Exercises Week 14

Aaron Palumbo

November 27, 2015

Contents

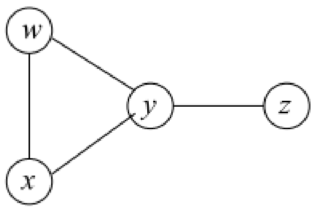

Figure 10.20: A social graph

10.5.1 (Section 10.5.5)

Suppose graphs are generted by picking a probabliity p and choosing each edge independently with probability p, as in Example 10.21. For the graph of Fig. 10.20, what value of p gives the maximum likelihood of seeing that graph? What is the probability this graph is generated?

There are $\binom{4}{2} = 6$ pairs of nodes. The probability this graph is generated is:

$$p^4(1-p)^2$$

To find the value of p that maximizes this:

$$\begin{aligned}
\frac{d}{dp}p^4(1-p)^2 &= 0 \\
p^4(2(1-p)(-1)) + (1-p)^2(4p^3) &= 0 \\
2p^5 - 2p^4 + (1 - 2p + p^2)(4p^3) &= 0 \\
2p^5 - 2p^4 + 4p^3 - 8p^4 + 4p^5 &= 0 \\
2(3p^5 - 5p^4 + 2p^3) &= 0 \\
p^3(3p^2 - 5p + 2) &= 0 \\
p^3(3p-2)(p-1) &= 0
\end{aligned}$$

The potential maximum values are 0, 2/3, and 1. 0 and 1 will make the probability 0, so the value of p that maximizes the probability is 2/3.

The probability this graph is generated is $(2/3)^4(1-2/3)^2 = 0.0219479$

10.7.1 (Section 10.7.6)

How many triangles are there in the graphs:

```r
library(dplyr)

find_triangles <- function(edges) {
  names(edges) <- c("A", "B")
  # for edges = E and |><| = 'natural join' do:
  # [ E(X, Y) |><| E(Y, Z) ] |><| E(X, Z)
  return(

    inner_join(
      # E(X, Y) |><| E(Y, Z)
      inner_join(edges, edges, by=c("B" = "A")),
      # . |><| E(X, Z)
      edges, by="A"
    ) %>%
      setNames(., c("X", "Y", "Z1", "Z2")) %>%
      # if Z1 == Z2 there exists X -> Y -> Z and X -> Z
      # this is a triangle
      filter(Z1 == Z2) %>%
      # clean up
      select(X, Y, Z1) %>%
      setNames(., c("i", "j", "k"))

  )
}
```

10.7.1 (a)

Figure 10.1.

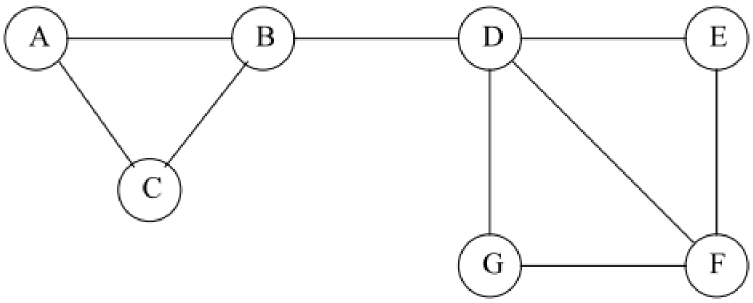

Figure 10.1: Example of a small social network

```r
# Sparse matrix representation
E = as.data.frame(matrix(c("A", "B",
                           "A", "C",
                           "B", "C",
                           "B", "D",
                           "D", "E",
                           "D", "F",
                           "D", "G",
                           "E", "F",
                           "F", "G"),
                         ncol = 2,
                         byrow = TRUE))
colnames(E) <- c("A", "B")

knitr::kable(find_triangles(E))
```

i	j	k
A	B	C
D	E	F
D	F	G

10.7.1 (b)

Figure 10.9

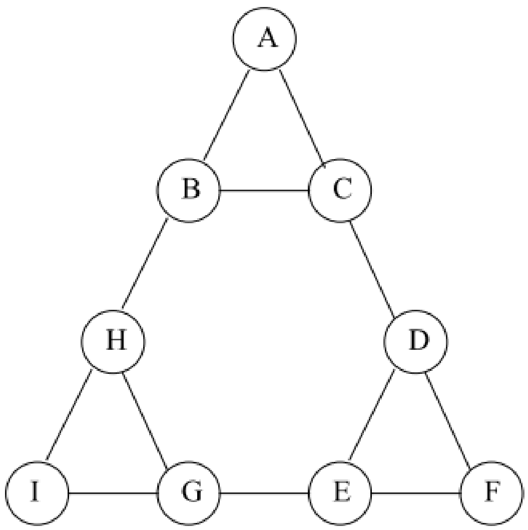

Figure 10.9: Graph for exercises

```
# Sparse matrix representation
E = as.data.frame(
  matrix(c("A", "B",
           "A", "C",
           "B", "C",
           "B", "H",
           "C", "D",
           "D", "E",
           "D", "F",
           "E", "F",
           "E", "G",
           "G", "H",
           "G", "I",
           "H", "I"
            ),
         ncol = 2,
         byrow = TRUE)
  )
colnames(E) <- c("A", "B")

knitr::kable(find_triangles(E))
```

i	j	k
A	B	C
D	E	F
G	H	I

10.7.1 (c)

Figure 10.2

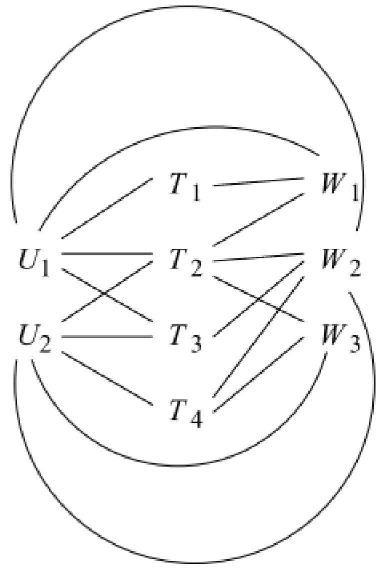

Figure 10.2: A tripartite graph representing users, tags, and Web pages

```r
# Sparse matrix representation
E = as.data.frame(
  matrix(c("T1", "U1",
           "T1", "W1",
           "T2", "U1",
           "T2", "U2",
           "T2", "W1",
           "T2", "W2",
           "T2", "W3",
           "T3", "U1",
           "T3", "U2",
           "T3", "W2",
           "T4", "U2",
           "T4", "W2",
           "T4", "W3",
           "U1", "W1",
           "U1", "W2",
           "U2", "W2",
           "U2", "W3"
           ),
         ncol = 2,
         byrow = TRUE)
  )
colnames(E) <- c("A", "B")
```

```r
knitr::kable(find_triangles(E))
```

i	j	k
T1	U1	W1
T2	U1	W1
T2	U1	W2
T2	U2	W2
T2	U2	W3
T3	U1	W2
T3	U2	W2
T4	U2	W2
T4	U2	W3