

## Example: Returns Calculations

Let us look at the seemingly simple task of calculating *returns*

$$R_{it} = \frac{R_{i,t} - R_{i,t-1}}{R_{i,t-1}}$$

And use a very simple example to illustrate

$$\mathbf{P} = \{1, 2, 4\}$$

The returns here are (This is a simple calculation)

$$\mathbf{R} = \{1, 1\}$$

Investigate this simple example in  $R$

(Always good to check out in the context of a very simple example that you get what you think you get. )

Check that we get this using various methods for calculation  
Let us first use the time series object `ts` in R:

```
> prices <- ts(c(1,2,4),frequency=1)
> returns <- diff(prices)/prices[-length(prices)]
> returns
Time Series:
Start = 2
End = 3
Frequency = 1
[1] 1 1
```

This works because we are dividing the differences by all but the last element of the price vector.

But does it work when we use the zoo library?

```
> library(zoo)
> dates <- c(as.Date("2001-12-31"),as.Date("2002-12-31"),as.Date("2003-12-31"))
> prices <- zoo(c(1,2,4),order.by=dates)
> prices
2001-12-31 2002-12-31 2003-12-31
           1           2           4
> returns <- diff(prices)/prices[-length(prices)]
> returns
2002-12-31
           0.5
```

Why does this not work?

Check what we are actually dividing here

```
> print(diff(prices))
2002-12-31 2003-12-31
           1         2
> print(prices[-length(prices)])
2001-12-31 2002-12-31
           1         2
```

So, zoo silently matches date when doing the above calculation, which is not what was desired.

If we want to keep using zoo, but still require returns, one way to fix this is to remove the dates from the divisor:

```
> returns <- diff(prices)/as.vector(prices[-length(prices)])  
> returns  
2002-12-31 2003-12-31  
          1          1
```

Let us also check what xts does:

```
> prices <- xts(c(1,2,4),order.by=dates)
> prices
           [,1]
2001-12-31    1
2002-12-31    2
2003-12-31    4
> returns <- diff(prices)/prices[-length(prices)]
> returns
           e1
2001-12-31  NA
2002-12-31  0.5
```

which is also very wrong.

```
> print(diff(prices))
      [,1]
2001-12-31  NA
2002-12-31   1
2003-12-31   2
> print(prices[-length(prices)])
      [,1]
2001-12-31   1
2002-12-31   2
```

Here, the trick of replacing the prices does not work either

```
> returns <- diff(prices)/as.vector(prices[-length(prices)])
```

Warning message:

```
In ‘/.default‘(diff(prices), as.vector(prices[-length(prices)])):  
longer object length is not a multiple of shorter object length
```

```
> returns
```

```
      [,1]
```

```
2001-12-31    NA
```

```
2002-12-31    0.5
```

```
2003-12-31    2.0
```



But we can get the correct answer as follows

```
> returns <- na.omit(diff(prices))/coredata(prices)[-length(prices)]
> returns
      [,1]
2002-12-31    1
2003-12-31    1
```

Another alternative involves the quantmod package

```
> library(quantmod)
> returns <- na.omit(annualReturn(prices,leading=FALSE))
> returns
```

	yearly.returns
2002-12-31	1
2003-12-31	1

Well, it seems like getting returns are actually less trivial than one should think