Basic tseries

Contents

- Clear workspace
- The tseries object
- Manipulating IRIS tseries
- Panel tseries
- Exploring tseries properties
- Functions operating on tseries

Clear workspace

StartLineNumber=4;

```
close all;
clear all;
```

The tseries object

IRIS stores timeseries data in the tseries object which is taylored for macroeconomic timseries.

```
a = tseries(qq(2000,1):qq(2001,4),@randn); a
b = tseries(mm(2000,1):mm(2001,12),@randn); b
c = tseries(yy(2000):yy(2005),@randn);
```

a The tseries function takes two arguements:

1. A sample period, in IRIS date format
2. Numeric values to be associated with the corresponding dates

b The mm, qq, yy functions create monthly, quarterly and annual dates in IRIS date format, respectively. The arguments are a 4 digit year and the desired period, eg. Jan 2000 is mm(2000,1), whereas 2010Q3 is qq(2010,3) and the year 1991 is simply yy(1991).

Manipulating IRIS tseries

Once you have created an IRIS tseries you can easily reference a sub-range, or subset a sub-range

```
aa = a{qq(2000,2):qq(2000,4)} c
bb = a(qq(2000,2):qq(2000,4))

aa = a;
aa(qq(2001,1):qq(2001,2)) = [1 2]

bb = a;
bb(qq(2001,1):qq(2001,2)) = 3 d
```

c When referencing a sub-range of a `tseries`, use {} to return a `tseries` object and () to return a plain vanilla vector

d When overwriting a sub-range of a `tseries`, always use (). On the right-hand side, either provide a vector of the approriate length, or a single scalar, which will be repeated over the given range.

```
aa =

       tseries object: 3-by-1

       2000Q2: 0.84055
       2000Q3: -0.40861
       2000Q4: -0.95864
   {''}

       user data: empty


bb =

   0.8406
  -0.4086
  -0.9586


aa =

       tseries object: 8-by-1

       2000Q1: 0.18402
       2000Q2: 0.84055
       2000Q3: -0.40861
       2000Q4: -0.95864
       2001Q1:       1
       2001Q2:       2
       2001Q3:  -1.043
       2001Q4: -1.6052
   {''}

       user data: empty


bb =

       tseries object: 8-by-1

       2000Q1: 0.18402
       2000Q2: 0.84055
       2000Q3: -0.40861
```

```
    2000Q4: -0.95864
    2001Q1:       3
    2001Q2:       3
    2001Q3:  -1.043
    2001Q4: -1.6052
{''}

    user data: empty
```

Panel tseries

You can concatenate `tseries` of the same frequency horizontaly to create panel `tseries` objects

```
a = tseries(qq(1991,1):qq(1992,3),ones(7,1));
b = tseries(qq(1991,3):qq(1992,1),@rand);
c = [a b]
```

e Notice that IRIS takes care of appending the approriate number of `NaN` when concatenating `tseires`.

```
c =

    tseries object: 7-by-2

    1991Q1: 1       NaN
    1991Q2: 1       NaN
    1991Q3: 1    0.16724
    1991Q4: 1    0.67966
    1992Q1: 1   0.079569
    1992Q2: 1       NaN
    1992Q3: 1       NaN
''    ''

    user data: empty
```

Exploring tseries properties

Use the `get` function to explore your `tseries`. You can get the complete list of queries by typing

```
help tseries.get
```

```
get(a,'start')
get(a,'freq')
x = get(c,'nanrange')
dat2char(x)
dat2str(x)
```

f Use the `dat2char` or `dat2str` functions to translate dates from IRIS date format to human readable form.

```
ans =

  7.9640e+003


ans =

     4


x =

  1.0e+003 *

    7.9660   7.9670   7.9680


ans =

1991Q3
1991Q4
1992Q1


ans =

    '1991Q3'   '1991Q4'   '1992Q1'
```

Functions operating on `tseries`

You can use most mathematical functions with `tseires` objects intuitively and you will get the expected answer

```
a+b
log(b)
a*b
a.*b g
a+c h
b*c

bb = pct(b,2) i
cc = diff(b,2) j

b = tseries(mm(2000,1):mm(2002,12),@randn);
bb = convert(b,'q') k
cc = convert(b,'y')
dd = convert(cc,'q')
```

g Multiplying two one-dimentional `tseries` produces the element-by-element product

h When working with panel `tseries`, $a + c$ is equivalent to $[a + c\{:,1\}, a + c\{:,2\}]$ and $b * c$ returns $[b. * c\{:,1\}, b. * c\{:,2\}]$

i `pct(x,k)` calculates percent change of `x` relative to k periods ago

j `diff(x,k)` calcualtes simple difference of `x` relative to k periods ago

k `convert`, as the name suggests, converts IRIS `tseries` from one frequency to another. The default behavior is to use the mean for high-to-low frequency conversion and the cubic spline for low-to-high frequency conversion. Type

```
help tseries/convert
```

to learn more about the IRIS `convert` function.

```
ans =

       tseries object: 3-by-1

       1991Q3: 1.1672
       1991Q4: 1.6797
       1992Q1: 1.0796
   {''}

       user data: empty


ans =

       tseries object: 3-by-1

       1991Q3: -1.7883
       1991Q4: -0.38616
       1992Q1: -2.5311
   {''}

       user data: empty


ans =

       tseries object: 3-by-1

       1991Q3: 0.16724
       1991Q4: 0.67966
       1992Q1: 0.079569
   {''}
```

```
       user data: empty


ans =

       tseries object: 3-by-1

       1991Q3: 0.16724
       1991Q4: 0.67966
       1992Q1: 0.079569
   {''}

       user data: empty


ans =

       tseries object: 7-by-2

       1991Q1: 2       NaN
       1991Q2: 2       NaN
       1991Q3: 2    1.1672
       1991Q4: 2    1.6797
       1992Q1: 2    1.0796
       1992Q2: 2       NaN
       1992Q3: 2       NaN
   ''   ''

       user data: empty


ans =

       tseries object: 3-by-2

       1991Q3: 0.16724   0.027968
       1991Q4: 0.67966    0.46194
       1992Q1: 0.079569 0.0063312
   ''   ''

       user data: empty


bb =

       tseries object: 1-by-1

       1991Q3: 110.1799
   {''}
```

```
        user data: empty


cc =

        tseries object: 1-by-1

        1991Q3: 0.087669
    {''}

        user data: empty


bb =

        tseries object: 12-by-1

        2000Q1:  0.28817
        2000Q2: -0.48813
        2000Q3: -0.43719
        2000Q4:   0.2396
        2001Q1:  -1.0988
        2001Q2:    0.826
        2001Q3: -0.019939
        2001Q4:  0.59868
        2002Q1: -0.59312
        2002Q2: -0.45562
        2002Q3:  0.55241
        2002Q4:  0.17218
    {''}

        user data: empty


cc =

        tseries object: 3-by-1

        2000Y: -0.099387
        2001Y: 0.076482
        2002Y: -0.081039
    {''}

        user data: empty


dd =
```

```
    tseries object: 12-by-1

    2000Q1: -0.24952
    2000Q2: -0.14465
    2000Q3: -0.059046
    2000Q4: 0.0064393
    2001Q1: 0.050944
    2001Q2: 0.073608
    2001Q3: 0.074146
    2001Q4: 0.055137
    2002Q1: 0.016294
    2002Q2: -0.043242
    2002Q3: -0.12433
    2002Q4: -0.22783
{''}

    user data: empty
```