Estimating the Number of Essential Genes using Occugene

Oliver Will
April 24, 2017

This vignette contains code from a chapter of the forthcoming book, A Osterman and S Gerdes. Gene Essentiality: Protocols and Bioinformatics. Humana Press. This package has similar functionality as the R package negenes written by Karl Broman.


We model the number of insertions per clone as a Multinomial\((n, p_1, \ldots, p_k)\) random vector. The number of knockouts in the library follows the occupancy distribution of the multinomial random variable. We compute the expected number of genes hit if there were no essential genes.

```r
> library("occugene")
> n <- 60
> p <- c(seq(10,1,-1),seq(10,1,-1),18)/124
> p <- p/sum(p)
> eMult(n,p)
[1] 17.74773
> varMult(n,p)
[1] 1.744004
```

We approximate the moments of the occupancy distribution using Monte Carlo integration.

```r
> eMult(n,p,iter=100,seed=4)
[1] 17.64
> varMult(n,p,iter=100,seed=4)
```

1
We load an example hit table and experimental results to analyze. The format of the hit table is different than what is used in \textit{negenes} because we wish to track the order of insert locations.

```r
> data(sampleAnnotation)
> data(sampleInsertions)
> print(sampleAnnotation)

<table>
<thead>
<tr>
<th>idNum</th>
<th>first</th>
<th>last</th>
<th>orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>11</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>21</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>38</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>45</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>48</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>53</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>57</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>60</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>62</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>73</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>83</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>92</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>107</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>110</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>17</td>
<td>115</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>119</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>19</td>
<td>122</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>124</td>
<td>0</td>
</tr>
</tbody>
</table>

> print(sampleInsertions)

<table>
<thead>
<tr>
<th>position</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
</tbody>
</table>
```
We estimate the number of genes that will be knocked out in the next 10 clones using the Efron and Thisted estimator.

```r
> orf <- cbind(a.data$first, a.data$last)
> clone <- experiment$position
> etDelta(10, orf, clone)
```

$expected

[1] 0.1190665

$variance

[1] 0.02936508

We use the Will and Jacobs’ bootstrap to estimate the number of knockouts made in the next 10 clones.

```r
> orf <- cbind(a.data$first, a.data$last)
> clone <- experiment$position
> fFit(orf, clone, FALSE)
```

Nonlinear regression model

- model: noOrfs ~ b0 + b1 * exp(-b2 * x)
- data: cumul

<table>
<thead>
<tr>
<th>b0</th>
<th>b1</th>
<th>b2</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.34393</td>
<td>12.05475</td>
<td>0.06668</td>
</tr>
</tbody>
</table>

residual sum-of-squares: 15.62

Number of iterations to convergence: 6
Achieved convergence tolerance: 7.669e-07

```r
> unbiasDelta0(10, orf, clone, iter=10, seed=4, alpha=0.05, TR=F)
```

$delta0

[1] 0.2165025

$CI

[1] -0.1282142  0.4791953

We estimate the number of essential genes using the Will and Jacobs’ bootstrap.

```r
> unbiasB0(orf, clone, iter=10, seed=4, alpha=0.05, TR=F)
```
Finally, we convert occugene’s data format into the format for negenes.

```r
> newOrf <- occup2Negenes(orf, clone)
> print(newOrf)
```

```
      n.sites n.sites2 counts counts2
     1       10      0      5         0
     2       9       0      3         0
     3       8       0      0         0
     4       7       0      4         0
     5       4       2      0         0
     6       3       0      0         0
     7       4       0      2         0
     8       3       0      2         0
     9       2       0      0         0
    10       1       0      1         0
    11      10       0      7         0
    12       9       0      0         0
    13       8       0      4         0
    14       7       0      5         0
    15       4       2      0         0
    16       3       0      0         0
    17       4       0      6         0
    18       3       0      0         0
    19       2       0      3         0
    20       1       0      1         0
```

We outlined the basic features of the occugene package in this Sweave document.