Workshop: High-performance computing for economists

Lars Vilhuber[1]   John M. Abowd[1]   Richard Mansfield[1]
Kevin L. McKinney

[1]Cornell University, Economics Department,

August 20-22, 2013: Day 1

Basic subroutine programming

Goal

- ▶ Show the basics of proper subroutine programming
- ▶ Advantages, pitfalls
- ▶ Examples in R
- ▶ Tomorrow: generalization and differences in other programming languages

Control structures in programming languages

Mostly generic

- ▶ `if, else`: testing a condition [R, SAS]
- ▶ `for`: execute a loop a fixed number of times [R, in SAS: `do`]
- ▶ `while`: execute a loop while a condition is true [R,SAS]
- ▶ `until`: execute a loop until a condition is true [SAS]
- ▶ `repeat`: execute an infinite loop [R]
- ▶ `break`: break the execution of a loop [R, SAS]
- ▶ `next`: skip an interation of a loop [R]
- ▶ `return`: exit a function [R]

Control structures: if

Control structures: if

... in R

```
if(<condition>) {
## do something
} else {
## do something else
}
if(<condition1>) {
## do something
} else if(<condition2>) {
## do something different
} else {
## do something different
}
```

Control structures: if

... in R

```
if(<condition>) {
## do something
} else {
## do something else
}
if(<condition1>) {
## do something
} else if(<condition2>) {
## do something different
} else {
## do something different
}
```

... in SAS

```
if (<condition>) then do;
## do something
end; else do;
## do something else
end;
if (<condition1>) then do
## do something
else if (<condition2>) th
## do something different
end; else do;
## do something different
end;
```

Control structures: for

Run through a fixed sequence of numbers (or in R, a sequence of vectors)

Control structures: for

Run through a fixed sequence of numbers (or in R, a sequence of vectors)

simple loop in R

```
for(i in 1:10) {
print(i)
}
```

Control structures: for

Run through a fixed sequence of numbers (or in R, a sequence of vectors)

simple loop in R

```
for(i in 1:10) {
print(i)
}
```

... in SAS

```
do i = 1 to 10;
put i;
end;
```

Control structures: for

Across programming languages, some flexibility:

Control structures: for

Across programming languages, some flexibility:

Equivalent loops in R

... in SAS

```
x <- c("a", "b", "c", "d")
for(i in 1:4) {
print(x[i])
}
for(i in x) {
print(i)
}
for(i in 1:4) print(x[i])
```

```
do i = 1 to 10;
put i;
end;
```