Online Appendix for "Effects of heterogeneity and homophily on cooperation"

Appendix A: Stata Simulation Code with Explanations

```
// Explanations come after "//"

clear
set obs   100000  // N of Monte-Carlo integration points
set seed  123

// defining PD outcome parameters using the standard PD notation: T,R,P,S
scalar P  = 2
scalar R  = 3
scalar S  = 1.5
scalar T  = 3.5

// XY is the critical threshold above which a person cooperates.
// In the current paper, XY=1/3
scalar XY = (T-R)/(R-S)

// setting the distribution of ingroup theta and outgroup theta
scalar sd1     = 0.3
scalar sd2     = 0.3
scalar cor12   = 0.3

matrix C = (sd1*sd1       , cor12*sd1*sd2  \  ///
            cor12*sd1*sd2 , sd2*sd2           )

// t1= ingroup theta, t2=outgroup theta
// adjust means(a b) to set the means t1 and t2

corr2data t1 t2, cov(C) means(0.2 0.1)

// ----------------------------------------------------------------
// ----------------------- iteration 1 ---------------------------
// ----------------------------------------------------------------

count if t1>=XY
gen      pin_1 = r(N)/_N   //initial value of avg. ingroup cooperation
count if t2>=XY
gen      pou_1 = r(N)/_N   //initial value of avg. outgroup cooperation

// expected utility terms
// uc1_1 indicate exp. utiliy of playing with ingroup and cooperating
// uc2_1 indicate exp. utiliy of playing with outgroup and cooperating
// ud1_1 indicate exp. utiliy of playing with ingroup and defecting
// ud2_1 indicate exp. utiliy of playing with outgroup and defecting
// *_1 the last 1 indicates values of iteration 1

gen uc1_1 = pin_1*(R + t1*R) + (1-pin_1)*(S + t1*T)
gen uc2_1 = pou_1*(R + t2*R) + (1-pou_1)*(S + t2*T)
gen ud1_1 = pin_1*(T + t1*S) + (1-pin_1)*(P + t1*P)
gen ud2_1 = pou_1*(T + t2*S) + (1-pou_1)*(P + t2*P)

// Predicted Behavior
// the first letter indicates whether subject would coop. or defect with ingr.
// the second letter indicates whether subject would coop. or defect with ingr.
// the third character, i.e., 1 or 2 indicate whether subject plays
// ingroup (1) or outgroup (2)

gen cc1_1 = uc1_1>=uc2_1 & t1>=XY & t2>=XY
gen cc2_1 = uc1_1< uc2_1 & t1>=XY & t2>=XY
gen dd1_1 = ud1_1>=ud2_1 & t1< XY & t2< XY
gen dd2_1 = ud1_1< ud2_1 & t1< XY & t2< XY
gen cd1_1 = uc1_1>=ud2_1 & t1>=XY & t2<=XY
gen cd2_1 = uc1_1< ud2_1 & t1>=XY & t2<=XY
gen dc1_1 = ud1_1>=uc2_1 & t1< XY & t2> XY
gen dc2_1 = ud1_1< uc2_1 & t1< XY & t2> XY
```

```
gen ou_1 = cc2_1 + dd2_1 + dc2_1 + cd2_1 // playing outgroup in iteration 1
gen nou_1 =  r(mean)                     // proportion playing outgroup iter. 1

gen in_1 = cc1_1 + dd1_1 + dc1_1 + cd1_1
summ in_1              // playing ingroup in iteration 1
gen nin_1 = r(mean)    // proportion playing ingroup in iteration 1

gen pinm_1 = pin_1     // mean ingroup cooperation in iteration 1
gen poum_1 = pou_1     // mean ingroup cooperation in iteration 1


// asserting that subject plays one strategy

assert cc1_1+cc2_1+dd1_1+dd2_1+cd1_1+cd2_1+dc1_1+dc2_1==1

// ----------------------------------------------------------------
// Iterating until average cooperation rates and proportion -------
// playing in/outgroup converges ---------------------------------
// ----------------------------------------------------------------

// At the moment 15 iterations, can be increased if needed

forvalues z = 2/15 {

local k = `z' - 1

gen uc1_`z' = pinm_`k'*(R + t1*R ) + (1-pinm_`k')*(S + t1*T )
gen uc2_`z' = poum_`k'*(R + t2*R ) + (1-poum_`k')*(S + t2*T )
gen ud1_`z' = pinm_`k'*(T + t1*S ) + (1-pinm_`k')*(P + t1*P )
gen ud2_`z' = poum_`k'*(T + t2*S ) + (1-poum_`k')*(P + t2*P )

gen cc1_`z' = uc1_`z'>= uc2_`z' & t1>=XY & t2>=XY
gen cc2_`z' = uc1_`z'<  uc2_`z' & t1>=XY & t2>=XY
gen dd1_`z' = ud1_`z'>= ud2_`z' & t1< XY & t2< XY
gen dd2_`z' = ud1_`z'<  ud2_`z' & t1< XY & t2< XY
gen cd1_`z' = uc1_`z'>= ud2_`z' & t1>=XY & t2<=XY
gen cd2_`z' = uc1_`z'<  ud2_`z' & t1>=XY & t2<=XY
gen dc1_`z' = ud1_`z'>= uc2_`z' & t1< XY & t2> XY
gen dc2_`z' = ud1_`z'<  uc2_`z' & t1< XY & t2> XY

gen  ou_`z' = cc2_`z' + dd2_`z' + dc2_`z' + cd2_`z'
summ ou_`z'
gen  nou_`z' = r(mean)

gen  in_`z' = cc1_`z' + dd1_`z' + dc1_`z' + cd1_`z'
summ in_`z'
gen  nin_`z' = r(mean)

gen junk_`z' = cc1_`z' + cd1_`z'
summ junk_`z'
gen pin_`z' = r(mean)/nin_`z' // ingroup cooperation level in iteration z
drop junk_`z'

gen junk_`z' = cc2_`z' + dc2_`z'
summ junk_`z'
gen pou_`z' = r(mean)/nou_`z'  // outgroup cooperation level in iteration z
drop junk_`z'

// To improve on convergence and also to make the process similar to the
// protocol implemented in the experiment, the cooperation levels in ingroup and
// outgroup environments calculated as weighted averages of previous iterations:

forvalues x = 1/`z' {
gen pnou_`x' = pou_`x'*nou_`x'
gen pnin_`x' = pin_`x'*nin_`x'
}

egen junk2_`z' = rowtotal(pnou_*)
egen junk3_`z' = rowtotal(nou_*)

gen poum_`z'  = junk2_`z'/junk3_`z'
```

```
capture drop junk2 junk3 junk*

egen junk2_`z' = rowtotal(nin_*)
egen junk3_`z' = rowtotal(pnin_*)
gen  pinm_`z'   = junk3_`z'/junk2_`z'
capture drop junk2 junk3 junk* pnou* pnin*

gen coop_c`z' = cc1_`z'+cc2_`z'+cd1_`z'+dc2_`z'
}

// Without partner choice:

gen  c1_nc = t1>XY
gen  c2_nc = t2>XY
summ c1_nc
gen pin_nc = r(mean)
summ c2_nc
gen pou_nc = r(mean)
gen coop_nc = 0.5*pin_nc + 0.5*pou_nc

// Table 2 entry (one row for each t1-t2 distribution):

tabstat t1 t2 coop_c15 pin_15 pou_15 nou_15 coop_nc pin_nc pou_nc , ///
        stat(mean) f(%9.2f)
```

Appendix B: Inducing Group Identity Using the Minimal Group Paradigm

Subjects received five pairs of paintings by Paul Klee and Wassily Kandinsky. Depending on their relative preferences, half of the subjects were classified as Klees and the other half as Kandinskys.

To enhance group identity, two additional steps were taken. First, subjects received a collective quiz on which they had to guess the painter, Klee or Kandinsky, of two additional paintings. Participants received 20 tokens (£.8) if the majority of their group members identified the painters of both paintings correctly. They received an additional 20 tokens (£.8) if their group gave at least as many correct answers as the other group. To prevent the possibility that identification with a low-performing group may be weaker, the results of the quiz were revealed only at the end of the experiment. Second, the subjects allocated tokens between ingroup members and outgroup member by making decisions in 10 binary other-other Dictator Games adapted from Aksoy and Weesie (2012).

In the control condition, subjects reviewed and stated their preferences for the same five pairs of paintings, and they were privately informed about which of the two painters they liked more. However, they were not divided into groups. The control condition included the same painter quiz and the other-other allocation task, without any reference to groups.

Appendix C: Manipulation Check

Table C1: Descriptive statistics for the items used for manipulation check: means, in parentheses standard deviations, and in square brackets standard errors of the means of items measuring identification with the ingroup and the outgroup.

	Ingroup	Outgroup
Belongingness	3.93	2.02
	(1.95)	(1.32)
	[.18]	[.12]
Commonality	3.99	2.66
	(1.66)	(1.51)
	[.15]	[.14]
Closeness	3.88	2.40
	(1.79)	(1.37)
	[.16]	[.12]
Liking	4.17	2.88
	(1.79)	(1.57)
	[.16]	[.14]