Modelling Evolutionary Trees
CSC8622

Pedro Pinto da Silva   Alexander Kell

November 4, 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     11 Heloderma horridum     1 Colaptes campestroides     TRUE 1.227838
##   Termination    Length
## 1    1.437527 0.2096885
```

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

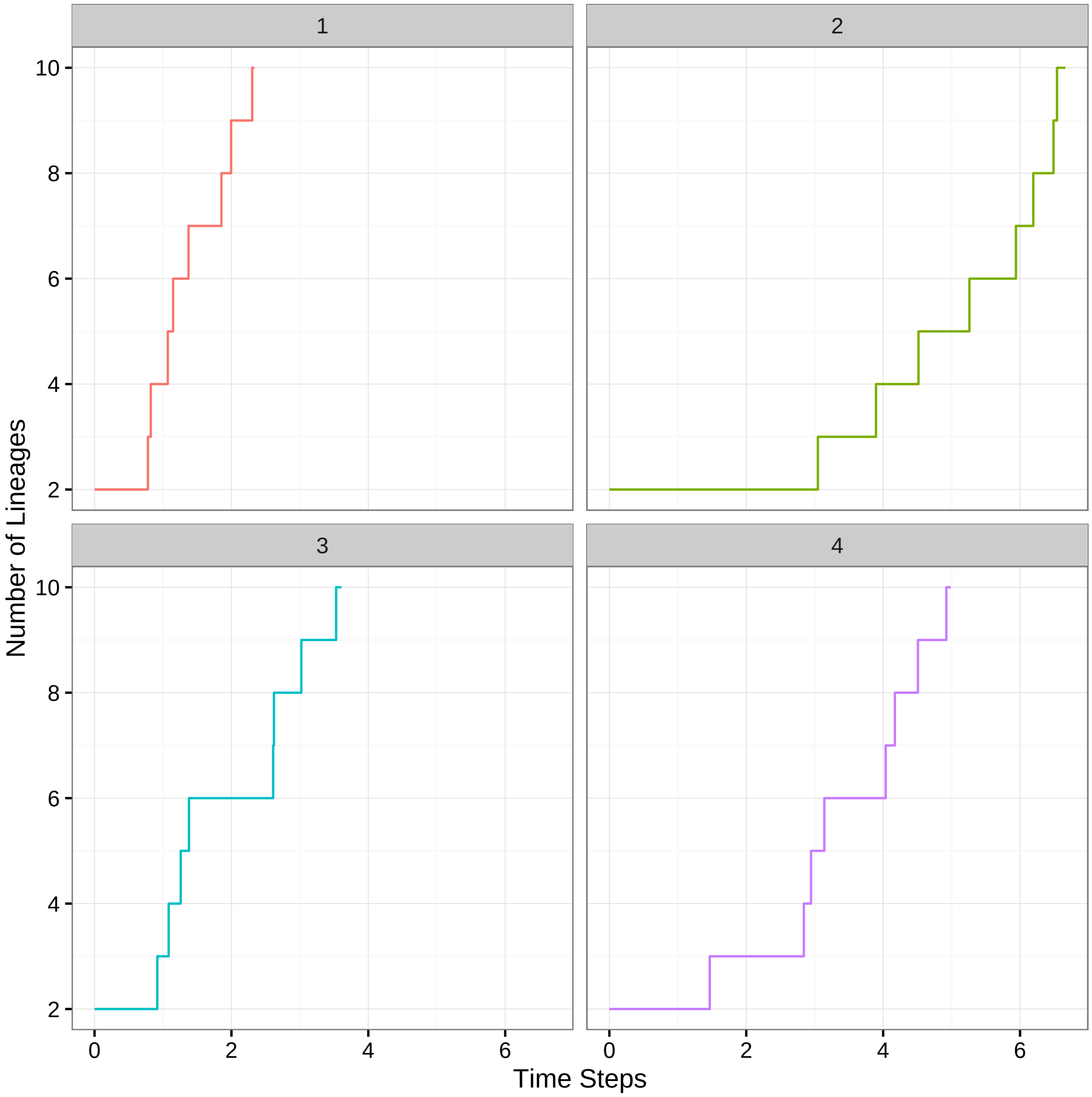
```
##    Parent Child isExtant     Birth Termination     Length
## 1      11     1     TRUE 1.2278382   1.4375268 0.20968852
## 2      11     2     TRUE 1.2278382   1.4375268 0.20968852
## 3       7     3     TRUE 1.3232742   1.4375268 0.11425259
## 4       8     4     TRUE 0.7246738   1.4375268 0.71285294
## 5       7     5    FALSE 1.3232742   1.3440240 0.02074988
## 6      15     6     TRUE 0.4224356   1.4375268 1.01509113
## 7      17     7    FALSE 0.8144395   1.3232742 0.50883463
## 8      18     8    FALSE 0.2340444   0.7246738 0.49062939
## 9      19     9    FALSE 0.0000000   0.6764591 0.67645907
## 10      5    10     TRUE 1.3440240   1.4375268 0.09350271
## 11     17    11    FALSE 0.8144395   1.2278382 0.41339871
## 12     15    12     TRUE 0.4224356   1.4375268 1.01509113
## 13      9    13     TRUE 0.6764591   1.4375268 0.76106769
## 14      9    14     TRUE 0.6764591   1.4375268 0.76106769
## 15      18     15     FALSE 0.2340444    0.4224356 0.18839120
## 16       5     16      TRUE 1.3440240    1.4375268 0.09350271
## 17       8     17     FALSE 0.7246738    0.8144395 0.08976572
## 18      19     18     FALSE 0.0000000    0.2340444 0.23404442
```

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  0.2340444         3
## 3  0.4224356         4
## 4  0.6764591         5
## 5  0.7246738         6
## 6  0.8144395         7
## 7  1.2278382         8
## 8  1.3232742         9
## 9  1.3440240        10
## 10 1.4375268        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