QuantilePeer: An R package for Simulating and Estimation Quantile Peer Effect Models

Aristide Houndetoungan

2026-10-31

Contents

1 Introduction **2**

2 A brief description of the model **2**

3 Estimating quantile peer effects **3**
3.1 Data simulation 4
3.2 Instruments 5
3.3 Estimation 6
3.4 Specification tests 13

4 CES-based peer effect models **16**
4.1 A brief description 16
4.2 Data simulation 16
4.3 Estimation 18

5 Other specifications **19**

6 Conclusion **21**

1 Introduction

This vignette provides a quick introduction to the **QuantilePeer** package. The package is designed to estimate quantile peer effect models, allowing researchers to assess peer effects at multiple quantiles of the peer outcome distribution (Houndetoungan, 2025). This modeling approach is more flexible than the standard linear-in-means (LIM) model, which assumes that all peers exert the same influence on individuals. The quantile peer effect model allows peer influence to vary across different quantiles of the outcome distribution, making it a useful framework for testing the assumptions underlying the LIM model.

The package also provides routines for estimating peer effect models with a constant elasticity of substitution (CES) social norm (Boucher et al., 2024). The CES-based approach includes a substitution parameter that allows peers with higher or lower outcomes to exert more influence on individual outcomes.

This documentation is organized as follows. Section 2 provides a brief overview of quantile peer effect models. Section 3 illustrates how to simulate and estimate these models. Section 4 presents, simulates, and estimates CES-based peer effect models. Section 5 discusses additional model specifications that can also be estimated using this package.

To cite **QuantilePeer**, kindly run `citation("QuantilePeer")` in **R**. This will display the citation information, including the BibTeX entry for LaTeX users. Please also cite the paper associated with the package (Houndetoungan, 2025).

2 A brief description of the model

Let $\mathcal{N}$ be a set of n agents indexed by the integer $i \in [1,\ n]$. Agents are connected through a network that is characterized by an adjacency matrix $\mathbf{A} = [a_{ij}]$ of dimension $n \times n$, where $a_{ij} = 1$ if agent j is a friend of agent i, and $a_{ij} = 0$ otherwise. In weighted networks, a_{ij} can be a nonnegative variable (not necessarily binary) that measures the intensity of the outgoing link from i to j. The model can also accommodate such networks. Note that the network is generally constituted in many independent subnets (eg: schools).

Let $\mathcal{T}$ be a set of quantile levels. The reduced-form specification of the quantile peer effect models is given by:

$$y_i = \sum_{\tau \in \mathcal{T}} \lambda_\tau q_{\tau,i}(\mathbf{y}_{-i}) + \boldsymbol{x}_i'\boldsymbol{\gamma} + \varepsilon_i, \tag{1}$$

where $\mathbf{y}_{-i} = (y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n)'$ is the vector of outcomes for individuals other than i, and $q_{\tau,i}(\mathbf{y}_{-i})$ denotes the sample τ-quantile of peer outcomes. The term ε_i is an idiosyncratic error term, λ_τ captures the effect of the τ-quantile of peer outcomes on y_i, and $\boldsymbol{\gamma}$ captures the effect of the exogenous variable $\boldsymbol{x}_i$ on y_i. Hyndman and Fan (1996) distinguish nine types of quantiles. The results developed in the paper hold for all these types. However, both

the simulations and empirical analysis use *Type 7*, which relies on linear interpolation when the quantile level does not correspond exactly to a peer's rank.[1] If the network matrix is weighted, a sample *weighted quantile* can be used, where the outcome for friend j of individual i is weighted by a_{ij}.

One issue in linear peer effect models is that individual preferences with conformity and spillover (complementarity or substitution) lead to the same reduced form (see Boucher and Fortin, 2016). However, it is possible to disentangle both types of preferences using isolated individuals (Boucher et al., 2024). Isolated individuals are those who have no friends, although they may or may not be considered friends by others. The structural specification of the model differs between isolated and non-isolated individuals, allowing for the separate identification of peer effects arising from spillovers and conformity.

For isolated i, the specification is similar to a standard linear-in-means (LIM) model without social interactions, given by:

$$y_i = \mathbf{x}_i'\boldsymbol{\gamma} + \varepsilon_i. \tag{2}$$

If i is non-isolated, the specification is given by:

$$y_i = \sum_{\tau\in\mathcal{T}} \lambda_\tau q_{\tau,i}(\mathbf{y}_{-i}) + (1-\lambda_2)\mathbf{x}_i'\boldsymbol{\gamma} + \varepsilon_i, \tag{3}$$

where λ_2 determines whether preferences exhibit conformity or anti-conformity. Specifically, let $\lambda := \sum_{\tau\in\mathcal{T}} \lambda_\tau$ denote the total peer effects at all quantile levels. These total effects can be decomposed as $\lambda = \lambda_1 + \lambda_2$, where λ_1 captures total spillover effects and λ_2 captures total conformity effects. As in Boucher et al. (2024), when $\lambda_1 > 0$, preferences exhibit complementarity; when $\lambda_1 < 0$, preferences exhibit substitution. In contrast, when $\lambda_2 > 0$, preferences are conformist. Anti-conformity may also arise when $\lambda_2 < 0$. If peer effects are solely due to spillovers, then $\lambda_1 \neq 0$ and $\lambda_2 = 0$. Conversely, if peer effects arise only from conformity, then $\lambda_1 = 0$ and $\lambda_2 \neq 0$. The quantile peer effect model allows for the decomposition of total peer effects at different quantile levels, measured by the parameters λ_τ's.

To identify λ_1 and λ_2, it is important to observe a sufficient number of isolated individuals in the network. This enables the identification of $\boldsymbol{\gamma}$ in Equation (2), which in turn makes it possible to identify λ_2 in Equation (3). The underlying assumption here is that the $\boldsymbol{\gamma}$ parameter is the same in both equations. Alternatively, assuming that a subset of the components of $\boldsymbol{\gamma}$ are the same is sufficient for the identification of λ_2. When the network contains no isolated nodes, the λ_τ's can still be identified, but it is not possible to determine whether they reflect conformity or spillover effects.

3 Estimating quantile peer effects

The key functions discussed in this section include:

[1] For example, when an agent has only two friends, the sample median of peer outcomes is simply the average of the two friends' outcomes. The first decile is a weighted average of the two outcomes, where the friend with the lower outcome receives a weight of 0.9.

1. `qpeer.sim`: simulates data from models with quantile peer effects;
2. `qpeer.inst`: computes instruments for models with quantile peer effects;
3. `qpeer`, `linpeer`, and `genpeer`: estimate models with quantile peer effects;
4. `qpeer.test`: performs specification tests.

Most of these functions are also implemented as classes, with associated `summary` and `print` methods.

3.1 Data simulation

Throughout this documentation, I use simulated data. To begin, I first create a network matrix $\mathbf{A}$ and two exogenous variables, x_1 and x_2. Importantly, I include some isolated nodes to ensure the identification of the structural model.

```
library(QuantilePeer)
set.seed(123)  # Set seed for reproducibility
ngr  <- 50  # Number of subnets
nvec <- rep(30, ngr)  # Size of subnets
n    <- sum(nvec)

# Network matrix
A <- lapply(1:ngr, function(g) {
  Ag <- matrix(rbinom(nvec[g]^2, 1, 0.3), nvec[g], nvec[g])
  diag(Ag) <- 0
  # Adding isolated nodes (important for the structural model)
  niso <- sample(0:nvec[g], 1, prob = (nvec[g] + 1):1 / sum((nvec[g] + 1):1))
  if (niso > 0) {
    Ag[sample(1:nvec[g], niso), ] <- 0
  }
  Ag
})

X   <- cbind(rnorm(n), rpois(n, 2)); colnames(X) <- c("X1", "X2")
```

Using the network matrix and the exogenous variables, I can now generate the dependent variables. I consider the quantile levels $(0,\ 1/3,\ 2/3,\ 1)$, which define four quantiles, and two types of dependent variables. The first is generated from the reduced-form model (without a conformity parameter), while the second is based on the structural model. The following code assigns values to the model parameters and simulates the corresponding dependent variables.

```
tau       <- seq(0, 1, 1/3) #quantile level
lambdatau <- c(0.1, 0.25, 0.2, 0.15) #lambda_tau
```

```r
lambda2   <- 0.2 # lambda_2
gamma     <- c(2, -0.5, 1)

# First dependent variable (reduced form without conformity)
y1        <- qpeer.sim(formula = ~ X, A = A, tau = tau, lambda = lambdatau,
                       gamma = gamma, structural = FALSE,
                       epsilon = rnorm(n, 0, 0.4))
y1        <- y1$y #qpeer.sim returns a list of several object including y

# Second dependent variable (structural form with conformity)
y2        <- qpeer.sim(formula = ~ X, A = A, tau = tau,
                       lambda = c(lambda2, lambdatau), gamma = gamma,
                       structural = TRUE, epsilon = rnorm(n, 0, 0.4))
y2        <- y2$y
```

Note that we can also include contextual variables, such as averages of x among peers, as additional exogenous variables.

3.2 Instruments

I propose two instrument sets for quantile peer outcomes. The first type of instruments ($\mathbf{Z}_1$) is the set of quantiles of x among peers. The second type ($\mathbf{Z}_2$) also consists of quantiles of x among peers, but with a key distinction that the values of x are ordered using the values of the peers' dependent variable (see the detailed discussion in the paper). The second type of instruments can yield more efficient estimators but may also be endogenous. Their validity can be tested using a procedure similar to that of Hausman (1978). It is also possible to combine $\mathbf{Z}_1$ and $\mathbf{Z}_2$ to strengthen the instruments.

Instruments can be computed using `qpeer.inst`. The type of instruments is specified through the `formula` argument. If `formula` is defined without a dependent variable (i.e., an expression of the form `~ X1 + X2 + ...`), then the first type of instruments is computed. In contrast, if `formula` includes a dependent variable (i.e., an expression of the form `y ~ X1 + X2 + ...`), then the second type of instruments is computed.

For the first type of instruments, it is important to use a finer subdivision of quantile levels than τ. By including many quantile levels of the characteristics x, one obtains a comprehensive representation of the distribution of peer characteristics x, which helps effectively approximate their outcomes and the quantiles of their outcomes. Using a finer subdivision of quantile levels is not important for the second type of instruments.

```r
# First instrument set
Z1  <- qpeer.inst(formula = ~ X, A = A, tau = seq(0, 1, 0.1),
                  max.distance = 2, checkrank = TRUE) # finer subdivision
```

```
Z1  <- Z1$instruments #qpeer.inst returns a list of several object

# Second instrument set: y1 is used to order X
Z21 <- qpeer.inst(formula = y1 ~ X, A = A, tau = tau, max.distance = 2,
                  checkrank = TRUE)
qy1 <- Z21$qy #quantile of y among peers
Z21 <- Z21$instruments

# Second instrument set: y2 is used to order X
Z22 <- qpeer.inst(formula = y2 ~ X, A = A, tau = tau, max.distance = 2,
                  checkrank = TRUE)
qy2 <- Z22$qy #quantile of y among peers
Z22 <- Z22$instruments
```

As in the standard linear model, one can use the quantiles of x within direct friends and long-distance friends (such as friends of friends) to strengthen the instruments. Setting `max.distance = 2` means that the quantiles of x are computed among both direct friends and friends of friends. The `checkrank` argument ensures that the resulting instrument set is a full-rank matrix by removing columns that are linear combinations of others.

3.3 Estimation

Quantile peer effects are estimated using the General Method of Moments (GMM). Estimates can be obtained using the `qpeer` function. I begin with the reduced-form specification, using both types of instruments for each dependent variable.

```
QtR1 <- qpeer(formula = y1 ~ X, excluded.instruments = ~ Z1, A = A,
              tau = tau)
summary(QtR1, diagnostic = TRUE)
```

```
## Formula: y1 ~ X
## Excluded instruments: ~Z1
##
## Model: Reduced Form
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##               Estimate Std. Error   t value  Pr(>|t|)
## y1_q(total)  0.6916431  0.0062631 110.43111    <2e-16 ***
## y1_q1        0.0821477  0.0173895   4.72397 2.313e-06 ***
## y1_q2        0.2434535  0.0308467   7.89237 2.887e-15 ***
## y1_q3        0.2278337  0.0414335   5.49879 3.824e-08 ***
## y1_q4        0.1382082  0.0181092   7.63193 2.309e-14 ***
## (Intercept)  1.9919391  0.0231787  85.93830    <2e-16 ***
## XX1         -0.5032540  0.0106806 -47.11830    <2e-16 ***
```

```
## XX2          1.0109511  0.0076039 132.95082    <2e-16 ***
##
## Diagnostic tests:
##                           df1  df2 statistic p-value
## Weak instruments (y1_q1)   44 1453  58.13612  <2e-16 ***
## Weak instruments (y1_q2)   44 1453 345.98901  <2e-16 ***
## Weak instruments (y1_q3)   44 1453 595.86773  <2e-16 ***
## Weak instruments (y1_q4)   44 1453 805.51161  <2e-16 ***
## Kleibergen-Paap rk Wald    41   NA 643.21681  <2e-16 ***
## Wu-Hausman                  4 1489   1.25374  0.2863
## Hansen J                   40   NA  36.44111  0.6312
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma: 0.41127
## R-Squared: 0.98808, Adjusted R-squared: 0.98804
## Degree of freedoms of residuals: 1493
```

```
QtR2 <- qpeer(formula = y1 ~ X, excluded.instruments = ~ Z1 + Z21, A = A,
              tau = tau)
summary(QtR2)
```

```
## Formula: y1 ~ X
## Excluded instruments: ~Z1 + Z21
##
## Model: Reduced Form
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##               Estimate Std. Error   t value  Pr(>|t|)
## y1_q(total)  0.7001141  0.0041503 168.68936    <2e-16 ***
## y1_q1        0.1067169  0.0094624  11.27799    <2e-16 ***
## y1_q2        0.2242823  0.0216333  10.36745    <2e-16 ***
## y1_q3        0.2258518  0.0286079   7.89475 2.887e-15 ***
## y1_q4        0.1432631  0.0126108  11.36038    <2e-16 ***
## (Intercept)  1.9884601  0.0227889  87.25575    <2e-16 ***
## XX1         -0.5026656  0.0105528 -47.63331    <2e-16 ***
## XX2          1.0118756  0.0074986 134.94141    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma: 0.40668
## R-Squared: 0.98835, Adjusted R-squared: 0.9883
## Degree of freedoms of residuals: 1493
```

```
QtR3 <- qpeer(formula = y2 ~ X, excluded.instruments = ~ Z1, A = A,
              tau = tau)
summary(QtR3)
```

```
## Formula: y2 ~ X
## Excluded instruments: ~Z1
##
## Model: Reduced Form
## Estimator: IV
```

```
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##               Estimate Std. Error   t value  Pr(>|t|)
## y2_q(total)  0.6525717  0.0073755  88.47786    <2e-16 ***
## y2_q1        0.1466308  0.0242797   6.03924 1.548e-09 ***
## y2_q2        0.2714595  0.0449646   6.03718 1.568e-09 ***
## y2_q3        0.1509579  0.0511722   2.95000  0.003178  **
## y2_q4        0.0835235  0.0213909   3.90463 9.437e-05 ***
## (Intercept)  2.2236946  0.0224068  99.24187    <2e-16 ***
## XX1         -0.4316099  0.0103426 -41.73109    <2e-16 ***
## XX2          0.8752302  0.0073696 118.76244    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma: 0.39876
## R-Squared: 0.98002, Adjusted R-squared: 0.97994
## Degree of freedoms of residuals: 1493
```

```
QtR4 <- qpeer(formula = y2 ~ X, excluded.instruments = ~ Z1 + Z21, A = A,
              tau = tau)
summary(QtR4)
```

```
## Formula: y2 ~ X
## Excluded instruments: ~Z1 + Z21
##
## Model: Reduced Form
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##               Estimate Std. Error   t value  Pr(>|t|)
## y2_q(total)  0.6467303  0.0049670 130.20570    <2e-16 ***
## y2_q1        0.1277322  0.0131509   9.71278    <2e-16 ***
## y2_q2        0.2909670  0.0304924   9.54229    <2e-16 ***
## y2_q3        0.1454283  0.0365219   3.98195 6.835e-05 ***
## y2_q4        0.0826028  0.0156838   5.26676 1.389e-07 ***
## (Intercept)  2.2243059  0.0221615 100.36793    <2e-16 ***
## XX1         -0.4317768  0.0102679 -42.05101    <2e-16 ***
## XX2          0.8747741  0.0073026 119.78966    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma: 0.39602
## R-Squared: 0.9803, Adjusted R-squared: 0.98022
## Degree of freedoms of residuals: 1493
```

Note that the `formula` argument does not include the quantiles of peer outcomes, but only the outcome and the exogenous variables. The quantiles are computed internally by the function based on the levels specified in the `tau` argument. Additionally, the `excluded.instruments` argument should not contain any instruments that are already included as explanatory variables.

The output of the qpeer function is an object of class qpeer, to which the summary method can be applied. An important argument of the summary method is diagnostics, a logical value indicating whether diagnostic tests for the instrumental variable regression should be performed. These tests include an F-test and the rank test of Kleibergen and Paap (2006) for weak instruments, the Wu–Hausman test for endogeneity, and Hansen's J-test for overidentifying restrictions (when the number of instruments exceeds the number of endogenous regressors). The diagnostic test results are displayed for the first estimation shown above.

The reduced-form estimation for y2 is likely to be inconsistent, since the reduced-form model assumes that preferences exhibit either spillover or conformity, but not both, whereas y2 is generated from a structural model that includes both effects. The following code replicates the estimations using the structural specification.

```
QtS1 <- qpeer(formula = y1 ~ X, excluded.instruments = ~ Z1, A = A,
              tau = tau, structural = TRUE)
summary(QtS1, diagnostic = TRUE)
```

```
## Formula: y1 ~ X
## Excluded instruments: ~Z1
##
## Model: Structural
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##                    Estimate Std. Error   t value  Pr(>|t|)
## y1_q(spillover)   0.6939012  0.0122862  56.47791    <2e-16 ***
## y1_q(conformity) -0.0008894  0.0135270  -0.06575    0.9476
## y1_q(total)       0.6930118  0.0067219 103.09724    <2e-16 ***
## y1_q1             0.0881780  0.0166601   5.29276 1.205e-07 ***
## y1_q2             0.2314494  0.0301058   7.68787 1.488e-14 ***
## y1_q3             0.2364275  0.0416608   5.67506 1.386e-08 ***
## y1_q4             0.1369570  0.0178447   7.67493 1.665e-14 ***
## (Intercept)       1.9830049  0.0311950  63.56812    <2e-16 ***
## XX1              -0.4903197  0.0185511 -26.43082    <2e-16 ***
## XX2               1.0139227  0.0128029  79.19507    <2e-16 ***
##
## Diagnostic tests:
##                           df1 df2 statistic   p-value
## Weak instruments (y1_q1)   44 916  13.88460    <2e-16 ***
## Weak instruments (y1_q2)   44 916  53.12490    <2e-16 ***
## Weak instruments (y1_q3)   44 916  58.74438    <2e-16 ***
## Weak instruments (y1_q4)   44 916  57.44843    <2e-16 ***
## Kleibergen-Paap rk Wald    41  NA 122.61117 4.575e-10 ***
## Wu-Hausman                  4 954   1.25223    0.2872
## Hansen J                   42  NA  41.09511    0.5106
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma (isolated): 0.41916, (non-isolated): 0.40519
## R-Squared: 0.98623, Adjusted R-squared: 0.98616
## Degree of freedoms of residuals: 1492
```

```r
QtS2 <- qpeer(formula = y1 ~ X, excluded.instruments = ~ Z1 + Z21, A = A,
              tau = tau, structural = TRUE)
summary(QtS2)
```

```
## Formula: y1 ~ X
## Excluded instruments: ~Z1 + Z21
##
## Model: Structural
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##                     Estimate Std. Error   t value Pr(>|t|)
## y1_q(spillover)   0.7013939  0.0108886  64.41544   <2e-16 ***
## y1_q(conformity) -0.0014503  0.0131863  -0.10998   0.9124
## y1_q(total)       0.6999436  0.0051369 136.25799   <2e-16 ***
## y1_q1             0.1078357  0.0092427  11.66715   <2e-16 ***
## y1_q2             0.2221531  0.0221412  10.03347   <2e-16 ***
## y1_q3             0.2267459  0.0291700   7.77325 7.55e-15 ***
## y1_q4             0.1432088  0.0126217  11.34626   <2e-16 ***
## (Intercept)       1.9830049  0.0311950  63.56812   <2e-16 ***
## XX1              -0.4903197  0.0185511 -26.43082   <2e-16 ***
## XX2               1.0139227  0.0128029  79.19507   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma (isolated): 0.41916, (non-isolated): 0.39946
## R-Squared: 0.98835, Adjusted R-squared: 0.98829
## Degree of freedoms of residuals: 1492
```

```r
QtS3 <- qpeer(formula = y2 ~ X, excluded.instruments = ~ Z1, A = A,
              tau = tau, structural = TRUE)
summary(QtS3)
```

```
## Formula: y2 ~ X
## Excluded instruments: ~Z1
##
## Model: Structural
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##                     Estimate Std. Error   t value  Pr(>|t|)
## y2_q(spillover)   0.5091646  0.0098656  51.61001    <2e-16 ***
## y2_q(conformity)  0.1935020  0.0114013  16.97199    <2e-16 ***
## y2_q(total)       0.7026665  0.0068303 102.87517    <2e-16 ***
## y2_q1             0.1076362  0.0189718   5.67348 1.399e-08 ***
## y2_q2             0.1750139  0.0363926   4.80906 1.516e-06 ***
## y2_q3             0.3292068  0.0434045   7.58462 3.331e-14 ***
## y2_q4             0.0908096  0.0174036   5.21787  1.81e-07 ***
## (Intercept)       2.0264225  0.0313213  64.69787    <2e-16 ***
## XX1              -0.4976590  0.0186262 -26.71822    <2e-16 ***
## XX2               0.9915772  0.0128547  77.13725    <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma (isolated): 0.42086, (non-isolated): 0.40267
## R-Squared: 0.98361, Adjusted R-squared: 0.98354
## Degree of freedoms of residuals: 1492
```

```
QtS4 <- qpeer(formula = y2 ~ X, excluded.instruments = ~ Z1 + Z22, A = A,
              tau = tau, structural = TRUE)
summary(QtS4)
```

```
## Formula: y2 ~ X
## Excluded instruments: ~Z1 + Z22
##
## Model: Structural
## Estimator: IV
## Fixed effects: No
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##                     Estimate Std. Error   t value Pr(>|t|)
## y2_q(spillover)   0.5085494  0.0086698  58.65754   <2e-16 ***
## y2_q(conformity)  0.1900717  0.0110625  17.18161   <2e-16 ***
## y2_q(total)       0.6986212  0.0052681 132.61233   <2e-16 ***
## y2_q1             0.0992477  0.0103602   9.57973   <2e-16 ***
## y2_q2             0.2375855  0.0245254   9.68731   <2e-16 ***
## y2_q3             0.2337188  0.0301858   7.74267 9.77e-15 ***
## y2_q4             0.1280691  0.0126337  10.13712   <2e-16 ***
## (Intercept)       2.0264225  0.0313213  64.69787   <2e-16 ***
## XX1              -0.4976590  0.0186262 -26.71822   <2e-16 ***
## XX2               0.9915772  0.0128547  77.13725   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma (isolated): 0.42086, (non-isolated): 0.3928
## R-Squared: 0.98457, Adjusted R-squared: 0.98449
## Degree of freedoms of residuals: 1492
```

In the new results, the estimates appear reliable. For the dependent variable `y1`, the conformity parameter is not significant, as the data were simulated under the assumption of complementarity.

The `qpeer` function offers several useful options, including the ability to change the type of GMM estimator, control for subnet fixed effects, and account for heteroskedasticity. The GMM estimator type is specified using the `estimator` argument. The default value, `"IV"`, corresponds to the standard instrumental variables (IV) estimator. It is also possible to use the GMM estimator with the identity matrix as the weighting matrix (`estimator = "gmm.identity"`) or with the optimal GMM weighting matrix (`estimator = "gmm.optimal"`). Jackknife IV estimators (type 1 and type 2) can be obtained by setting `estimator` to `"JIVE"` and `"JIVE2"`, respectively. Jackknife estimators can be especially useful when the number of instruments is large, as they help reduce the bias associated with the standard IV estimator (see Mikusheva and Sun, 2022).

The `fixed.effects` argument specifies how to control for subnet fixed effects. The default value is `FALSE` or `"no"`, indicating that no fixed effects are included. Two levels of subnet fixed effects are supported: a single fixed effect per subnet (`fixed.effects = "join"`) and separate fixed effects per subnet for isolated and non-isolated individuals (`fixed.effects = "separate"`).[2] For the structural specification, fixed effects must be specified separately for each type; that is, they are necessarily double per subnet.

The `HAC` argument specifies the assumed covariance structure of the errors. By default, homoscedasticity is assumed (`HAC = "iid"`). To allow for heteroskedasticity at the individual level, set `HAC = "hetero"`. To account for heteroskedasticity and within-subnet correlation, where errors may be correlated among individuals in the same subnet, set `HAC = "cluster"`. Below are examples of model specifications that incorporate these additional options.

```
QtR5  <- qpeer(formula = y1 ~ X, excluded.instruments = ~ Z1, A = A,
               tau = tau, structural = FALSE, estimator = "gmm.optimal",
               HAC = "cluster", fixed.effects = "separate")
summary(QtR5)
```

```
## Formula: y1 ~ X
## Excluded instruments: ~Z1
##
## Model: Reduced Form
## Estimator: GMM (Weight: Optimal)
## Fixed effects: Separate
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##               Estimate Std. Error    t value  Pr(>|t|)
## y1_q(total)  0.6948934  0.0068706  101.14065    <2e-16 ***
## y1_q1        0.0877290  0.0072879   12.03759    <2e-16 ***
## y1_q2        0.3084151  0.0162134   19.02218    <2e-16 ***
## y1_q3        0.1438027  0.0208479    6.89770 5.285e-12 ***
## y1_q4        0.1549466  0.0062258   24.88778    <2e-16 ***
## XX1         -0.4980287  0.0033646 -148.01908    <2e-16 ***
## XX2          1.0131613  0.0026961  375.78856    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: Cluster
## R-Squared: 0.93462, Adjusted R-squared: 0.9344
## Degree of freedoms of residuals: 1397
```

```
QtS5 <- qpeer(formula = y2 ~ X, excluded.instruments = ~ Z1, A = A,
              tau = tau, structural = TRUE, estimator = "iv",
              HAC = "cluster", fixed.effects = "separate")
summary(QtS5)
```

```
## Formula: y2 ~ X
```

[2] As discussed by Houndetoungan et al. (2024), including two fixed effects per subnetwork may also be necessary to identify peer effects in unobserved effort, particularly when the dependent variable y is a proxy for that effort (e.g., academic effort and grade point average).

```
## Excluded instruments: ~Z1
##
## Model: Structural
## Estimator: IV
## Fixed effects: Separate
## Quantile type: 7
## Quantile levels (4): 0 0.3333333 0.6666667 1
##
## Coefficients:
##                     Estimate Std. Error   t value  Pr(>|t|)
## y2_q(spillover)   0.4943859  0.0243361  20.31495    <2e-16 ***
## y2_q(conformity)  0.1939695  0.0129421  14.98744    <2e-16 ***
## y2_q(total)       0.6883554  0.0220983  31.14965    <2e-16 ***
## y2_q1             0.1128382  0.0255863   4.41010 1.033e-05 ***
## y2_q2             0.1617763  0.0545694   2.96460  0.003031  **
## y2_q3             0.3127926  0.0548240   5.70540 1.161e-08 ***
## y2_q4             0.1009484  0.0197665   5.10706 3.272e-07 ***
## XX1              -0.4953984  0.0240556 -20.59385    <2e-16 ***
## XX2               0.9951260  0.0119127  83.53512    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: Cluster
## R-Squared: 0.93366, Adjusted R-squared: 0.93339
## Degree of freedoms of residuals: 1397
```

3.4 Specification tests

Several specification tests have been discussed above. These include weak instrument, endogeneity, and overidentification tests, which can be performed by setting `diagnostic` to `TRUE` in the `summary` method. In this section, I discuss monotonicity tests for quantile peer effects, the validity of type 2 instruments (see the discussion of the two types of instruments in Section 3.2), and an encompassing test to choose between competing sets of quantile levels.

These tests can be performed using the function `qpeer.test`. The argument `which` indicates the type of test. For instance, `which` can take the values `"uniform"`, `"increasing"`, and `"decreasing"` to test whether the λ_τ's are uniform, increasing, or decreasing. The uniform test is based on a standard Wald test for the equality of all λ_τ's. The increasing and decreasing tests are based on Kodde and Palm (1986). Here is an example:

```
qpeer.test(QtR1, which = "uniform")
```

```
## Testing quantile peer effect monotonicity
##
## Quantile peer effects:
##   lambda_tau:
##   Null hypothesis: lambda_tau is uniform
##   Statistic: 34.14108 -- p-value: 1.85e-07
```

```
qpeer.test(QtS5, which = "decreasing")
```

```
## Testing quantile peer effect monotonicity
##
## Quantile peer effects:
##   lambda_tau:
```

```
##   Null hypothesis: lambda_tau is decreasing
##   Statistic: 21.66732 -- p-value: 9.313e-10
```

The monotonicity tests can be useful for selecting a more parsimonious and precise model, as they involve fewer parameters. For instance, if the null hypothesis that the λ_τ's are uniform is not rejected, then a standard LIM peer effect model can be used instead of a quantile model. Moreover, if the λ_τ's are monotonic (increasing or decreasing), the constant elasticity of substitution (CES)-based model can be used (see Section 4).

To test the endogeneity of $\mathbf{Z}_2$, the argument `which` can be set to `"wald"` or `"sargan"`. The former is a Wald-style test that compares the estimates obtained using $\mathbf{Z}_1$ and $\mathbf{Z}_2$. If the null hypothesis that both estimates are equal is not rejected, then $\mathbf{Z}_2$ is considered exogenous. The latter is an overidentification-style test that assesses the validity of the additional information in $\mathbf{Z}_2$ that is not captured by $\mathbf{Z}_1$. These tests can be used even if $\mathbf{Z}_2$ does not nest $\mathbf{Z}_1$. The following examples illustrate the test:

```
qpeer.test(QtR1, QtR2, which = "wald")
```

```
## Testing instrument validity (Wald Test)
##
## Coefficient differences:
##                  Estimate   Std. Error     t value Pr(>|t|)
## y1_q1       -0.0245692433 0.0144388786 -1.70160328  0.08883 .
## y1_q2        0.0191712199 0.0216246987  0.88654275   0.3753
## y1_q3        0.0019818668 0.0294948703  0.06719361   0.9464
## y1_q4       -0.0050548880 0.0127851763 -0.39537101   0.6926
## (Intercept)  0.0034790404 0.0024595287  1.41451507   0.1572
## XX1         -0.0005883950 0.0004258552 -1.38167866   0.1671
## XX2         -0.0009244596 0.0005560624 -1.66251040  0.09641 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Null hypothesis: Z2 is exogenous
##   Statistic: 6.242505 -- p-value: 0.1818
```

```
qpeer.test(QtS3, QtS4, which = "sargan")
```

```
## Testing instrument validity (J-Sargan Test)
## Null hypothesis: Z2 is exogenous
##   Statistic: 36.9902 -- p-value: 0.003375
```

If the estimations based on the type 2 instrument set are not rejected, then the user can report these estimates, as the type 2 instrument set is likely stronger.

Choosing suitable quantile levels is important. The `qpeer.test` function offers an encompassing test to choose between two competing specifications. Assume that the quantile peer effect model is estimated with two sets of quantile levels, $\mathcal{T}_1$ and $\mathcal{T}_2$. By setting `which` to `"encompassing"`, one can test whether one model performs worse; that is, whether it fails to replicate the features captured by the other. The `model1` and `model2` arguments are used to specify both models estimated using the `qpeer` function. The null hypothesis is that `model1` is not worse.

```r
# Estimating QtS6 with a misspecified tau
QtS6 <- qpeer(formula = y2 ~ X, excluded.instruments = ~ Z1, A = A,
              tau = c(0, 1), structural = TRUE, estimator = "iv",
              HAC = "cluster", fixed.effects = "separate")
qpeer.test(model1 = QtS6, model2 = QtS5, which = "encompassing", boot = 100)
```

```
## Encompassing Test
##
## Coefficients for delta:
##                      Estimate Std. Error     t value  Pr(>|t|)
## y2_q(conformity)  0.000413983 0.01061311  0.03900675    0.9689
## y2_q1            -0.206169216 0.03433621 -6.00442520  1.92e-09 ***
## y2_q2             0.161776289 0.05354477  3.02132781  0.002517  **
## y2_q3             0.312792586 0.05495287  5.69201580 1.255e-08 ***
## y2_q4            -0.170513535 0.02310051 -7.38137628 1.568e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Null hypothesis: Model 1 is not worse
##   Robust F statistic: 44.96761 -- p-value: < 2e-16
##   KP Wald rank statistic: 179.8704 -- p-value: < 2e-16
```

In the preceding code, the null hypothesis is that the specification which includes only the minimum and the maximum as quantile levels does not perform worse. The hypothesis is rejected, which is expected since the data are simulated with more quantile levels. Consequently, the model with only two quantile levels is not a good choice.

In the following code, the null hypothesis is that the model with four quantile levels is not worse than the competing model with two quantile levels:

```r
qpeer.test(model1 = QtS5, model2 = QtS6, which = "encompassing", boot = 100)
```

```
## Encompassing Test
##
## Coefficients for delta:
##                      Estimate   Std. Error   t value Pr(>|t|)
## y2_q(conformity) 9.256957e-17 2.121584e-16 0.4363230   0.6626
## y2_q1            5.775534e-17 1.123332e-16 0.5141429   0.6072
## y2_q2            3.547553e-17 6.074190e-17 0.5840373   0.5592
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Null hypothesis: Model 1 is not worse
##   Robust F statistic: 0.3619924 -- p-value: 0.6964
##   KP Wald rank statistic: 0.7239848 -- p-value: 0.6963
```

Here, the null hypothesis is not rejected. However, this does not directly imply that the other model with four quantiles is well specified. The model should also be compared with alternative specifications that include, for example, five or six quantile levels. Indeed, encompassing tests only compare two models and do not imply that one of them is correctly specified. Of course, in this example, we know that the model with four quantile levels is well specified because we know the data-generating process. Therefore, the model with four

quantile levels cannot be rejected, in general, against other well-specified competing models.

4 CES-based peer effect models

This section introduces peer effect models with a CES social norm (see Boucher et al., 2024). These models include a substitution parameter that determines whether peers with high or low outcomes have a greater influence.

4.1 A brief description

Boucher et al. (2024) present a flexible social norm using the CES function. They replace the average outcome among peers in the standard LIM model by the CES function. Specifically, the model is given by:

$$y_i = \left(\sum_{j \neq i} a_{ij} y_j^{\rho} \right)^{\frac{1}{\rho}} + \boldsymbol{x}_i' \boldsymbol{\gamma} + \varepsilon_i. \tag{4}$$

This model nests the standard LIM model. When $\rho = 1$, Equation (4) becomes the standard LIM model. If $\rho > 1$, peers with high outcomes are more important in explaining y_i. In contrast, if $\rho < 1$, peers with low outcomes are more important. To see why, observe that the CES function is strictly convex when $\rho > 1$ and strictly concave when $\rho < 1$. Therefore, peer effects are either uniform, strictly increasing, or strictly decreasing in peer outcomes. This model does not accommodate situations where only peers with high and low outcomes matter while those with moderate outcomes are not influential, and vice versa.

When $\rho \to +\infty$, only the peer with the highest outcome matters; when $\rho \to -\infty$, only the peer with the lowest outcome matters. These limiting cases correspond to models where the average peer outcome is replaced by the maximum or minimum peer outcome (see Tao and Lee, 2014; Tatsi, 2015).

As in Section 2, CES-based models can describe preferences that exhibit both spillover and conformity effects. See, for example, Equations (2) and (3). The specification of the outcome for isolated agents is similar to Equation (2), and that for non-isolated agents is similar to Equation (3), with the CES function replacing the quantile functions. Disentangling conformity from spillover effects requires the presence of isolated agents in the network.

4.2 Data simulation

The function `cespee.sim` can be used to simulate the CES-based peer effect model. This is useful, for instance, for conducting counterfactual analyses involving changes in the intercept (e.g., providing subsidies or introducing a tax), in the network structure, or in the exogenous characteristics $\boldsymbol{x}$.

The following code defines the sample size, simulates the network matrix and the matrix of explanatory variables $\mathbf{X}$, assigns values to the model parameters, and simulates the outcomes. As in Section 3, I simulate two outcomes: one generated from the reduced-form model (with only spillovers) and the other from the structural model. One important limitation of the CES specification is that it does not accommodate zero or negative outcomes. To simulate data from this model, $x_i'\gamma + \varepsilon_i$ must be strictly positive for all i. Consequently, I increase the intercept in γ to ensure this condition is satisfied.

```
ngr  <- 50  # Number of subnets
nvec <- rep(30, ngr)  # Size of subnets
n    <- sum(nvec)

# Network matrix
A <- lapply(1:ngr, function(g) {
  Ag <- matrix(rbinom(nvec[g]^2, 1, 0.3), nvec[g], nvec[g])
  diag(Ag) <- 0
  # Adding isolated nodes (important for the structural model)
  niso <- sample(0:nvec[g], 1, prob = (nvec[g] + 1):1 / sum((nvec[g] + 1):1))
  if (niso > 0) {
    Ag[sample(1:nvec[g], niso), ] <- 0
  }
  Ag
  rs   <- rowSums(Ag); rs[rs == 0] <- 1
  Ag   <- Ag/rs # rowSums are normalized to one
})

X   <- cbind(rnorm(n), rpois(n, 2)); colnames(X) <- c("X1", "X2")

lambda  <- 0.55
lambda2 <- 0.2
gamma   <- c(2.5, -0.5, 1)
rho     <- -3

# First dependent variable (reduced form without conformity)
y1      <- cespeer.sim(formula = ~ X, A = A, rho = rho,
                       lambda = lambda, gamma = gamma, structural = FALSE,
                       epsilon = rnorm(n, 0, 0.4))
y1      <- y1$y #cespeer.sim returns a list of several object including y

# Second dependent variable (structural form with conformity)
y2      <-  cespeer.sim(formula = ~ X, A = A, rho = rho,
```

```
                        lambda = c(lambda2, lambda), gamma = gamma,
                        structural = TRUE, epsilon = rnorm(n, 0, 0.4))
y2      <- y2$y
```

4.3 Estimation

The model can be estimated using the function `cespeer`. As suggested by Boucher et al. (2024), the instrument for the social norm can be constructed using an exogenous prediction of the outcome. This prediction can be obtained from a simple linear regression of the outcome on the exogenous characteristics, as shown in the following code:

```
yhat1 <- fitted(lm(y1 ~ X))
yhat2 <- fitted(lm(y2 ~ X))
```

Using the function `cespeer`, the exogenous prediction can be specified via the `instrument` argument. The remaining arguments are similar to those of the `qpeer` function, except that the quantile level vector `tau` is not required. Additionally, the function includes a `fixed.effects` argument to indicate the type fixed effects, and a `HAC` argument to account for heteroskedasticity and within-group correlation in the error terms.

```
cesR1 <- cespeer(formula = y1 ~ X, instrument = ~ yhat1, A = A,
                 structural = FALSE)
summary(cesR1)
```

```
## Formula: y1 ~ X
## Excluded instrument: ~yhat1
##
## Model: Reduced Form
## Fixed effects: No
##
## Coefficients:
##               Estimate Std. Error   t value  Pr(>|t|)
## rho         -3.2161914  0.4177989  -7.69794 1.377e-14 ***
## A:y1         0.5510609  0.0093259  59.08906    <2e-16 ***
## (Intercept)  2.5224384  0.0225614 111.80341    <2e-16 ***
## XX1         -0.5177238  0.0104664 -49.46524    <2e-16 ***
## XX2          1.0040681  0.0073507 136.59536    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma: 0.40411
## CES parameter -- testing whether rho = 1: prob =  0
```

```
cesS1 <- cespeer(formula = y2 ~ X, instrument = ~ yhat2, A = A,
                 structural = TRUE)
summary(cesS1)
```

```
## Formula: y2 ~ X
## Excluded instrument: ~yhat2
##
```

```
## Model: Structural
## Fixed effects: No
##
## Coefficients:
##                     Estimate Std. Error   t value Pr(>|t|)
## rho              -3.3463398  0.3582790  -9.34004   <2e-16 ***
## A(spillover):y2   0.3580073  0.0073713  48.56750   <2e-16 ***
## A(conformity):y2  0.2026839  0.0108273  18.71968   <2e-16 ***
## A(total):y2       0.5606912  0.0100889  55.57534   <2e-16 ***
## (Intercept)       2.4610251  0.0299567  82.15277   <2e-16 ***
## XX1              -0.5285101  0.0171679 -30.78479   <2e-16 ***
## XX2               1.0119264  0.0119310  84.81465   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: IID, sigma (isolated): 0.4133, (non-isolated): 0.41848
## CES parameter -- testing whether rho = 1: prob =  0
```

The estimates appear reasonable, including the estimate of the CES parameter ρ. The output of the `summary` function includes a test of whether the ρ parameter is equal to one; that is, whether the standard LIM model can be used instead of the CES specification. As expected, the null hypothesis is rejected, since the data were simulated by setting $\rho = -3$.

Since $\rho < 1$, peers with low outcomes play a more important role in explaining agent outcomes. For instance, if the user were to estimate a quantile peer effect model on the same data, the function `qpeer.test` can be used to test whether the λ_τ's are decreasing:

```
Z    <- qpeer.inst(formula = ~ X, A = A, tau = seq(0, 1, 0.1),
                   max.distance = 2, checkrank = TRUE)$instruments
QtR1 <- qpeer(formula = y1 ~ X, excluded.instruments = ~ Z, A = A,
              tau = seq(0, 1, 1/3))
qpeer.test(QtR1, which = "decreasing")
```

```
## Testing quantile peer effect monotonicity
##
## Quantile peer effects:
##   lambda_tau:
##   Null hypothesis: lambda_tau is decreasing
##   Statistic: 2.027122 -- p-value: 0.002813
```

The null hypothesis is not rejected, suggesting that the CES model can be used in this case to analyze the outcome. The CES model involves fewer parameters and thus provides more precise estimates than the quantile model.

5 Other specifications

I discuss two important functions in this section for estimating other specifications of peer effect models. The first function is `linpeer`, which estimates the standard LIM model. The second function is `genpeer`, which allows the user to define their own social norm.

For the standard LIM model, instruments can be **AX** when the model does not include

contextual effects, or $\mathbf{A}^2\mathbf{X}$ when it does (see Bramoullé et al., 2009). It is also possible to include higher powers of $\mathbf{A}$ to strengthen the instruments (see Houndetoungan and Maoude, 2024). To compute $\mathbf{A}^k\mathbf{X}$ for $k \geq 1$, I use the function `peer.avg` from the **PartialNetwork** package (Boucher and Houndetoungan, 2024). In the following code, I use both $\mathbf{AX}$ and $\mathbf{A}^2\mathbf{X}$ as instruments.

```
library(PartialNetwork)
AX  <- peer.avg(A, X)
AAX <- peer.avg(A, AX)
Ot1 <- linpeer(formula = y2 ~ X, excluded.instruments = ~ AX + AAX, A = A,
               structural = TRUE, estimator = "gmm.optimal",
               fixed.effects = "separate", HAC = "cluster")
summary(Ot1, diagnostic = TRUE)
```

```
## Formula: y2 ~ X
## Excluded instruments: ~AX + AAX
##
## Model: Structural
## Estimator: GMM (Weight: Optimal)
## Fixed effects: Separate
##
## Coefficients:
##                    Estimate Std. Error   t value  Pr(>|t|)
## A(spillover):y2   0.3313329  0.0478694   6.92161 4.466e-12 ***
## A(conformity):y2  0.1871388  0.0136018  13.75839    <2e-16 ***
## A(total):y2       0.5184717  0.0413902  12.52643    <2e-16 ***
## XX1              -0.5181891  0.0162509 -31.88684    <2e-16 ***
## XX2               1.0133287  0.0122158  82.95207    <2e-16 ***
##
## Diagnostic tests:
##                          df1 df2  statistic p-value
## Weak instruments           4 917  970.40830  <2e-16 ***
## Kleibergen-Paap rk Wald    4  NA 3881.63320  <2e-16 ***
## Wu-Hausman                 1 920    0.00014  0.9904
## Hansen J                   4  NA   10.94442  0.0272   *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## HAC: Cluster
## R-Squared: 0.90358, Adjusted R-squared: 0.90339
## Degree of freedoms of residuals: 1400
```

Furthermore, the package provides the generic function `genpeer`, which allows users to define their own endogenous variables. This option is useful for estimating, for example, the effects of quantiles among girl and boy peers separately, combining average peer effects with quantile peer effects in the same model, or specifying other custom social norms.

The `endogenous.variables` argument must be specified as a formula that includes the desired endogenous variables. The example below illustrates how to estimate peer effects using both the average outcome of peers and the minimum and maximum peer outcomes.

```r
Ay2 <- peer.avg(A, y2)
qy2 <- qpeer.inst(formula = y2 ~ 1, A = A, tau = c(0, 1))$qy #min and max
Ot2 <- genpeer(formula = y2 ~ X, excluded.instruments = ~ Z + AX + AAX,
               endogenous.variables = ~ Ay2 + qy2, A = A, structural = TRUE,
               estimator = "gmm.optimal", fixed.effects = "separate",
               HAC = "iid") # includes average, min, and max of peers
summary(Ot2, diagnostic = TRUE)
```

```
## Formula: y2 ~ X
## Endogenous variables: ~Ay2 + qy2
## Excluded instruments: ~Z + AX + AAX
##
## Model: Structural
## Estimator: GMM (Weight: Optimal)
## Fixed effects: Separate
##
## Coefficients:
##                     Estimate Std. Error   t value  Pr(>|t|)
## Peers(conformity)  0.1969587  0.0123191  15.98807    <2e-16 ***
## Ay2                0.2423510  0.0645759   3.75296 0.0001748 ***
## qy2y2_q1           0.3700117  0.0356916  10.36692    <2e-16 ***
## qy2y2_q2          -0.0027204  0.0259380  -0.10488    0.9165
## XX1               -0.5181891  0.0179111 -28.93119    <2e-16 ***
## XX2                1.0133287  0.0125286  80.88134    <2e-16 ***
##
## Diagnostic tests:
##                              df1 df2  statistic  p-value
## Weak instruments (Ay2)        48 873  146.17469   <2e-16 ***
## Weak instruments (qy2y2_q1)   48 873   19.89947   <2e-16 ***
## Weak instruments (qy2y2_q2)   48 873   50.48365   <2e-16 ***
## Kleibergen-Paap rk Wald       46  NA 6107.18649   <2e-16 ***
## Wu-Hausman                     3 916    2.74833  0.04182   *
## Hansen J                      46  NA   77.26621 0.002648  **
## ---
## Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
##
## HAC: IID, sigma (isolated): 0.41346, (non-isolated): 0.44667
## R-Squared: 0.93417, Adjusted R-squared: 0.93395
## Degree of freedoms of residuals: 1398
```

6 Conclusion

Thank you for reading this documentation. If you encounter any issues, please report them via the Issues page on GitHub. If you use **QuantilePeer** in your research or publications, please cite it. You can run `citation("QuantilePeer")` in **R** to obtain the citation information, including the BibTeX entry for LaTeX users. Please also cite the paper associated with the package (Houndetoungan, 2025).

References

Boucher, V. and Fortin, B. (2016). Some challenges in the empirics of the effects of networks.

Boucher, V. and Houndetoungan, A. (2024). *PartialNetwork: Estimating Peer Effects Using Partial Network Data*. R package version 1.0.4.

Boucher, V., Rendall, M., Ushchev, P., and Zenou, Y. (2024). Toward a general theory of peer effects. *Econometrica*, 92(2):543–565.

Bramoullé, Y., Djebbari, H., and Fortin, B. (2009). Identification of peer effects through social networks. *Journal of Econometrics*, 150(1):41–55.

Hausman, J. A. (1978). Specification tests in econometrics. *Econometrica: Journal of the Econometric Society*, pages 1251–1271.

Houndetoungan, A. (2025). Quantile peer effect models. *arXiv preprint arXiv:25*.

Houndetoungan, A., Kouame, C., and Vlassopoulos, M. (2024). Identifying peer effects in networks with unobserved effort and isolated students. *arXiv preprint arXiv:2405.06850*.

Houndetoungan, A. and Maoude, A. H. (2024). Inference for two-stage extremum estimators. *arXiv preprint arXiv:2402.05030*.

Hyndman, R. J. and Fan, Y. (1996). Sample quantiles in statistical packages. *The American Statistician*, 50(4):361–365.

Kleibergen, F. and Paap, R. (2006). Generalized reduced rank tests using the singular value decomposition. *Journal of Econometrics*, 133(1):97–126.

Kodde, D. A. and Palm, F. C. (1986). Wald criteria for jointly testing equality and inequality restrictions. *Econometrica: Journal of the Econometric Society*, pages 1243–1248.

Mikusheva, A. and Sun, L. (2022). Inference with many weak instruments. *The Review of Economic Studies*, 89(5):2663–2686.

Tao, J. and Lee, L.-f. (2014). A social interaction model with an extreme order statistic. *The Econometrics Journal*, 17(3):197–240.

Tatsi, E. (2015). Endogenous social interactions: Which peers matter? In *VfS Annual Conference 2015 (Muenster): Economic Development-Theory and Policy*, number 113168. Verein für Socialpolitik/German Economic Association.