

Ficha 4

```
//codigo ex do prof
var prop = {"ps":36.34,"psd":27.76,"cdu":6.33,"cds":4.22, "be":9.52,
  "pan":3.32,"chega":1.29,"il":1.29, "livre":1.09, "indecisos":8.84}
var pares = _.toPairs(prop)
print(pares)
var tuplo = _.unzip(pares)
var partidos = tuplo[0]
var percentagens = tuplo[1]
print(partidos)
print(percentagens)
var dic = _.fromPairs(_.zip("partidos percentagens".split(" "), tuplo))
print(dic)
print(dic.partidos)
```

run

```
[[{"ps",36.34}, {"psd",27.76}, {"cdu",6.33}, {"cds",4.22}, {"be",9.52}, {"pan",3.32}, {"chega",1.29}, {"il",1.29}, {"livre",1.09}, {"indecisos",8.84}]
["ps","psd","cdu","cds","be","pan","chega","il","livre","indecisos"]
[36.34,27.76,6.33,4.22,9.52,3.32,1.29,1.29,1.09,8.84]
{"partidos":["ps","psd","cdu","cds","be","pan","chega","il","livre","indecisos"],"percentagens":[36.34,27.76,6.33,4.22,9.52,3.32,1.29,1.29,1.09,8.84]}
["ps","psd","cdu","cds","be","pan","chega","il","livre","indecisos"]
[36.34,27.76,6.33,4.22,9.52,3.32,1.29,1.29,1.09,8.84]
```

Exercicio 1

```
var pessoa = 1000;
var sondagens = function (pessoas) {
  var perc = {"ps":36.34,"psd":27.76,"cdu":6.33,"cds":4.22, "be":9.52,
    "pan":3.32,"chega":1.29,"il":1.29, "livre":1.09, "indecisos":8.84}
  var per = _.values(perc)
  var percent = map(function (x){x/100}, per)
  var sond = multinomial({ps:percent,n:pessoas})
  var dic = _.fromPairs(_.zip(_.keys(perc), sond))
  return dic
}
```

run

```
{"ps":352,"psd":274,"cdu":64,"cds":49,"be":104,"pan":37,"chega":17,"il":10,"livre"
4,"indecisos":79}
```

Exercicio 2



```
var pessoa = 1000;

var sondagens = function (pessoas) {
  var perc = {"ps":36.34,"psd":27.76,"cdu":6.33,"cds":4.22, "be":9.52,
    "pan":3.32,"chega":1.29,"il":1.29, "livre":1.09, "indecisos":8.84}
  var per = _.values(perc)
  var percent = map(function (x){x/100}, per)
  var sond = multinomial({ps:percent,n:pessoas})
  var dic =_.fromPairs(_.zip(_.keys(perc), sond))
  return dic
}

var s = sondagens (pessoa)

var modelo = function () {
  var percent = mapN(function (){uniform(0,1)}, 10)
  var per = map(function (x){x/(sum(percent))}, percent)
  var dist = Multinomial({ps:per,n:pessoa})
  observe (dist, _.values(s))
  var dici =_.fromPairs(_.zip(_.keys(s), per))
  return dici
}
```

run

X



Exercicio 3



```
var pessoa = 1000;

var sondagens = function (pessoas) {
  var perc = {"ps":36.34,"psd":27.76,"cdu":6.33,"cds":4.22, "be":9.52,
    "pan":3.32,"chega":1.29,"il":1.29, "livre":1.09, "indecisos":8.84}
  var per = _.values(perc) //lista com os valores do elementos de perc; a prob de cada
  var percent = map(function (x){x/100}, per) //é só passar para percentagem
  var sond = multinomial({ps:percent,n:pessoas})
  var dic =_.fromPairs(_.zip(_.keys(perc), sond))
  return dic
}

var s = sondagens (pessoa)

var modelo = function () {
  var percent = mapN(function (){uniform(0,1)}, 10)
  var per = map(function (x){x/(sum(percent))}, percent)
  var dist = Multinomial({ps:per,n:pessoa})
  observe (dist, _.values(s))
  var dici =_.fromPairs(_.zip(_.keys(s), per))
  //print(_(dici))
  return dici
}

viz.marginals(Infer(modelo))
```

run

▼

```
x Uniform({ a: 0, b: 1 })
```

X

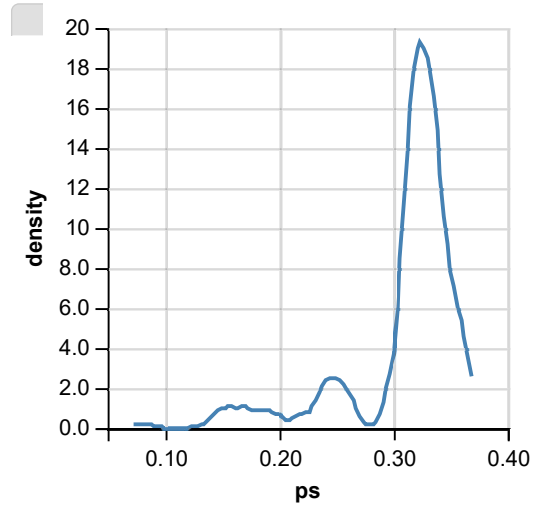
Enumerate can only be used with distributions that have finite support...quit enumerate

Using "rejection"

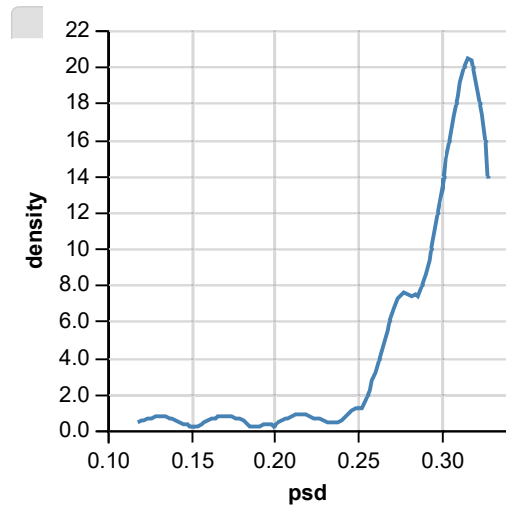
0.00 samples/sec is below threshold...quit rejection

Using "MCMC"

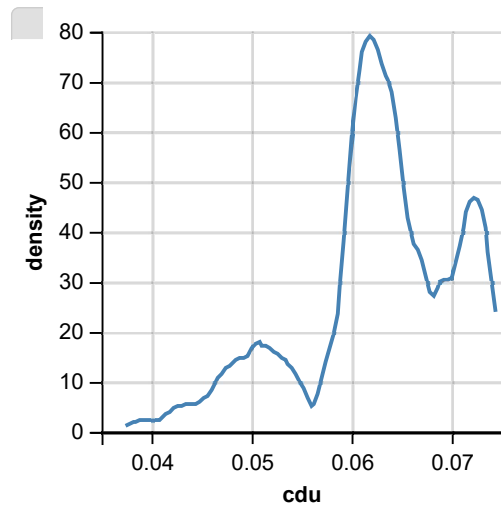
ps:



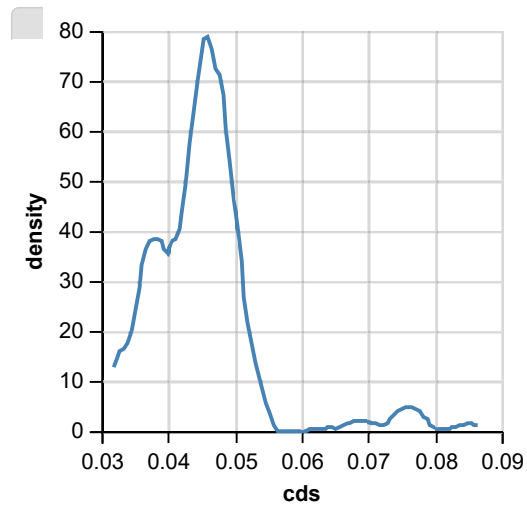
psd:



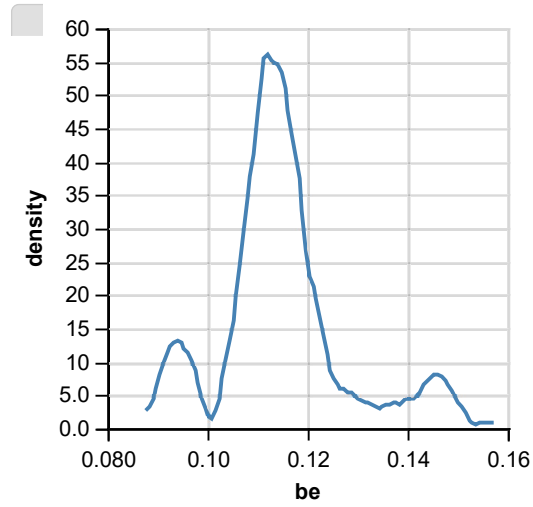
cdu:



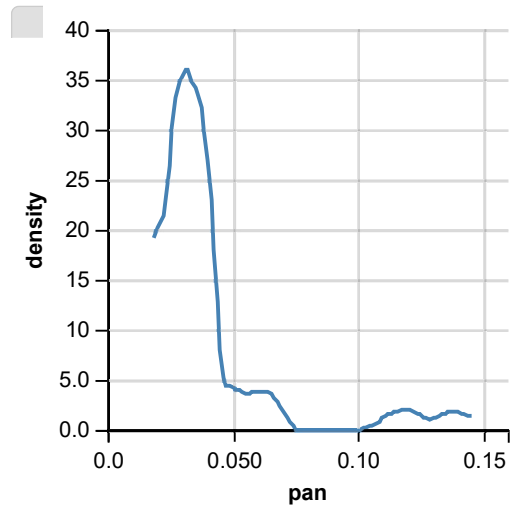
cds:



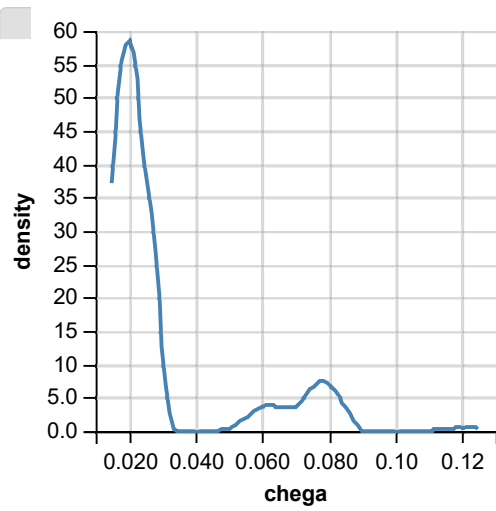
be:



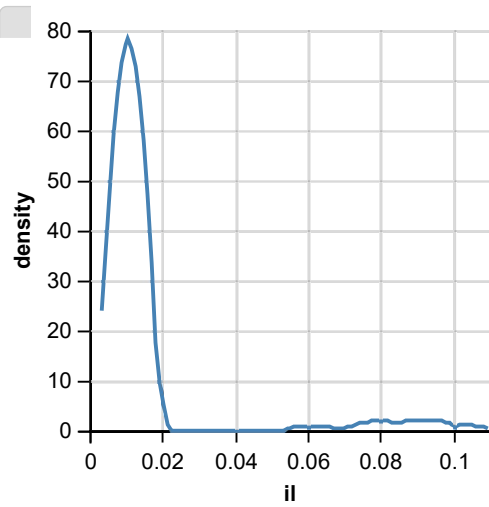
pan:



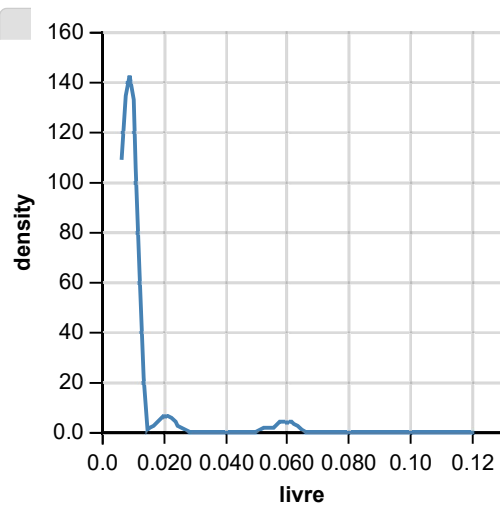
chega:



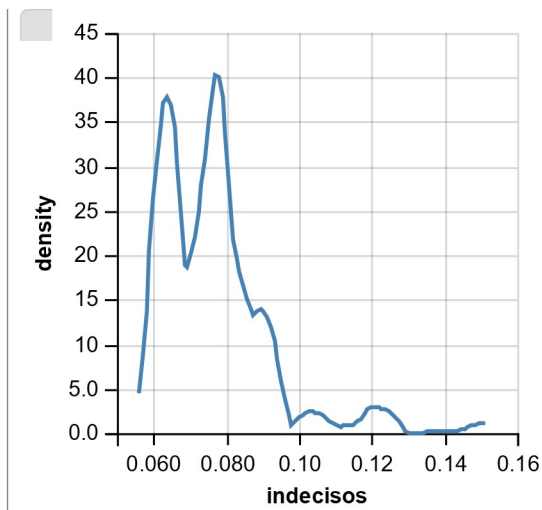
il:



livre:



indecisos:



Exercicio 4

```

var pessoa = 1000;
var sondagens = function (pessoas) {
  var perc = {"ps":36.34,"psd":27.76,"cdu":6.33,"cds":4.22, "be":9.52,
    "pan":3.32,"chega":1.29,"il":1.29, "livre":1.09, "indecisos":8.84}
  var per = _.values(perc) //lista com os valores do elementos de perc; a prob de cada
  var percent = map(function (x){x/100}, per) //é só passar para percentagem
  var sond = multinomial({ps:percent,n:pessoas})
  var dic = _.fromPairs(_.zip(_.keys(perc), sond)) //zip dos valores e keys; atribuição
  return dic
}

var s = sondagens (pessoa)

var modelo = function () {
  var percent = mapN(function (){uniform(0,1)}, 10)
  var per = map(function (x){x/(sum(percent))}, percent)
  var dist = Multinomial({ps:per,n:pessoa})
  observe (dist, _.values(s))
  var dici = _.fromPairs(_.zip(_.keys(s), per))
  return dici
}

var estimar_intervalo = function(dist, margem, low, high) {
  expectation(marginalize(dist, margem), function(p) {low < p && p < high})
}

var HDI = function(dist, margem, low, high, delta) {
  var p = estimar_intervalo(dist, margem, low, high)
  if (p <= 0.95) return [low, high]
  var A = estimar_intervalo(dist, margem, low + delta, high)
  var B = estimar_intervalo(dist, margem, low, high - delta)
  return A > B ? HDI(dist, margem, low + delta, high, delta)
    : HDI(dist, margem, low, high - delta, delta)
}

var print_intervals = function(dist, margens) {
  map(function(m) {
    print(m + ": " + HDI(dist, m, 0, 1, 0.005))
  })
}

```

```

    }, margens)

}
var dist = Infer(modelo)
viz.marginals(dist)
var dummy = print_intervals(dist, ["ps","psd","cdu","cds","be","pan","chega","il",
                                  "livre", "indecisos"])

```

run



x Uniform({ a: 0, b: 1 })

X

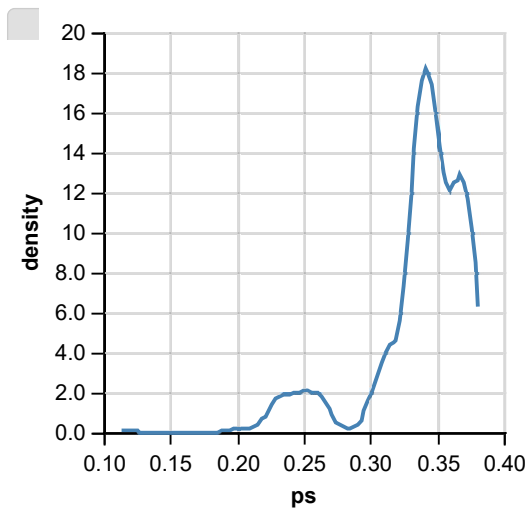
Enumerate can only be used with distributions that have finite support...quit enumerat
e

Using "rejection"

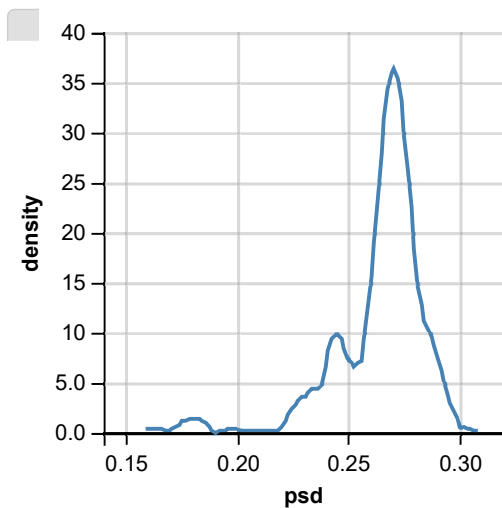
0.00 samples/sec is below threshold...quit rejection

Using "MCMC"

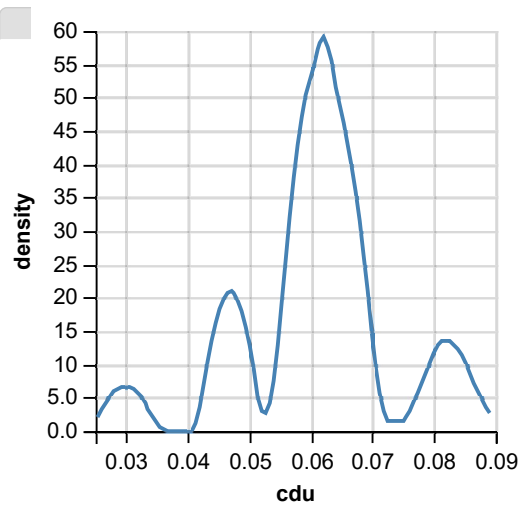
ps:



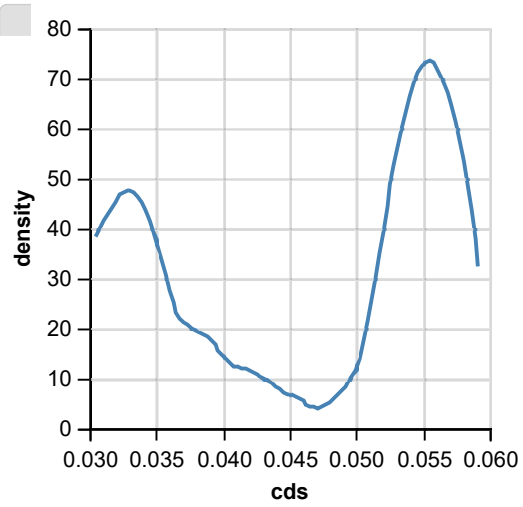
psd:



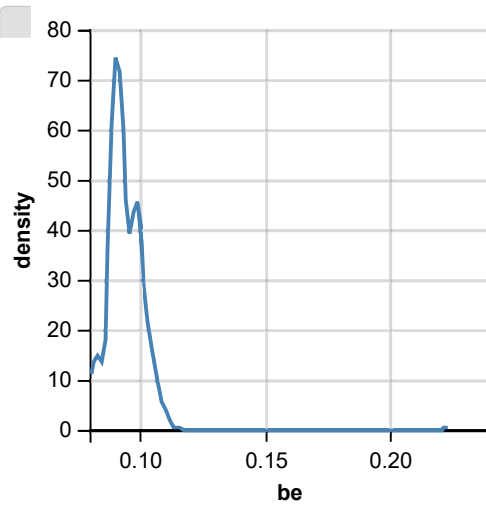
cdu:



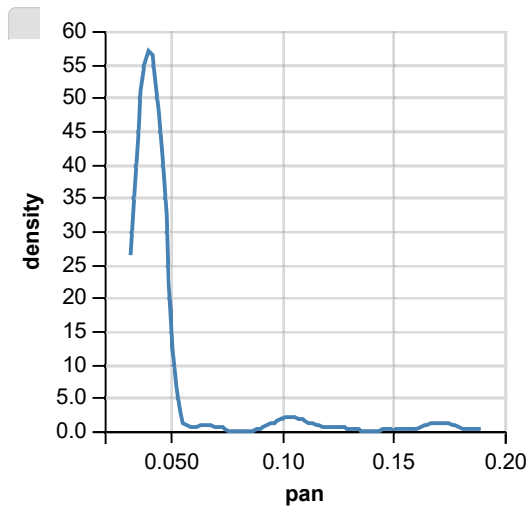
cds:



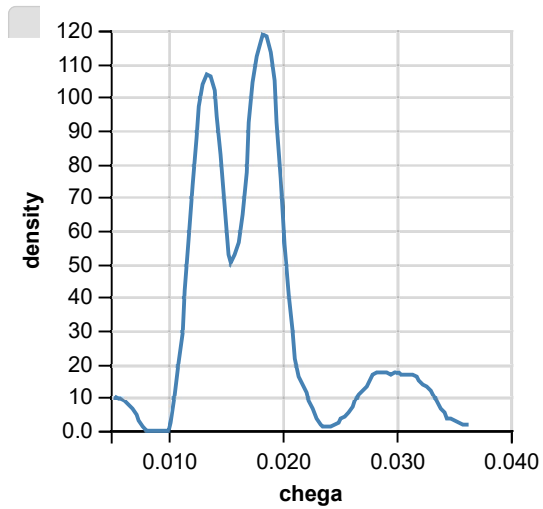
be:



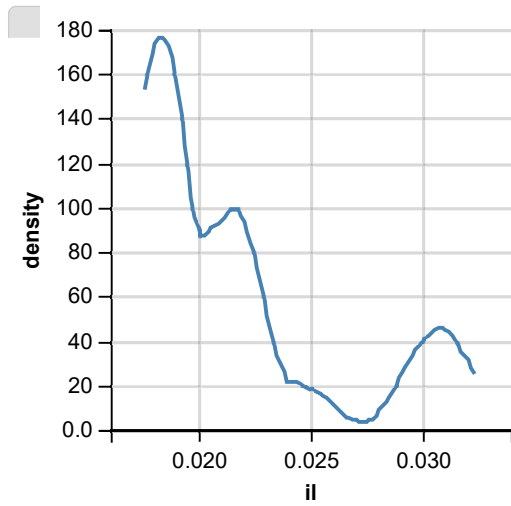
pan:



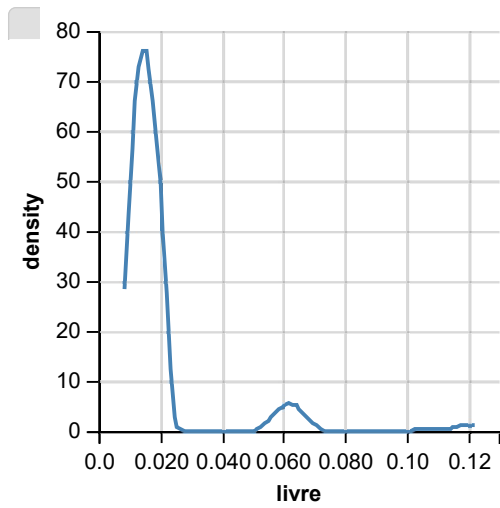
chega:



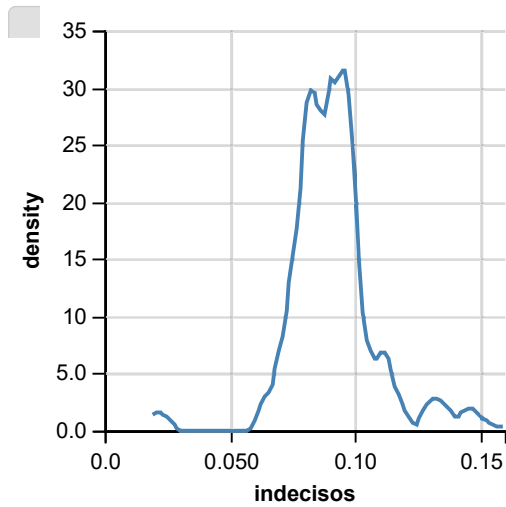
il:



livre:



indecisos:



ps: 0.24000000000000013,0.37999999999999945

psd: 0.23000000000000012,0.2949999999999994

cdu: 0.035,0.08499999999999919

cds: 0.035,0.05999999999999917

be: 0.085,0.1049999999999992

pan: 0.030000000000000002,0.1049999999999992

chega: 0.01,0.029999999999999184

il: 0.015,0.029999999999999184

livre: 0.005,0.05999999999999917

indecisos: 0.0649999999999999,0.12999999999999923