

Regresión Armónica Dinámica Lineal

Marcos Bujosa

Marzo de 2007

Abstract

Clase práctica de uso de los programas que componen la “*toolbox*” **LDHR** para GNU/Octave

Primero copie los datos de texto plano (ASCII) en su directorio de trabajo

Comenzaremos descomponiendo la serie del Índice de Producción Industrial de España (datos de enero de 1975 a noviembre de 2004).

Para ello debemos “cargar” la variable IPI.

```
1a <codigoLDHR.m 1a>≡ 1b>
    load ipi.txt
```

La función `timefmt2` genera una columna de fechas, de manera que las siguientes tres líneas, hacen lo siguiente:

línea 1. genera una columna de fechas de datos mensuales (12), que comienzan en el primer mes de 1975 (“1975”, “1”), y en total hay tantas fechas como datos tiene `ipi` (`length(ipi)`)

línea 2. define una matriz cuya primera columna son las fechas y la segunda los datos del IPI

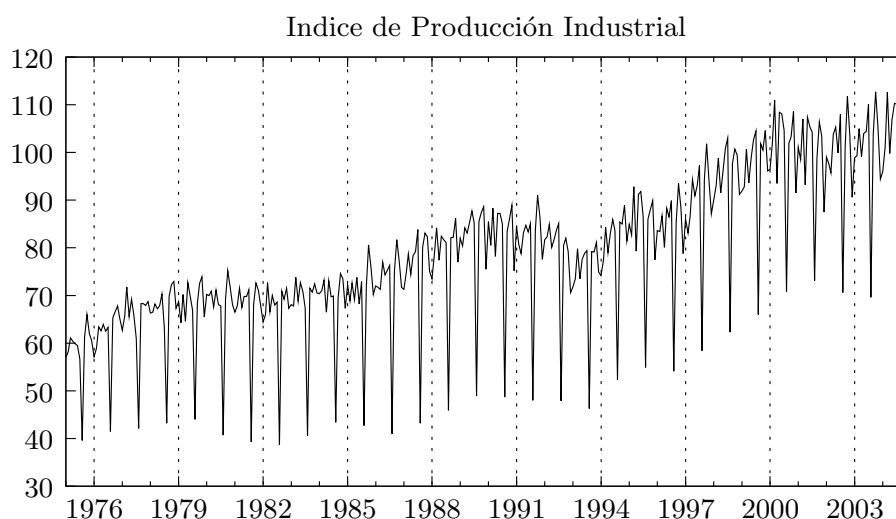
línea 3. guarda dicha matriz en un fichero de texto con mismo nombre y extensión `.txt`

```
1b <codigoLDHR.m 1a>+≡ <1a 1c>
    fechasIPI = timefmt2 ([ 12, 1975, 1, length(ipi) ]);
    ipi_fechas=[fechasIPI, ipi];
    save ipi_fechas.txt ipi_fechas
```

Ahora podemos importar dicho fichero (con fechas incluidas) desde algún otro programa¹.

Si queremos ver el gráfico de la serie desde GNU Octave debemos teclear

```
1c <codigoLDHR.m 1a>+≡ <1b 1d>
    plot(ipi)
```

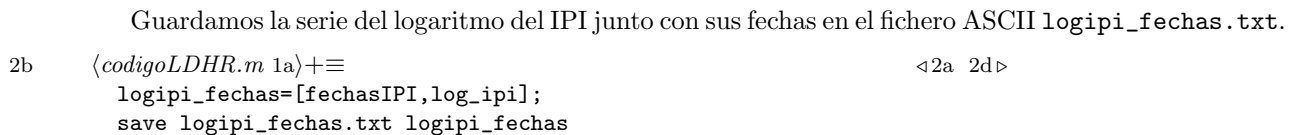


Analizaremos la serie del IPI en logaritmos, por ello definimos la nueva variable

```
1d <codigoLDHR.m 1a>+≡ <1c 2a>
    log_ipi=log(ipi);
```

¹por ejemplo, para poder realizar los gráficos de este documento con un eje temporal se ha empleado el programa GNUplot y los ficheros `*.txt` generados desde GNUOctave

2a	$\langle \text{codigoLDHR.m } 1\text{a} \rangle + \equiv$	$\triangleleft 1\text{d } 2\text{b} \triangleright$
	<code>plot(log_ipi)</code>	



```
2c      <figlogipi.gp 2c>≡
        set xdata time
        set timefmt "%Y%m"
        set format x "%Y"
        set size 1.2,1 #.75
        set origin 0,0
        set grid xtics
        set title 'Índice de Producción Industrial'
        plot ["197501":"200411"] 'logipi_fechas.txt' using 1:2 notitle w l l
```

2d $\langle \text{codigoLDHR.m 1a} \rangle + \equiv$ $\langle 2b \ 5a \rangle$
 help autodhr

Argumentos de salida Lo que aquí nos interesa son los cuatro primeros argumentos, es decir

VAR Las varianzas asociadas a las innovaciones de los modelos de

- el componente irregular e
- la tendencia T
- Los distintos componentes estacionales

P Los periodos de los componentes del modelo DHR

- **Inf**: tendencia
- **12**: Componente estacional de periodo 12
- **6**: Componente estacional de periodo 6
- Etc.

TVP Modelo DHR para cada componente. **TVP** es una matriz de dos filas, y tantas columnas como componentes DHR tenga el modelo. Los elementos de la matriz indican el módulo de las raíces del proceso AR(1) o AR(2) que modula la amplitud de las oscilaciones de cada componente. Por ejemplo, si **P** es

$$P = (\text{Inf} \quad 12 \quad 6 \quad 4 \quad 3 \quad 2.4 \quad 2)$$

entonces la matriz **TVP**

$$TVP = \begin{pmatrix} 1 & 1 & 1 & 1 & .9 & 0 & .9 \\ 1 & 0 & .7 & 0 & 0 & 0 & .8 \end{pmatrix}$$

indica un modelo IRW para la tendencia y un RW para los componentes de periodos 12, y 4; un modelo SRW con parámetro de suavidad 0.7 para el componente de periodo 6; el componente de periodo 3 es un AR(1) con coeficiente 0.9 y el componente de periodo 2 es un AR(2) con coeficientes 0.9 y 0.8. Puesto que sólo hay ceros en la columna correspondiente al componente de periodo 2.4, no existe dicho componente en el modelo.

oar Es el orden autor-regresivo empleado en la estimación del espectro de la serie analizada.

MCNN Toma el valor 1 si la estimación del modelo identificado ha sido realizada mediante un algoritmo de “mínimos cuadrados no negativos”. Toma el valor 0 en caso contrario.

NVR Toma el valor de los hiper-parámetros NVR. Además, muestra por pantalla una tabla resumen de los modelos identificados para los distintos órdenes auto-regresivos; junto con algunos gráficos informativos sobre el ajuste espectral y la ganancia de los filtros.

Argumentos de entrada El número de argumentos de entrada introducidos en la función `autodhr` puede variar. Los dos primeros argumentos son indispensables

`y` la serie a analizar.

`p` la periodicidad de los datos (12 si son mensuales, 4 si son trimestrales, 1 si son anuales, 3 si son cuatrimestrales, etc.).

Con esa información el programa empleará los parámetros de decisión por defecto para identificar un modelo DHR para la serie.

Argumentos de entrada opcionales El resto de parámetros son opcionales. Si no se incluyen, o bien se introducen como matrices vacías, el programa empleará los valores por defecto. Por ejemplo

```
[VAR,P,TVP,oar]=autodhr(y,12,[],[],PaP,TVPaP)
```

empleará para el tercer, cuarto y séptimo argumentos de entrada los valores por defecto (nótese que el séptimo ni aparece). En el caso

```
[VAR,P,TVP,oar]=autodhr(y,12)
```

empleará todos los argumentos de entrada opcionales con los valores fijados por defecto (*que se muestran entre paréntesis en la ayuda mostrada con el comando `help autodhr`*).

El resto de argumentos de entrada son:

Arg. opc. 1: `PD`: Parámetros de Decisión para la selección del modelo. Es un vector con cinco números:

`1im-1` Criterio para decidir cuando suponer que una raíz es unitaria. El valor por defecto 0.05 significa que una discrepancia de 0.05 o menos respecto a la unidad significa que se supondrá que la raíz es unitaria, es decir, raíces de módulo superior a 0.95 se tomarán como raíces de módulo uno.

`1im-0` Criterio para decidir cuando suponer que una raíz no debe tomarse en cuenta en el modelo DHR. El valor por defecto 0.45 significa que una raíz de módulo inferior a 0.45 no será tomada en cuenta en el modelo de componentes DHR.

`1im-T` Criterio para decidir que frecuencias deben incluirse en el componente de tendencia. El valor 36 significa que aquellas raíces asociados a periodos de longitud mayor o igual a 36 deben incorporarse a la tendencia; es decir, si la serie es mensual, ciclos de 36 periodos (meses) o más deben estar incorporados en la tendencia. Valor por defecto `p*3`

`1im-S` Realiza la función análoga para los componentes estacionales. Un valor muy elevado significa exigir que los picos espectrales estén muy muy cerca de las frecuencias estacionales para que sean incorporados al modelo DHR. Un numero bajo significa que picos espectrales distantes de las frecuencias estacionales serán incorporados a los componentes DHR estacionales.

`nps` Es el máximo número de parámetros de suavidad de los modelos DHR. Valores posibles son 1 o 2 (use siempre el valor 2).

Arg. opc. 2: `RO` Es el rango de órdenes auto-regresivos empleados en la estimación del espectro. Por ejemplo el vector `[20:30]` significa que el algoritmo debe emplear AR-espectros de ordenes 20, 21, ..., 30 en el proceso de identificación del modelo DHR.

Arg. opc. 3: `PaP` Son los Periodos “a Priori” de los componentes del modelo. Por defecto Tendencia, y componente estacional de periodo `p` (segundo argumento de `autodhr`) y sus todos sus armónicos. Por ejemplo,

```
PaP=12./[0:6]
```

o lo que es lo mismo

```
PaP=[Inf 12 6 4 3 2.4 2]
```

indica al programa que debe buscar modelos con tendencia, componente estacional de periodo 12 y “todos” sus armónicos.

Arg. opc. 4: `TVPaP` Modelos DHR “a priori” de los componentes “a priori” (`TVPaP`). Por defecto modelos IRW para todos los componentes excepto el de periodo 2.

Arg. opc. 5: `fijo` No trata de identificar el modelo plausible más parecido al modelo DHR “a priori” indicado por los argumentos `PaP` y `TVPaP`, sino que fuerza la estimación de dicho modelo DHR “a priori”. Los valores posibles son 1 para forzar la estimación del modelo “a priori”, y 0 para no forzar (0 es el valor por defecto).

Análisis del IPI

1 Identificación del modelo estándar

Definamos el modelo “estándar” para el logaritmo del IPI (serie mensual):

```
5a  <codigoLDHR.m 1a>+≡ <2d 5b>
      PaP=12./(0:6)
      TVPaP=[1 1 1 1 1 1 1;1 0 0 0 0 0]
```

Identifiquemos y estimemos el modelo DHR:

```
5b  <codigoLDHR.m 1a>+≡ <5a 5c>
      [VAR,P,TVP,oar]=autodhr(log_ipi,12,[],[],PaP,TVPaP)
```

Podemos calcular los ratios de varianzas (NVR) del siguiente modo:

```
5c  <codigoLDHR.m 1a>+≡ <5b 5d>
      NVR=VAR(2:8)./VAR(1)
```

Función dhrfilt y estimación de los componentes

Con esta función estimamos los componentes DHR del modelo obtenido mediante la función `autodhr`.

```
5d  <codigoLDHR.m 1a>+≡ <5c 6a>
      filt=0;
      [trend,season,cycle,irreg]=dhrfilt(log_ipi,P,TVP,VAR,12,filt,0);
```

Argumentos de entrada

y Serie temporal a filtrar

P Vector con los periodos de los componentes DHR. En este caso $P=[12 \ 6 \ 4 \ 3 \ 2.4 \ 2]$.

TVP Matriz que describe el modelo DHR

VAR Vector con las varianzas de las innovaciones de los componentes DHR. La primera varianza corresponde al componente irregular.

p Periodicidad de los datos. En el caso de datos mensuales $p=12$; en el caso de datos trimestrales $p=4$; para datos anuales $p=1, \dots$

filt Tipo de filtrado

filt=0 componentes suavizados empleando la función **e4trend** de la toolbox **E4**.

filt=1 componentes sin suavizar estimados empleando la función **e4trend** de la toolbox **E4**.

filt=2 componentes estimados y suavizados mediante el filtro de intervalo fijo. Esta opción permite obtener las estimaciones de cada uno de los componentes DHR por separado (es decir, el componente de periodo 12, el de periodo 6, ...).

g Si $g=0$ no se muestran los gráficos de las estimaciones (comportamiento por defecto).

Argumentos de salida

trend es una matriz cuya primera columna es la tendencia-ciclo estimada. En el caso de que haya más columnas (esto sólo ocurre cuando se han estimado la tendencia y el ciclo por separado), la segunda corresponde a la tendencia y las siguientes a los componentes del ciclo.

season es una matriz cuya primera columna es el componente estacional completo; y las siguientes columnas (de existir) son los distintos componentes DHR estacionales (solo se muestran con el procedimiento de filtrado **filt=2**).

cycle es una matriz cuya primera columna es el ciclo completo estimado, y las restantes columnas son los distintos componentes DHR cíclicos (en general esta matriz es un vector de ceros)

irreg es el componente irregular estimado.

Por tanto, el componente estacional completo es

```
6a <codigoLDHR.m 1a>+≡ <5d 6b>
    DHR_estacional=season(:,1);
```

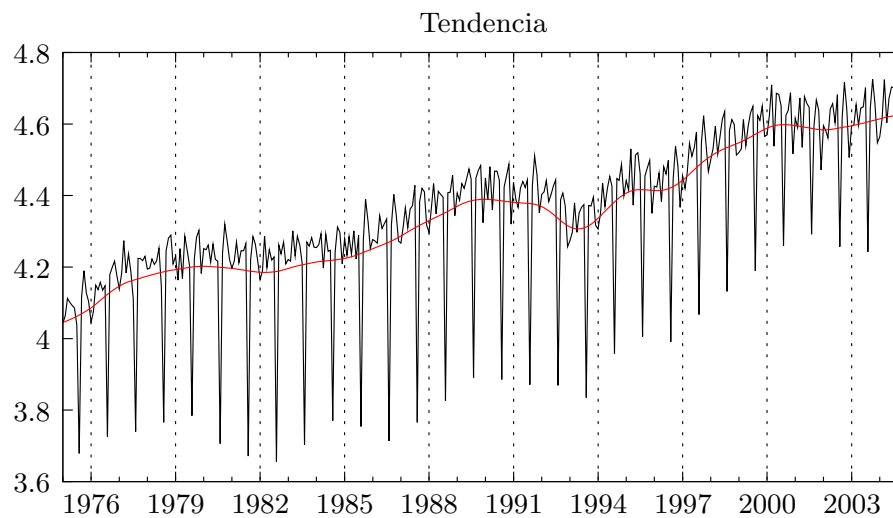
y el componente ciclo tendencia es

```
6b <codigoLDHR.m 1a>+≡ <6a 6c>
    DHR_tendencia=trend(:,1);
```

veamos los distintos componentes estimados. Primero la serie y su tendencia

```
6c <codigoLDHR.m 1a>+≡ <6b 7a>
    logipi_COMP_fechas=...
    [timefmt2(12,1975,1,size(log_ipi,1)),log_ipi,trend(:,1),season(:,1),irreg];
    save logipi_COMP_fechas.txt logipi_COMP_fechas

    plot([log_ipi,trend(:,1)])
```



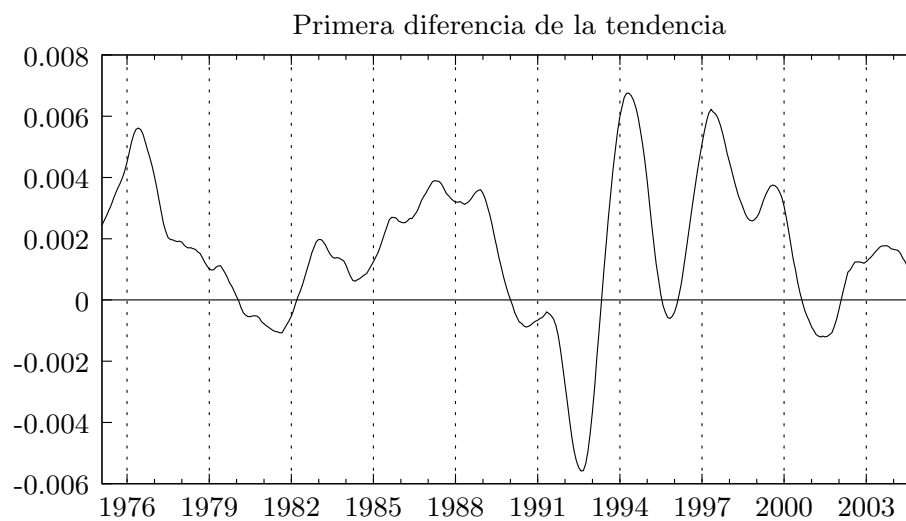
la tendencia estimada incorpora información sobre el ciclo. Para poder ver visualizar claramente dicha información es necesario diferenciar la tendencia

7a `<codigoLDHR.m 1a>+≡` `<6c 7b>`

```
dtrend=diff(trend(:,1));

logipi_DT_fechas=[timefmt2(12,1975,2,size(dtrend,1)),dtrend];
save logipi_DT_fechas.txt logipi_DT_fechas

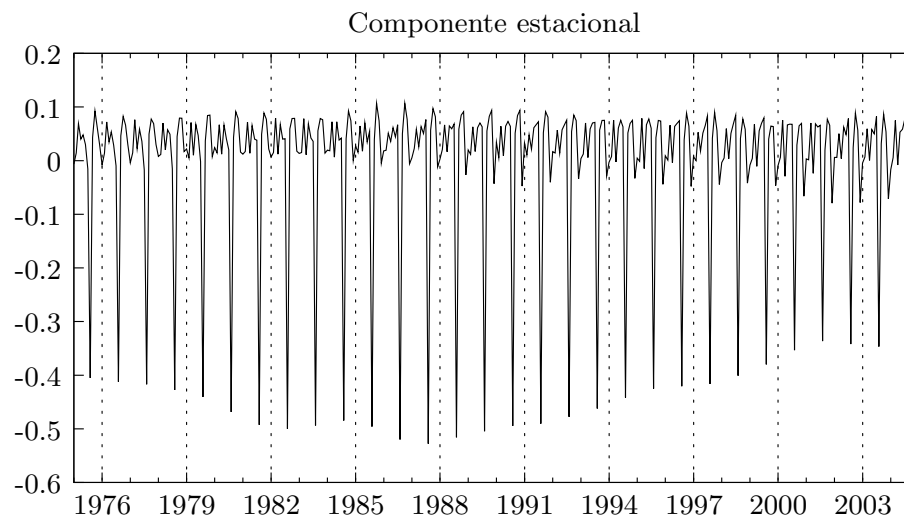
plot(dtrend)
```



el gráfico del componente estacional completo es

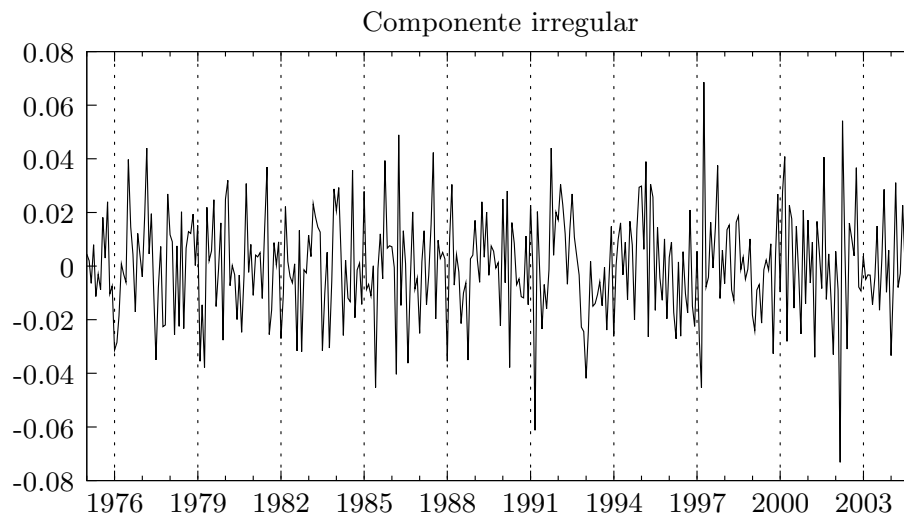
7b `<codigoLDHR.m 1a>+≡` `<7a 8a>`

```
plot(season(:,1))
```



y el del componente irregular

8a `<codigoLDHR.m 1a>+≡` `plot(irreg)` `<7b 8b>`



1. Pruebe a sustituir el cero del último argumento de entrada por un uno.
2. Pruebe, además, a sustituir el argumento *filt* por un 2.

2 Forzando el modelo estándar

Si queremos forzar el modelo “estándar” (empleando el mismo orden AR [`oar=34`] para acelerar la estimación) tecleamos:

8b `<codigoLDHR.m 1a>+≡` `[VAR2,P2,TVP2]=autodhr(log_ipi,12,[],[34],PaP,TVPaP,1)` `<8a 8c>`

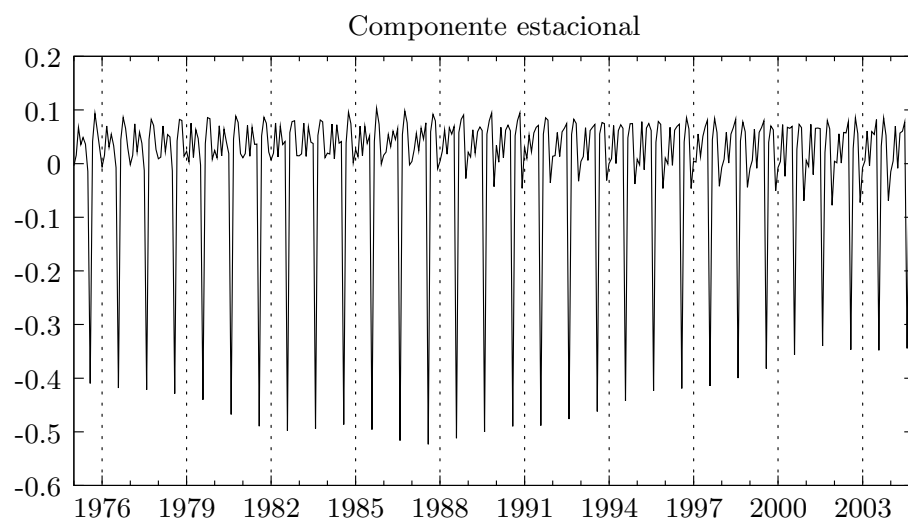
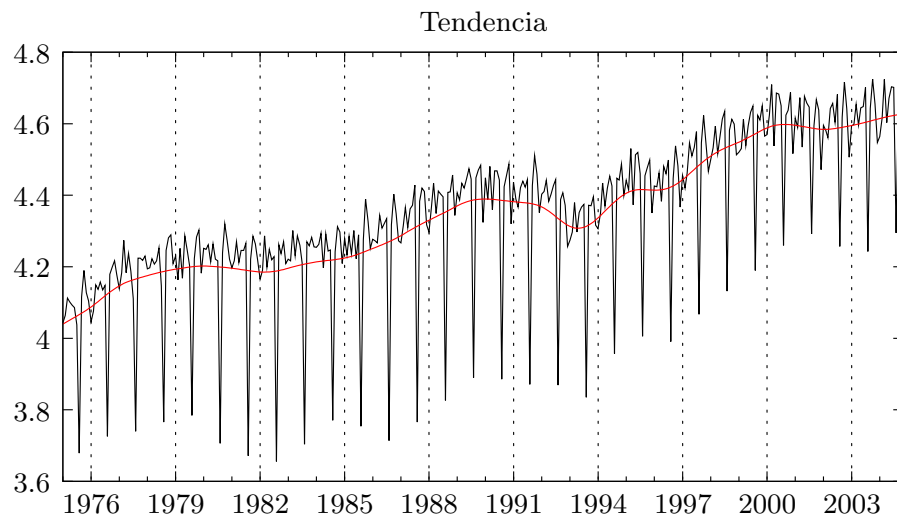
donde el nuevo vector NVR es:

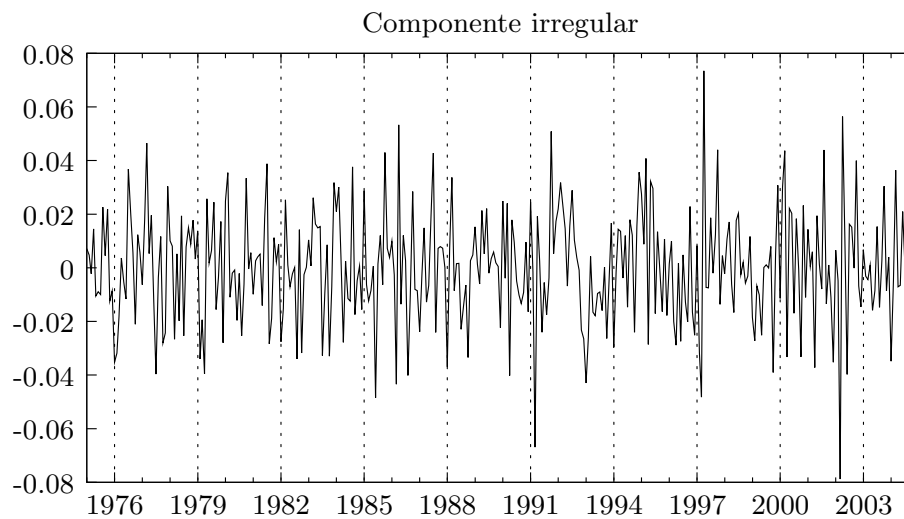
8c `<codigoLDHR.m 1a>+≡` `NVR2=VAR2(2:8)./VAR2(1)` `<8b 9>`

veamos los nuevos componentes estimados:

```
9  <codigoLDHR.m 1a>+≡ <8c 10a>
    [trend2,season2,cycle2,irreg2]=dhrfilt(log_ipi,P2,TVP2,VAR2,12,filt,1);
    logipi_COMP2_fechas=...
        [timefmt2(12,1975,1,size(log_ipi,1)),log_ipi,trend2(:,1),season2(:,1),irreg2];
    save logipi_COMP2_fechas.txt logipi_COMP2_fechas

    % plot([log_ipi,trend2(:,1)])
    % plot(season2(:,1))
    % plot(irreg2)
```





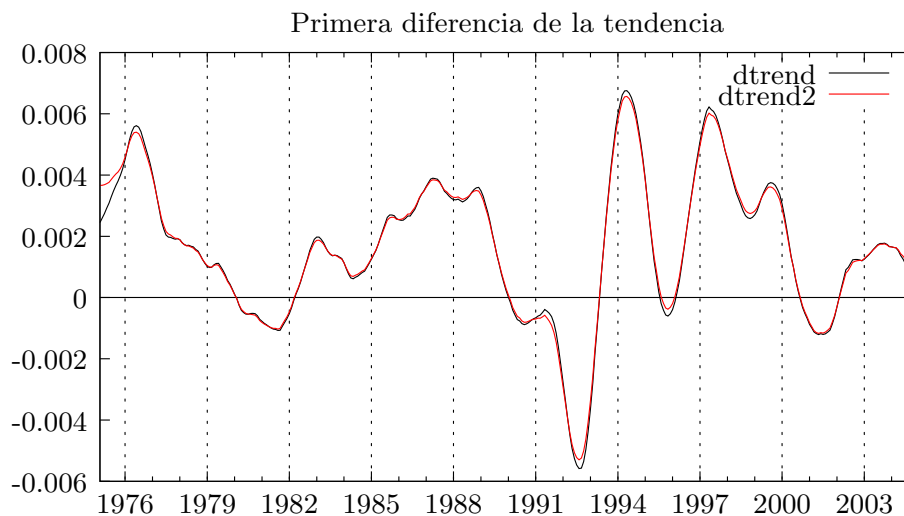
Puesto que hemos forzado un modelo IRW para la tendencia, en lugar del SRW (smooth random walk) estimado anteriormente, es de esperar que la primera diferencia de la tendencia sea algo más suave:

10a `<codigoLDHR.m 1a>+≡` <9 10b>

```
dtrend2=diff(trend2(:,1));

logipi_DT2_fechas=[timefmt2(12,1975,2,size(dtrend,1)),dtrend,dtrend2];
save logipi_DT2_fechas.txt logipi_DT2_fechas

plot([dtrend,dtrend2])
```



Se puede observar que en este ejemplo particular las diferencias son mínimas; ello es debido a que el modelo SRW con parámetro de suavidad .94 para la tendencia no es muy distinto de un modelo IRW.

3 Variando los parámetros de decisión para forzar la identificación

Otra forma de forzar el modelo “estándar” pero de manera menos estricta es variar los parámetros de decisión PD

10b `<codigoLDHR.m 1a>+≡` <10a 11a>

```
[VAR3,P3,TVP3,oar3]=autodhr(log_ipi,12,[.1,.45,36,500,2],[],PaP,TVPaP)
NVR3=VAR3(2:8)./VAR3(1)
```

en el caso anterior el primer parámetro de decisión es 0.1 en lugar de su valor por defecto 0.05. Con ello garantizamos que raíces de módulo mayor a 0.9 sean tomadas como “unitarias” (ie, de módulo uno). Con los valores por defecto, únicamente raíces de módulo mayor a .95 son tomadas como “unitarias”. Por esto la segunda raíz asociada a la tendencia que originalmente era 0.941 se mantenía como distinta de la unidad (Modelo SRW), pero con el nuevo valor del parámetro de decisión pasa a ser tomada como “unitaria” (Modelo IRW).

4 Identificación libre

Hasta ahora hemos solicitado al programa de ordenador que estime directamente el modelo “estándar” o bien que trate de identificar un modelo parecido al modelo “estándar”. Esto lo hemos hecho introduciendo los siguientes parámetros:

```
PaP=12./(0:6)
TVPaP=[1 1 1 1 1 1 1;1 0 0 0 0 0 0]
```

Sin embargo, el modelo “a priori” por defecto es más general que este, ya que permite identificar modelos de tipo AR, RW, SRW, e IRW para “todos” los componentes exceptuando el de periodo 2. Es decir, que para una serie mensual como el IPI, los parámetros por defecto son:

```
PaP=12./(0:6)
TVPaP=[1 1 1 1 1 1 1;1 1 1 1 1 1 0]
```

```
11a <codigoLDHR.m 1a>+≡ <10b 11b>
[VAR4,P4,TVP4,oar4]=autodhr(log_ipi,12)
NVR4=VAR4(2:8)./VAR4(1)
```

En este caso obtenemos el mismo modelo que con el procedimiento inicial (Ojo! esto no es así en todos los casos, pudiendo aparecer modelos más complejos para los componentes estacionales).

Análisis del valor añadido bruto de la industria (SIN CONSTRUCCIÓN)

Cargamos la serie trimestral del valor añadido bruto de la industria (muestra: 1980-I a 2004-III).

```
11b <codigoLDHR.m 1a>+≡ <11a 11c>
load vabindus.txt
```

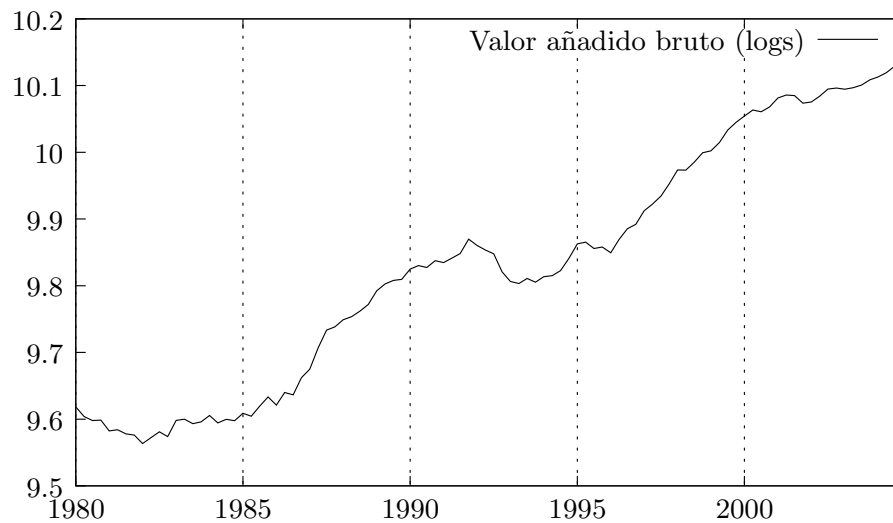
Tomamos logaritmos como hicimos con el IPI

```
11c <codigoLDHR.m 1a>+≡ <11b 11d>
log_vabindus=log(vabindus);
```

Para ver la apariencia de la serie en logaritmos tecleamos

```
11d <codigoLDHR.m 1a>+≡ <11c 12a>
plot(log_vabindus)

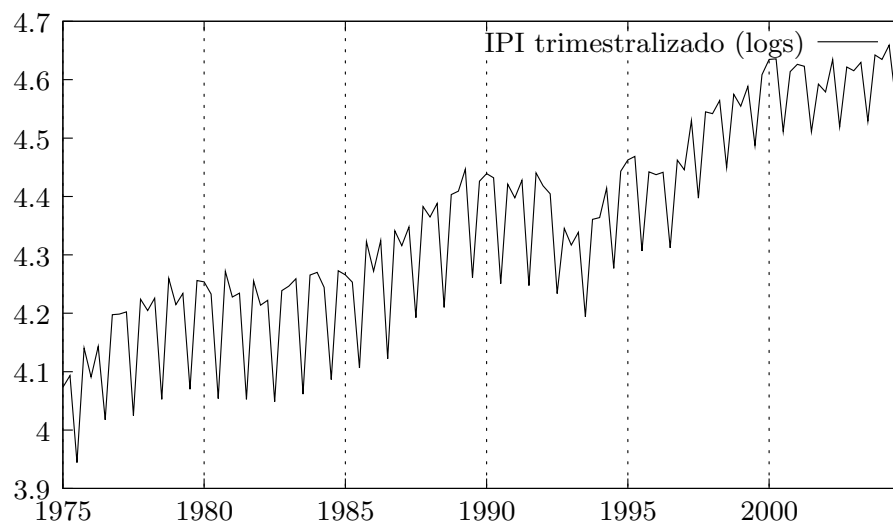
logvab_fechas=[timefmt2(4,1980,1,size(log_vabindus,1)),log_vabindus];
save logvab_fechas.txt logvab_fechas
```



Queremos comprobar hasta que punto “la derivada” (en realidad “primera diferencia”) de la tendencia del IPI y del Valor añadido bruto son coincidentes o si por el contrario una adelanta a la otra en la datación del ciclo. El problema a la hora de realizar esta comprobación, es que el IPI es mensual, y el VAB es cuatrimestral; por ello es necesario trimestralizar el logaritmo del IPI.

12a `<codigoLDHR.m 1a>+≡` `[logipi_trim] = acumula(log_ipi,3,0); %trimestralizacion` `<11d 12b>`

12b `<codigoLDHR.m 1a>+≡` `<12a 12c>`
`logipi_trim_fechas=[timefmt2(4,1975,1,size(logipi_trim,1)),logipi_trim];`
`save logipi_trim_fechas.txt logipi_trim_fechas`
`plot(logipi_trim)`



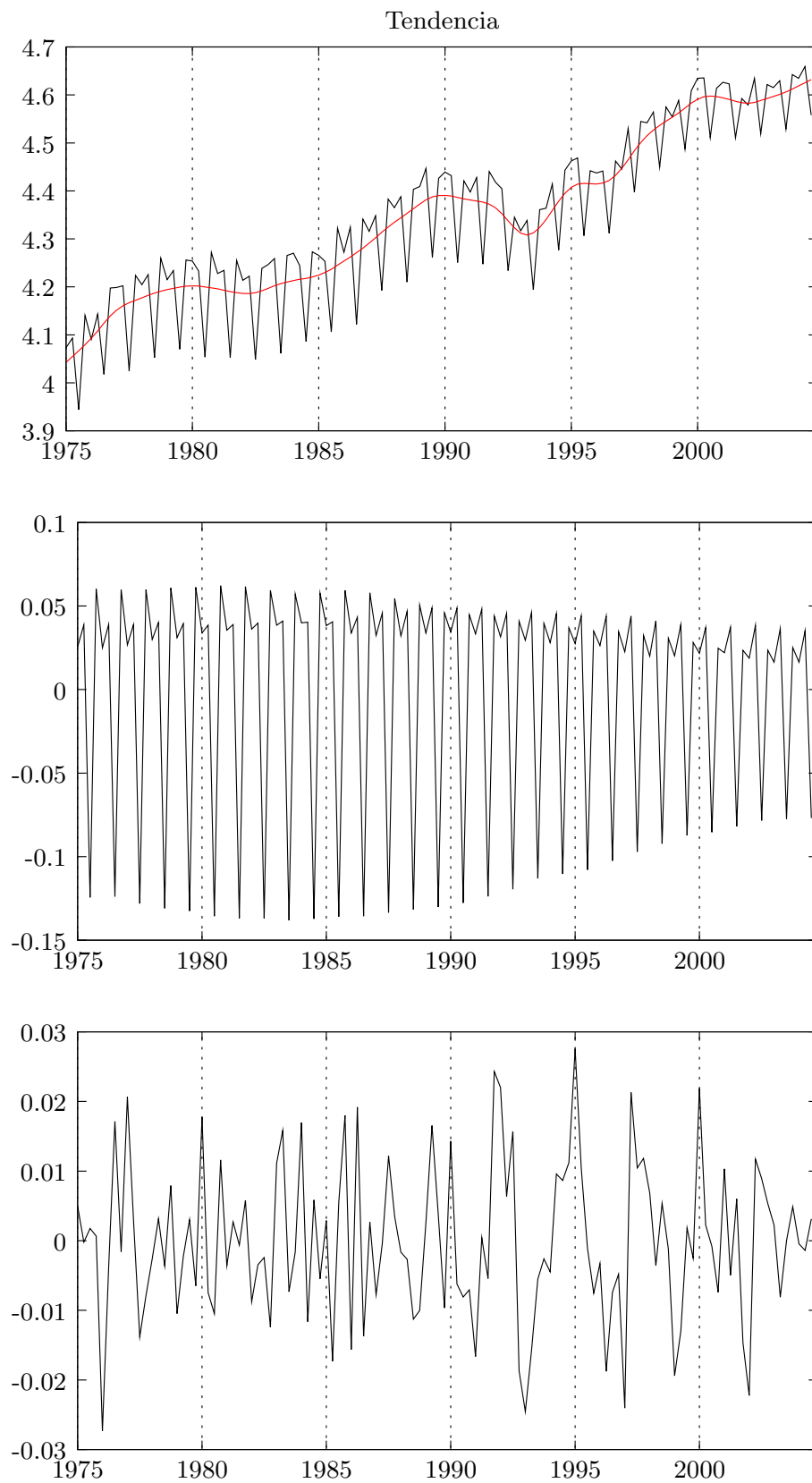
5 Forzando el modelo estándar para la serie trimestral

Estimamos (sin identificación previa) el modelo DHR “estándar” de manera forzada:

12c `<codigoLDHR.m 1a>+≡` `<12b 13>`
`[VAR5,P5,TVP5,oar5]=autodhr(logipi_trim,4,[],[],[],[1 1;1 0],1)`

`[trend5,season5,cycle5,irreg5]=dhrfilt(logipi_trim,P5,TVP5,VAR5,4,filt,1);`
`logipi_COMP5_fechas=...`
`[timefmt2(4,1975,1,size(logipi_trim,1)),logipi_trim,trend5(:,1),season5(:,1),irreg5];`
`save logipi_COMP5_fechas.txt logipi_COMP5_fechas`

`% plot([logipi_trim,trend5(:,1)])`
`% plot(season5(:,1))`
`% plot(irreg5)`



13

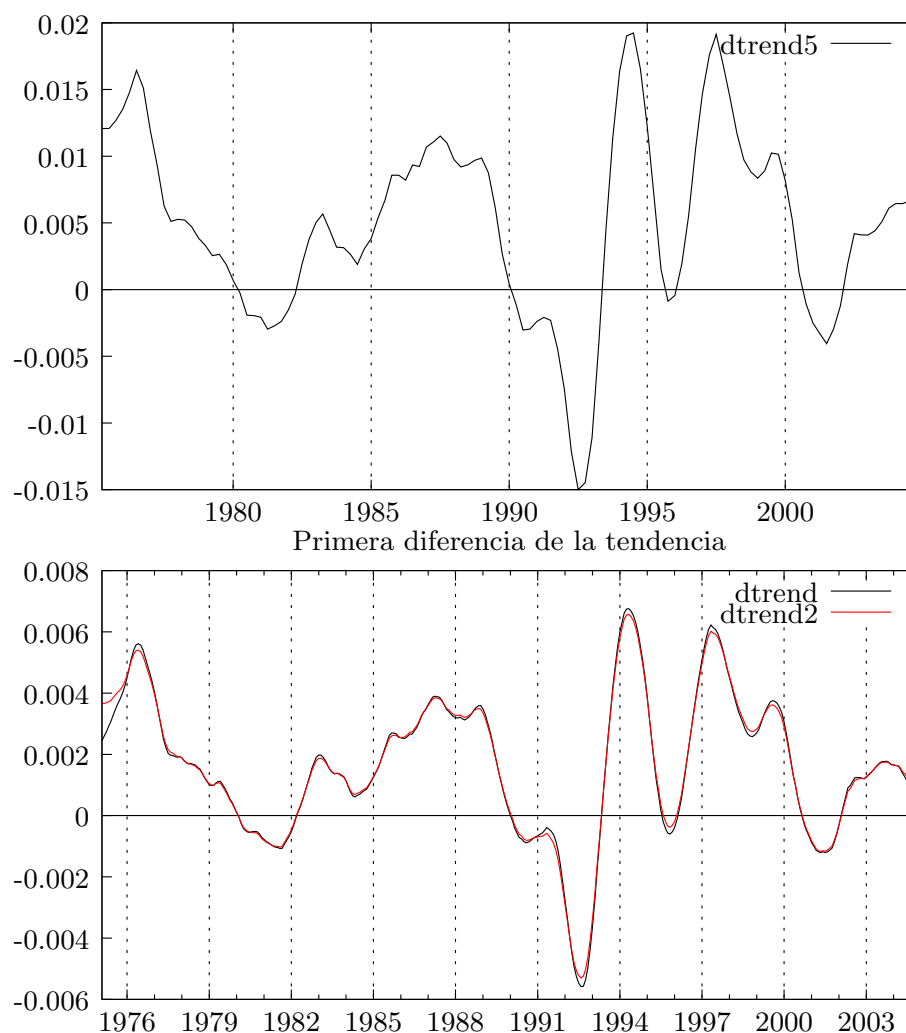
```
<codigoLDHR.m 1a>+≡
dtrend5=diff(trend5(:,1));
```

<12c 14>

```
logipi_DT5_fechas=[timefmt2(4,1975,2,size(dtrend5,1)),dtrend5];
save logipi_DT5_fechas.txt logipi_DT5_fechas
```

```
plot(dtrend5)
```

```
plot([dtrend,dtrend2])
```



6 Filtrando la tendencia del Valor Añadido Bruto

Queremos obtener un componente de tendencia de $y = \log_vabindus$ que contenga información sobre las mismas frecuencias que están incluidas en la tendencia obtenida para el IPI trimestralizado.

Por ello vamos a filtrar $\log_vabindus$ con el mismo modelo de tendencia que hemos empleado para el IPI trimestralizado; es decir una tendencia IRW con un valor del NVR de $VAR5(2)/VAR5(1)$. Por ello introducimos los siguientes valores para la función `dhrfilt`:

- serie a filtrar: $y = \log_vabindus$
- periodos de los componentes: $P = [Inf]$
- modelo DHR: $TVP = [1; 1]$
- varianzas: $VAR = VAR5(1:2)$
- periodicidad de la serie: $p = 4$

```
14  <codigoLDHR.m 1a>+≡                                     <13 15>
    [trend6,season6,cycle6,irreg6]=dhrfilt(log_vabindus,[Inf],[1;1],VAR5(1:2),4,filt,1);

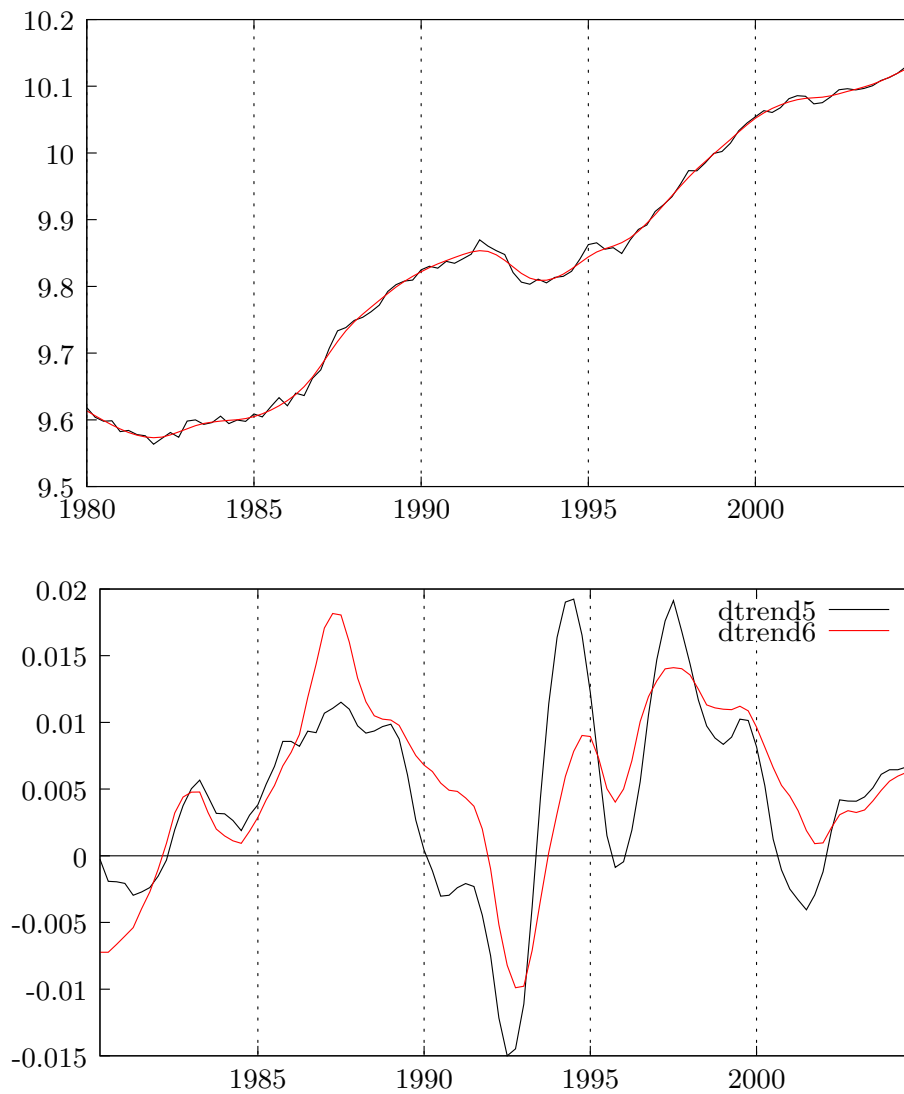
    logvab_COMP6_fechas=...
        [timefmt2(4,1980,1,size(log_vabindus,1)),log_vabindus,trend6(:,1),irreg6];
    save logvab_COMP6_fechas.txt logvab_COMP6_fechas
    dtrend6=diff(trend6(:,1));
    logvab_DT6_fechas=[timefmt2(4,1980,2,size(dtrend6,1)),dtrend6];
    save logvab_DT6_fechas.txt logvab_DT6_fechas
```

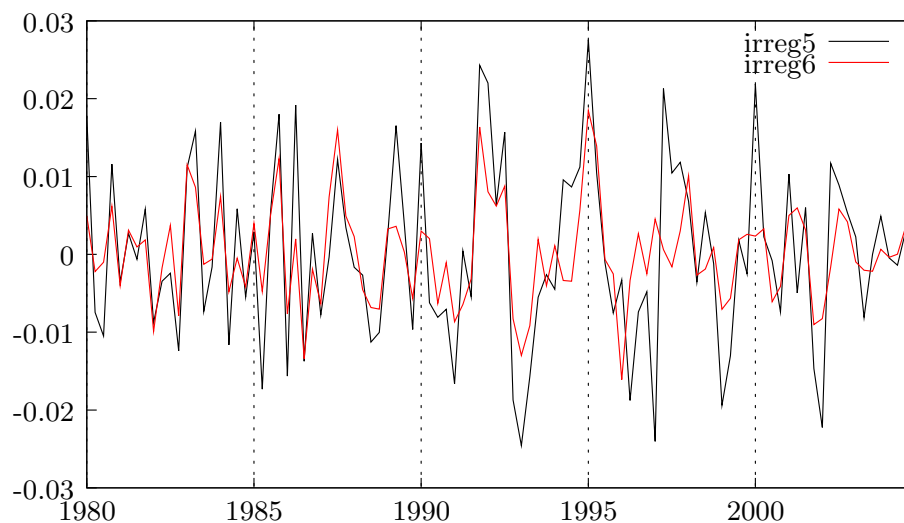
Una vez guardadas las la tendencia, su primera diferencia y el componente irregular, podemos representar la tendencia junto a la serie `log_vabindus`, así como la primera diferencia y el componente irregular conjuntamente con los obtenidos para el IPI trimestral (teniendo en cuenta que la muestra del IPI es más larga, por lo que debemos incluir sólo las fechas comunes de ambas series).

15

```
<codigoLDHR.m 1a>+≡
%plot([log_vabindus,trend6(:,1)])
%plot([irreg5(21:119), irreg6])

plot([dtrend5(21:118) dtrend6])
<14 16a>
```





como podemos observar, los ciclos son coincidentes, así como los componentes irregulares.

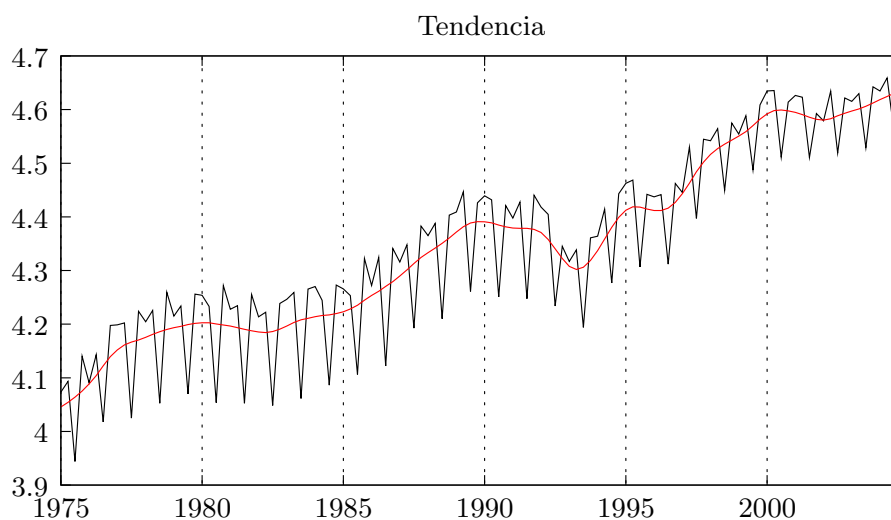
7 Repetición con identificación libre del IPI trimestral

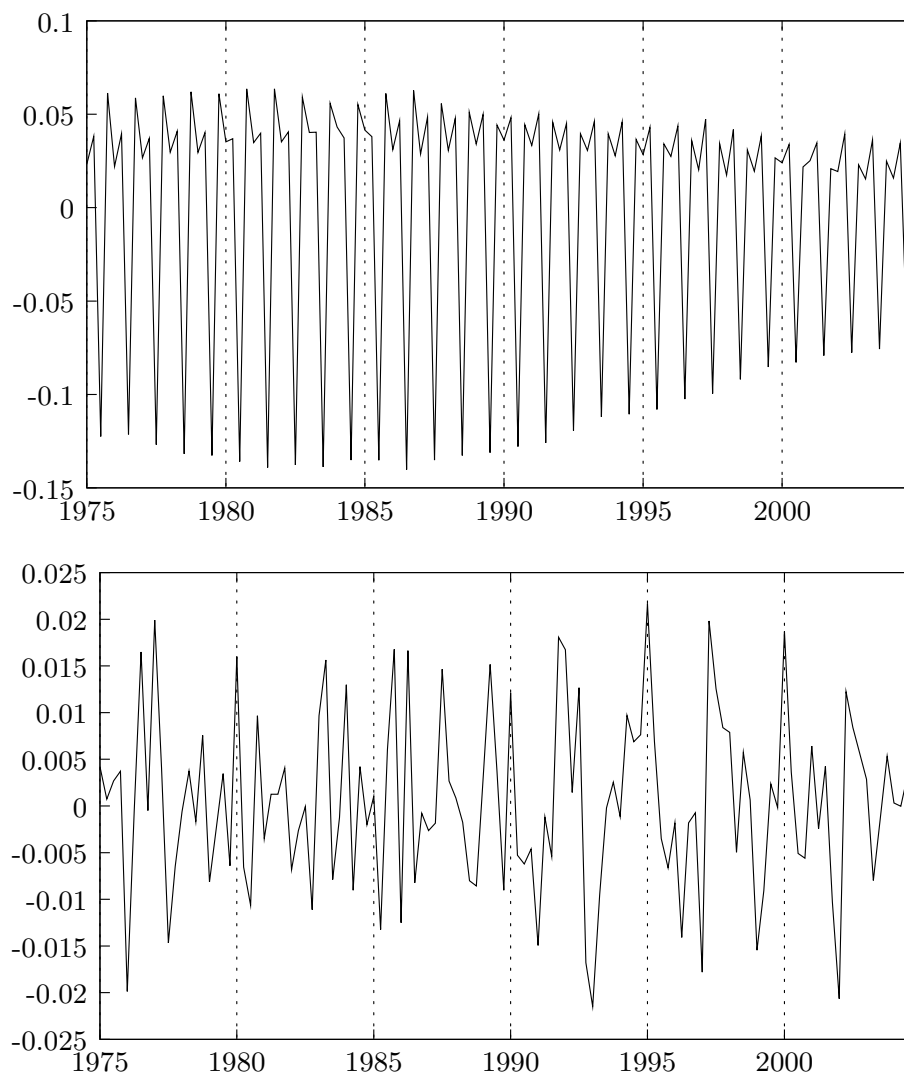
Identificamos y estimamos el modelo DHR de manera libre, es decir, permitiendo modelos SRW (fijamos un AR(25) para mayor rapidez)

```
16a <codigoLDHR.m 1a>+≡ <15 16b>
[VAR7,P7,TVP7,oar7]=autodhr(logipi_trim,4,[],25)

[trend7,season7,cycle7,irreg7]=dhrfilt(logipi_trim,P7,TVP7,VAR7,4,filt,1);
logipi_COMP7_fechas=...
[timefmt2(4,1975,1,size(logipi_trim,1)),logipi_trim,trend7(:,1),season7(:,1),irreg7];
save logipi_COMP7_fechas.txt logipi_COMP7_fechas

16b <codigoLDHR.m 1a>+≡ <16a 17>
% plot([logipi_trim,trend7(:,1)])
% plot(season7(:,1))
% plot(irreg7)
```



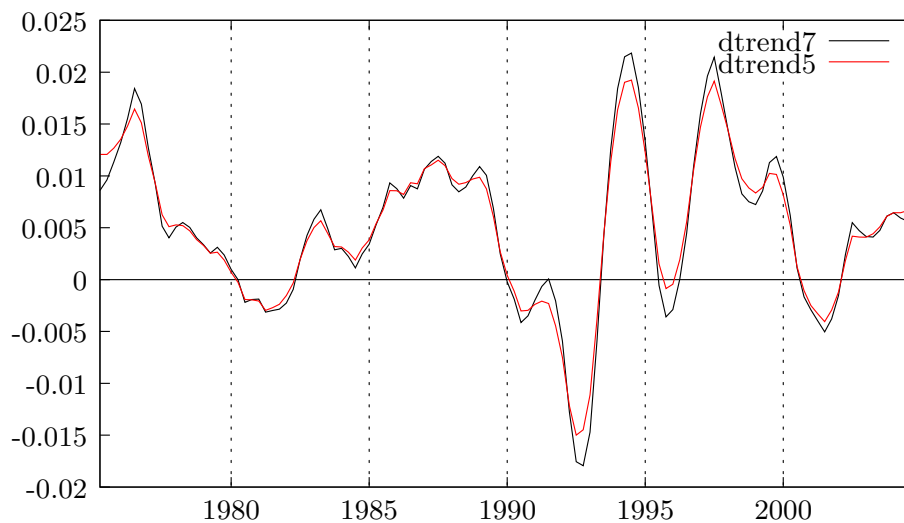


Se puede observar que la tendencia estimada sin dejar de ser “suave”, es algo más “volátil”.

```
17  <codigoLDHR.m 1a>+≡                                     <16b 18a>
    dtrend7=diff(trend7(:,1));

    logipi_DT7_fechas=[timefmt2(4,1975,2,size(dtrend7,1)),dtrend7];
    save logipi_DT7_fechas.txt logipi_DT7_fechas

    plot([dtrend5,dtrend7])
```



Una notable característica de esta tendencia es que replica bastante bien el ciclo que obtuvimos con el IPI mensual.

8 Filtrado del Valor Añadido Bruto con el nuevo modelo de tendencia

18a `<codigoLDHR.m 1a>+≡` `<17 18b>`

```
[trend8,season8,cycle8,irreg8]=dhrfilt(log_vabindus,P7(1),TVP7(:,1),VAR7(1:2),4,filt,1);

logvab_COMP8_fechas=...
    [timefmt2(4,1980,1,size(log_vabindus,1)),log_vabindus,trend8(:,1),irreg8];
save logvab_COMP8_fechas.txt logvab_COMP8_fechas
dtrend8=diff(trend8(:,1));
logvab_DT8_fechas=[timefmt2(4,1980,2,size(dtrend8,1)),dtrend8];
save logvab_DT8_fechas.txt logvab_DT8_fechas
```

Veamos los gráficos con este procedimiento alternativo:

18b `<codigoLDHR.m 1a>+≡` `<18a>`

```
% plot([vabindus,trend8(:,1)])
% plot([irreg7(21:119), irreg8])

plot([dtrend7(21:118) dtrend8])
```

