Laborator 10

Petculescu Mihai-Silviu

Repartiţii Clasice

```
# d<nume>
calculeaza densitatea atunci cand vorbim de o variabila continua sau functia de
masa, atunci cand avem o repartitie discreta (P(X = k))

# p<nume>
calculeaza functia de repartitie, i.e. F(x) = P(X <= x)

# q<nume>
reprezinta functia cuantila, cu alte cuvinte valoarea pentru care functia de
repartitie are o anumita probabilitate; in cazul continuu, daca p<nume>(x) = p
atunci q<nume>(p) = x iar in cazul discret intoarce cel mai mic intreg u pentru
care P(X <= u) >= p

# r<nume>
genereaza observatii independente din repartitia data
```

Repartiţia Uniformă

```
> runif(10, 3, 5)
 [1] 3.785430 3.093727 3.161389 4.853334 4.531957 3.745534 4.951405 4.227713
 [9] 4.775125 3.095654

> dunif(c(3.1, 3.7, 3.95, 4.86), 3, 5)
 [1] 0.5 0.5 0.5 0.5

> punif(c(3.1, 3.7, 3.95, 4.86), 3, 5)
 [1] 0.050 0.350 0.475 0.930
```

Repartiţia Normală

```
> rnorm(10, mean = 0, sd = sqrt(2))
 [1]  1.37201493  0.16832573 -1.81833870 -1.63850658 -1.09559909 -0.01664415
 [7] -1.88757810 -0.85869165  0.90567103 -1.28345503

> dnorm(seq(-2, 2, length.out = 15), mean = 3, sd = 5)
 [1] 0.04839414 0.05115647 0.05390019 0.05660592 0.05925368 0.06182308
 [7] 0.06429362 0.06664492 0.06885700 0.07091058 0.07278734 0.07447021
[13] 0.07594361 0.07719368 0.07820854

> pnorm(seq(-1, 1, length.out = 15), mean = 3, sd = 1)
 [1] 3.167124e-05 5.736006e-05 1.018892e-04 1.775197e-04 3.033834e-04
 [6] 5.086207e-04 8.365374e-04 1.349898e-03 2.137367e-03 3.320943e-03
[11] 5.063995e-03 7.579219e-03 1.113549e-02 1.606229e-02 2.275013e-02

> qnorm(c(0.01, 0.025, 0.05, 0.25, 0.5, 0.75, 0.95, 0.975, 0.99), mean = 0, sd =
1)
[1] -2.3263479 -1.9599640 -1.6448536 -0.6744898  0.0000000  0.6744898  1.6448536
[8]  1.9599640  2.3263479
```

Repartiţia Exponenţială

```
> rexp(15, rate = 5)
 [1] 0.09883625 0.16874284 0.07949068 0.20611032 0.16603631 0.01965043
 [7] 0.10250783 0.04407111 0.06351975 0.08510264 0.04419076 0.06358394
[13] 0.21125171 0.86614666 0.55913708

> dexp(seq(0, 5, length.out = 20), rate = 5)
 [1] 5.000000e+00 1.341312e+00 3.598237e-01 9.652719e-02 2.589462e-02
 [6] 6.946555e-03 1.863500e-03 4.999070e-04 1.341063e-04 3.597568e-05
[11] 9.650925e-06 2.588981e-06 6.945263e-07 1.863153e-07 4.998141e-08
[16] 1.340814e-08 3.596899e-09 9.649130e-10 2.588499e-10 6.943972e-11

> pexp(seq(0, 5, length.out = 15), rate = 5)
 [1] 0.0000000 0.8323228 0.9718843 0.9952856 0.9992095 0.9998675 0.9999778
 [8] 0.9999963 0.9999994 0.9999999 1.0000000 1.0000000 1.0000000 1.0000000
[15] 1.0000000

> qexp(c(0.01, 0.025, 0.05, 0.25, 0.5, 0.75, 0.95, 0.975, 0.99), rate = 5)
[1] 0.002010067 0.005063562 0.010258659 0.057536414 0.138629436 0.277258872
[7] 0.599146455 0.737775891 0.921034037
```

Repartiţia Cauchy

```
> rcauchy(15, location = 0, scale = 2)
 [1] -1.80489619  0.95442004 13.06165753  0.11998785 -0.05515141  0.09601801
 [7] -2.03789327  0.73434825 -1.65880111  6.54091969 15.22332612  5.29126741
[13]  2.42290833  0.65076813 -0.87890203

> dcauchy(seq(-5, 5, length.out = 20), location = 1, scale = 3)
 [1] 0.02122066 0.02450975 0.02852541 0.03345265 0.03951056 0.04693392
 [7] 0.05591721 0.06648594 0.07825871 0.09012539 0.10006665 0.10558334
[13] 0.10494052 0.09835367 0.08782920 0.07584810 0.06425529 0.05399054
[19] 0.04532934 0.03819719
```

```
> pcauchy(seq(-5, 5, length.out = 15), location = 1, scale = 3)
 [1] 0.1475836 0.1643213 0.1848605 0.2104166 0.2425988 0.2833834 0.3347507
 [8] 0.3975836 0.4697759 0.5451672 0.6158581 0.6764416 0.7255627 0.7644587
[15] 0.7951672

> qcauchy(c(0.01, 0.025, 0.05, 0.25, 0.5, 0.75, 0.95, 0.975, 0.99), location = 1,
scale = 3)
[1] -94.46155 -37.11861 -17.94125  -2.00000   1.00000   4.00000  19.94125
[8]  39.11861  96.46155
```

Repartiţia Gama

```
> rgamma(15, shape = 2, rate = 2)
 [1] 0.3161372 1.4655756 0.6114103 1.7457244 0.8275230 0.1643775 1.1141751
 [8] 0.3762099 0.6089281 2.2397701 1.9685723 2.4453056 1.3807334 2.2222921
[15] 2.4706962

> dgamma(seq(0, 5, length.out = 20), shape = 1, rate = 3)
 [1] 3.000000e+00 1.362251e+00 6.185761e-01 2.808853e-01 1.275455e-01
 [6] 5.791632e-02 2.629886e-02 1.194188e-02 5.422615e-03 2.462321e-03
[11] 1.118100e-03 5.077110e-04 2.305433e-04 1.046860e-04 4.753619e-05
[16] 2.158541e-05 9.801583e-06 4.450739e-06 2.021008e-06 9.177070e-07

> pgamma(seq(0, 5, length.out = 15), shape = 1, rate = 3)
 [1] 0.0000000 0.6574811 0.8826808 0.9598160 0.9862362 0.9952856 0.9983852
 [8] 0.9994469 0.9998106 0.9999351 0.9999778 0.9999924 0.9999974 0.9999991
[15] 0.9999997

> qgamma(c(0.01, 0.025, 0.05, 0.25, 0.5, 0.75, 0.95, 0.975, 0.99), shape = 1,
rate = 3)
[1] 0.003350112 0.008439269 0.017097765 0.095894024 0.231049060 0.462098120
[7] 0.998577425 1.229626485 1.535056729
```

Repartiţia Beta

```
> rbeta(15, shape1 = 2.5, shape2 = 1)
 [1] 0.8431672 0.8825630 0.9030902 0.9792175 0.6171661 0.1357871 0.7792012
 [8] 0.4951123 0.7984270 0.3408624 0.5171854 0.7403253 0.8885471 0.6706276
[15] 0.9870174

> dbeta(seq(0, 1, length.out = 20), shape1 = 1, shape2 = 3)
 [1] 3.000000000 2.692520776 2.401662050 2.127423823 1.869806094 1.628808864
 [7] 1.404432133 1.196675900 1.005540166 0.831024931 0.673130194 0.531855956
[13] 0.407202216 0.299168975 0.207756233 0.132963989 0.074792244 0.033240997
[19] 0.008310249 0.000000000

> pbeta(seq(0, 1, length.out = 15), shape1 = 1, shape2 = 3)
 [1] 0.0000000 0.1993440 0.3702624 0.5149417 0.6355685 0.7343294 0.8134111
 [8] 0.8750000 0.9212828 0.9544461 0.9766764 0.9901603 0.9970845 0.9996356
[15] 1.0000000

> qbeta(c(0.01, 0.025, 0.05, 0.25, 0.5, 0.75, 0.95, 0.975, 0.99), shape1 = 1,
shape2 = 3)
[1] 0.003344507 0.008403759 0.016952428 0.091439704 0.206299474 0.370039475
[7] 0.631596850 0.707598226 0.784556531
```