An Eviews program to perform the fractional Dickey-Fuller test

Bensalma, Ahmed

Ecole Nationale Supérieure de Statistique et d'Economie Appliquée (ENSSEA)

27 April 2021

Supplementary Material 1

for the main paper entitled:

"**Fractional Dickey-Fuller test with or without prehistorical influence**"

Ahmed BENSALMA

Department of Statistics

Laboratoire de Modélisation de Phénomènes Stochastiques (LAMOPS)
Ecole Nationale Supérieure de Statistique et d'Economie Appliquée (ENSSEA)

An Eviews program to perform the fractional Dickey-Fuller test

Ahmed Bensalma

Department of Statistics,

Laboratoire de Modélisation de Phénomènes Stochastiques (LAMOPS)

Ecole Nationale Supérieure de Statistique et d'Economie Appliquée (ENSSEA)

Pôle universitaire de Koléa, Tipaza, Algeria

bensalma.ahmed@gmail.com

Key Words: ARFIMA; Dickey-Fuller test; Fractional Dickey-Fuller test; fractional integration parameter; type II fractional Brownian motion, Fracdiff, EViews add-in.

ABSTRACT
This paper demonstrates how sequential fractional Dickey-Fuller (FDF in short) test can be implemented in EViews. We first briefly introduce how to use the fracdiff an EViews add-in to compute the fractional difference of the Nile data. Next, we give the program that executes the sequential FDF testing on the Nile data series.

1 Introduction

Let consider an $ARFIMA(0, d, 0)$ process defined by

$$y_t = (1 - L)^{-d} u_t^*, \ t = 1, 2, \cdots, n, \tag{1.1}$$

with initial conditions $y_t = 0$, if $t < 1$ and where

$$u_t^* = \begin{cases} u_t, & \text{if } t \geq 1 \\ 0, & \text{otherwise} \end{cases}$$

where u_t are independent and identically distributed (i.i.d) random variables and L is backward shift operator $Ly_t = y_{t-1}$. The fractional integration operator $(1 - L)^{-d}$ is defined by its Maclaurin series (by its binomial expansion, if d is an integer):

$$(1 - L)^{-d} = \sum_{j=0}^{\infty} \frac{\Gamma(d + j)}{\Gamma(d)\Gamma(j + 1)} L^j$$

where

$$\Gamma(z) = \begin{cases} \int_0^\infty s^{z-1} e^{-z} ds & \text{If } z > 0 \\ \infty & z = 0. \end{cases}$$

If $z < 0$, $\Gamma(z)$ is defined by the recursion formula $z\Gamma(z) = \Gamma(z + 1)$.
In EViews 9, a general procedure to compute the fractional difference of a given series $\{y_t, \ t = 1, \cdots, n\}$ is to apply the formula

$$x_t = (1 - L)^d y_t = \sum_{j=0}^{t-1} \frac{\Gamma(-d - j)}{\Gamma(-d)\Gamma(j + 1)} y_{t-j}. \tag{1.2}$$

For example, to compute the fractional difference of the demeaned Nile data with $d = 0.3$, we can use the following command lines,

1. series y=Nile-@mean(Nile)
2. y.fracdiff(d=0.3)

The second command line, simply take the difference of order 0.3 and saves the output as y_diff.

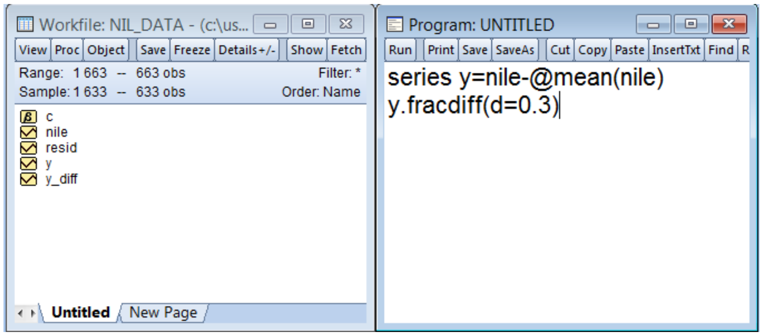

Figure 1: Compute fractional difference of the Nile series

For another naming output, we can use the third command line,

3. rename y_diff x

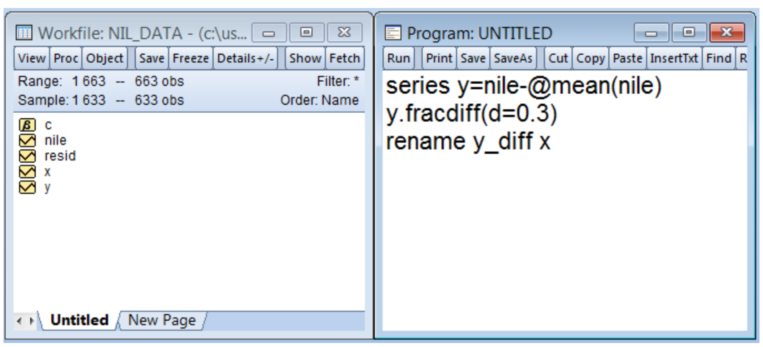

Figure 2: Compute fractional difference of the Nile series

If one want to compute many fractional difference series of the Nile data for a sequence of different values of d, for example, $d_1 = 0.1$, $d_2 = 0.2$, $d_3 = 0.3$, $\cdots$, $d_{10} = 1$ we can use the following command lines

1.	series y=nile-@mean(nile)
2.	for !i=1 to 10
3.	!d=0.1*!i
4.	y.fracdiff(d=!d)
5.	rename y_diff x!i
6.	next

The output is x_1, x_2, $\cdots$, x_{10}, where

$$x_{i,t} = (1 - L)^{0.1*i} y_t$$

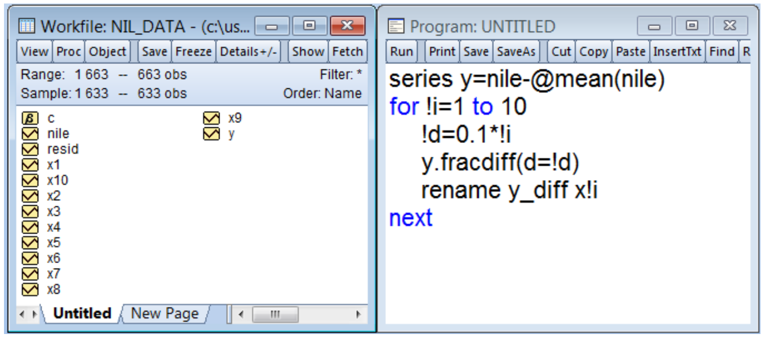

Figure 3: Compute meany different fractional difference of Nile series

2 Fractional Dickey-Fuller test

If $\{y_t,\ t = 1, \cdots, n\}$ is a sample of an $ARFIMA(0, d, 0)$ process with $d \in (-0.5, +\infty)$ then we can use the process $x_t = (1 - L)^{d_0 - 1} y_t$ to test the null hypothesis

$$H_0 : d \geq d_0, \text{ with } d_0 \in (-0.5, +\infty), \tag{1.3}$$

by using the regression model

$$(1 - L)^{d_0} y_t = \rho(1 - L)^{d_0 - 1} y_{t-1} + \varepsilon_t, \quad t = 1, 2, \cdots, n. \tag{1.4}$$

Bensalma (2018) show that the domains of limit probability density function of

$$DF_n = n\widehat{\rho} = \frac{\sum_{t=1}^{n} x_{t-1}\Delta x_t}{\sum_{t=1}^{n} (x_{t-1})^2} \quad \text{and} \quad DF_t = t_{\widehat{\rho}} = \frac{\sum_{t=1}^{n} x_{t-1}\Delta x_t}{\left(\widehat{\sigma}_{\varepsilon}^2 \sum_{t=1}^{n} (x_{t-1})^2\right)^{0.5}}$$

are

$$\begin{cases} \mathbb{R}^-, & \text{if } d < d_0, \\ \mathbb{R} & \text{if } d = d_0, \\ \mathbb{R}^+ & \text{if } d > d_0. \end{cases} \tag{1.5}$$

$x_t = (1 - L)^{d_0 - 1} y_t$ is an $I(d - d_0 + 1)$ process and then we have the following three cases

$$(d - d_0 + 1) \begin{cases} < 1 \text{ if } d < d_0, \\ = 1 \text{ if } d = d_0, \\ > 1 \text{ if } d > d_0. \end{cases}$$

The limiting distribution of DF_n and DF_t is described in the following theorem.
Theorem 1: (Bensalma (2018)) Let be a sample of an $ARFIMA(0, d, 0)$ process. If a regression model (1.4) is fitted *to a sample of size* (n) *then, as* $n \longrightarrow \infty$, *we have*

1. $DF_n \to -\infty$ and $DF_t \to -\infty$ if $d < d_0$.

2. $DF_n \Longrightarrow \frac{0.5\left[W^2(1) - 1\right]}{\int_0^1 W^2(r)dr}$ and $DF_t \Longrightarrow \frac{0.5\left[W^2(1) - 1\right]}{\left(\int_0^1 W^2(r)dr\right)^{0.5}}$ if $d = d_0$.

3. $DF_n \Longrightarrow \frac{\frac{1}{2} W_{d-d_0}^2(1)}{\int_0^1 W_{d-d_0+1}^2(r)dr}$ and $DF_t \Longrightarrow +\infty$ if $d > d_0$.

   where $W(\cdot)$ *is standard Brownian motion and* $W_d(\cdot)$ *is a type 2 fractional Brownian motion defined by*

$$W_d(r) = \frac{1}{\Gamma(d)} \int_0^r (r - s)^{d-1} dW(s), \quad r \in [0, 1] \blacksquare$$

Proof: See *Bensalma (2018, 2021).*
In practice, when the null is composite, for a given critical value of size α (i.e. $cv(\alpha)$), the probability of type I error is controlled by imposing the following constraint

$$P\left[type\ I\ error\right] = \underset{d\geq d_0}{Sup} P\left[DF_t < cv(\alpha)\right] \leq \alpha.$$

Given that the limit of probability density of DF_t have a remarkable arrangement (1.5) it is easy to show that

$$\underset{d\geq d_0}{Sup} P\left[DF_t < cv(\alpha)\right] = P_{d=d_0}\left[DF_t < cv(\alpha)\right]$$

This later result combined with the second result of theorem above, which show that DF_t have the Dickey-Fuller limit distribution when $d = d_0$, demonstrate that to perform the fractional Dickey-Fuller test we can use the usual tabulated value of the standard Dickey-Fuller test. Our testing procedure will not enable us to apprehend the case of $H_0 : d \geq 0.5$, but also the general case of $H_0 : d \geq d_0$, with $d_0 \in (-0.5, +\infty)$. Moreover, if we use upward or downward testing sequence for a set of values $d_0^1 < d_0^2 < \cdots < d_0^l$, it is possible to determine an overlap domain of the parameter d.

3 Empirical application with Eviews

In this section, we consider the well-known series of annual minima of the Nile, as studied by Hurst (1951) and reproduced in Beran (1994). The sample size is $n = 633$ annual observations $(622 - 1284\ AD)$

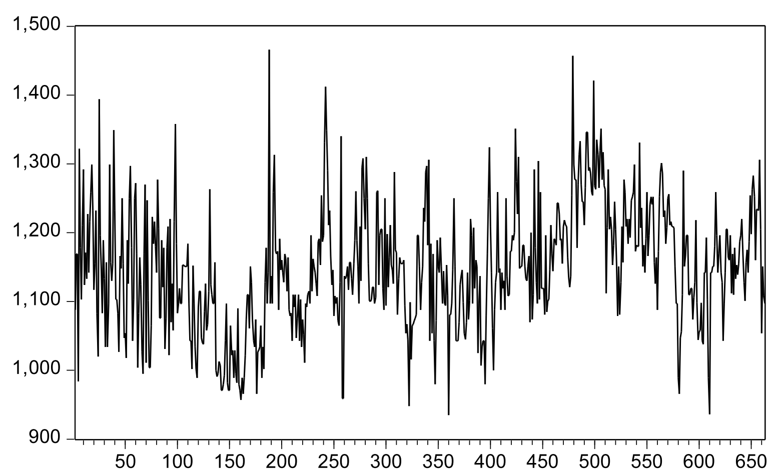

Figure 4: Nile series

In the following, we apply the sequential F-DF test to the demeaned Nile series, namely

$$y_t = Nile_t - \overline{Nile}$$

where $\overline{Nile} = \frac{\sum_{t=1}^{633} Nile_t}{633}$. We use upward testing sequence for a set of values

$$d_0^i \in \{0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,\ 1\}.$$

For a particular value d_0^i from this later set, we test the hypothesis

$$H_{0i} : d \geq d_0^i$$

by using the $t_{\widehat{\rho}_i} = DF_t$ calculated via the estimation of the following autoregression model

$$(1-L)^{d_0^i} y_t = (1-L)^{d_0^i - 1} y_{t-1} + \varepsilon_t.$$

The Table 1 summarize the sequential upward testing procedure of the standard FDF test on the y_t series.

d_0^i	FDF test on	$t_{\widehat{\rho}_i} = DF_t$	Dickey-Fuller $cv(5\%)$	Reject or Accept $H_{0i} : d \geq d_0^i$
0	$\Delta^{-1} y_t$	-0.1986	-1.94	accept
0.1	$\Delta^{-0.9} y_t$	-0.2746	-1.94	accept
0.2	$\Delta^{-0.8} y_t$	-0.4201	-1.94	accept
0.3	$\Delta^{-0.7} y_t$	-0.7148	-1.94	accept
0.4	$\boldsymbol{\Delta}^{-0.6} \mathbf{y}_t$	$\mathbf{-1.2174}$	$\mathbf{-1.94}$	**accept**
0.5	$\boldsymbol{\Delta}^{-0.5} \mathbf{y}_t$	$\mathbf{-2.0029}$	$\mathbf{-1.94}$	**reject**
0.6	$\Delta^{-0.4} y_t$	-3.2481	-1.94	reject
0.7	$\Delta^{-0.3} y_t$	-4.9185	-1.94	reject
0.8	$\Delta^{-0.2} y_t$	-7.1955	-1.94	reject
0.9	$\Delta^{-0.1} y_t$	-10.0671	-1.94	reject
1	x_t	-13.3303	-1.94	reject
Conclusion	$0.4 \leq d < 0.5$			

The table 1 show that we can apply a downward testing sequence, in this case we take the largest value, (the maximum value of d_0^i, i.e. $d \geq 1$), under consideration as the first maintained hypothesis and then decrease the order of differenced each time the current null is rejected. The table 1 show, also, that an upward testing sequence can be applied. In this case, we take the smallest value of d_0^i, (i.e. $d \geq 0$) under consideration as the first maintained hypothesis and then increase the order of differenced each time the current alternative is accepted. Table 1 show that the lower and upper bound of fractional integration order of the demeaned Nile data are 0.4 and 0.5 respectively.
The program that executes this sequential testing procedure is shown in Appendix. The first 30 lines constitute the main program. The rest of the program consists of an understandable formatting of the results in a table. The figure 5, show how the results are displayed in EViews after the execution of the EViews sequential FDF program

```
**************************************************
Sequential fractional Dickey-Fuller test with EViews
**************************************************

series y=nile-@mean(nile)
vector (10) Accept_Reject_H0
'_____________________________
' Set of sequential values of d0
'_____________________________
for !i=1 to 10
!d0=0.1*!i
!d=-1+0.1*!i
'_____________________________
'Compute x(t) = (1 - L)^(d0-1) y(t)
'_____________________________
y.fracdiff(d=!d, )
rename y_diff x!i
'_______________________________________________________________________
'Testing the null H0 : d >= d0 by means the t-stat of c(1)
'coefficient in the model (1 - L)^d0 y_t = c(1) * (1 - L)^(d0-1) y_(t-1) + e_t
'_______________________________________________________________________
equation eq!i.ls d(x!i) x!i(-1)
if eq!i.@tstat(1)>-1.94 then Accept_Reject_H0(!i)=1 else Accept_Reject_H0(!i)=0
endif
delete x!i
next
'_____________________________________________
'Find the lower and upper bound of d
'_____________________________________________
for !i= 1 to 9
if Accept_Reject_H0(!i)=1 and Accept_Reject_H0(!i+1)=0 then
scalar Lower_bound_of_d=0.1*!i
scalar Upper_bound_of_d=0.1*(!i+1)
endif
next
'______________________
' display results in table
'______________________
table tab1
setcolwidth(tab1,1,20)
tab1(1,1)="Table 1"
tab1(2,1)="Sequential testing procedure,"
tab1(3,1)="of the standard Dickey-Fuller test"
tab1(4,1)="applied to the nile series: H0 : d >= d0,i"
tab1(4,2)=" "
tab1(4,3)=" "
tab1(4,4)=" "
tab1(4,5)=" "
setline(tab1,5)
```

```
tab1(6,1)="d0,i"
tab1(6,2)="d0,i-1"
tab1(6,3)="DFt"
tab1(6,4)="DF cv(5%)" 'critical value of the Dickey-Fuller distribution
tab1(6,5)="Accept_Reject_H0"
setline(tab1,7)
tab1(8,1)="0.1"
tab1(8,2)="0.9"
tab1(8,3)=eq1.@tstat(1)
tab1(8,4)="-1.94"
tab1(8,5)=Accept_Reject_H0(1)
setline(tab1,9)
tab1(10,1)="0.2"
tab1(10,2)="0.8"
tab1(10,3)=eq2.@tstat(1)
tab1(10,4)="-1.94"
tab1(10,5)=Accept_Reject_H0(2)
setline(tab1,11)
tab1(12,1)="0.3"
tab1(12,2)="0.7"
tab1(12,3)=eq3.@tstat(1)
tab1(12,4)="-1.94"
tab1(12,5)=Accept_Reject_H0(3)
setline(tab1,13)
tab1(14,1)="0.4"
tab1(14,2)="0.6"
tab1(14,3)=eq4.@tstat(1)
tab1(14,4)="-1.94"
tab1(14,5)=Accept_Reject_H0(4)
setline(tab1,15)
tab1(16,1)="0.5"
tab1(16,2)="0.5"
tab1(16,3)=eq5.@tstat(1)
tab1(16,4)="-1.94"
tab1(16,5)=Accept_Reject_H0(5)
setline(tab1,17)
tab1(18,1)="0.6"
tab1(18,2)="0.4"
tab1(18,3)=eq6.@tstat(1)
tab1(18,4)="-1.94"
tab1(18,5)=Accept_Reject_H0(6)
setline(tab1,19)
tab1(20,1)="0.7"
tab1(20,2)="0.3"
tab1(20,3)=eq7.@tstat(1)
tab1(20,4)="-1.94"
tab1(20,5)=Accept_Reject_H0(7)
setline(tab1,21)
tab1(22,1)="0.8"
```

```
tab1(22,2)="0.2"
tab1(22,3)=eq8.@tstat(1)
tab1(22,4)="-1.94"
tab1(22,5)=Accept_Reject_H0(8)
setline(tab1,23)
tab1(24,1)="0.9"
tab1(24,2)="0.1"
tab1(24,3)=eq9.@tstat(1)
tab1(24,4)="-1.94"
tab1(24,5)=Accept_Reject_H0(9)
setline(tab1,25)
tab1(26,1)="1"
tab1(26,2)="0"
tab1(26,3)=eq10.@tstat(1)
tab1(26,4)="-1.94"
tab1(26,5)=Accept_Reject_H0(10)
setline(tab1,27)
tab1(28,1)="Lower bound of d="+@str(Lower_bound_of_d)
tab1(29,1)="Upper bound of d="+@str(Upper_bound_of_d)
tab1(30,1)=@str(Lower_bound_of_d)+"=< d <"+@str(Upper_bound_of_d)
show tab1
```

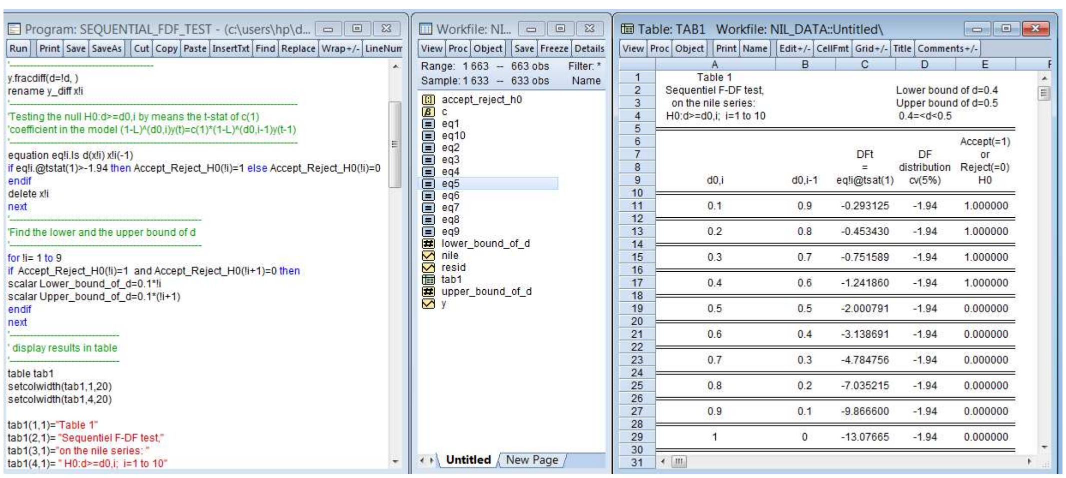

Figure 5: Output of the execution of the Eviews sequential FDF program

Table 1 Sequentiel F-DF test, on the nile series: H0:d>=d0,i; i=1 to 10			Lower bound of d=0.4 Upper bound of d=0.5 0.4=<d<0.5	
d0,i	d0,i-1	DFt = eq!i@tsat(1)	DF distribution cv(5%)	Accept or Reject H0
0.1	0.9	−0.293125	-1.94	1.000000
0.2	0.8	−0.453430	-1.94	1.000000
0.3	0.7	−0.751589	-1.94	1.000000
0.4	0.6	−1.241860	-1.94	1.000000
0.5	0.5	−2.000791	-1.94	0.000000
0.6	0.4	−3.138691	-1.94	0.000000
0.7	0.3	−4.784756	-1.94	0.000000
0.8	0.2	−7.035215	-1.94	0.000000
0.9	0.1	−9.866600	-1.94	0.000000
1	0	−13.07665	-1.94	0.000000

Table 1: Output of the execution of the Eviews sequential FDF program

References

[1] Bensalma, A. (2016) 'A consistent test for unit root against fractional alternative', International Journal of Operational Research, Vol. 27, Nos. 1/2, pp.252–274.

[2] Bensalma, A. (2018) 'Testing the fractional integration parameter revisited: a fractional Dickey-Fuller test', International Journal of Mathematics in Operational Research, Vol. 12, No. 4, pp. 471-506.

[3] Bensalma, A.(2021) 'Fractional Dickey Fuller test with or without prehitorical influence' hal-0307575 (v1)

[4] Bensalma, A.(2021) 'Fractional Dickey Fuller test with or without prehitorical influence' MPRA Paper No. 107408