HW 6 Report

Yongchao Qiao

Introduction

Exponential random graph models true generative statistical model of network structure and characteristics. They can be used to test inferential hypotheses and predict network properties, like diameter, degree distribution, etc. Also, they are popular to handle the complex dependencies of networks, to handle different types of predictors/covariates, to predict overall network characteristics and to be used through well-implemented software. Typically, ERGMs are estimated using Monte Carlo Markov Chain (MCMC) maximum-likelihood estimation (MLE). There can be a wide variety of predictors included in the ERGMs, such as node-level predictors, dyadic predictors, relational predictors and local structure predictors. For node-level predictors, *nodefactor* and *nodecov* in R code, having a particular characteristic is hypothesized to affect the likelihood of observing a tie. Generally, the characteristics of both actors (nodes) in a dyad may influence the probability of observing a tie between them, then dyadic predictors, *nodemix*, *nodematch* and *absdiff* in R code, are found. Information about other ties (or relationships) in predicting some specific ties, so relational predictor, *edgecov* in R code, can be used. Also, information about local structural network properties can also be used as useful covariates, so local structure predictors, geometrically weighted degree distribution (*GWDegree*), geometrically weighted edgewise shared partnerships (*GWESP*) and geometrically weighted dyadwise shared partnerships (*GWDSP*), will be applied. The underlying MCMC simulation algorithm can be used to provide useful information for judging model fit.

I Data fr_w1

1. Change the network to be an undirected one (tie or no tie)

   First, we use `as.undirected` function with `mode = "collapse"` to change the network to be an undirected one, named w1, within Igraph form. Then we use function `as.matrix(get.adjacency(w1))` to get the adjacency matrix of w1. Next, we use `w1net = network(w1_matrix, matrix.type = "adjacency",directed = FALSE)` to build the undirected statnet network since `ergm` function can only apply to statnet network. Finally, we create the attributes, like `vertex.names`, `gender` and `smoke` using the values of the same attributes in igraph fr_w1. And the codes of these steps are show below:

```
w1 = as.undirected(fr_w1, mode = "collapse")

w1_matrix = as.matrix(get.adjacency(w1))
w1net = network(w1_matrix, matrix.type = "adjacency",directed = FALSE)

w1net %v% "vertex.names" <-V(w1)$vertex.names
w1net %v% "gender" <-V(w1)$gender
w1net %v% "smoke" <-V(w1)$smoke
```

Then we can see the undirected statnet network w1net:

```
##  Network attributes:
##   vertices = 37
##   directed = FALSE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges= 125
##     missing edges= 0
##     non-missing edges= 125
## Vertex attribute names:
##     gender smoke vertex.names
## No edge attributes
```

2. Model 1

We fit the w1net data with edges under the control option with seed=40, which is the null model for data w1net.

```
w1mod1 <- ergm(w1net ~ edges,control=control.ergm(seed=40))

summary(w1mod1)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##       Estimate Std. Error MCMC % z value Pr(>|z|)
## edges -1.46511    0.09924      0  -14.76   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 643.2  on 665  degrees of freedom
## AIC: 645.2    BIC: 649.7    (Smaller is better.)
```

The estimate coefficient of edges is -1.46511 with p-value smaller than 0.0001, which means the predictor, edges, is significant for this model.

3. Model 2

We fit the w1net data with edges, node-level predictor, nodefactor, for two node attributes, gender and smoke and the dyadic predictor, nodematch, under the control option with seed=40, which is the second model for data w1net.

```
w1mod2 <- ergm(w1net ~ edges + nodefactor('gender') + nodefactor('smoke') +
nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),control=con
trol.ergm(seed=40))
summary(w1mod2)
## ==========================
```

```
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                -5.13139    0.68433      0  -7.498   <1e-04 ***
## nodefactor.gender.2  0.28844    0.12093      0   2.385   0.0171 *
## nodefactor.smoke.1   0.01115    0.29119      0   0.038   0.9695
## nodematch.gender     4.23680    0.59251      0   7.151   <1e-04 ***
## nodematch.smoke      0.21447    0.34849      0   0.615   0.5383
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 467.9  on 661  degrees of freedom
## AIC: 477.9    BIC: 500.4    (Smaller is better.)
```

The estimate coefficient of *edges* is -5.13139 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.28844, with p-value being 0.0171, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.01115, with p-value being 0.9695, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 4.23680, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.21447, with p-value being 0.5383, which means the predictor, *nodematch.smoke*, is not significant for this model.

4. Model 3

We fit the w1net data with edges, node-level predictor, nodefactor, for gender and smoke two node attributes, the dyadic predictor, nodematch with diff=FALSE, and local structure predictor, gwesp, with option alpha=0.7 and fixed=TRUE, under the control option with seed=40, which is the third model for data w1net.

```
w1mod3 <- ergm(w1net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),
   control=control.ergm(seed=40))
summary(w1mod3)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                -5.66572    0.63351      0  -8.943  < 1e-04 ***
## nodefactor.gender.2  0.22353    0.07726      0   2.893  0.00381 **
## nodefactor.smoke.1   0.09210    0.28854      0   0.319  0.74958
## nodematch.gender     2.39416    0.53118      0   4.507  < 1e-04 ***
```

```
## nodematch.smoke      0.24109    0.39951      0   0.603  0.54620
## gwesp.fixed.0.7      0.92982    0.17528      0   5.305  < 1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 442.9  on 660  degrees of freedom
## AIC: 454.9    BIC: 481.9    (Smaller is better.)
```

The estimate coefficient of *edges* is -5.66572 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.22353, with p-value being 0.00381, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.09210 with p-value being 0.74958, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 2.39416, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.24109, with p-value being 0.54620, which means the predictor, *nodematch.smoke*, is not significant for this model.

The estimate coefficient of *gwesp.fixed.0.7* is 0.92982, with p-value smaller than 0.0001, which means the predictor, *gwesp.fixed.0.7*, is significant for this model.

Then for these three models we compare the AIC and BIC as below:

Table 1 The AICs and BICs of three model for data 1

	Model 1	Model 2	Model 3
AIC	645.2	477.9	454.9
BIC	649.7	500.4	481.9

From the result we can see, the model 3 obtained the smallest AIC and BIC among these three models. So model 3 is the best model for data 1.

II Data fr_w2

1. Change the network to be an undirected one (tie or no tie)

First, we use `as.undirected` function with `mode = "collapse"` to change the network to be an undirected one, named w2, within Igraph form. Then we use function `as.matrix(get.adjacency(w2))` to get the adjacency matrix of w2. Next, we use `w2net = network(w2_matrix, matrix.type = "adjacency",directed = FALSE)` to build the undirected statnet network since `ergm` function can only apply to statnet network. Finally, we create the attributes, like `vertex.names`, `gender` and `smoke` using the values of the same attributes in igraph fr_w2. And the codes of these steps are show below:

```
w2 = as.undirected(fr_w2, mode = "collapse")

w2_matrix = as.matrix(get.adjacency(w2))
w2net = network(w2_matrix, matrix.type = "adjacency",directed = FALSE)
```

```r
w2net %v% "vertex.names" <-V(w2)$vertex.names
w2net %v% "gender" <-V(w2)$gender
w2net %v% "smoke" <-V(w2)$smoke
```

Then we can see the undirected statnet network w2net:

```
##  Network attributes:
##   vertices = 37
##   directed = FALSE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges= 127
##     missing edges= 0
##     non-missing edges= 127
##  Vertex attribute names:
##     gender smoke vertex.names
## No edge attributes
```

2. Model 1

We fit the w2net data with edges under the control option with seed=40, which is the null model for data w2net.

```r
w2mod1 <- ergm(w2net ~ edges,control=control.ergm(seed=40))

summary(w2mod1)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##       Estimate Std. Error MCMC % z value Pr(>|z|)
## edges -1.44553    0.09864      0  -14.65   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 649.0  on 665  degrees of freedom
## AIC: 651    BIC: 655.5    (Smaller is better.)
```

The estimate coefficient of edges is -1.44553 with p-value smaller than 0.0001, which means the predictor, edges, is significant for this model.

3. Model 2

We fit the w2net data with edges, node-level predictor, nodefactor, for two node attributes, gender and smoke and the dyadic predictor, nodematch, under the control option with seed=40, which is the second model for data w2net.

```r
w2mod2 <- ergm(w2net ~ edges + nodefactor('gender') + nodefactor('smoke') +
nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),control=con
trol.ergm(seed=40))
summary(w2mod2)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                     Estimate Std. Error MCMC % z value Pr(>|z|)
## edges               -4.75522    0.54445      0  -8.734   <1e-04 ***
## nodefactor.gender.2  0.29047    0.12314      0   2.359   0.0183 *
## nodefactor.smoke.1  -0.05896    0.22267      0  -0.265   0.7912
## nodematch.gender     3.74819    0.46738      0   8.020   <1e-04 ***
## nodematch.smoke      0.47655    0.28675      0   1.662   0.0965 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 479.9  on 661  degrees of freedom
## AIC: 489.9    BIC: 512.4    (Smaller is better.)
```

The estimate coefficient of *edges* is -4.75522 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.29047, with p-value being 0.0183, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is -0.05896, with p-value being 0.7912, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 3.74819, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.47665, with p-value being 0.0965, which means the predictor, *nodematch.smoke*, is not significant for this model.

4. Model 3

We fit the w1net data with edges, node-level predictor, nodefactor, for gender and smoke two node attributes, the dyadic predictor, nodematch with diff=FALSE, and local structure predictor, gwesp, with option alpha=0.7 and fixed=TRUE, under the control option with seed=40, which is the third model for data w2net.

```r
w2mod3 <- ergm(w2net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),control=control.ergm(seed=40))
summary(w2mod3)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                     Estimate Std. Error MCMC % z value Pr(>|z|)
```

```
## edges                -5.61828    0.47518      0 -11.824   <1e-04 ***
## nodefactor.gender.2  0.21629    0.08687      0   2.490   0.0128 *
## nodefactor.smoke.1   0.08886    0.16529      0   0.538   0.5908
## nodematch.gender     2.05300    0.37088      0   5.536   <1e-04 ***
## nodematch.smoke      0.47879    0.26856      0   1.783   0.0746 .
## gwesp.fixed.0.7      1.01180    0.19398      0   5.216   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 447.7  on 660  degrees of freedom
## AIC: 459.7    BIC: 486.7    (Smaller is better.)
```

The estimate coefficient of *edges* is -5.661828 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.21629, with p-value being 0.0128, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.08886 with p-value being 0.5908, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 2.05300, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.47879, with p-value being 0.0746, which means the predictor, *nodematch.smoke*, is not significant for this model.

The estimate coefficient of *gwesp.fixed.0.7* is 1.01180, with p-value smaller than 0.0001, which means the predictor, *gwesp.fixed.0.7*, is significant for this model.

Then for these three models we compare the AIC and BIC as below:

Table 2 The AICs and BICs of three model for data 2

	Model 1	Model 2	Model 3
AIC	651	489.9	459.7
BIC	655.5	512.4	486.7

From the result we can see, the model 3 obtained the smallest AIC and BIC among these three models. So model 3 is the best model for data 2.

III Data fr_w3

1. Change the network to be an undirected one (tie or no tie)

First, we use `as.undirected` function with `mode = "collapse"` to change the network to be an undirected one, named w3, within Igraph form. Then we use function `as.matrix(get.adjacency(w3))` to get the adjacency matrix of w3. Next, we use `w3net = network(w3_matrix, matrix.type = "adjacency",directed = FALSE)` to build the undirected statnet network since `ergm` function can only apply to statnet network. Finally, we create the attributes, like `vertex.names`, `gender` and `smoke` using the values of the same attributes in igraph fr_w3. And the codes of these steps are show below:

```
w3 = as.undirected(fr_w3, mode = "collapse")

w3_matrix = as.matrix(get.adjacency(w3))
w3net = network(w3_matrix, matrix.type = "adjacency",directed = FALSE)
```

```r
w3net %v% "vertex.names" <-V(w3)$vertex.names
w3net %v% "gender" <-V(w3)$gender
w3net %v% "smoke" <-V(w3)$smoke
```

Then we can see the undirected statnet network w3net:

```
##  Network attributes:
##   vertices = 37
##   directed = FALSE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges= 130
##     missing edges= 0
##     non-missing edges= 130
## Vertex attribute names:
##     gender smoke vertex.names
## No edge attributes
```

2. Model 1

We fit the w3net data with edges under the control option with seed=40, which is the null model for data w3net.

```r
w3mod1 <- ergm(w3net ~ edges,control=control.ergm(seed=40))

summary(w3mod1)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##       Estimate Std. Error MCMC % z value Pr(>|z|)
## edges -1.41660    0.09776      0  -14.49   <1e-04 ***
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 657.6  on 665  degrees of freedom
## AIC: 659.6    BIC: 664.1    (Smaller is better.)
```

The estimate coefficient of edges is -1.44660 with p-value smaller than 0.0001, which means the predictor, edges, is significant for this model.

3. Model 2

We fit the w3net data with edges, node-level predictor, nodefactor, for two node attributes, gender and smoke and the dyadic predictor, nodematch, under the control option with seed=40, which is the second model for data w3net.

```r
w3mod2 <- ergm(w3net ~ edges + nodefactor('gender') + nodefactor('smoke') +
nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE), control=co
```

```r
ntrol.ergm(seed=40))
summary(w3mod2)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                 -4.9921     0.5090      0  -9.807  < 1e-04 ***
## nodefactor.gender.2   0.3190     0.1232      0   2.590 0.009611 **
## nodefactor.smoke.1    0.1504     0.1883      0   0.798 0.424588
## nodematch.gender      3.6412     0.4324      0   8.422  < 1e-04 ***
## nodematch.smoke       0.9043     0.2601      0   3.476 0.000508 ***
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 482.9  on 661  degrees of freedom
## AIC: 492.9    BIC: 515.4    (Smaller is better.)
```

The estimate coefficient of *edges* is -4.9921 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.3190, with p-value being 0.009611, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.1504, with p-value being 0.424588, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 3.6412, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.9043, with p-value being 0.000508, which means the predictor, *nodematch.smoke*, is significant for this model.

4. Model 3

We fit the w3net data with edges, node-level predictor, nodefactor, for gender and smoke two node attributes, the dyadic predictor, nodematch with diff=FALSE, and local structure predictor, gwesp, with option alpha=0.7 and fixed=TRUE, under the control option with seed=40, which is the third model for data w3net.

```r
w3mod3 <- ergm(w3net ~ edges + nodefactor('gender') + nodefactor('smoke') +
nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),control=control.ergm(seed=40))
summary(w3mod3)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                -5.73005    0.44862      0 -12.773  < 1e-04 ***
## nodefactor.gender.2  0.24337    0.09519      0   2.557 0.010567 *
## nodefactor.smoke.1   0.20977    0.13386      0   1.567 0.117106
```

```
## nodematch.gender     2.12840    0.36289      0   5.865  < 1e-04 ***
## nodematch.smoke      0.82233    0.21960      0   3.745 0.000181 ***
## gwesp.fixed.0.7      0.90520    0.18836      0   4.806  < 1e-04 ***
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 456.5  on 660  degrees of freedom
## AIC: 468.5    BIC: 495.6    (Smaller is better.)
```

The estimate coefficient of *edges* is -5.73005 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.24337, with p-value being 0.010567, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.20977 with p-value being 0.117106, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 2.12840, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.82233, with p-value being 0.000181, which means the predictor, *nodematch.smoke*, is significant for this model.

The estimate coefficient of *gwesp.fixed.0.7* is 0.90520, with p-value smaller than 0.0001, which means the predictor, *gwesp.fixed.0.7*, is significant for this model.

Then for these three models we compare the AIC and BIC as below:

Table 3 The AICs and BICs of three model for data 3

	Model 1	Model 2	Model 3
AIC	659.6	492.9	468.5
BIC	664.1	515.4	495.6

From the result we can see, the model 3 obtained the smallest AIC and BIC among these three models. So model 3 is the best model for data 3.

IV Data fr_w4

1. Change the network to be an undirected one (tie or no tie)

First, we use `as.undirected` function with `mode = "collapse"` to change the network to be an undirected one, named w4, within Igraph form. Then we use function `as.matrix(get.adjacency(w4))` to get the adjacency matrix of w4. Next, we use `w4net = network(w4_matrix, matrix.type = "adjacency",directed = FALSE)` to build the undirected statnet network since `ergm` function can only apply to statnet network. Finally, we create the attributes, like `vertex.names`, `gender` and `smoke` using the values of the same attributes in igraph fr_w4. And the codes of these steps are show below:

```
w4 = as.undirected(fr_w4, mode = "collapse")

w4_matrix = as.matrix(get.adjacency(w4))
w4net = network(w4_matrix, matrix.type = "adjacency",directed = FALSE)
```

```
w4net %v% "vertex.names" <-V(w4)$vertex.names
w4net %v% "gender" <-V(w4)$gender
w4net %v% "smoke" <-V(w4)$smoke
```

Then we can see the undirected statnet network w4net:

```
##  Network attributes:
##   vertices = 37
##   directed = FALSE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges= 133
##     missing edges= 0
##     non-missing edges= 133
##  Vertex attribute names:
##     gender smoke vertex.names
## No edge attributes
```

2. Model 1

We fit the w4net data with edges under the control option with seed=40, which is the null model for data w4net.

```
w4mod1 <- ergm(w4net ~ edges,control=control.ergm(seed=40))

summary(w4mod1)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##       Estimate Std. Error MCMC % z value Pr(>|z|)
## edges -1.38817    0.09693      0  -14.32   <1e-04 ***
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 666.0  on 665  degrees of freedom
## AIC: 668    BIC: 672.5    (Smaller is better.)
```

The estimate coefficient of edges is -1.38817 with p-value smaller than 0.0001, which means the predictor, edges, is significant for this model.

3. Model 2

We fit the w4net data with edges, node-level predictor, nodefactor, for two node attributes, gender and smoke and the dyadic predictor, nodematch, under the control option with seed=40, which is the second model for data w4net.

```
w4mod2 <- ergm(w4net ~ edges + nodefactor('gender') + nodefactor('smoke') +
nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),control=con
trol.ergm(seed=40))
summary(w4mod2)
## ==========================
```

```
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                 -5.0763     0.5051      0 -10.049  < 1e-04 ***
## nodefactor.gender.2    0.2541     0.1231      0   2.063 0.039075 *
## nodefactor.smoke.1     0.3214     0.1692      0   1.899 0.057504 .
## nodematch.gender       3.6871     0.4342      0   8.492  < 1e-04 ***
## nodematch.smoke        0.9242     0.2397      0   3.856 0.000115 ***
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 485.8  on 661  degrees of freedom
## AIC: 495.8    BIC: 518.3    (Smaller is better.)
```

The estimate coefficient of *edges* is -5.0763 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.2541, with p-value being 0.039075, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.3214, with p-value being 0.057504, which means the predictor, *nodefactor.smoke.1*, is not significant for this model.

The estimate coefficient of *nodematch.gender* is 3.6871, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.9242, with p-value being 0.000115, which means the predictor, *nodematch.smoke*, is significant for this model.

4. Model 3

We fit the w4net data with edges, node-level predictor, nodefactor, for gender and smoke two node attributes, the dyadic predictor, nodematch with diff=FALSE, and local structure predictor, gwesp, with option alpha=0.7 and fixed=TRUE, under the control option with seed=40, which is the third model for data w4net.

```
w4mod3 <- ergm(w4net ~ edges + nodefactor('gender') + nodefactor('smoke') +
nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE), control=control.ergm(seed=40))
summary(w4mod3)
## ==========================
## Summary of model fit
## ==========================
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                -5.89436    0.44252      0 -13.320   <1e-04 ***
## nodefactor.gender.2   0.19569    0.07793      0   2.511   0.0120 *
## nodefactor.smoke.1    0.24465    0.10862      0   2.252   0.0243 *
## nodematch.gender      2.02448    0.33909      0   5.970   <1e-04 ***
## nodematch.smoke       0.81565    0.20380      0   4.002   <1e-04 ***
## gwesp.fixed.0.7       1.03217    0.19176      0   5.382   <1e-04 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##      Null Deviance: 923.3  on 666  degrees of freedom
##  Residual Deviance: 455.4  on 660  degrees of freedom
## AIC: 467.4    BIC: 494.4    (Smaller is better.)
```

The estimate coefficient of *edges* is -5.89436 with p-value smaller than 0.0001, which means the predictor, *edges*, is significant for this model.

The estimate coefficient of *nodefactor.gender.2* is 0.19569, with p-value being 0.0120, which means the predictor, *nodefactor.gender.2,* is significant for this model.

The estimate coefficient of *nodefactor.smoke.1* is 0.24465 with p-value being 0.0243, which means the predictor, *nodefactor.smoke.1*, is significant for this model.

The estimate coefficient of *nodematch.gender* is 2.02448, with p-value smaller than 0.0001, which means the predictor, *nodematch.gender*, is significant for this model.

The estimate coefficient of *nodematch.smoke* is 0.81565, with p-value smaller than 0.0001, which means the predictor, *nodematch.smoke*, is significant for this model.

The estimate coefficient of *gwesp.fixed.0.7* is 1.03217, with p-value smaller than 0.0001, which means the predictor, *gwesp.fixed.0.7*, is significant for this model.

Then for these three models we compare the AIC and BIC as below:

Table 4 The AICs and BICs of three model for data 4

	Model 1	Model 2	Model 3
AIC	688	495.8	467.4
BIC	672.5	518.3	494.4

From the result we can see, the model 3 obtained the smallest AIC and BIC among these three models. So model 3 is the best model for data 4.

V Model 3 for 4 data sets

We use the results of model 3 from previous steps and compare the estimates and p-values among the results for four data sets. The results are shown below:

Table 5 The estimates and p-values of model #3 for four data stes

	Data fr_w1	Data fr_w2	Data fr_w3	Data fr_w4
edges	-5.66572 (<0.0001) ***	-5.6182 (<0.0001) ***	-5.73005 (<0.0001) ***	-5.89436 (<0.0001) ***
nodefactor.gender.2	0.22353 (0.00381) **	0.21629 (0.0128) *	0.24337 (0.010567) *	0.19569 (0.0120) *
nodefactor.smoke.1	0.09210 (0.74958)	0.08886 (0.5980)	0.20977 (0.11706)	0.24465 (0.0243) *
nodematch.gender	2.39416 (<0.0001) ***	2.05300 (<0.0001) ***	2.12840 (<0.0001) ***	2.02448 (<0.0001) ***
nodematch.smoke	0.24109 (0.54620)	0.47879 (0.0746) .	0.82233 (0.000181) ***	0.81565 (<0.0001) ***
gwesp.fixed.0.7	0.92980 (<0.0001) ***	1.01180 (<0.0001) ***	0.90520 (<0.0001) ***	1.03217 (<0.0001) ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Typically, the estimate coefficients of *edges* for four data sets are similar with all p-values smaller than 0.0001, which means the predictor, *edges*, is significant for model#3 in all four data sets

The estimates coefficients of *nodefactor.gender.2* for four data sets are around 0.2, with all p-values less than 0.05, which means the predictor, *nodefactor.gender.2,* is significant for model#3 in all four data sets

The estimate coefficients of *nodefactor.smoke.1* for first three data sets are 0.09210, 0.08886 and 0.20977, respectively, with all p-values greater than 0.05, which means the predictor, *nodefactor.smoke.1*, is not significant for model#3 in first three data sets. However, the estimates coefficient of *nodefactor.smoke.1* for the fourth data set is 0.24465, with all p-values less than 0.05, which means the predictor, *nodefactor.smoke.1*, is significant for model#3 in the fourth data set.

The estimate coefficients of *nodematch.gender* are 2.39416, 2.0530, 2.1284 and 2.02448 respectively, with all p-values less than 0.001, which means the predictor, *nodematch.gender*, is significant for model#3 in all four data sets

The estimate coefficients of *nodematch.smoke* for first two data sets are 0.24109 and 0.47879 respectively, with all p-values greater than 0.05, which means the predictor, *nodematch.smoke*, is not significant for model#3 in first two data sets. However, the estimate coefficients of *nodematch.smoke* for the other two data sets are 0.82233 and 0.81565 respectively, with all p-values less than 0.001, which means the predictor, *nodematch.smoke*, is significant for model#3 in the other two data sets.

The estimate coefficients of *gwesp.fixed.0.7* for four data sets are around 1 with all p-values smaller than 0.0001, which means the predictor, *gwesp.fixed.0.7*, is significant for model#3 in all four data sets

Appendix

```
library(igraph)
library("UserNetR")
library(ergm)

data(Coevolve) # Load the data
fr_w1 <- Coevolve$fr_w1 # Get the data 1
fr_w2 <- Coevolve$fr_w2 # Get the data 2
fr_w3 <- Coevolve$fr_w3 # Get the data 3
fr_w4 <- Coevolve$fr_w4 # Get the data 4
```

Data fr_w1

```
fr_w1 # Show the data 1

w1 = as.undirected(fr_w1, mode = "collapse") # Transform the originaL direc
ted graph to the undirected graph, with option where one undirected edge wi
ll be created for each pair of vertices which are connected with at least o
ne directed edge, no multiple edges will be created
w1_matrix = as.matrix(get.adjacency(w1)) # Get the adjacency matraix and tr
ansform it into matrix form storing in w1_matrix
w1net = network(w1_matrix, matrix.type = "adjacency",directed = FALSE) # Cr
eate the undirected network by adjacency matrix witn function network
w1net %v% "vertex.names" <-V(w1)$vertex.names # Set the w1net's node attrib
ute named "vertex.names" using vakues from the w1
w1net %v% "gender" <-V(w1)$gender # Set the w1net's node attribute named "g
ender" using vakues from the w1
w1net %v% "smoke" <-V(w1)$smoke # Set the w1net's node attribute named ""sm
oke" using vakues from the w1
w1net # Show the w1net
```

model 1

```
w1mod1 <- ergm(w1net ~ edges,  control=control.ergm(seed=40))
# Fitting the w1net with edges predictors and set the seed=40 to ensure the
 same results over multiple runs.
summary(w1mod1) # Show the fitting results
```

model 2

```
w1mod2 <- ergm(w1net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),
   control=control.ergm(seed=40))
# Fitting the w1net with edges predictors, nodel-level predictor, nodefacto
r, for gender and smoke and dyadic predictor, nodematch, for gender and smo
ke, with seting the seed=40 to ensure the same results over multiple runs.
```

```
summary(w1mod2) # Show the fitting results
```

model 3

```
w1mod3 <- ergm(w1net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),
   control=control.ergm(seed=40))# Fitting the w1net with edges predictors,
 nodel-level predictor, nodefactor, for gender and smoke, dyadic predictor,
 nodematch, for gender and smoke and local structure predictor, gwesp, with
 alpha = 0.7 as well as fixed = TRUE, along with seting the seed=40 to ensur
e the same results over multiple runs.
summary(w1mod3) #Show the  fitting result
```

Data fr_w2

```
fr_w2 # Show the data 2

w2 = as.undirected(fr_w2, mode = "collapse") # Transform the originaL direc
ted graph to the undirected graph, with option where one undirected edge wi
ll be created for each pair of vertices which are connected with at least o
ne directed edge, no multiple edges will be created
w2_matrix = as.matrix(get.adjacency(w2)) # Get the adjacency matraix and tr
ansform it into matrix form storing in w2_matrix
w2net = network(w2_matrix, matrix.type = "adjacency",directed = FALSE) # Cr
eate the undirected network by adjacency matrix witn function network
w2net %v% "vertex.names" <-V(w2)$vertex.names # Set the w1net's node attrib
ute named "vertex.names" using vakues from the w2
w2net %v% "gender" <-V(w2)$gender # Set the w1net's node attribute named "g
ender" using vakues from the w2
w2net %v% "smoke" <-V(w2)$smoke # Set the w1net's node attribute named ""sm
oke" using vakues from the w2
w2net # Show the w1net
```

model 1

```
w2mod1 <- ergm(w2net ~ edges,  control=control.ergm(seed=40))
# Fitting the w2net with edges predictors and set the seed=40 to ensure the
 same results over multiple runs.
summary(w2mod1) # Show the fitting results
```

model 2

```
w2mod2 <- ergm(w2net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),
```

```
   control=control.ergm(seed=40))
# Fitting the w2net with edges predictors, nodel-level predictor, nodefacto
r, for gender and smoke and dyadic predictor, nodematch, for gender and smo
ke, with seting the seed=40 to ensure the same results over multiple runs.

summary(w2mod2) # Show the fitting results
```

model 3

```
w2mod3 <- ergm(w2net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),
   control=control.ergm(seed=40))# Fitting the w2net with edges predictors,
 nodel-level predictor, nodefactor, for gender and smoke, dyadic predictor,
 nodematch, for gender and smoke and local structure predictor, gwesp, with
 alpha = 0.7 as well as fixed = TRUE, along with seting the seed=40 to ensur
e the same results over multiple runs.
summary(w2mod3) #Show the  fitting result
```

Data fr_w3

```
fr_w3 # Show the data 3

w3 = as.undirected(fr_w3, mode = "collapse") # Transform the original direc
ted graph to the undirected graph, with option where one undirected edge wi
ll be created for each pair of vertices which are connected with at least o
ne directed edge, no multiple edges will be created
w3_matrix = as.matrix(get.adjacency(w3)) # Get the adjacency matraix and tr
ansform it into matrix form storing in w3_matrix
w3net = network(w3_matrix, matrix.type = "adjacency",directed = FALSE) # Cr
eate the undirected network by adjacency matrix witn function network
w3net %v% "vertex.names" <-V(w3)$vertex.names # Set the w1net's node attrib
ute named "vertex.names" using vakues from the w3
w3net %v% "gender" <-V(w3)$gender # Set the w1net's node attribute named "g
ender" using vakues from the w3
w3net %v% "smoke" <-V(w3)$smoke # Set the w1net's node attribute named ""sm
oke" using vakues from the w3
w3net # Show the w1net
```

model 1

```
w3mod1 <- ergm(w3net ~ edges,  control=control.ergm(seed=40))
# Fitting the w3net with edges predictors and set the seed=40 to ensure the
 same results over multiple runs.
summary(w3mod1) # Show the fitting results
```

model 2

```
w3mod2 <- ergm(w3net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),
   control=control.ergm(seed=40))
# Fitting the w3net with edges predictors, nodel-level predictor, nodefacto
r, for gender and smoke and dyadic predictor, nodematch, for gender and smo
ke, with seting the seed=40 to ensure the same results over multiple runs.

summary(w3mod2) # Show the fitting results
```

model 3

```
w3mod3 <- ergm(w3net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),
   control=control.ergm(seed=40))# Fitting the w3net with edges predictors,
 nodel-level predictor, nodefactor, for gender and smoke, dyadic predictor,
 nodematch, for gender and smoke and local structure predictor, gwesp, with
 alpha = 0.7 as well as fixed = TRUE, along with seting the seed=40 to ensur
e the same results over multiple runs.
summary(w3mod3) #Show the  fitting result
```

Data fr_w4

```
fr_w4 # Show the data 4

w4 = as.undirected(fr_w4, mode = "collapse") # Transform the original direc
ted graph to the undirected graph, with option where one undirected edge wi
ll be created for each pair of vertices which are connected with at least o
ne directed edge, no multiple edges will be created
w4_matrix = as.matrix(get.adjacency(w4)) # Get the adjacency matraix and tr
ansform it into matrix form storing in w4_matrix
w4net = network(w4_matrix, matrix.type = "adjacency",directed = FALSE) # Cr
eate the undirected network by adjacency matrix witn function network
w4net %v% "vertex.names" <-V(w4)$vertex.names # Set the w1net's node attrib
ute named "vertex.names" using vakues from the w4
w4net %v% "gender" <-V(w4)$gender # Set the w1net's node attribute named "g
ender" using vakues from the w4
w4net %v% "smoke" <-V(w4)$smoke # Set the w1net's node attribute named ""sm
oke" using vakues from the w4
w4net # Show the w1net
```

model 1

```
w4mod1 <- ergm(w4net ~ edges,  control=control.ergm(seed=40))
# Fitting the w4net with edges predictors and set the seed=40 to ensure the
 same results over multiple runs.
summary(w4mod1) # Show the fitting results
```

model 2

```
w4mod2 <- ergm(w4net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE),
   control=control.ergm(seed=40))
# Fitting the w4net with edges predictors, nodel-level predictor, nodefacto
r, for gender and smoke and dyadic predictor, nodematch, for gender and smo
ke, with seting the seed=40 to ensure the same results over multiple runs.

summary(w4mod2) # Show the fitting results
```

model 3

```
w4mod3 <- ergm(w4net ~ edges + nodefactor('gender') + nodefactor('smoke') +
 nodematch('gender',diff=FALSE) + nodematch('smoke',diff=FALSE) + gwesp(0.
7, fixed=TRUE),
   control=control.ergm(seed=40))# Fitting the w4net with edges predictors,
 nodel-level predictor, nodefactor, for gender and smoke, dyadic predictor,
 nodematch, for gender and smoke and local structure predictor, gwesp, with
 alpha = 0.7 as well as fixed = TRUE, along with seting the seed=40 to ensur
e the same results over multiple runs.
summary(w4mod3) #Show the  fitting result



summary(w1mod3) #Show the  fitting result of model3 for data 1

summary(w2mod3)#Show the  fitting result of model3 for data 2

summary(w3mod3)#Show the  fitting result of model3 for data 3

summary(w4mod3)#Show the  fitting result of model3 for data 4
```