Making assorted networks by re-wiring

We will generate scale-free networks according to the parameters listed in Table 1 using the classic BA-algorithm. Following network generation, we will update the networks as following:

1. Assign nodes randomly as male (0) or female (1).
2. Calculate temporary value of sex-assortativity in the network (r_t).
3. If r_t is less than the desired r, randomly choose a percentage of type 0–1 edges (i.e, a male–female edge) and re-wire them.
4. Repeat step 3 until $|r_f - r_t| \leq \epsilon$ or until a max number of re-wirings is completed.

Table 1: Design of pilot study I for generating networks.

Variable	Value
Sex-assortativity, r	-0.4, - 0.2, 0, 0.2, 0.4
Degree distribution, $p(k)$	$\frac{k^{-\alpha}}{\zeta(\alpha)}$
Mean degree, $< k >$	10
Network size, N	500, $1 \cdot 10^3$

Step 1: Assign node sex

Step 2: Calculate r_t

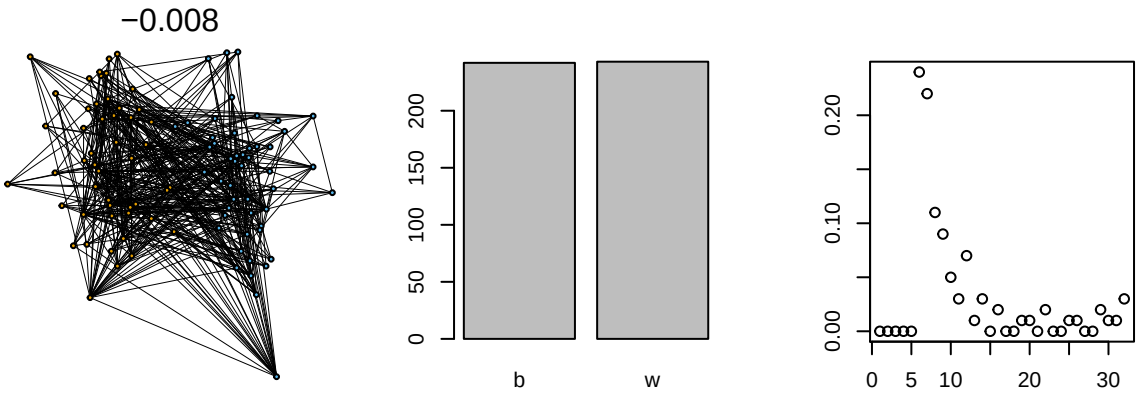
Step 3: Re-wire if less than r_f

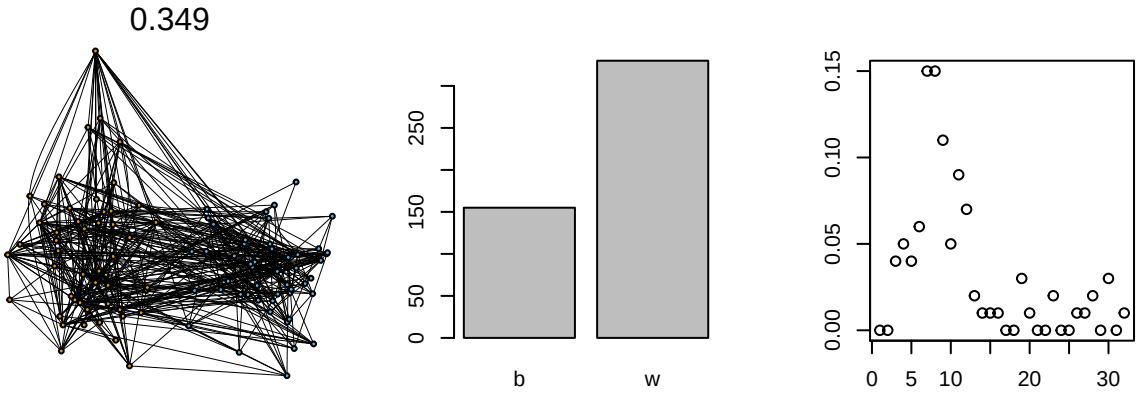
TEST ALGORITHM #1

This test algorithm shows a major issue: It produces self and multiple loops and removing them lowers the mean degree pretty substantially (at least with graph size of 100).

igraph has a function to re-wire edges that prevents self and multiple loops but we need to modify the use slightly so that it's only re-wiring edges that are between sexes.

TEST ALGORITHM #2






```
transitivity(Gg0)
```

```
## [1] 0.1674525
```

```
transitivity(Gg)
```

```
## [1] 0.1722716
```

```
diameter(Gg0)
```

```
## [1] 3
```

```
diameter(Gg)
```

```
## [1] 4
```

```
mean(degree(Gg0))
```

```
## [1] 9.7
```

```
mean(degree(Gg))
```

```
## [1] 9.7
```

```
assortativity_degree(Gg0)
```

```
## [1] -0.1252495
```

```
assortativity_degree(Gg)
```

```
## [1] -0.08816195
```

Notes:

- need to generalize for neg assort. coefficient (could base iteration progress on whether r_t is getting closer to r_f) rather than if r_t is getting bigger. This would also clean up one level of the if statements.
- also should begin to structure this code into a few nice functions rather than sad for loop.