

Introductory Programming in R

By Asef Nazari

asef.nazari@monash.edu

Faculty of IT

Monash university

6. Simulation

6.1 Generating Random Numbers

Here are functions for probability distribution in R. They help us simulate variables from given probability distributions.

- `rnorm`: generates random normal variables
- `pnorm`: evaluate the cumulative distribution of Normal distribution
- `dnorm`: evaluates normal probability density
- `qnorm`: quantiles

For each probability density function, there are four functions related to them:

- `d` for density
- `r` for random number generator
- `p` for cumulative distribution
- `q` for quantile function

Examples:

- `dnorm(x, mean=0, sd=1, log=FALSE)`
- `pnorm(q, mean=0, sd=1, lower.tail=TRUE, log.p=FALSE)`
- `dnorm(p, mean=0, sd=1, lower.tail=TRUE, log.p=FALSE)`
- `dnorm(n, mean=0, sd=1)`

If F is the cumulative distribution function for a standard normal distribution, then $\text{pnorm}(q) = F(q)$ and $\text{qnorm}(p) = F^{-1}(p)$

In [14]:

```
#Simulation
# rnorm, dnorm, pnorm,
x <- rnorm(10)
x
```

```
0.133336360814841  0.804189509744908  -0.0571067743838088
0.503607972233726  1.08576936214569  -0.69095383969683
-1.28459935387219  0.046726172188352  -0.235706556439501
-0.542888255010254
```

In [15]:

```
x <- rnorm(10,20,2)
x
```

```
19.1333793650864 18.7010567064075 21.4535014947709 22.3038235081744
21.9843207308916 19.1409737810162 22.4766082017068 19.4413074362915
23.5158061796214 21.1214921817761
```

In [16]:

```
summary(x)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
18.70	19.22	21.29	20.93	22.22	23.52

In [17]:

```
set.seed(1)
rnorm(5)
rnorm(5)
set.seed(1)
rnorm(5)
rnorm(5)
```

```
-0.626453810742332 0.183643324222082 -0.835628612410047
1.59528080213779 0.329507771815361

-0.820468384118015 0.487429052428485 0.738324705129217
0.575781351653492 -0.305388387156356

-0.626453810742332 0.183643324222082 -0.835628612410047
1.59528080213779 0.329507771815361

-0.820468384118015 0.487429052428485 0.738324705129217
0.575781351653492 -0.305388387156356
```

In [18]:

```
ppois(2,2) ##cumulative distribution
##Pr(x<=2)
ppois(4,2) ##Pr(x<=4)
```

```
0.676676416183063
```

```
0.947346982656289
```

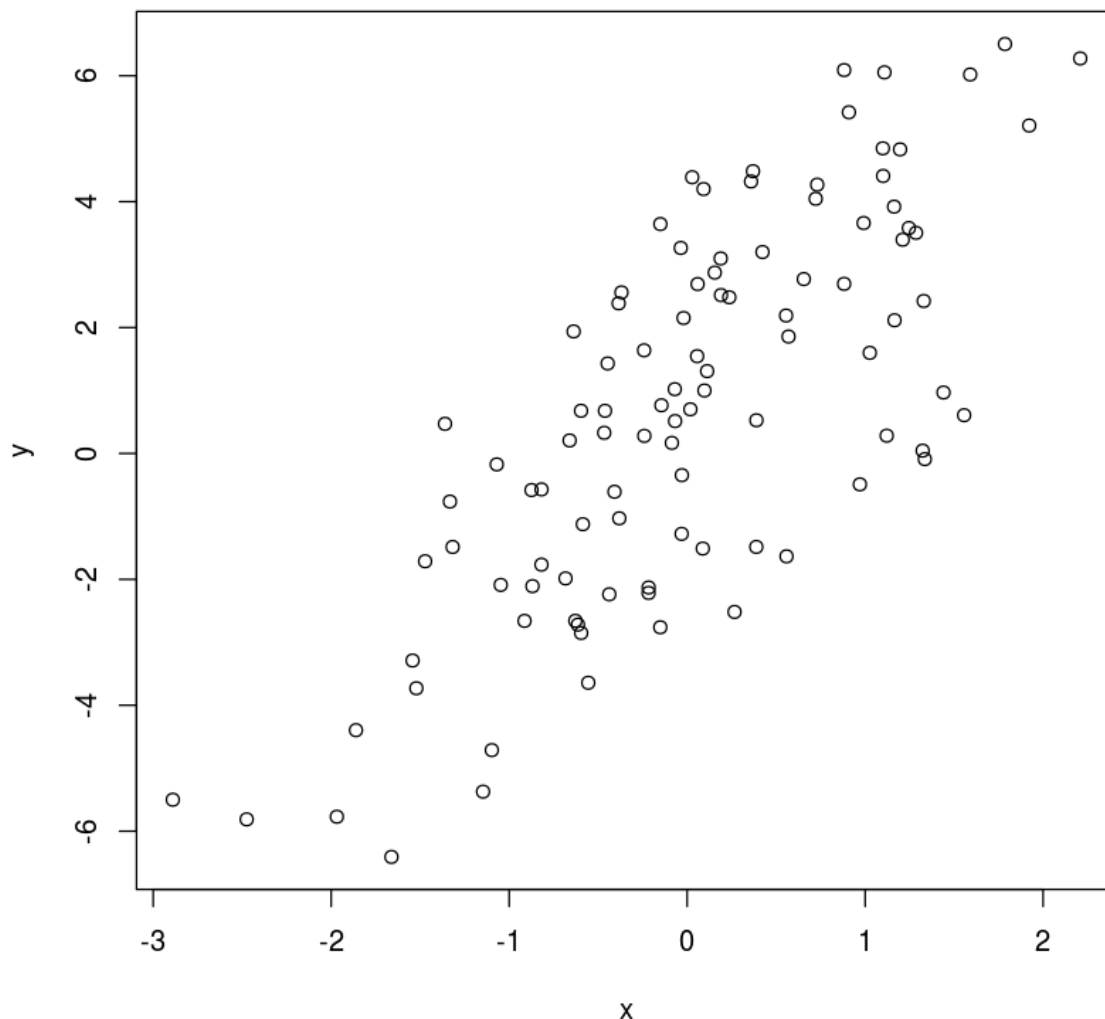
In [19]:

```
set.seed(20)
x <- rnorm(100)
e <- rnorm(100,0,2)
y <- 0.5+2*x+e
summary(y)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-6.4080	-1.5400	0.6789	0.6893	2.9300	6.5050

In [20]:

```
plot(x,y)
```



6.2 Random Sampling

The `sample()` function draws randomly from a specified set of (scalar) objects allowing you to sample from arbitrary distributions.

Summary:

- Drawing samples from specific probability distribution can be done with `r-` function
- Standard distributions are Normal, Poisson, Binomial, Exponential, Gamma, etc.
- the `sample()` function can be used to draw random samples from arbitrary vectors
- Setting the random number generator via `set.seed()` is critical for reproducibility.

In [21]:

```
set.seed(1)
sample(1:10, 4) # without replacement
```

3 4 5 7

In [22]:

```
sample(1:10,4)
```

```
3 9 8 5
```

In [23]:

```
sample(letters, 5)
```

```
'q' 'b' 'e' 'x' 'p'
```

In [24]:

```
sample(1:10) #permutation
```

```
4 7 10 6 9 2 8 3 1 5
```

In [25]:

```
sample(1:10)
```

```
2 3 4 1 9 5 10 8 6 7
```

In [26]:

```
sample(1:10, replace=TRUE) #sample with replacement
```

```
2 9 7 8 2 8 5 9 7 8
```

In []: