Modelling Evolutionary Trees
CSC8622

Pedro Pinto da Silva Alexander Kell

November 3, 2016

Part 1

Question (i)

```
buildTree = function(n=10, lambda=0.5) {
  nspecies = (2*n - 2)
  cols = c("parent", "child", "birth", "termination", "length")

  tree = matrix(NA, nrow = nspecies, ncol = length(cols))
  colnames(tree) = cols
  tree[,c("parent", "child")] = 0

  t = 0
  for(k in 1:(n-1)) {
    if(k == 1) {
      parent = 2*n - 1
    } else {
      candidates = which( ! (tree[, "child"] %in% tree[, "parent"]))
      # Length of candidates is always > 1 otherwise we would
      #  have to be careful with the behavior of sample
      #  (undesired behavior for length == 1)
      parent = sample(candidates, 1)
      t = t + rexp(1, rate = k*lambda)
    }

    childs = sample(which(tree[,"parent"]==0), 2)
    tree[childs, "child"] = childs
    tree[childs, "parent"] = parent
    tree[childs, "birth"] = t
    if(k > 1) {
      tree[parent, "termination"] = t
    }
  }
  t = t + rexp(1, rate = n*lambda)
  tree[is.na(tree[, "termination"]), "termination"] = t

  tree[, "length"] = tree[, "termination"] - tree[, "birth"]
  return(tree)
}
```

```r
isExtant = function(tree, index=1:nrow(tree)) {
  ! (tree[index, "child"] %in% tree[, "parent"])
}

loadSpecies = function(path="../aux/species.txt") {
  species = read.table(path, header=FALSE, sep = "+", stringsAsFactors = FALSE)$V1
  species[-which(species=="unavailable")]
}

# Yet Another Yule (YAY)
yay = function(n=10, lambda=0.5) {
  tree = buildTree(n)
  species = loadSpecies()
  if(length(species) > 0) {
    nomes = species[sample(1:length(species), nrow(tree)+1)]
  } else {
    nomes = paste("poney", 1:(nrow(tree)+1), sep="")
  }

  yule = data.frame(Parent      = tree[, "parent"],
                    ParentName  = nomes[tree[, "parent"]],
                    Child       = tree[, "child"],
                    ChildName   = nomes[tree[, "child"]],
                    isExtant    = isExtant(tree),
                    Birth       = tree[, "birth"],
                    Termination = tree[, "termination"],
                    Length      = tree[, "length"])
  yule[yule$Parent == 2*n-1, ]$ParentName = nomes[2*n-1]
  return(yule)
}

yule = yay()
head(yule, 1)
```

```
##   Parent       ParentName Child           ChildName isExtant     Birth
## 1     18 Perameles nasuta     1 Antechinus flavipes    FALSE 0.2019885
##   Termination   Length
## 1   0.9825305 0.780542
```

```r
yule[, -c(2,4)]
```

```
##    Parent Child isExtant     Birth Termination    Length
## 1      18     1    FALSE 0.2019885   0.9825305 0.7805420
## 2       6     2    FALSE 1.8371075   2.5503010 0.7131935
## 3      18     3    FALSE 0.2019885   1.2549904 1.0530019
## 4       2     4     TRUE 2.5503010   2.9562085 0.4059075
## 5       2     5     TRUE 2.5503010   2.9562085 0.4059075
## 6      16     6    FALSE 0.9115773   1.8371075 0.9255302
## 7       3     7    FALSE 1.2549904   1.5913406 0.3363502
## 8      16     8     TRUE 0.9115773   2.9562085 2.0446312
## 9       6     9     TRUE 1.8371075   2.9562085 1.1191010
## 10     14    10     TRUE 0.6480761   2.9562085 2.3081324
## 11      3    11     TRUE 1.2549904   2.9562085 1.7012181
## 12      1    12     TRUE 0.9825305   2.9562085 1.9736780
## 13      7    13     TRUE 1.5913406   2.9562085 1.3648679
## 14     19    14    FALSE 0.0000000   0.6480761 0.6480761
```

```
## 15      1    15     TRUE 0.9825305   2.9562085 1.9736780
## 16     14    16    FALSE 0.6480761   0.9115773 0.2635012
## 17      7    17     TRUE 1.5913406   2.9562085 1.3648679
## 18     19    18    FALSE 0.0000000   0.2019885 0.2019885
```

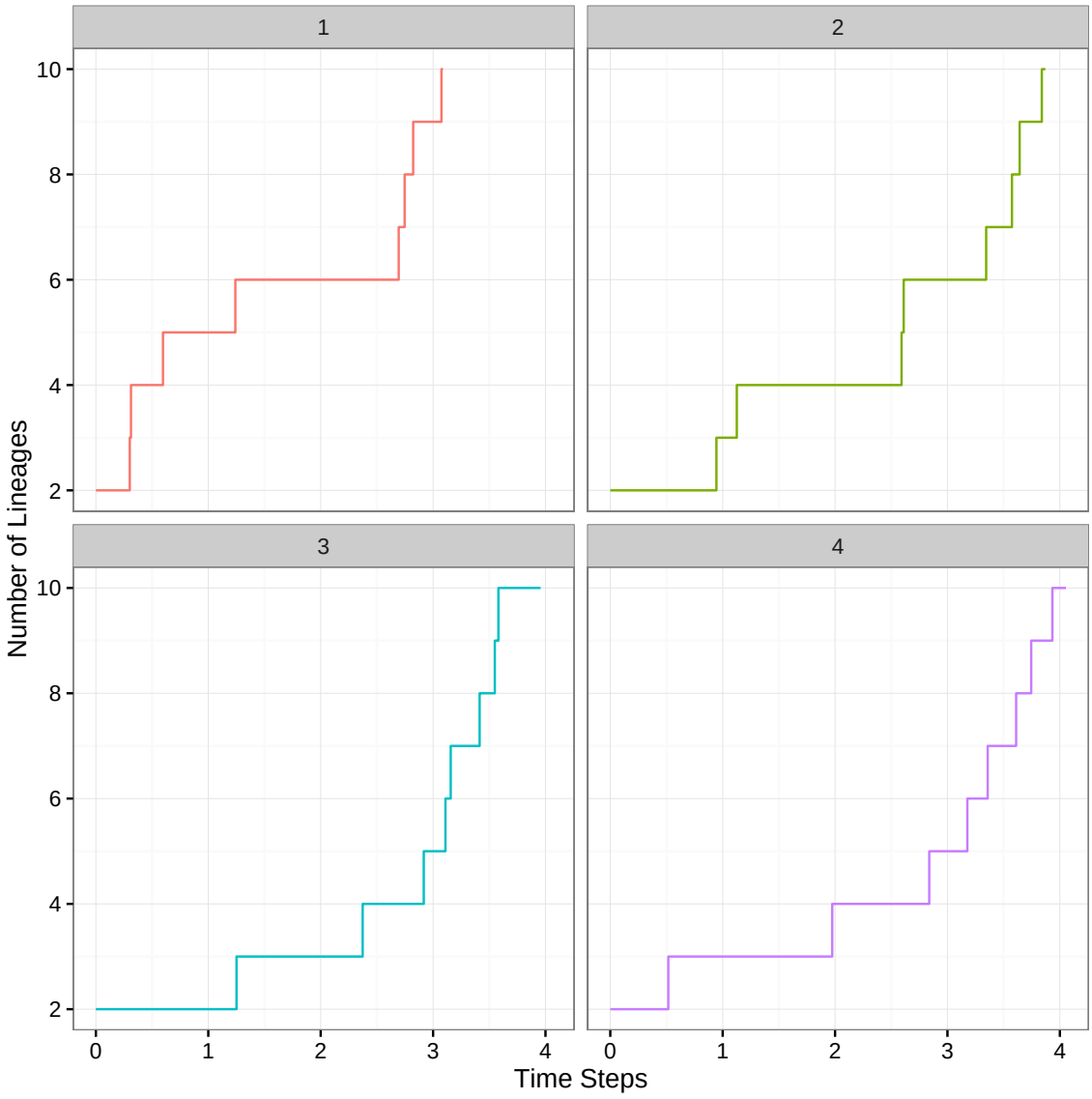
Question (ii)

```
evolutionOf = function(yule) {
  tstep = unique(sort(yule$Birth))
  tstep = c(tstep, max(yule$Termination))
  return(data.frame(tstep=tstep, nlineages=c(2:length(tstep), length(tstep))))
}

evolutionOf(yule)
```

```
##        tstep nlineages
## 1  0.0000000         2
## 2  0.2019885         3
## 3  0.6480761         4
## 4  0.9115773         5
## 5  0.9825305         6
## 6  1.2549904         7
## 7  1.5913406         8
## 8  1.8371075         9
## 9  2.5503010        10
## 10 2.9562085        10
```

Question (iii)

```
n = 10
lambda=0.5
four_yays = lapply(1:4, function(i) evolutionOf(yay(n, lambda)))
four_yays = rbind.fill(four_yays)
four_yays$group = ((as.numeric(rownames(four_yays)) - 1) %/% n) + 1

ggplot(four_yays, aes(x = tstep, y = nlineages))   +
  geom_step(aes(colour=factor(group))) +
  facet_wrap(~ group, ncol=2) +
  ylab("Number of Lineages") +
  xlab("Time Steps") +
  theme_bw() +
  scale_colour_discrete(guide = FALSE)
```



Part 2

Part 3