ACF of Observations
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
Lags

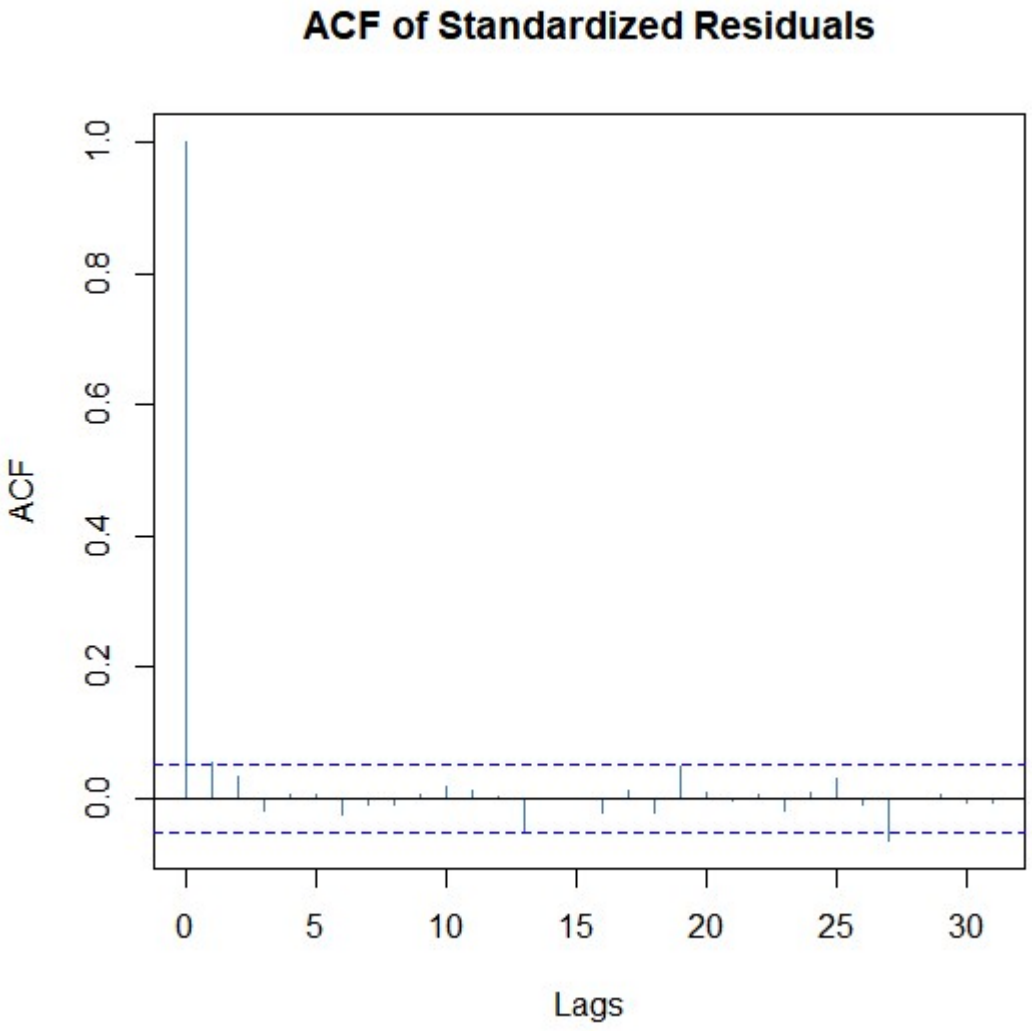

ACF of Squared Observations
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
Lags

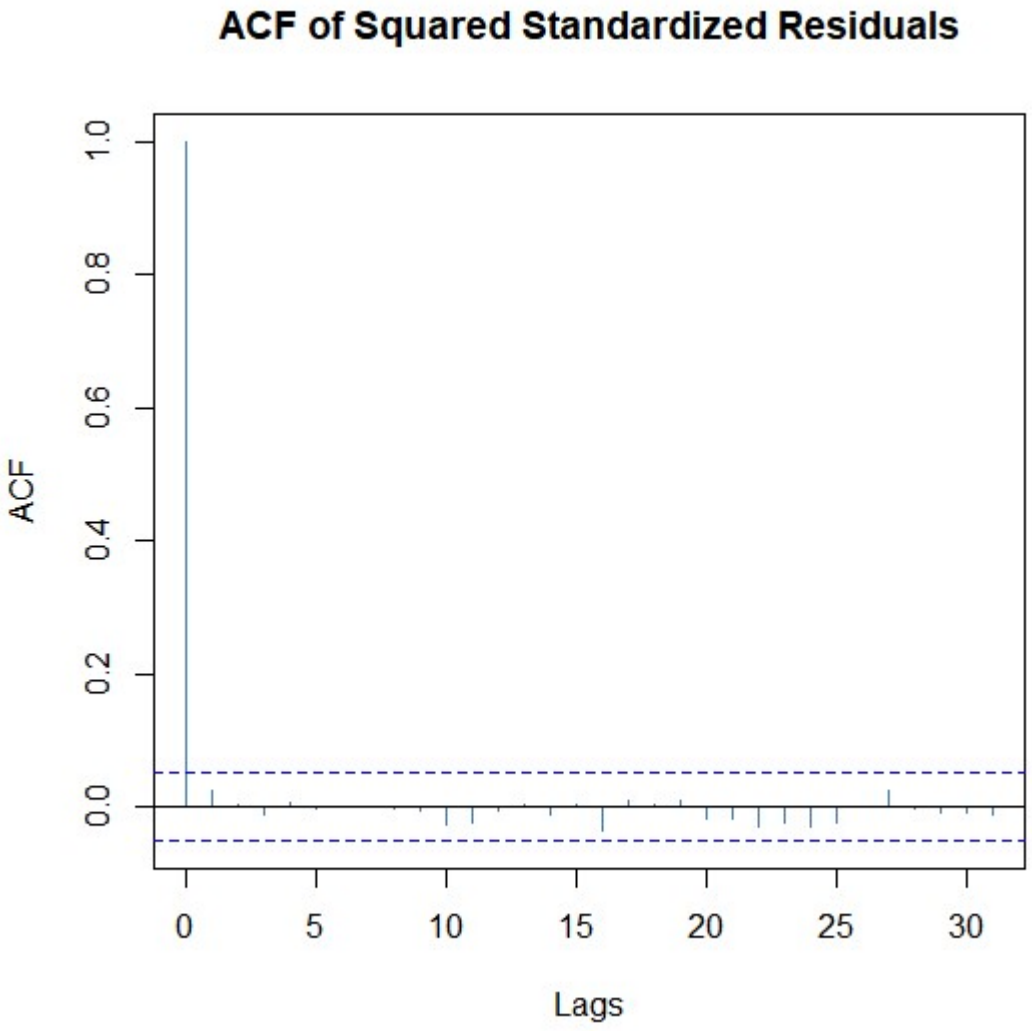

Cross Correlation
ACF
0.10
0.05
0.00
-0.05
-0.10
-30
-20
-10
0
10
20
30
Lags

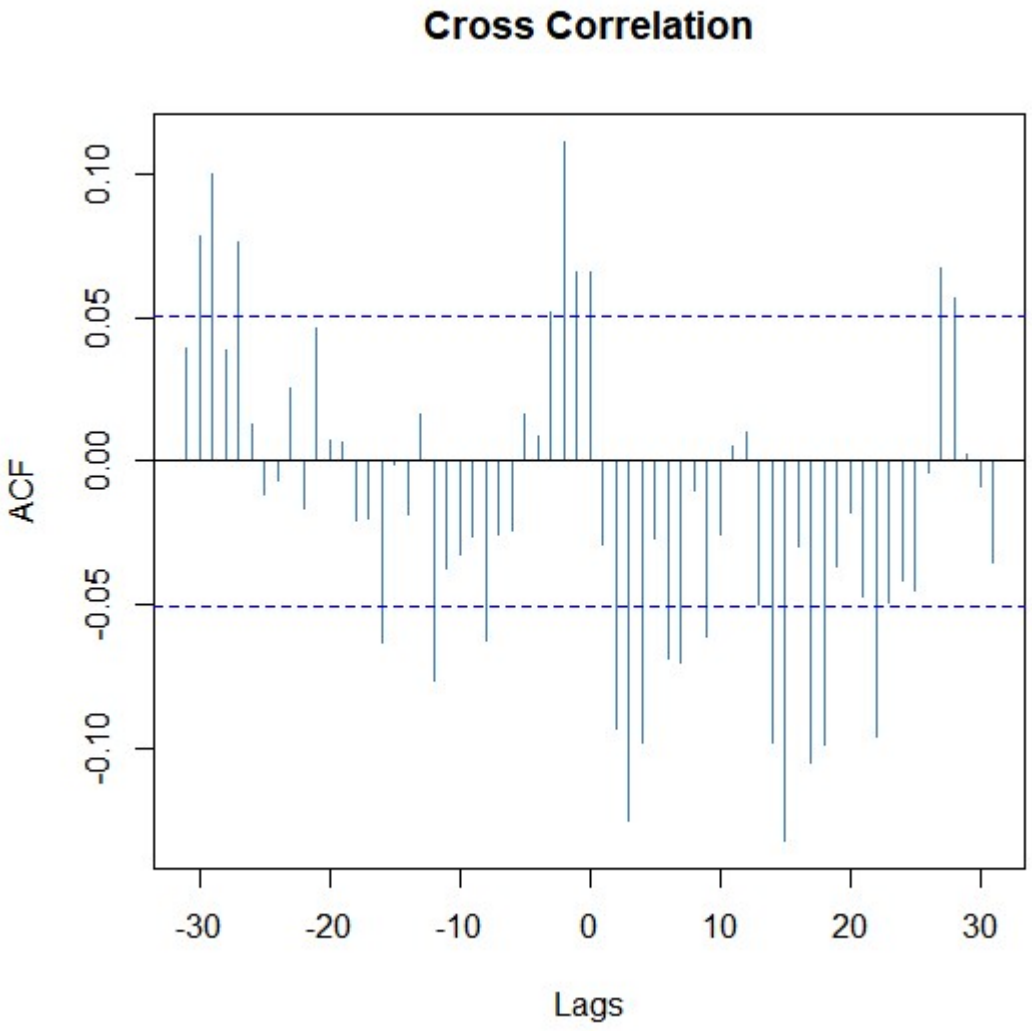

Residuals
res
0.10
0.05
0.00
-0.05
-0.10
0
500
1000
1500
Index

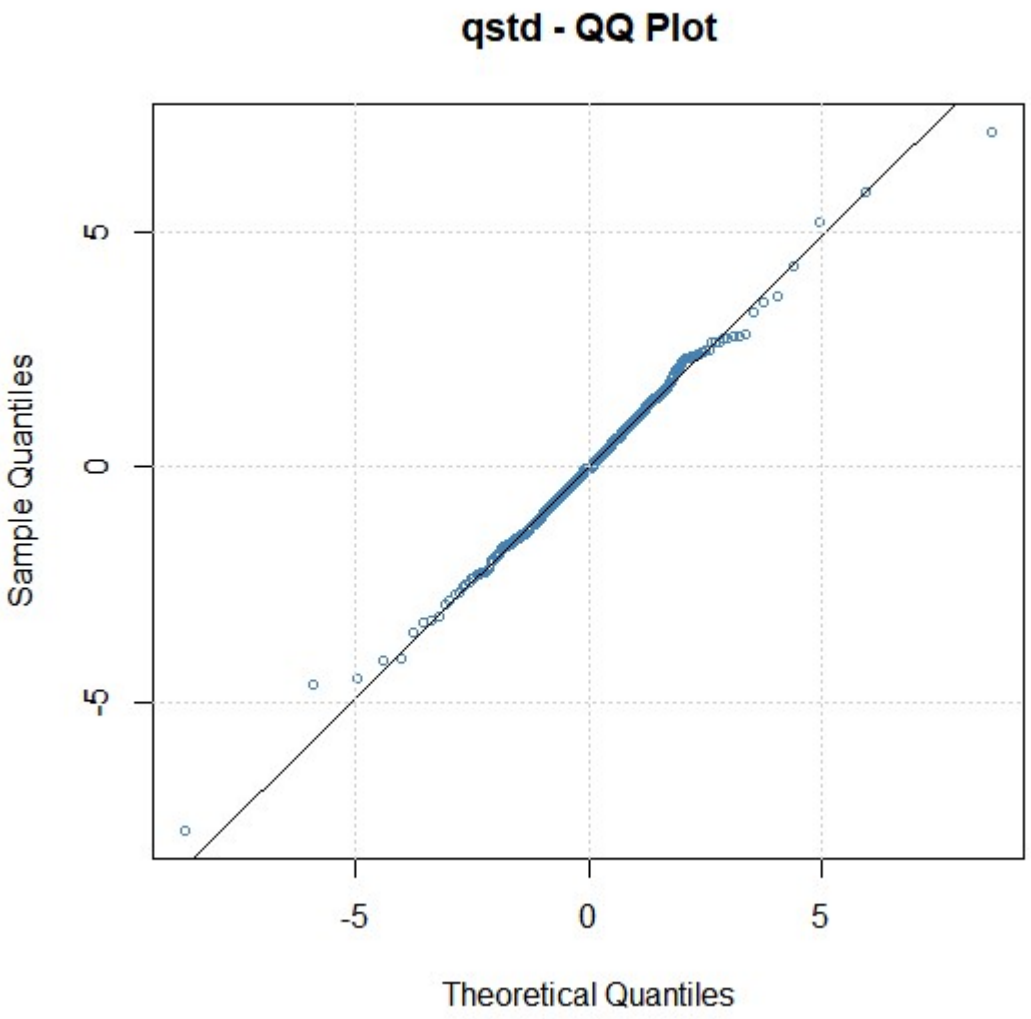

Conditional SD's
xcsd
0.01
0.02
0.03
0.04
0.05
0
500
1000
1500
Index


Standardized Residuals
sres
5
0
-5
0
500
1000
1500
Index


ACF of Standardized Residuals
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
30
Lags


ACF of Squared Standardized Residuals
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
20
25
30
Lags


Cross Correlation
ACF
0.10
0.05
0.00
-0.05
-0.10
-30
-20
-10
0
10
20
30
Lags



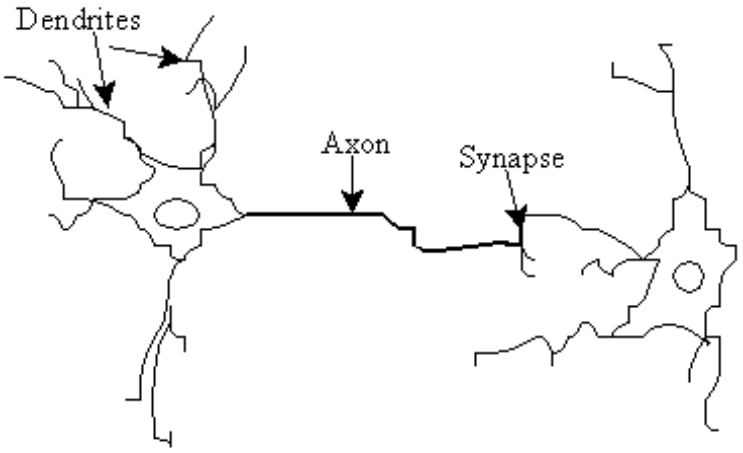
تمرين:

يترك تفسير النتائج و إختيار النموذج المناسب وكتابة شكل النموذج كتمرين للطالب.

الفصل السادس:

الشبكات العصبية الصناعية Artificial Neural Networks

الخلية العصبية الصناعية هي عبارة عن نموذج حسابي استوحي من الخلية العصبية الحقيقية. الخلايا العصبية الحقيقية تستقبل الإشارات من خلال نقاط الاشتباك العصبي *synapses* و التي توجد على الدندرايت dendrites أو غشاء الخلية العصبية. عندما تكون الإشارات القادمة قوية بشكل كاف (تتجاوز قيمة محددة) تنشط الخلية العصبية و تطلق إشارة خلال الموصل العصبي axon. هذه الإشارة قد ترسل نقطة إشتباك اخرى و قد تفعل خلية عصبية اخرى.



الشكل اعلاه يعطي محاكاة لخلية عصبية حقيقية. وفيه

Dendrites : تستقبل المعلومات

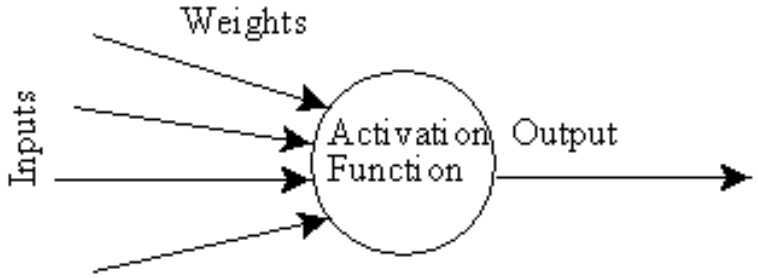
Cell Body: تعالج البيانات

Axon: يحمل المعلومات المعالجة لخلية اخرى

Synapse: توصيلة بين نهاية axon و dendrites لخلية اخرى

التعقيد الشديد في الخلايا العصبية الحقيقية لخص بشكل كبير عند نمذجة الخلايا العصبية الصناعية و التي تتكون بشكل أساسي من مدخلات inputs (و تحاكي synapes) التي

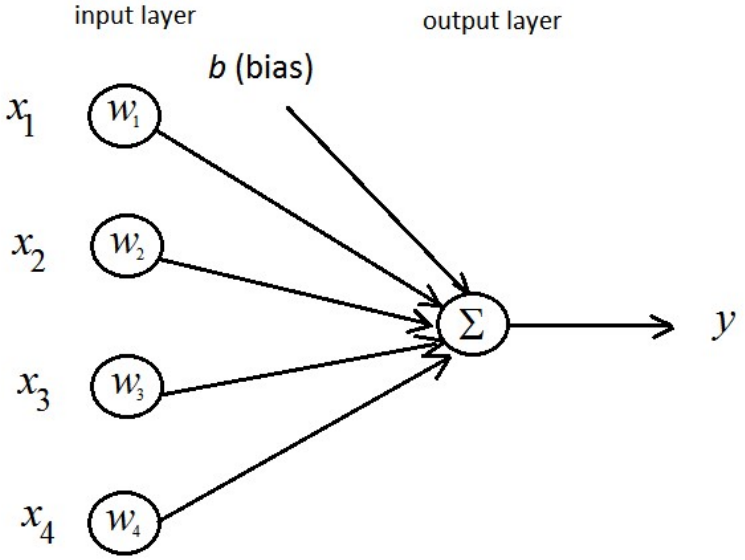
تضرب بأوزان weights (وتحاكي قوة الإشارات القادمة) ثم تحسب بمعادلة رياضية و التي تحدد تفعيل activation الخلية العصبية. دالة اخرى تحسب المخرجات output للخلية العصبية الصناعية.



الشكل أعلاه يمثل خلية عصبية صناعية.

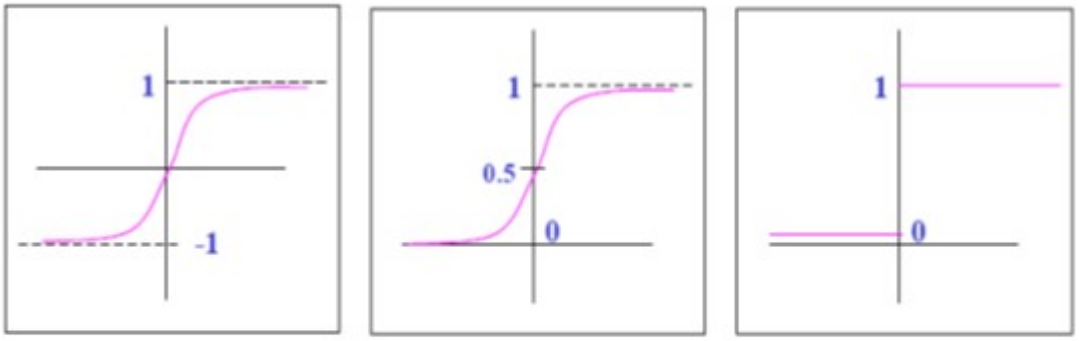
عن طريق ضبط أوزان الخلية العصبية نستطيع الحصول على مخرجات مطلوبة من مدخلات معينة. توجد خوارزمات تستطيع ضبط أوزان خلية عصبية للحصول على مخرجات مطلوبة من الشبكة. عملية ضبط الأوزان تسمى "تعلم" learning او "تدريب" training.

الشكل التالي يحاكي خلية عصبة صناعية



دوال التفعيل activation function

يوجد العديد من الإختيارات لدالة التفعيل فمثلا



tanh

$$f(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

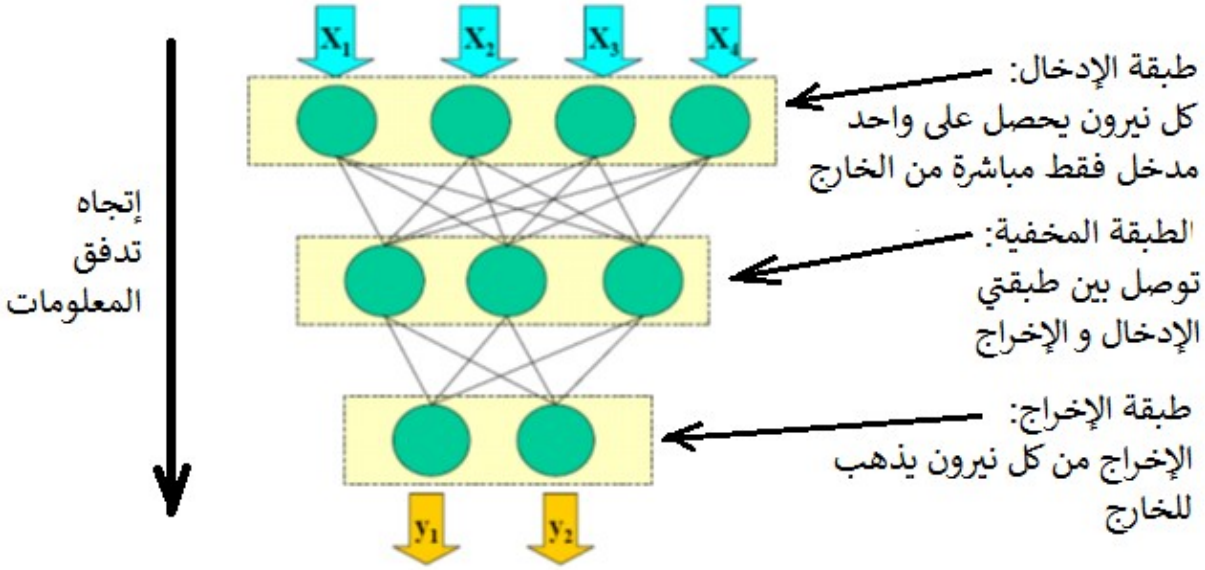
logistic

$$f(x)=\frac{e^{x}}{1+e^{x}}$$

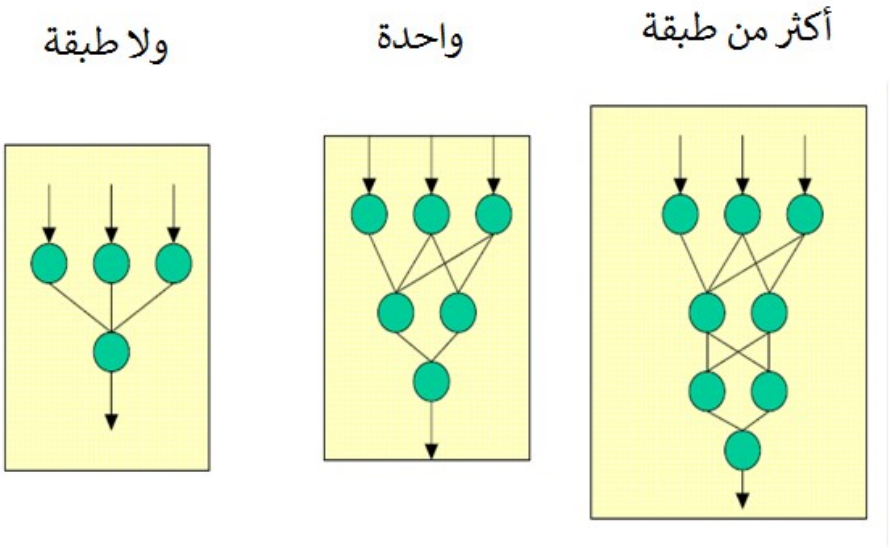
threshold

$$f(x)=\begin{cases}0, & if\ x<0\\ 1, & if\ x\geq 1\end{cases}$$

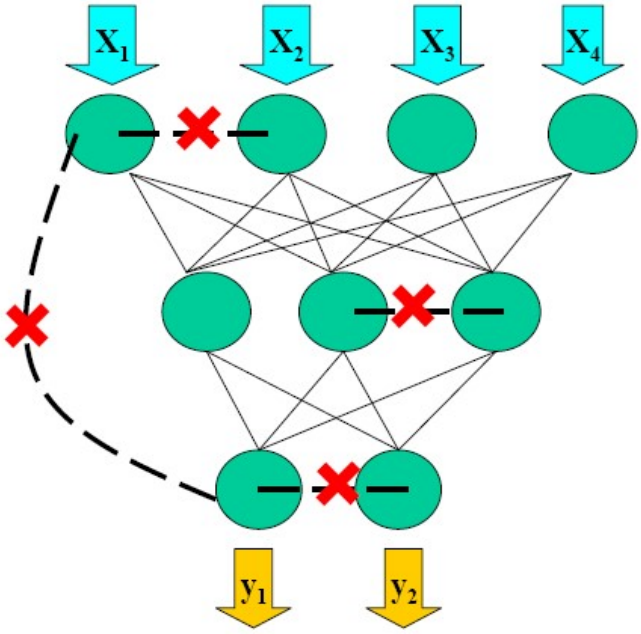
شبكة التغذية الأمامية Feed-forward Network



عدد الطبقات المخفية قد يكون



لاحظ التالي:



1) لايسمح بوصل بين نيرونين في نفس الطبقة

2) لا يسمح بوصل بين طبقتين يوجد بينهما طبقة في الوسط

3) يوصل نيرون في طبقة فقط للطبقة التالية له (ويسمى تغذية أمامية feed-forward)

مثال:

مدخلات: x_1, x_2, x_3

مخرجات: y حيث $y = f(x_1, x_2, x_3)$

في الشبكات العصبية الصناعية شكل الدالة $f(.)$ معقد جدا بحيث لايمكن كتابته ولكن يتميز بالتالي:

1- عدد نيرونات الإدخال

2- عدد الطبقات المخفية

3- عدد النيرونات في كل طبقة مخفية

4- عدد نيرونات الإخراج

5- الأوزان لكل التوصيلات

فتطبيق نموذج شبكة عصبية يعني تحديد قيم لكل هذه المعالم

ففي هذا المثال نجد التالي: (أنظر الشكل أدناه)

1- عدد نيرونات الإدخال = 3

2- عدد الطبقات المخفية = 1

3- عدد النيرونات في كل طبقة مخفية =2

4- عدد نيرونات الإخراج = 1

5- الأوزان لكل التوصيلات = مذكورة على كل توصيلة

ملاحظة: عدد نيرونات الإدخال و اخراج تتحدد من المشكلة المراد حلها

التنبؤ بإستخدام نموذج شبكة عصبية معينة:

مثال:

مدخلات: x_1, x_2, x_3

مخرجات: y حيث $y = f(x_1, x_2, x_3)$

و لتكن $x_1 = 1, \quad x_2 = -1, \quad x_3 = 1$

سوف نستخدم دالة التفعيل اللوجستية

$$f(x) = \frac{e^x}{1+e^x}$$

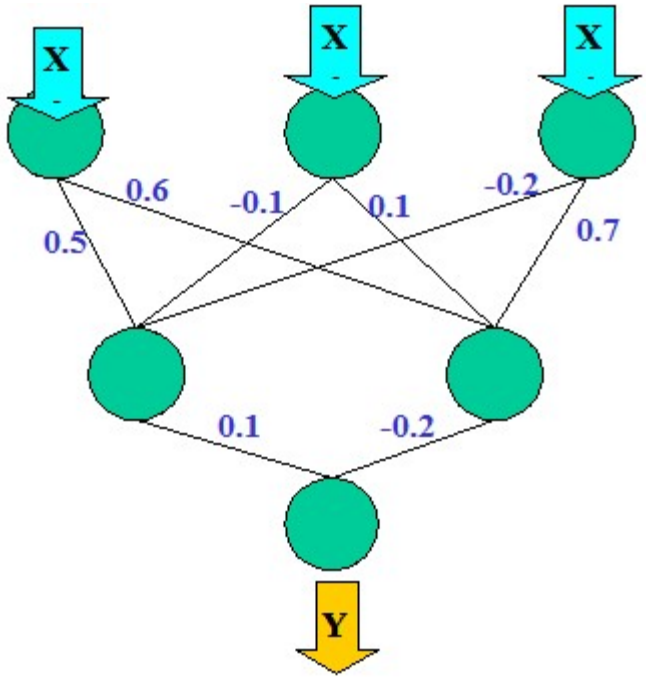
نريد التنبؤ عن قيمة y معطى الأوزان

$w_{11} = 0.5, w_{12} = 0.6$

$w_{21} = -0.1, w_{22} = 0.1$

$w_{31} = -0.2, w_{32} = 0.7$

$w_{46} = 0.1, w_{56} = -0.2$



قبل كل شيئ سوف نحسب

$$f\left(net_j\right)=f\left(\sum w_{ij}x_i\right)$$

للنيرونات المخفية

$$net_1=w_{11}x_1+w_{21}x_2+w_{31}x_3=0.2$$
$$net_2=w_{12}x_1+w_{22}x_2+w_{32}x_3=1.9$$

$$y_1=f\left(net_1\right)=f\left(0.2\right)=\frac{e^{0.2}}{1+e^{0.2}}=0.55$$

$$y_2=f\left(net_2\right)=f\left(1.9\right)=\frac{e^{1.9}}{1+e^{1.9}}=0.87$$

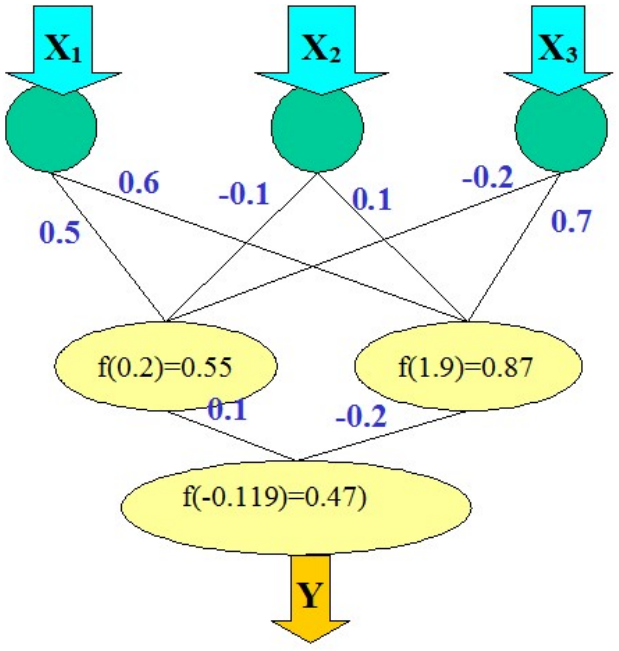
$$net_y=w_{46}y_1+w_{56}y_2=\left(0.55\right)\left(0.1\right)+\left(0.87\right)\left(-0.2\right)=0.119$$

نحسب الآن المخرج $\hat{y}$

$$\hat{y}=f\left(net_{\hat{y}}\right)=f\left(-0.119\right)=\frac{e^{-0.119}}{1+e^{-0.119}}=0.47$$

لنفترض ان القيمه الحقيقية $y=2$ فيكون الخطأ

$$\delta=y-\hat{y}=2-0.47=1.53$$



بناء نموذج شبكة عصبية Building Neural Network Model:

مدخلات: x_1, x_2, x_3

مخرجات: y حيث $y = f(x_1, x_2, x_3)$

عدد نيرونات الإدخال = 3

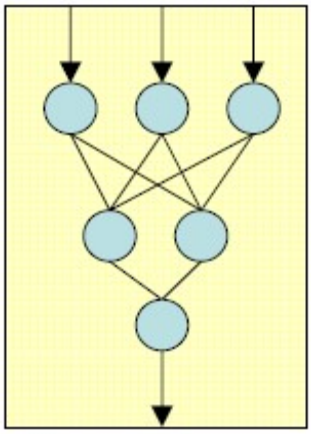
عدد نيرونات الإخراج = 1

عدد طبقات مخفية (لاتوجد إستراتيجية محددة) حاول = 1

عدد النيرونات في الطبقة المخفية (بالمحاولة و الخطأ) حاول =2

بهذا نكون قد حددنا الهيكل البنائي للشبكة. فكيف نوجد الأوزان.

معطى الهيكل البنائي للشبكة السابق



فيوجد 8 اوزان لكي نحددها $\mathbf{w} = \{w_1, w_2, \cdots, w_8\}$

بيانات التمرين training data : $\{y_i, x_{1i}, x_{2i}, \cdots, x_{pi}\}, \quad i = 1,...,n$

معطى أوزان معينة مختارة $\mathbf{w}$ سوف نحصل على قيم متنبأة $\{\hat{y}_1, \hat{y}_2, ..., \hat{y}_n\}$ وهي دوال لـ $\mathbf{w}$ و لهذا نختار $\mathbf{w}$ بحيث نقلل minimize

$$MMSE = \min_{i=1,...,n} \left(\sum (y_i - \hat{y}_i)^2 \right)$$

تدريب النموذج Training the Model:

1- نبدأ بمجموعة عشوائية من الأوزان

2- غذي إلى الأمام خلال الشبكة اول مشاهدة

$$x_1 \to network \to \hat{y}_1 \Rightarrow error = \left(y_1 - \hat{y}_1\right)$$

3- عدل الأوزان بحيث نقلل هذا الخطأ (الشبكة تنطبق على المشاهدة الأولى بشكل جيد)

4- غذي المشاهدة الثانية ومن ثم عدل الأوزان لكي تطبق المشاهدة الثانية بشكل جيد

5- استمر على هذا المنوال حتى المشاهدة الأخيرة

6- هذا يتمم دورة cycle واحدة خلال المشاهدات

7- قم بإجراء عدد من الدورات التدريبية حتى يصبح الخطأ الكلي للتنبؤ

$$\sum\left(y_i - \hat{y}_i\right)^2$$

أصغر مايمكن.

الإنتشار الخلفي Back Propagation:

1- كل وزن " يشارك في اللوم " "shares the blame" في خطأ التنبؤ مع الأوزان الاخرى

2- خوارزم الإنتشار الخلفي يقرر كيفية توزيع اللوم على جميع الأوزان و تضبيط الأوزان تبعا لذلك

3- جزء صغير من اللوم يقود لتضبيط بسيط

4- جزء كبير من اللوم يؤدي لتضبيط كبير

تضبيط الوزن خلال عملية الإنتشار الخلفي:

$\hat{y}_i$ التنبؤ للمشاهدة i هي دالة لأوزان الشبكة $\mathbf{w} = \{w_1, w_2, \cdots\}$ لهذا خطأ التنبؤ الكلي

$$E(\mathbf{w}) = \sum\left(y_i - \hat{y}_i\right)^2$$

طريقة نزول المنحدر Gradient Descent Method:

لكل وزن w_i صيغة تحديث الوزن هي

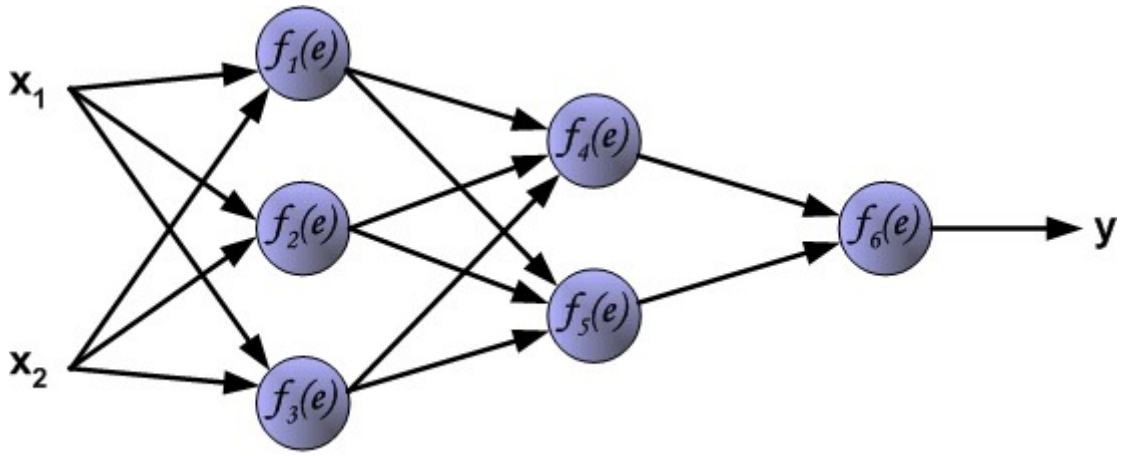
$$w_{new} = w_{old} + \alpha \frac{\delta E(\mathbf{w})}{\delta \mathbf{w}}|_{w_{old}}$$

حيث α معلم التعلم وهو بين 0 و 1 .

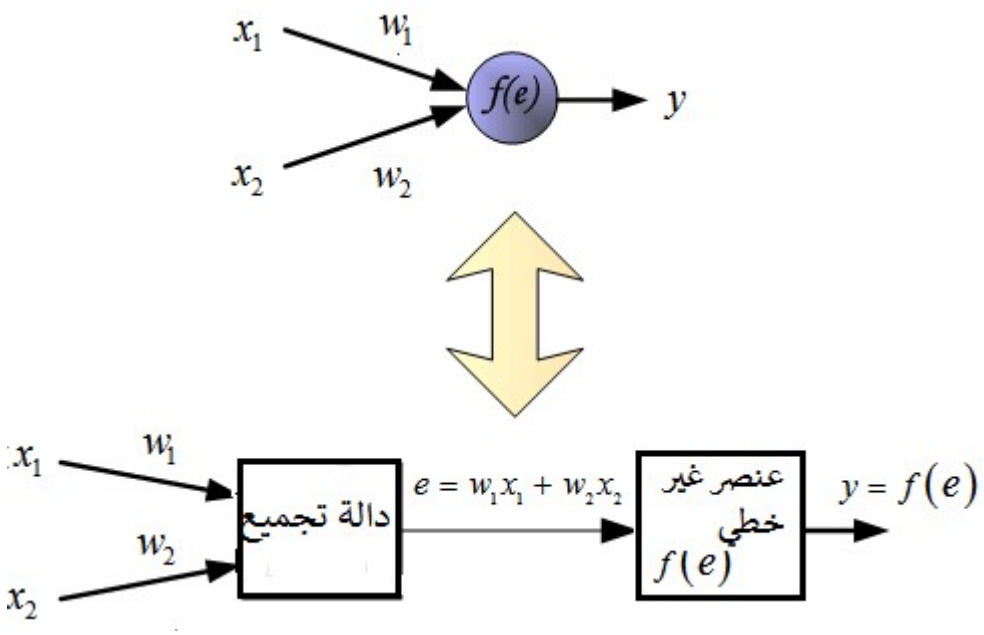
أساسيات تمرين شبكة عصبية متعددة الطبقات بواسطة الإنتشار الخلفي

Principles of Training Multi-layer Neural Network using Backpropagation

لوصف خوارزم الإنتشار الخلفي سوف نستخدم شبكة عصبية ذات ثلاثة طبقات حيث طبقة الإدخال تحوي مدخلين و طبقة الإخراج تحوي مخرج واحد كما في الشكل التالي:



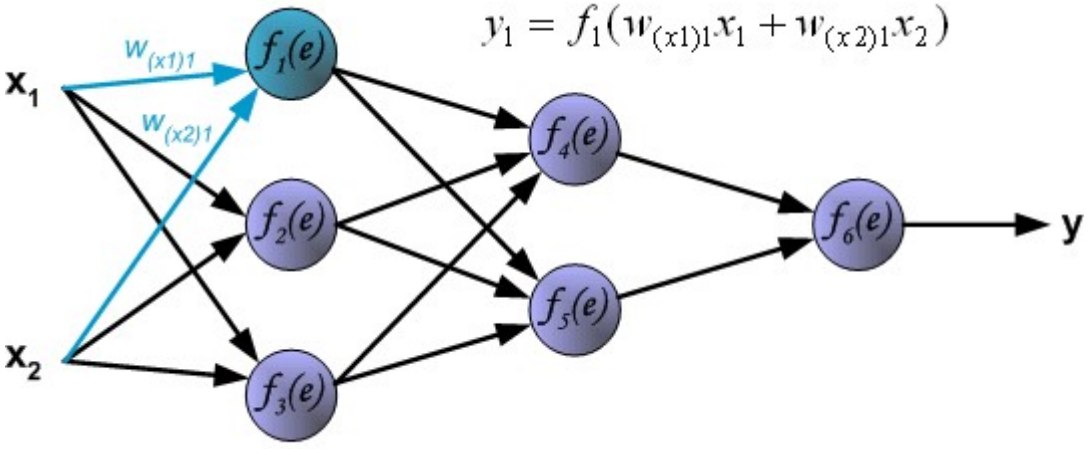
كل نيرون يتكون من وحدتين. الوحدة الاولى تضيف حاصل ضرب الأوزان مع المدخلات و الثانية تطبق دالة غير خطية و التي تسمى دالة تفعيل النيرون

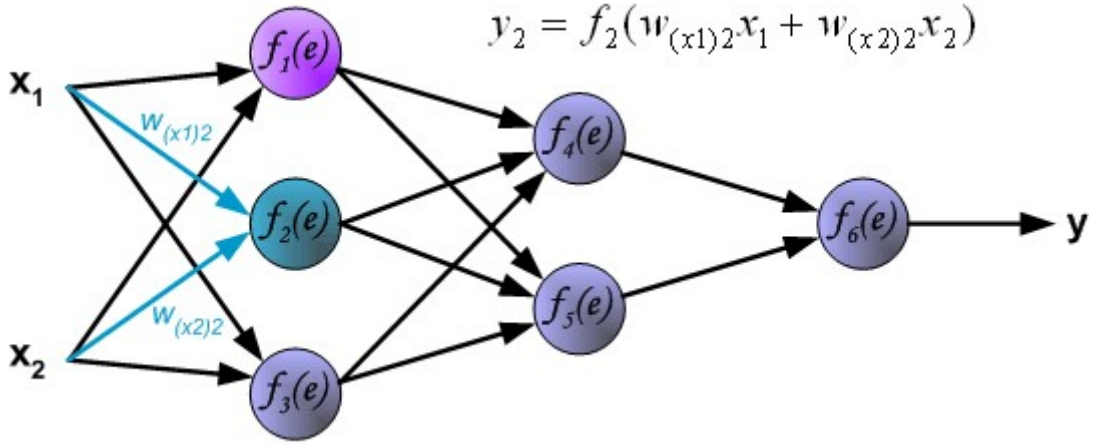


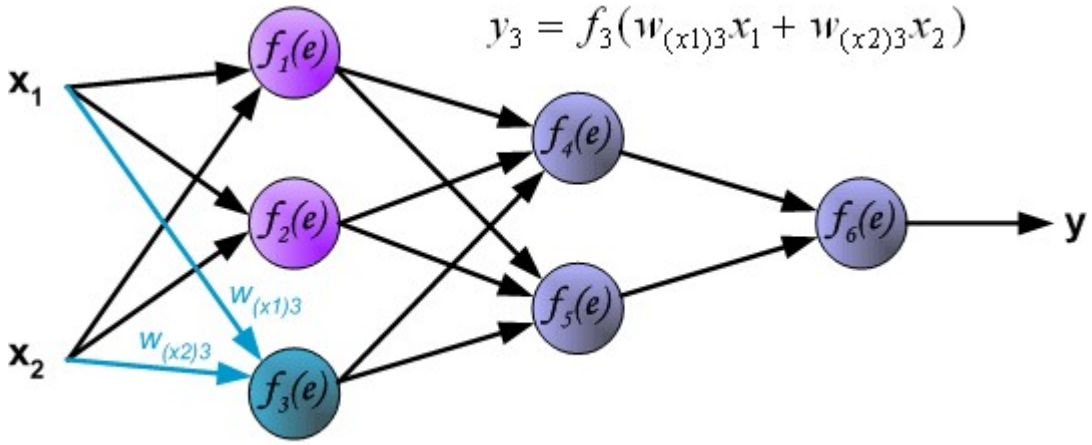
لتعليم الشبكة العصبية نحتاج إلى مجموعة بيانات تدريب. بيانات التدريب تتكون من مدخلات x_1 و x_2 . عملية تدريب الشبكة هي عملية تكرارية iterative ففي كل تكرار يتم تحديث الأوزان من خلال بيانات تدريب جديدة. التحديث يتم من خلال خوارزم الإنتشار الخلفي الذي نصفه كالتالي.

الرمز $w_{(xm)n}$ يمثل الوزن للتوصيلة بين المدخل x_m و النيرون n في طبقة الإدخال.

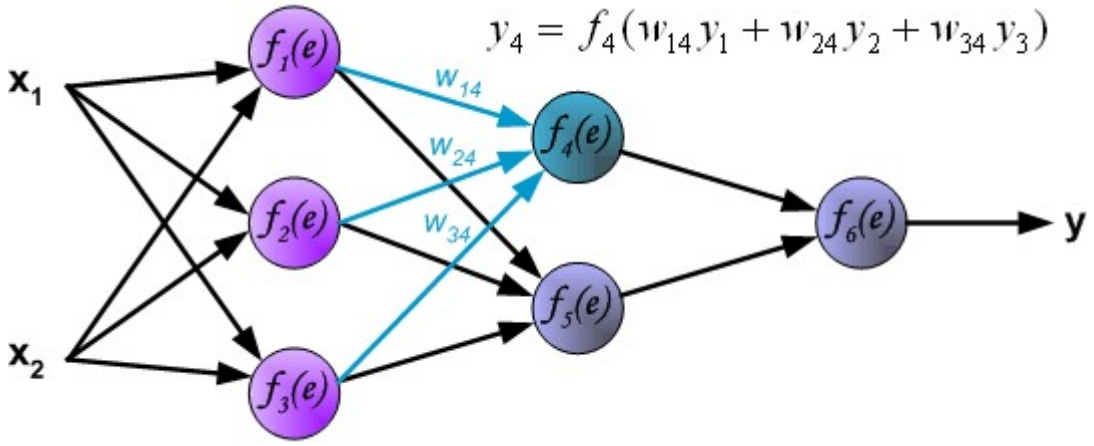
الرمز y_n المخرج من النيرون n.

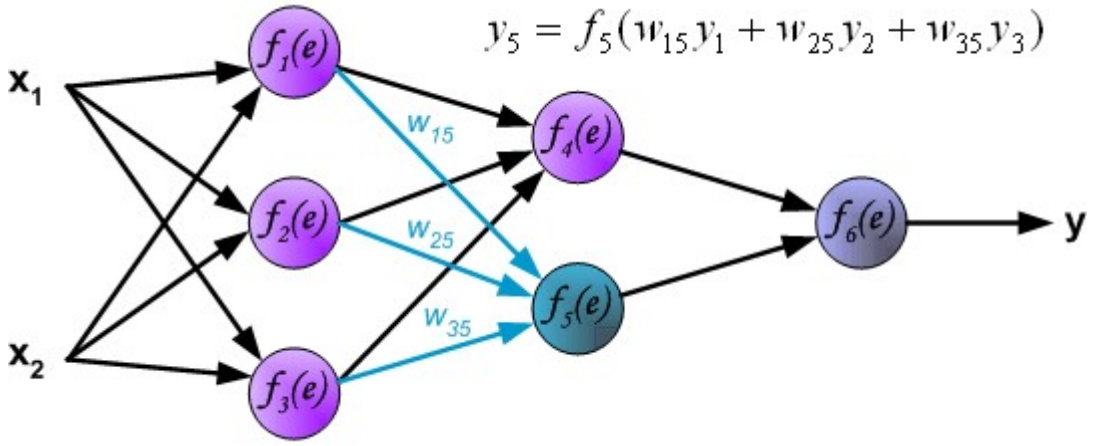




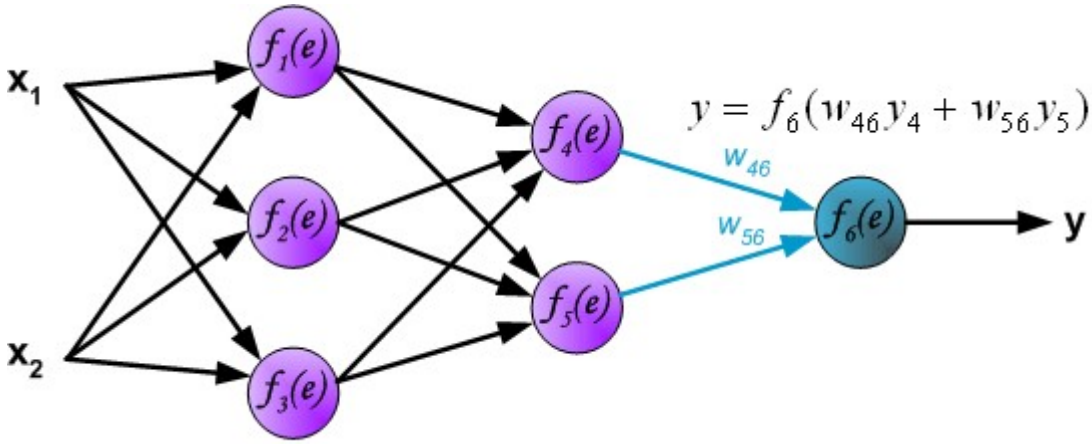


للإنتشار خلال الطبقة المخفية نستخدم الرمز w_{mn} للأوزان التي توصل بين المخرج من النيرون m و مدخل النيرون n في الطبقة التالية

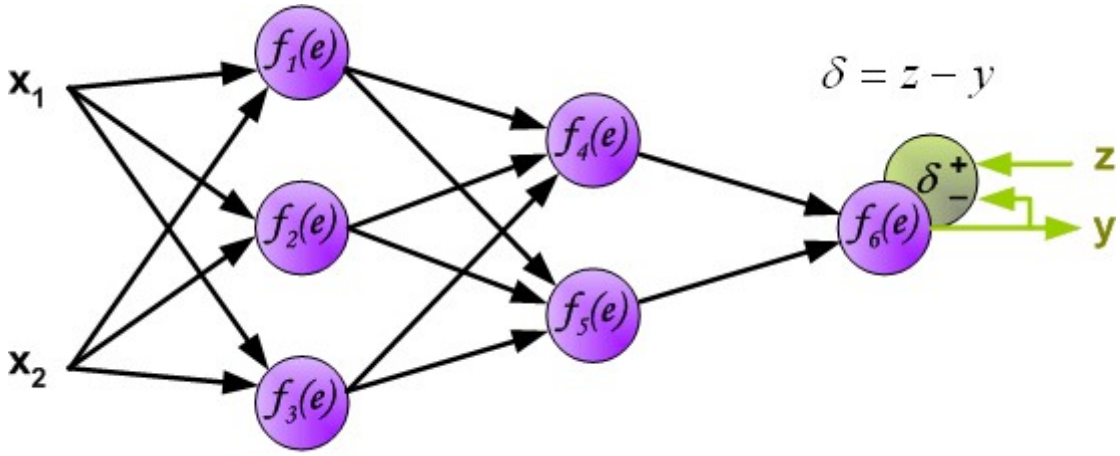




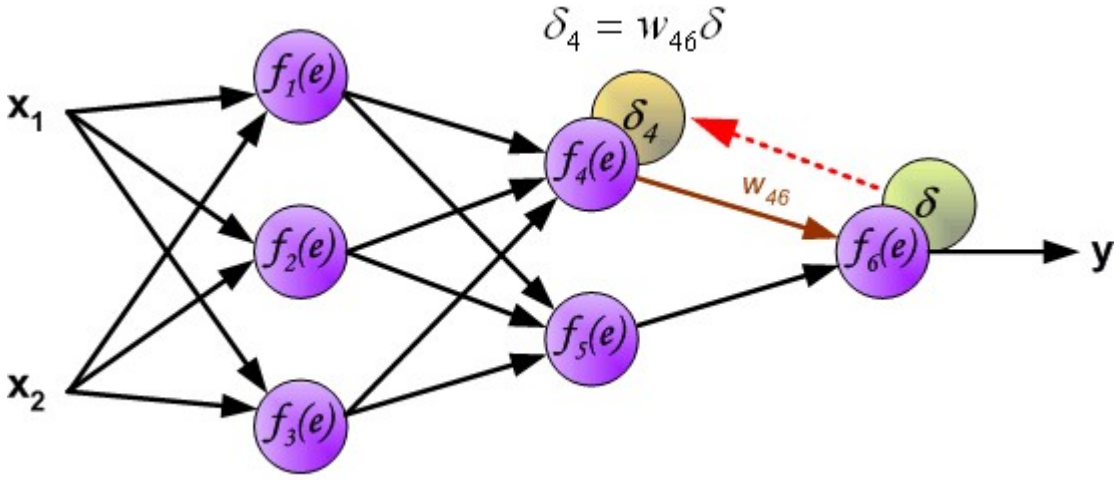
إنتشار البيانات خلال طبقة الإخراج

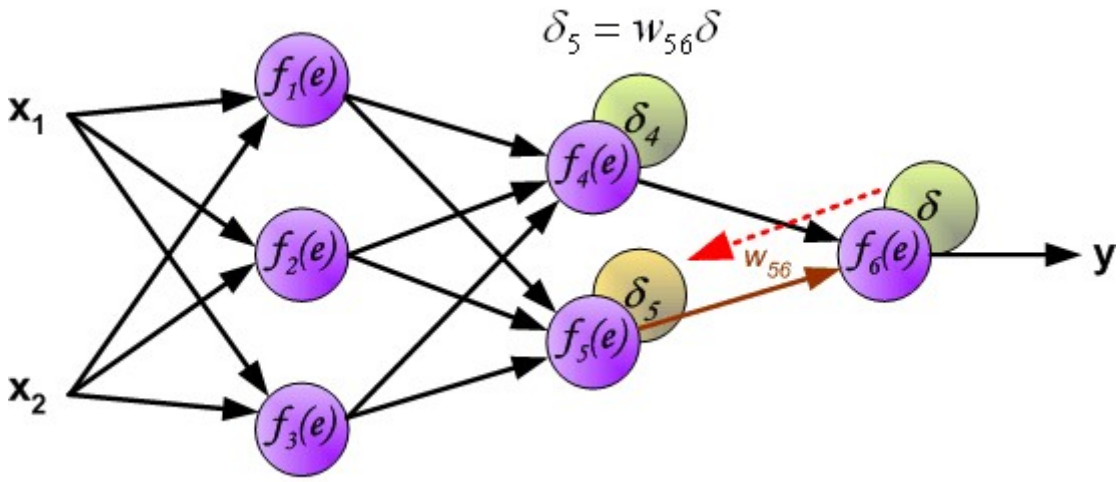


في الخطوة التالية للخوارزم المخرجات y تقارن مع القيمة المطلوبة و التي توجد في مجموعة بيانات التمرين و الفرق يسمى خطأ الإخراج δ من طبقة نيرون الإخراج

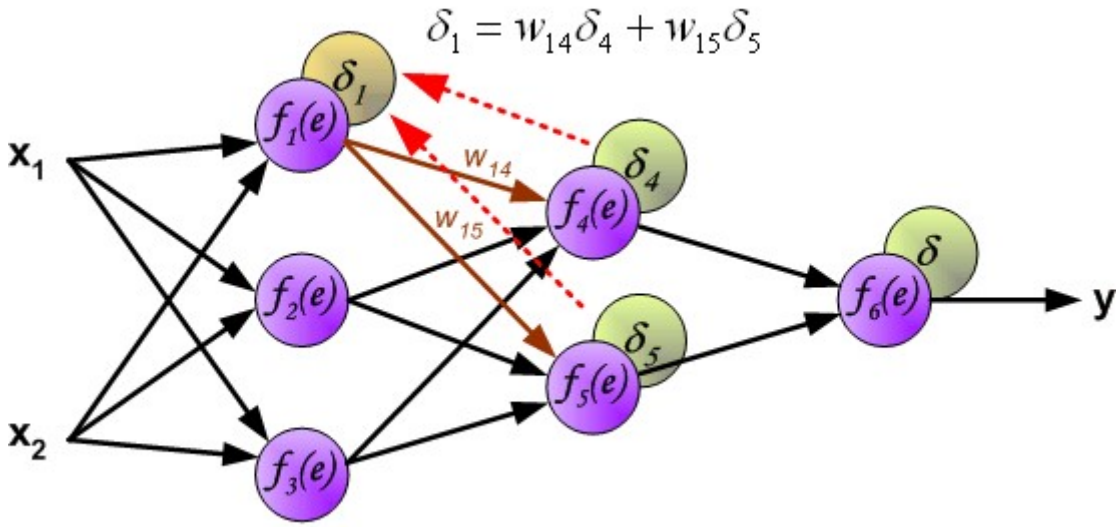


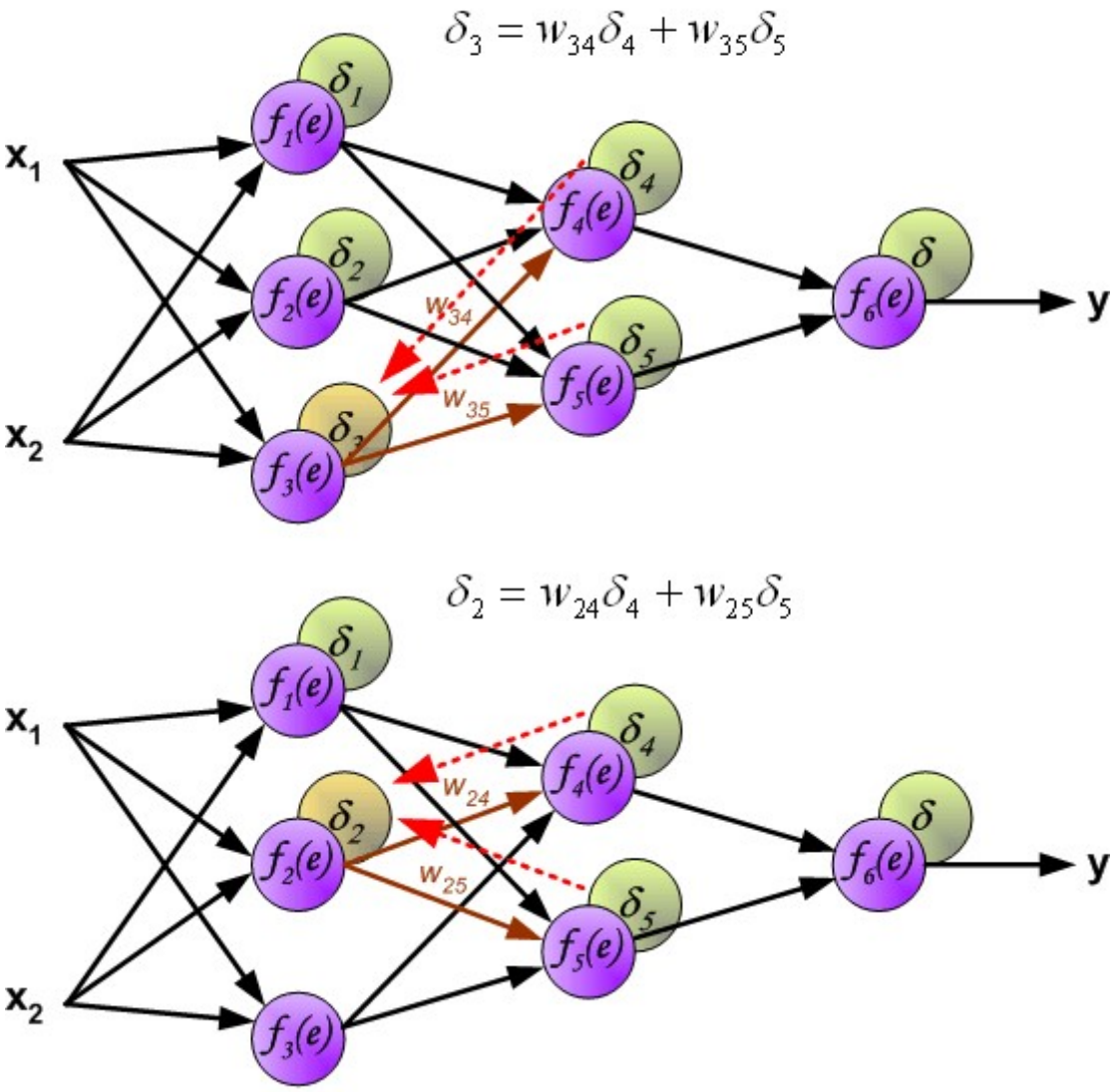
من المستحيل حساب الخطأ من النيرونات الداخلية مباشرة لأن القيم المخرجة من هذه النيرونات غير معروفه. المهم هنا هو نشر الخطأ δ إلى الخلف لكل النيرونات التي تمد هذه المخرجات





الأوزان w_{mn} و التي استخدمت لنشر الخطأ للخلف هي نفسها تلك التي استخدمت خلال حساب المخرجات فقط إتجاه سير البيانات قد تغير (البيانات تنتشر من المخرجات إلى المدخلات الواحدة بعد الاخرى. هذه الطريقة تستخدم لجميع طبقات الشبكة. تضاف أخطاء انتشار إذا اتت من بعض النيرونات .



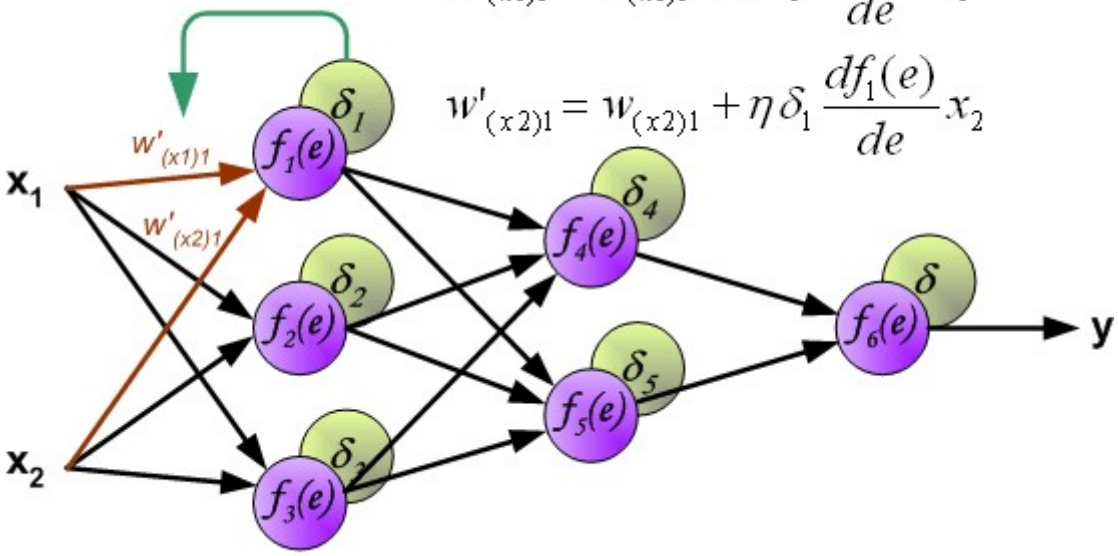


عندما يحسب الخطأ من كل نيرون فعندها يتم تعدل الأوزان عند كل نيرون إدخال.

الرمز $df(e)/de$ يمثل المشتقة لدالة التنشيط

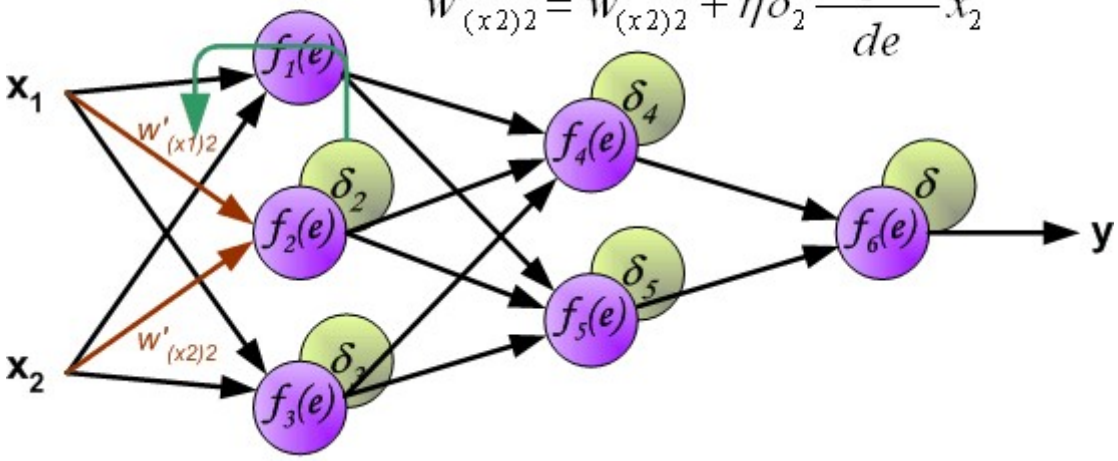
$$w'_{(x1)1} = w_{(x1)1} + \eta \delta_1 \frac{df_1(e)}{de} x_1$$

$$w'_{(x2)1} = w_{(x2)1} + \eta \delta_1 \frac{df_1(e)}{de} x_2$$



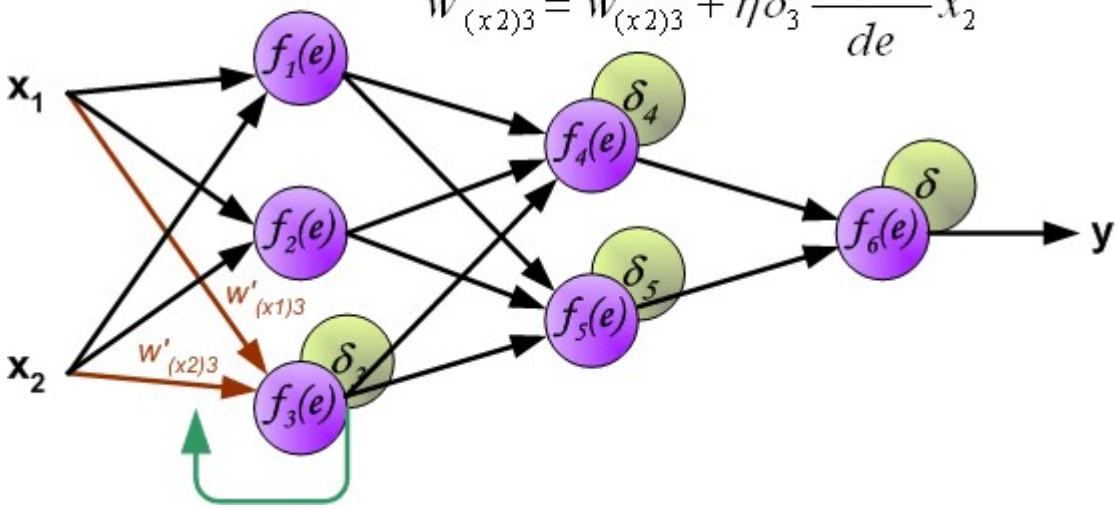
$$w'_{(x1)2} = w_{(x1)2} + \eta \delta_2 \frac{df_2(e)}{de} x_1$$

$$w'_{(x2)2} = w_{(x2)2} + \eta \delta_2 \frac{df_2(e)}{de} x_2$$



$$w'_{(x1)3} = w_{(x1)3} + \eta \delta_3 \frac{df_3(e)}{de} x_1$$

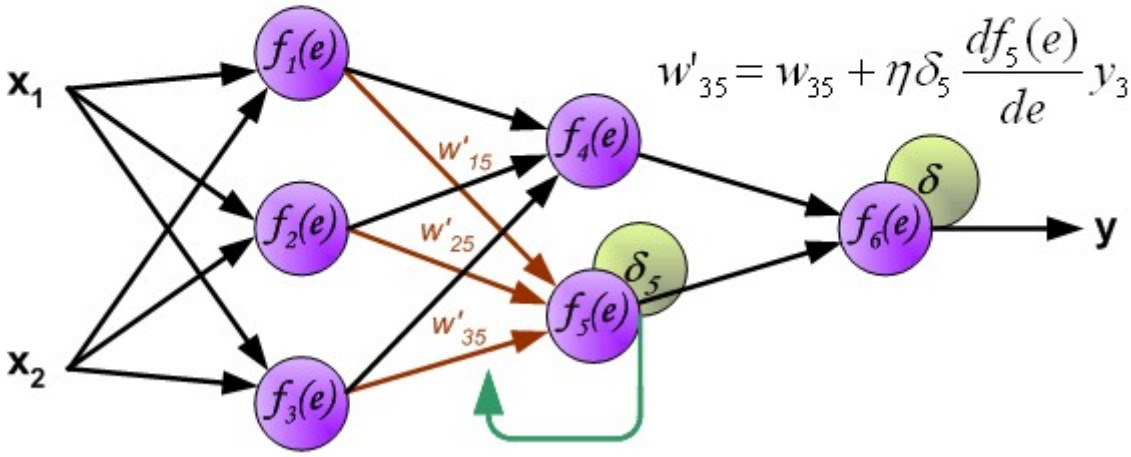
$$w'_{(x2)3} = w_{(x2)3} + \eta \delta_3 \frac{df_3(e)}{de} x_2$$



$$w'_{15} = w_{15} + \eta\delta_5 \frac{df_5(e)}{de} y_1$$

$$w'_{25} = w_{25} + \eta\delta_5 \frac{df_5(e)}{de} y_2$$

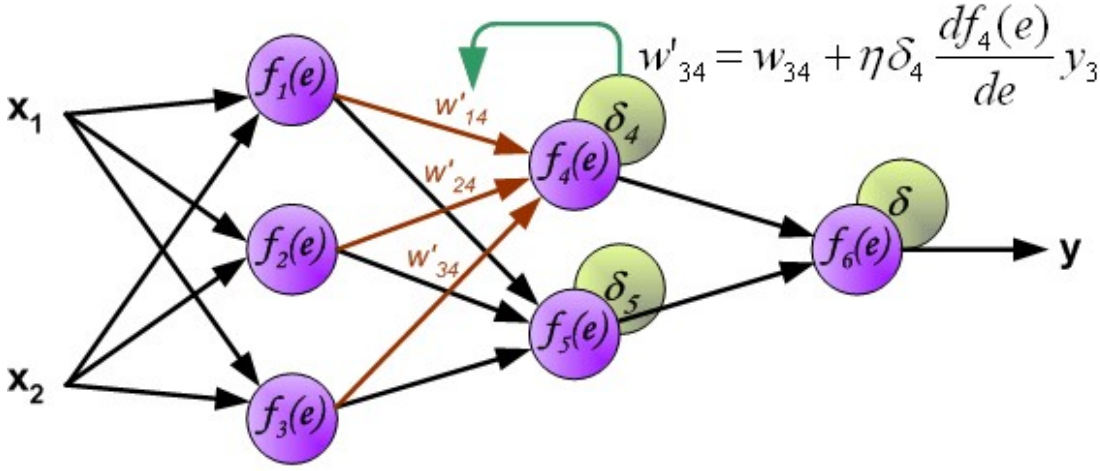
$$w'_{35} = w_{35} + \eta\delta_5 \frac{df_5(e)}{de} y_3$$

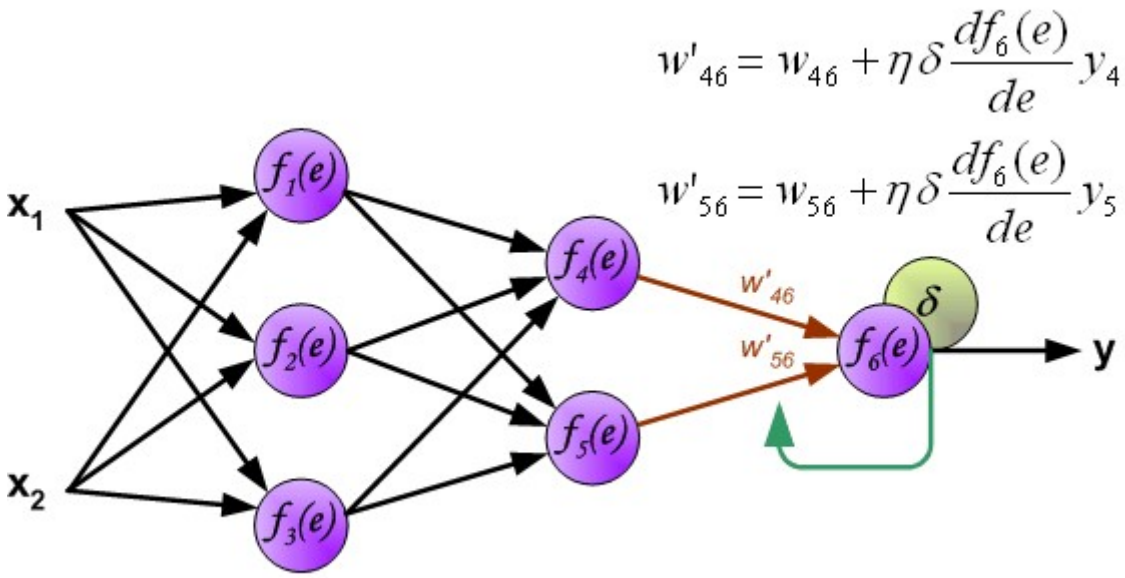


$$w'_{14} = w_{14} + \eta\delta_4 \frac{df_4(e)}{de} y_1$$

$$w'_{24} = w_{24} + \eta\delta_4 \frac{df_4(e)}{de} y_2$$

$$w'_{34} = w_{34} + \eta\delta_4 \frac{df_4(e)}{de} y_3$$





المعامل η يؤثر على سرعة التدريب و يحدد عن طريق التجربة و الخطأ. ومن الشايع هو البدء بقيمة صغيرة ثم زيادتها بالتدريج.

ملاحظة هامة:

في النمذجة بالشبكات العصبية في حالة وجود أكثر من متغير في النموذج يكون من الضروري معايرة او تطبيع البيانات normalize data قبل تمرين الشبكة عليها لأن الشبكة قد تجد صعوبة في التقارب converging قبل عدد التكرارات العظمى المسموح بها إذا لم تعاير البيانات.

أمثلة للتنبؤات المالية:

إحتساب الجداره الائتمانية او نظام تصنيف الائتمان Credit scoring وهي عملية تقوم بها البنوك لحساب جدارة الإقتراض لشخص creditworthness يحاول الحصول على قرض معتمدة على مقاييس شخصية مثل الدخل و العمر و الجنس أي

creditworthness = *f*(income,age,imployment,gender,...)

شكل الدالة *f*(.) يمكن تحديده بالإنحدار الخطي المتعدد linear regression أو الإنحدار اللوجستي logistic regression أو تعلم الآلة machine learnin أو أي خليط من الطرق السابقة. الشبكات العصبية تقع ضمن نطاق تعلم الآلة وسوف نستخدمها لتطوير نموذج بسيط.

سوف نستخدم المكتبة neuralnet والبيانات التالية التي يمكن تحميلها كالتالي

```
cd =
"https://gist.githubusercontent.com/Bart6114/8675941/raw/ac4cddcc0909c1
5ceada2d8c6a303206b10796d9/creditset.csv"
> library(neuralnet)
> dataset <- read.csv(cd)
> head(dataset)
  clientid      income         age          loan             LTI default10yr
1        1 66155.92510 59.01701507 8106.53213129 0.1225367511623           0
2        2 34415.15397 48.11715310 6564.74501768 0.1907515806612           0
3        3 57317.17006 63.10804949 8020.95329639 0.1399397996720           0
4        4 42709.53420 45.75197235 6103.64226014 0.1429105321411           0
5        5 66952.68885 18.58433593 8770.09923520 0.1309894999955           1
6        6 24904.06414 57.47160710   15.49859844 0.0006223320961           0
> tail(dataset)
     clientid      income         age        loan           LTI default10yr
1995     1995 24254.70079 37.75162224 2225.284643 0.09174653040           0
1996     1996 59221.04487 48.51817941 1926.729397 0.03253453905           0
1997     1997 69516.12757 23.16210447 3503.176156 0.05039371839           0
1998     1998 44311.44926 28.01716690 5522.786693 0.12463565930           1
1999     1999 43756.05660 63.97179584 1622.722598 0.03708566823           0
2000     2000 69436.57955 56.15261703 7378.833599 0.10626723906           0
> dim(dataset)
```

```
[1] 2000    6
```

مجموعة البيانات لـ 2000 شخص تحوي 6 متغيرات لمعلومات عن الدخل السنوي و عمر الشخص و قيمة القرض و نسبة القرض للدخل السنوي LTI و هل افلس خلال العشرة سنوات الماضية default10yr

هدفنا هو تطوير نموذج يتنبأ معتمدا على المتغيرات المدخلة LTI و age فيما سيحدث إفلاس خلال العشرة سنوات القادمة.

سوف نقسم البيانات لمجموعتين واحدة للتدريب و اخرى للإختبار و حيث ان الترتيب للبيانات عشوائي تماما فإننا لانحتاج إلى إختيار اسطر عشوائية و يمكننا فقط اخذ مثلا800 المشاهدة الأولى للتدريب

```
> trainset <- dataset[1:800, ]
```

و الجزء الأخير للإختبار

```
> testset <- dataset[801:2000, ]
```

نجري التطبيع على البيانات

```
> normalize <- function(x) {
+   x <- as.matrix(x)
+   minAttr=apply(x, 2, min)
+   maxAttr=apply(x, 2, max)
+   x <- sweep(x, 2, minAttr, FUN="-")
+   x=sweep(x, 2,  maxAttr-minAttr, "/")
+   attr(x, 'normalized:min') = minAttr
+   attr(x, 'normalized:max') = maxAttr
+   return (x)
+ }
> trainsetn = normalize(trainset)
> testn = normalize(testset)
```

الآن نبني الشبكة

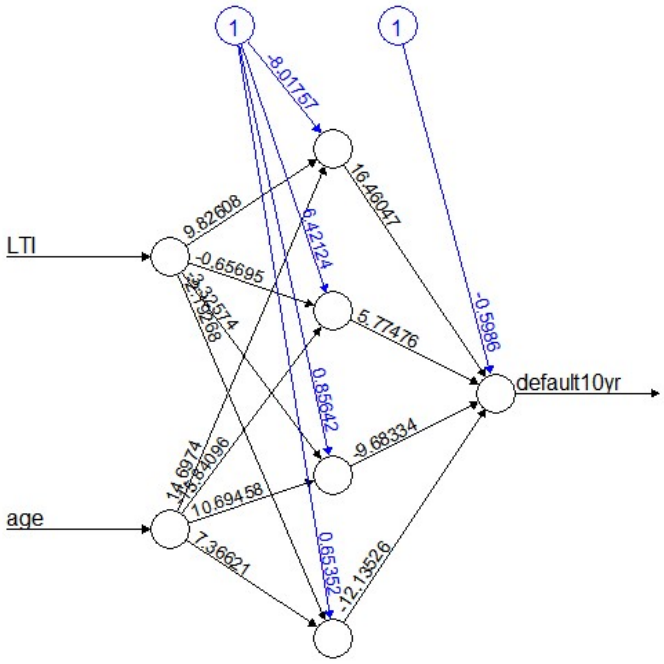
```
> creditnet <- neuralnet(default10yr ~ LTI + age, trainsetn, hidden =
+ 4,lifesign = "minimal",linear.output = FALSE, threshold = 0.1)
```

بعض النتائج

```
hidden: 4    thresh: 0.1    rep: 1/1    steps:     857
error: 2.70118  time: 0.45 secs
```

نرسم الشبكة

```
> plot(creditnet, rep = "best")
>
```



Error: 2.701178 Steps: 857

نختبر المخرجات الناتجة

```
> temp_test <- subset(testn, select = c("LTI", "age"))
> creditnet.results <- compute(creditnet, temp_test)
> results <- data.frame(actual = testset$default10yr, prediction =
creditnet.results$net.result)
> results[100:115, ]
    actual         prediction
900      0 0.0003257007294001
901      0 0.0022953190387399
902      0 0.0000337144136086
```

```
903      1 0.9988491825994084
904      0 0.0025831966743758
905      0 0.0000117541757410
906      0 0.0018152833848443
907      1 0.9174666266721554
908      0 0.3392054158734511
909      0 0.0000080682709607
910      0 0.0988541021097577
911      1 0.9934022761603937
912      0 0.0000003765359376
913      1 0.9999904775449217
914      0 0.0001342937348306
915      0 0.0027193873613529
```

نقرب القيم

```
> results$prediction <- round(results$prediction)
> results[100:115, ]
    actual prediction
900      0          0
901      0          0
902      0          0
903      1          1
904      0          0
905      0          0
906      0          0
907      1          1
908      0          0
909      0          0
910      0          0
911      1          1
912      0          0
913      1          1
914      0          0
915      0          0
>
```

نلاحظ تطابق القيم الحقيقية مع القيم المتنأة.

مثال للتنبؤ لمتسلسلة زمنية مالية:

المتسلسلة التالية هي قيم الإغلاق لسهم شركة سابك في السوق السعودية من 2017/1/1 وحتى 2017/23/5

```
> library(xlsx)
> library(forecast)
> sabicCls <-
read.xlsx("C:/Users/ibin/Desktop/sabicClose.xlsx",1,header=
TRUE )
> sc <- sabicCls[,2]
> sc <- as.ts(sc)
```

أولا سوف نستخدم مكتبة forecast و الدالة auto.arima لتطبيق نموذج

```
> model1 <- auto.arima(sc)
> summary(model1)
Series: sc
ARIMA(0,1,1)
```

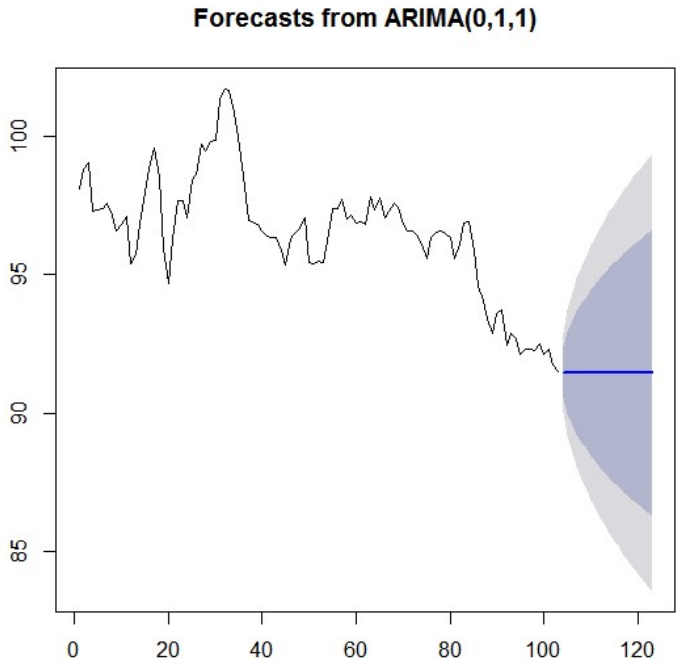
نلاحظ ان النموذج المطبق هو $ARIMA(0,1,1)$ حيث $\theta_1 = 0.2568$ و

$$s.e.(\theta_1) = 0.0937$$

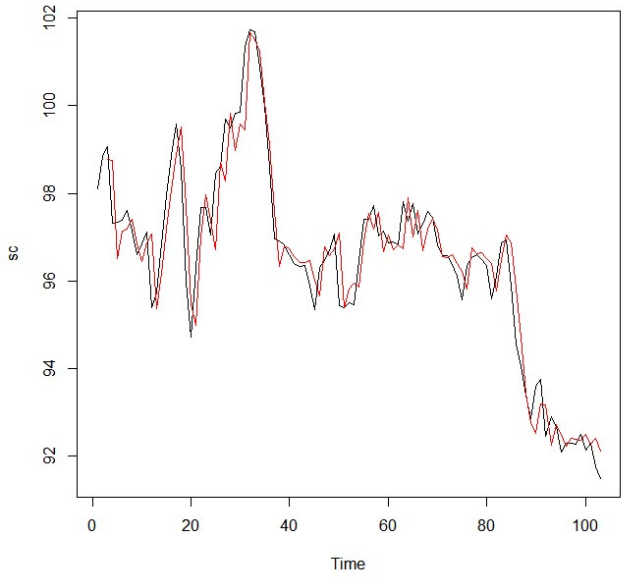
```
Coefficients:
          ma1
      0.2568
s.e.  0.0937

sigma^2 estimated as 0.5243:  log likelihood=-111.33
AIC=226.66   AICc=226.78   BIC=231.91

Training set error measures:
                      ME      RMSE       MAE          MPE
MAPE      MASE         ACF1
```

```
Training set -0.05053705 0.7170513 0.5375234 -0.05540766
0.5565022 0.9936098 -0.009867329
> fmodel1 <- forecast(model1, h = 20)
> summary(fmodel1)

Forecast method: ARIMA(0,1,1)

Model Information:
Series: sc
ARIMA(0,1,1)

Coefficients:
         ma1
      0.2568
s.e.  0.0937

sigma^2 estimated as 0.5243:  log likelihood=-111.33
AIC=226.66   AICc=226.78   BIC=231.91

Error measures:
                     ME      RMSE       MAE          MPE
MAPE      MASE         ACF1
Training set -0.05053705 0.7170513 0.5375234 -0.05540766
0.5565022 0.9936098 -0.009867329

Forecasts:
    Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
104       91.46168 90.53369 92.38967 90.04244 92.88092
105       91.46168 89.97122 92.95214 89.18221 93.74115
106       91.46168 89.56912 93.35424 88.56725 94.35611
107       91.46168 89.23859 93.68477 88.06176 94.86160
108       91.46168 88.95122 93.97214 87.62226 95.30110
```

```
109       91.46168 88.69351 94.22985 87.22813 95.69523
110       91.46168 88.45784 94.46552 86.86770 96.05566
111       91.46168 88.23936 94.68401 86.53356 96.38980
112       91.46168 88.03477 94.88859 86.22068 96.70268
113       91.46168 87.84173 95.08163 85.92545 96.99791
114       91.46168 87.65848 95.26488 85.64519 97.27817
115       91.46168 87.48366 95.43970 85.37782 97.54554
116       91.46168 87.31621 95.60715 85.12172 97.80164
117       91.46168 87.15526 95.76810 84.87558 98.04778
118       91.46168 87.00011 95.92325 84.63830 98.28506
119       91.46168 86.85018 96.07318 84.40900 98.51436
120       91.46168 86.70498 96.21838 84.18693 98.73643
121       91.46168 86.56408 96.35929 83.97144 98.95192
122       91.46168 86.42712 96.49625 83.76198 99.16138
123       91.46168 86.29378 96.62958 83.55806 99.36530
> plot(fmodel1)
>
```

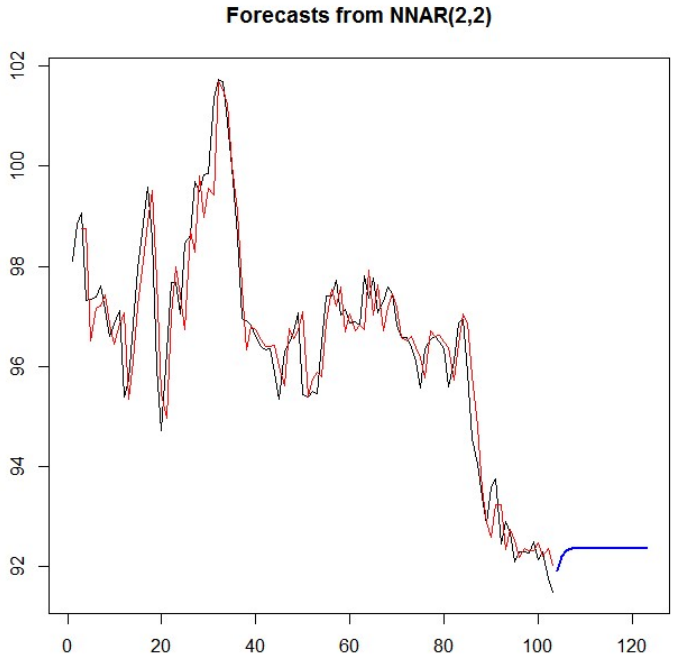


```
> fitsc <- nnetar(sc)
> plot.ts(sc)
> lines(fitsc$fitted, col="red")
>
```



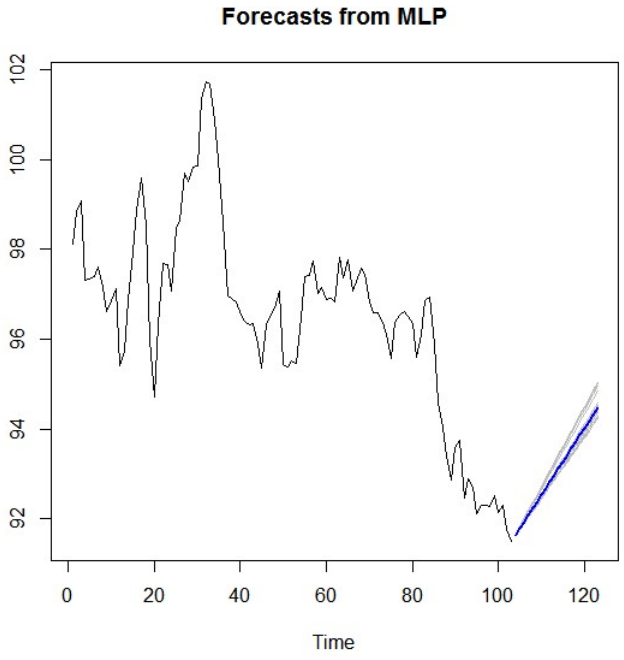

```
> fitsc <- nnetar(sc, lambda = 0)
> forecast(fitsc, h = 20)
    Point Forecast
104       91.94189
105       92.22807
106       92.35203
107       92.39515
108       92.40707
109       92.40925
110       92.40914
111       92.40879
112       92.40859
113       92.40850
114       92.40847
115       92.40846
```

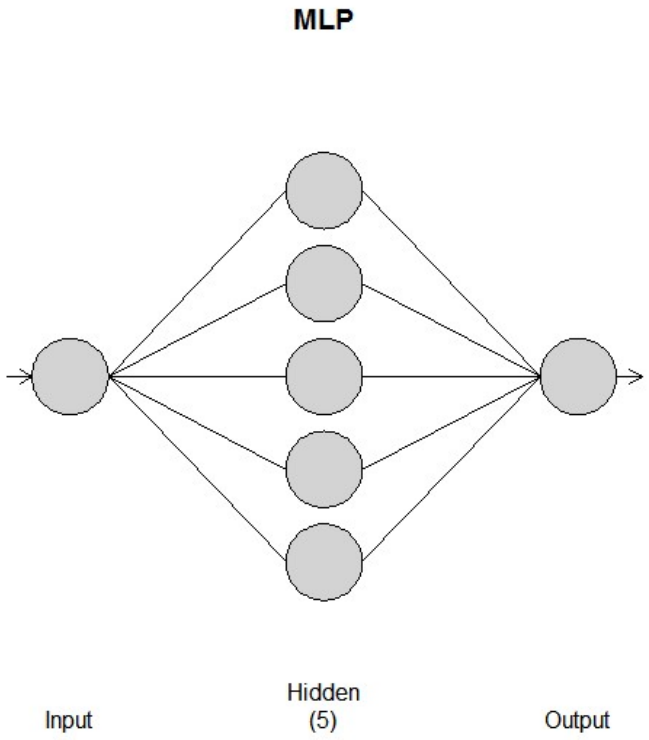
```
116       92.40846
117       92.40846
118       92.40846
119       92.40846
120       92.40846
121       92.40846
122       92.40846
123       92.40846

> plot(forecast(fitsc, h = 20) )
> lines(fitsc$fitted, col="red")
>
```



```
> library(TStools)
> library(xlsx)
> sabicCls <-
read.xlsx("C:/Users/ibin/Desktop/sabicClose.xlsx",1,header=TRUE )
> sv <- as.ts(sabicCls[,2])
> mlp.fit <- mlp(sv)
> plot(mlp.fit)
> print(mlp.fit)
MLP fit with 5 hidden nodes and 20 repetitions.
Series modelled in differences: D1.
Univariate lags: (1)
Forecast combined using the median operator.
MSE: 0.6028.
> mlp.frc <- forecast(mlp.fit,h= 20)
> mlp.frc
    Point Forecast
104    91.63721762
105    91.78455643
106    91.93192128
107    92.07929173
108    92.22666337
109    92.37403528
110    92.52140725
111    92.66877922
112    92.81615120
113    92.96352318
114    93.11089516
115    93.25826714
116    93.40563912
117    93.55301110
118    93.70038308
119    93.84775506
```

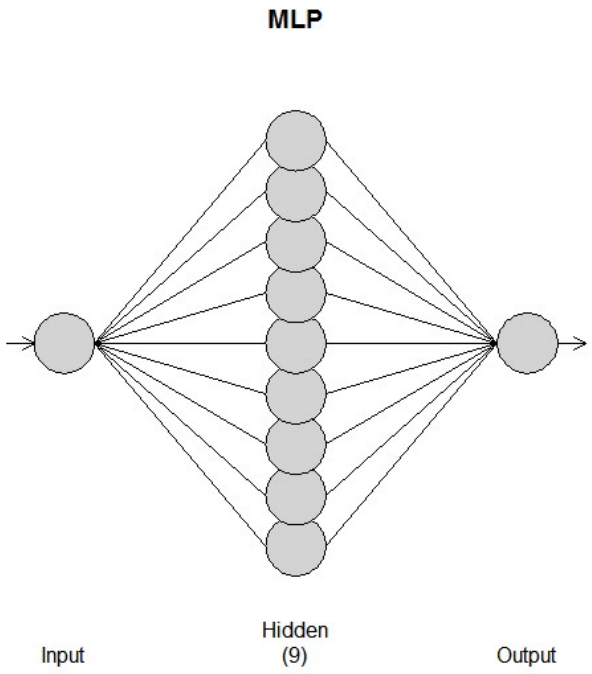
```
120    93.99512703
121    94.14249901
122    94.28987099
123    94.43724297
>
> plot(mlp.frc)
```



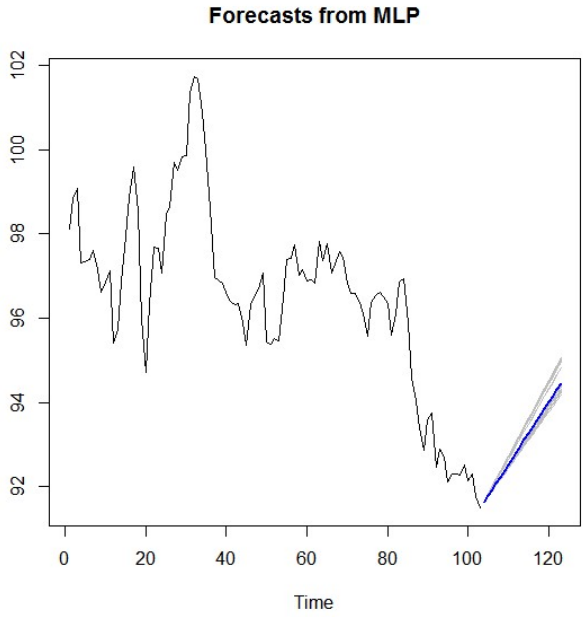



```
> mlp2.fit <- mlp(sv,hd.auto.type="valid",hd.max=10)
```

```
There were 21 warnings (use warnings() to see them)
> print(round(mlp2.fit$MSEH,4))
        MSE
H.1  0.0719
H.2  0.0714
H.3  0.0718
H.4  0.0712
H.5  0.0709
H.6  0.0697
H.7  0.0702
H.8  0.0705
H.9  0.0695
H.10 0.0696
> plot(mlp2.fit)
>
```



```
> mlp2.frc <- forecast(mlp2.fit,h= 20)
> mlp2.frc
    Point Forecast
104    91.63760763
105    91.78533574
106    91.93308947
107    92.08084864
108    92.22860898
109    92.37636957
110    92.52413021
111    92.67189086
112    92.81965151
113    92.96741216
114    93.11517281
115    93.26293347
116    93.41069412
117    93.55845477
118    93.70621542
119    93.85397608
120    94.00173673
121    94.14949738
122    94.29725803
123    94.44501869
>
> plot(mlp2.frc)
>
```



مثال آخر:

سوف نطبق شبكة عصبية على بيانات cereals و التي تعطي تقييم rating لانواع مختلفة من حبوب الإفطار إعتمادا على متغيرات مثل الكلوري calories و البروتين proteins و الدهن fat و الصوديوم sodium و الألياف fiber (المتغيرات : المحتوى الكربوني carbo و السكريات sugars و البوتاس potass و الفيتامينات vitamins و مدة الصلاحية shelf و الوزن weight و عدد اكواب الوجبة cups لن تشمل في التحليل و تترك كتمرين)

البيانات يمكن تحميلها من

https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2017/09/07122416/cereals.csv

```
> data = read.csv(file.choose(), header=T)
> head(data)
  calories protein fat sodium fiber   rating
1       70       4   1    130  10.0 68.40297
2      120       3   5     15   2.0 33.98368
```

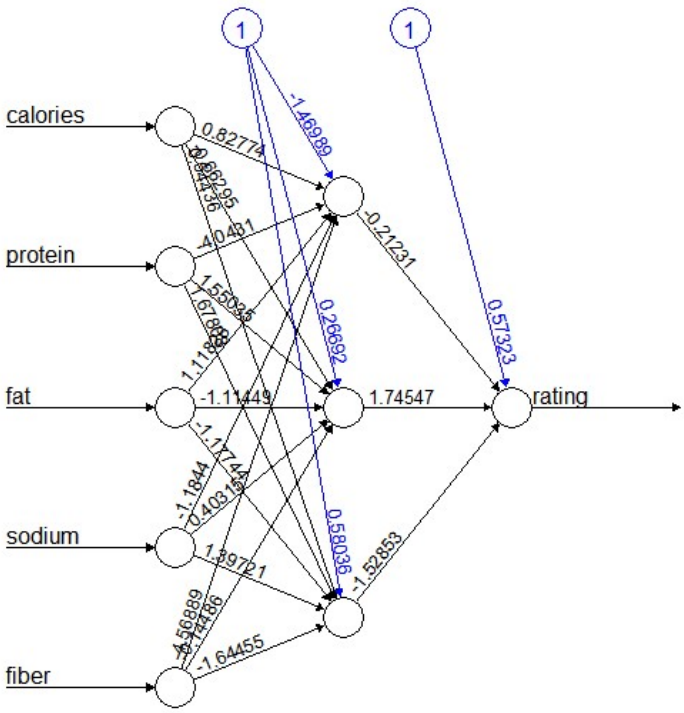
```
3       70        4    1    260    9.0 59.42551
4       50        4    0    140   14.0 93.70491
5      110        2    2    180    1.5 29.50954
6      110        2    0    125    1.0 33.17409
```

نعاين من البيانات بشكل عشوائي لتكوين مجموعتين التمرين و الإختبار سوف نستخدم 60% من البيانات للتمرين

```
> set.seed(80)
> samplesize = 0.60 * nrow(data)
> index = sample(seq_len(nrow(data)), size = samplesize )
> datatrain = data[ index, ]
> datatest = data[ -index, ]
```

من الضروري تدريج البيانات حتى لايؤثر متغير على التنبؤ لتأثير حجمه على الشبكة وسوف نستخدم min-max normalization و التي تحول البيانات إلى مجال موحد و بهذا يزال تأثير التدريج من المتغيرات

```
> max = apply(data , 2 , max)
> min = apply(data, 2 , min)
> scaled=as.data.frame(scale(data,center= min,scale=max-
min))
> library(neuralnet)
> trainNN = scaled[index , ]
> testNN = scaled[-index , ]
> NN = neuralnet(rating ~ calories + protein + fat + sodium
+ fiber, trainNN, hidden = 3 , linear.output = T )
> plot(NN)
```



Error: 0.088385 Steps: 146

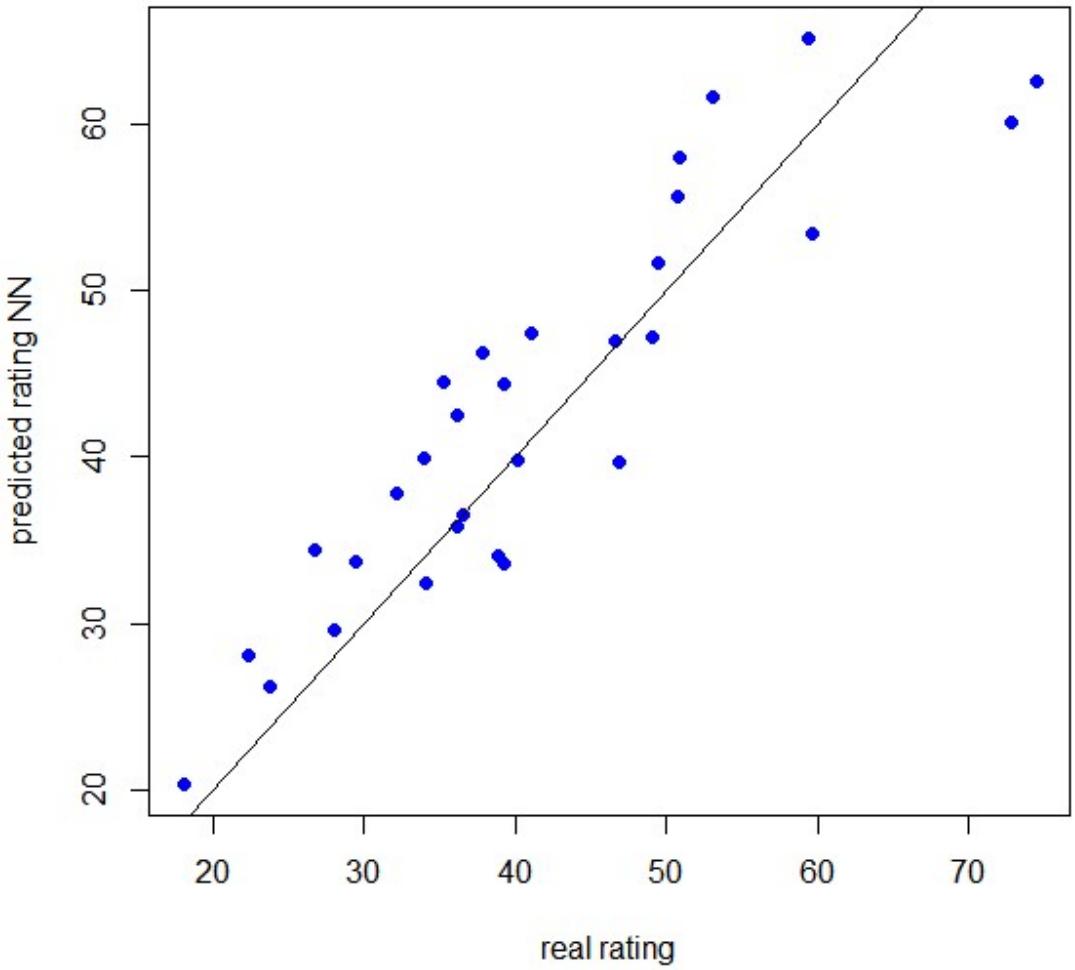
الشبكة تتكون من 5 عقد إدخال و 3 عقد مخفية و 1 عقدة إخراج.

الآن نتنبأ بإستخدام الشبكة

```
> predict_testNN = compute(NN, testNN[,c(1:5)])
> predict_testNN = (predict_testNN$net.result *
(max(data$rating) - min(data$rating))) + min(data$rating)
```

نرسم قيم المتغير rating مع القيم المتنبأة

```
> plot(datatest$rating, predict_testNN, col='blue', pch=16,
ylab = "predicted rating NN", xlab = "real rating")
> abline(0,1)
```



نحسب خطأ متوسط مربع الأساس Root Mean Square Error (RMSE) للشبكة

```
> RMSE.NN = (sum((datatest$rating - predict_testNN)^2) /
nrow(datatest)) ^ 0.5
> RMSE.NN
[1] 6.049574835
```

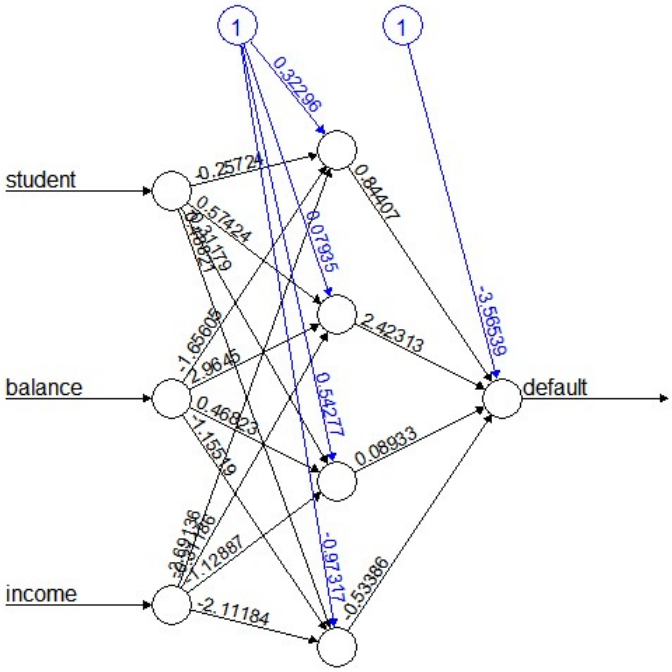
مثال آخر

يمكن تحميل البيانات من

https://raw.github.com/vincentarelbundock/Rdatasets/master/csv/ISLR/Default.csv

يترك تفسير البرنامج كتمرين

```
library(neuralnet)
cd3 = read.csv(file.choose())
names(cd3) = c("default", "student",  "balance", "income")
trainset <- cd3[1:6000, ]
testset <- cd3[6001:10000, ]
ctnet <- neuralnet(default ~ student + balance + income,
trainset, hidden = 4,lifesign = "minimal",linear.output =
FALSE, threshold = 0.1)
plot(ctnet, rep = "best")
```




```
temp_test <- subset(testset, select = c("student",
"balance","income"))
```

```
ctnet.results <- compute(ctnet, temp_test)
results <- data.frame(actual = testset$default, prediction
= ctnet.results$net.result)
results[100:115, ]
     actual   prediction
6100      0 0.0275078031
6101      0 0.0275078031
6102      0 0.0275078031
6103      0 0.0275078031
6104      0 0.0275078031
6105      0 0.0275078031
6106      0 0.0275078031
6107      0 0.0275078031
6108      0 0.0275078031
6109      0 0.0275078031
6110      0 0.0275078031
6111      0 0.0275078031
6112      0 0.0275078031
6113      0 0.0275078031
6114      0 0.0275078031
6115      0 0.0275078031
results$prediction <- round(results$prediction)
results[100:115, ]
     actual prediction
6100      0          0
6101      0          0
6102      0          0
6103      0          0
6104      0          0
6105      0          0
6106      0          0
6107      0          0
```

6108	0	0
6109	0	0
6110	0	0
6111	0	0
6112	0	0
6113	0	0
6114	0	0
6115	0	0

ملحق: الهشاشة

تعريف: الهشاشه Volatility هي الميول للتقلب بشكل كبير و متكرر على فترات عشوائية

الهشاشة العشوائية: Stochastic Volatility

التفاعل و الديناميكية بين الأسواق المالية حول العالم تغيرت بشكل مأساوي بسبب عالمية الإقتصاد. التقدم في الإتصالات وفي تجميع البيانات غيرت من طريقة معالجة و إستخدام المعلومات. في هذه الحقبة الأجهزة المالية Financial Instruments اصبحت معقدة بإضطراد و تأثيرها أصبح يطال الجميع. ولا ننسى الأزمة المالية الحديثة و التي هي مثال واضح أمامنا الآن للتحديات الجديدة التي تواجهنا و ستظل تواجهنا في عصر المعلومات.

المهارة التحليلية و القدرة على إستخراج المعلومات المفيدة من أحجام ضخمة من البيانات لكي نستوعب تعقيد الأجهزة المالية و لنقدر المخاطرة المالية Financial Risk الحاصلة أصبح ضرورة للإقتصاديين و المدراء الماليين و لمدراء المخاطر. و لكي نتوصل لهذا المستوى من المقدرة يجب أساسا معرفة بعض الشيء عن علم الحاسب و الإقتصاد و المالية و الرياضيات و الإحصاء.

وعلى هذا فإن الهندسة المالية Financial Engineering هي تجمع من هذه العلوم و طرقها النظرية و التطبيقية تتطور بسرعة.

تعريف الهشاشة الإحصائية:

الهشاشة هي مقياس إحصائي للتشتت في العائدات Returns لضمان مالي Security أو مؤشر سوق Market Index و يمكن قياس الهشاشة بالإنحراف المعياري بين العائدات من نفس الضمان المالي أو مؤشر السوق. ودائما كل ماكانت الهشاشة عالية كل مان الضمان المالي او المؤشر أكثر خطورة.

الخطورة Risk: وتعني عدم التأكد Uncertainty في العائدات المستقبلية من إستثمار Investment.

و تستخدم الهشاشة كمقياس للمخاطرة بالإضافة للقيمة المخاطرة Value at Risk (Var) و القصور المتوقع Expected Shortfall (وهو المقدار المطلوب والذي حان تسديده والذي يزيد على مقدرة المسدد).

العائد الصافي Net Return:

لنرمز بـ P_t لسعر الملكية Asset عند الزمن t فالعائد الصافي على الفترة من t-1 الى t يحسب من

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}} = \frac{P_t}{P_{t-1}} - 1$$

العائد الكلي Gross Returns:

ويحسب من

$$\frac{P_t}{P_{t-1}} = 1 + R_t$$

لوغارثم العائدات Log Returns ويسمى ايضاالعائدات المتراكمة بإستمرار Continuously Compounded Returns ويرمز لها بـ r_t وتحسب من

$$r_t = \log(1 + R_t) = \log\left(\frac{P_t}{P_{t-1}}\right) = p_t - p_{t-1}$$

حيث $p_t = \log(P_t)$ وهو لوغارثم السعر.

حساب الهشاشة لسهم شركة سابك: من 3/2/2013 وحتى 2/28/2016

ملاحظة: لايوجد فرق بين مرور الزمن من t-1 إلى t أو من t إلى t+1 أنظر لأي كتاب عن المتسلسلات الزمنية.

التاريخ	إقفال	Returns
2/28/2016	73.57	
2/25/2016	71.45	-0.02882
2/24/2016	70.99	-0.00644
2/23/2016	72.54	0.021834
2/22/2016	72.92	0.005238
2/21/2016	69.95	-0.04073
2/18/2016	70.45	0.007148
2/17/2016	69.15	-0.01845
2/16/2016	67.99	-0.01678
2/15/2016	66.66	-0.01956

2/14/2016	64.66	-0.03
2/11/2016	63.72	-0.01454
....		
3/16/2013	94	0
3/13/2013	93.5	-0.005319149
3/12/2013	93.25	-0.002673797
3/11/2013	93.25	0
3/10/2013	92.75	-0.00536193
3/9/2013	92.75	0
3/6/2013	92.5	-0.002695418
3/5/2013	92.25	-0.002702703
3/4/2013	92	-0.002710027
3/3/2013	91.75	-0.002717391
3/2/2013	92	0.002724796
	Standard Dev	0.018824

إذا هشاشة سهم سابك على الفترة المعطاة هي 1.8824%

تمرين:

أوجد الهشاشة لأسم عدة شركات قم بإختيارها من السوق المالية السعودية. رتب الشركات حسب الهشاشة و أختار الشركات الأقل هشاشة لغرض الإستثمار.

ملحق : المكتبة dlm في R

تقوم بمعالجة النماذج الحركية الخطية Dynamic Linear Models (DLM) والتي تعرف أيضا بنماذج فضاء الحالة الخطي Linear State Space Models و المكتبة تحتوي ايضا على دوال تقدير الأرجحية العظمى Maximum Likelihood Estimation لمعالم النماذج الحركية الخطية و لمرشح كالمن.

النموذج:

سوف نغير في الترميز لنتوافق مع مكتبة dlm كالتالي

النموذج الديناميكي الخطي DLM يحدد بالمعادلات التالية

$$\mathbf{y}_t = \mathbf{F}_t\boldsymbol{\theta}_t + \mathbf{v}_t, \quad \mathbf{v}_t \sim N(\mathbf{0}, \mathbf{V}_t)$$
$$\boldsymbol{\theta}_t = \mathbf{G}_t\boldsymbol{\theta}_{t-1} + \mathbf{w}_t, \quad \mathbf{w}_t \sim N(\mathbf{0}, \mathbf{W}_t)$$

مع توزيع مسبق لـ $\boldsymbol{\theta}_0$

$$\boldsymbol{\theta}_0 \sim N(\mathbf{m}_0, \mathbf{C}_0)$$

المتجه $\mathbf{y}_t$ ذا أبعاي $m \times 1$ هو متجه المشاهدات عند الزمن t بينما $\boldsymbol{\theta}_t$ متجه ذا ابعاد $p \times 1$ (لايمكن ملاحظته Unobservable) و يمثل حالة النظام عند الزمن t . $\mathbf{v}_t$ تمثل أخطاء المشاهدات و $\mathbf{w}_t$ أخطاء الحالة (او Evolution Errors) . المصفوفه $\mathbf{F}_t$ ذات أبعاد $m \times p$ و المصفوفة $\mathbf{G}_t$ ذات ابعاد $p \times p$ و مصفوفات تباين الأخطاء $\mathbf{V}_t$ و $\mathbf{W}_t$.

تعريف نماذج الحركية الخطيه في مكتبة dlm:

أبسط نموذج حركي خطي هو نموذج المشي العشوائي مع ضجة و الذي يسمى ايضا نموذج متعدد الحدود من الدرجة الأولى First Order Polynomial Model و يستخدم لنمذجة المشاهدات ذات المتغير الوحيد ومتجة الحالة ذا بعد واحد و يوصف بالعلاقة

$$y_t = \theta_t + v_t, \quad v_t \sim N(0,V)$$
$$\theta_t = \theta_{t-1} + w_t, \quad w_t \sim N(0,W)$$

النموذج ثابت (constant) اي ان جميع المصفوفات التي تصف حركية النظام لا تعتمد على الزمن (Time Invariant) . أيضا $\mathbf{F}_t = \mathbf{G}_t = [1]$ المعالم الوحيدة للنموذج هي تباينات المشاهدات و الحالة V و W و هذه غالبا ماتقدر من المشاهدات بإستخدام الأرجحية العظمى او تقنيات بيز (Bayesian Techniques) .

في المكتبة dlm يمثل نموذج حركي خطي ثابت كقائمة (list) من الأجزاء FF و V و GG و W وهي من طبقة dlm (class dlm) (أنظر ملحق R).

نموذج مشي عشوائي مضاف إلية ضجة مع $V = 0.8$ و $W = 0.1$ يعرف في مكتبة dlm في R كالتالي

```
> dlm(FF = 1, V = 0.8, GG = 1, W = 0.1, m0 = 0, C0 = 100)
```

لاحظ أن متوسطات و مصفوفة التباين لتوزيع θ_0 يجب أن تحدد وهنا هي قيم وحيدة m0 = 0 و C0 = 100 لأنها جزء اساسي لتحديد النموذج. طريقة اخرى لتعريف النموذج هي

```
> dlmModPoly(order = 1, dV = 0.8, dW = 0.1, C0 = 100)
```

الحد order يحدد درجة كثيرة الحدود و dV و dW عناصر القطر لكل من V و W .

تقدير الأرجحية العظمى:

الدالة dlmMLE تقوم بتقدير معالم النموذج وذلك بإمدادها بالمشاهدات و قيم اولية. هذه الدالة تحتاج إلى دالة لبناء شكل dlm و سوف نستعرض هذا لبيانات نهر النيل Nile و الذي يصفه نموذج المشي العشوائي مع الضجة السابق. تباينات المشاهدات و الحالة غير معروفة وسوف نمعلم التباينات على القياس اللوغارثمي لضمان الإيجابية كالتالي

```
> buildFun <- function(x) {
  dlmModPoly(1, dV = exp(x[1]), dW = exp(x[2]))}
```

نبدأ الأمثلية Optimization بقيمة اولية إختيارية (0,0) مقدر الأرجحية العظمى للمعالم توجد كالتالي

```
> library(dlm)
> data(Nile)
> dlmModPoly(order = 1, dV = 0.8, dW = 0.1, C0 = 100)
> buildFun <- function(x) {
+   dlmModPoly(1, dV = exp(x[1]), dW = exp(x[2]))}
>
> fit <- dlmMLE(Nile, parm = c(0,0), build = buildFun)
> fit$conv
[1] 0
> dlmNile <- buildFun(fit$par)
> V(dlmNile)
         [,1]
[1,] 15099.8
> W(dlmNile)
          [,1]
[1,] 1468.432
```

و بمقارنتها بالدالة structTS نجد

```
> StructTS(Nile, "level")

Call:
StructTS(x = Nile, type = "level")

Variances:
  level  epsilon
   1469    15099
>
```

المراجع

باللغة العربية:

1) طرق التنبؤ الإحصائي (الجزء الأول) : عدنان ماجد بري، جامعة الملك سعود

باللغة الإنجليزية:

1) Liu, L.-M. and Hanssens , D.M. ,1982 ,"Identification of Multiple-Input Transfer Function Models" , Communications in Statistics , A11 , 297-314 .

2) Makridakis, S. , Weelwright, S.C. and McGee, V.E., "Forecasting: Methods and Applications" , John Wiley & Sons , U.S.A. , 1983 .

3) Wei, W.W.S., "Time Series Analysis: Univariate and Multivariate Methods" , Addison-Wesley Publishing Company , U.S.A. , 1990 .

4) Brockwell, P.J and Davis, R.A,"Introduction to Time Series and Forecasting", 3rd. Ed. 2016, Springer.

5) Helmut, L.,"New Introduction to Multiple Time Series Analysis", Springer, 2005.

6) Petris, G., Petrone, S., and Campagnoli, P.,"Dynamic Linear Models with R", Springer, 2009.
7) Commandeur, J. and Koopman, S., "An Introduction to State Space Time Series Analysis", Oxford University Press, 2007.
8) Shumway, R.H. and Stoffer, D.S.,"Time Series Analysis and Its Application", 3rd. Ed., Springer 2011.
9) Cryer, J.D., and Chan, K.S.,"Time Series Analysis with Applications in R", 2nd. Ed., Springer, 2008.