LES DÉFIS DE L'ANALYSE DES RÉSEAUX DYNAMIQUES

UN EXEMPLE DE LA CO-DÉLINQUANCE

YANICK CHARETTE, PH.D.
ÉCOLE DE SERVICE SOCIAL

R À QUÉBEC, 26 MAI 2017

UNIVERSITÉ LAVAL

Charette, Y., & Papachristos, A. V. (2017). The network dynamics of co-offending careers. *Social Networks*.

OBJECTIFS

Analyser la continuité des partenariats de complicité de délinquance et les facteurs qui influencent celle-ci :

- Les caractéristiques des délinquants
- Les caractéristiques des partenariats
- Les processus supra-dyadiques

MÉTHODE ET DONNÉES

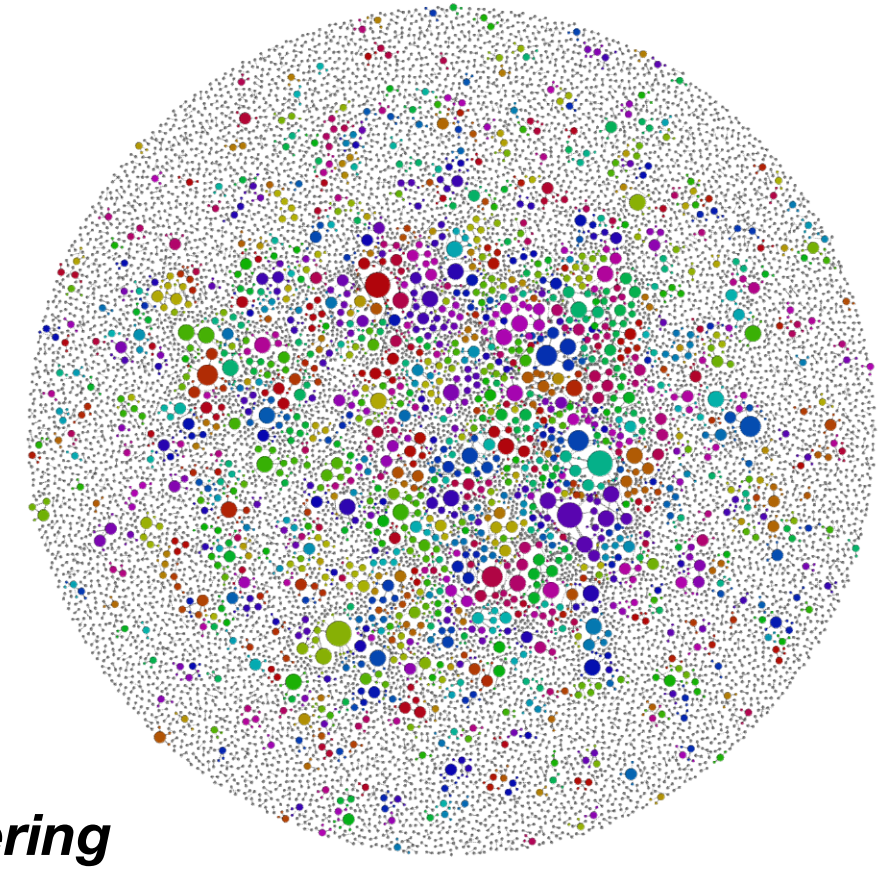
2 Sources de données

- **Réseau de co-arrestation**
  - Chicago (2006-2013)
  - n = 181 615 individus
  - n = 365 800 dyades (co-arrestations)

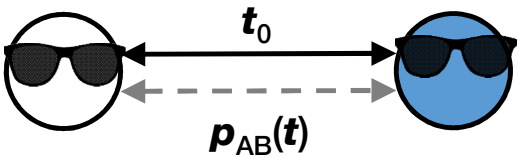
- **Victimisation**
  - n = 20 017

Analyses : Modèles à risques proportionnels avec covariées dépendantes temporellement et ***clustering***

FIGURE 1 : SOCIOGRAMME D'UN ÉCHANTILLON



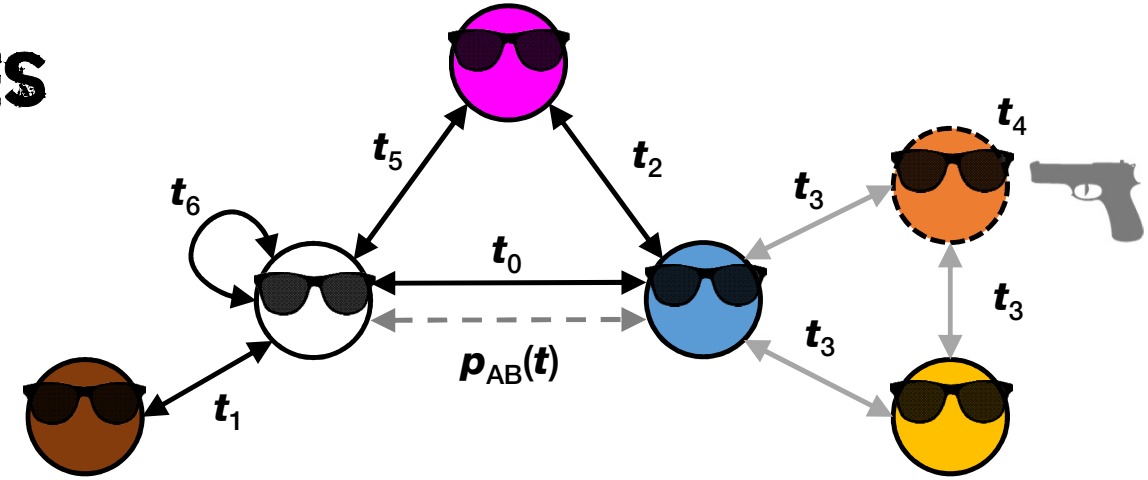
HOMOPHILIE

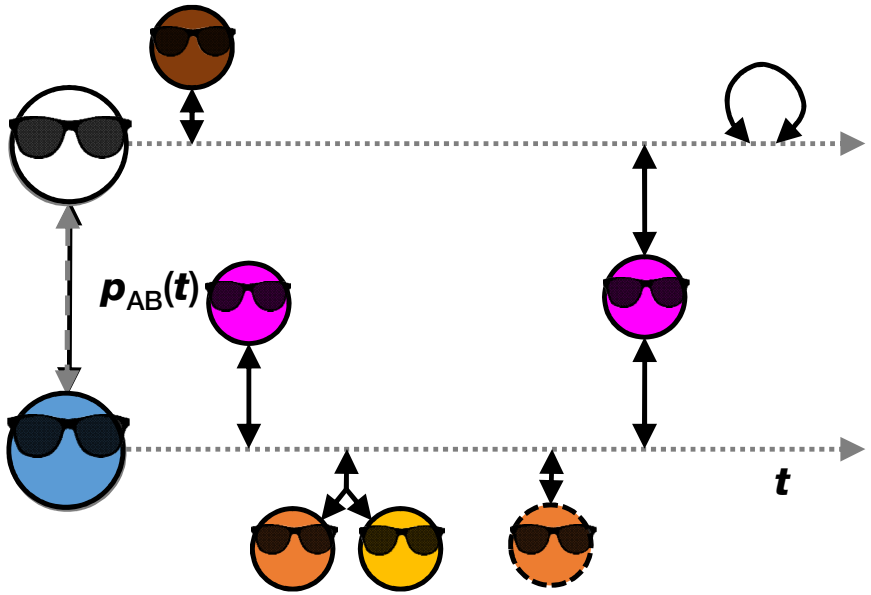


Informations individuelles

- Genre
- Ethnicité
- Âge
- Appartenance à un gang (même groupe)
- Distance géographique

DYNAMIQUES RÉSEAUX





STRUCTURE DES DONNÉES

- Défis avec les packages R actuels :
  - Peu appropriés pour les analyses de réseaux dynamiques (e.g. *sna*, *igraph*)
  - Ceux qui le sont, sont pour les mesures de temps discrètes (e.g. *siena*)

- Alternative simple, sympathique, efficace, et facilement personnalisable pour opérationnaliser les réseaux dynamiques avec *reshape*.

N=3 171 796

STRUCTURE DES DONNÉES

Importation des données:

```
library(data.table)

data <- data.table(rbind(c("WHI", 0, "01/01/2001" , "A"),
c("BLU", 0, "01/01/2001", "A"),
c("WHI", 1, "02/01/2001", "A"),
c("BRU", 1, "02/01/2001", "A"),
c("BLU", 2, "03/01/2001", "A"),
c("PIN", 2, "03/01/2001", "A"),
c("BLU", 3, "04/01/2001", "B"),
c("ORA", 3, "04/01/2001", "B"),
c("YEL", 3, "04/01/2001", "B"),
c("WHI", 5, "05/01/2001", "A"),
c("PIN", 5, "05/01/2001", "A"),
c("WHI", 6, "06/01/2001", "A")))

colnames(data) <- c("IND", "ARR", "DATE", "DELIT")

data$DATE <- as.Date(data$DATE, format="%d/%m/%Y")
```

IND	ARR	DATE	DÉLIT
WHI	0	01/01/2001	A
BLU	0	01/01/2001	A
WHI	1	02/01/2001	A
BRU	1	02/01/2001	A
BLU	2	03/01/2001	A
PIN	2	03/01/2001	A
BLU	3	04/01/2001	B
ORA	3	04/01/2001	B
YEL	3	04/01/2001	B
WHI	5	05/01/2001	A
PIN	5	05/01/2001	A
WHI	6	06/01/2001	A

N= 365 800

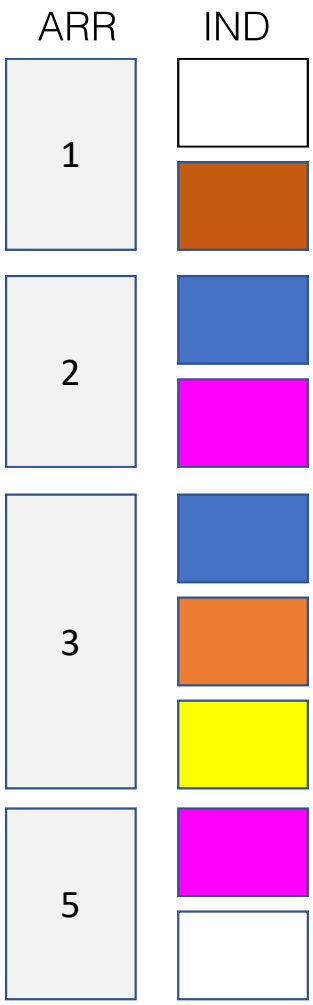
STRUCTURE DES DONNÉES

Création de dyades :

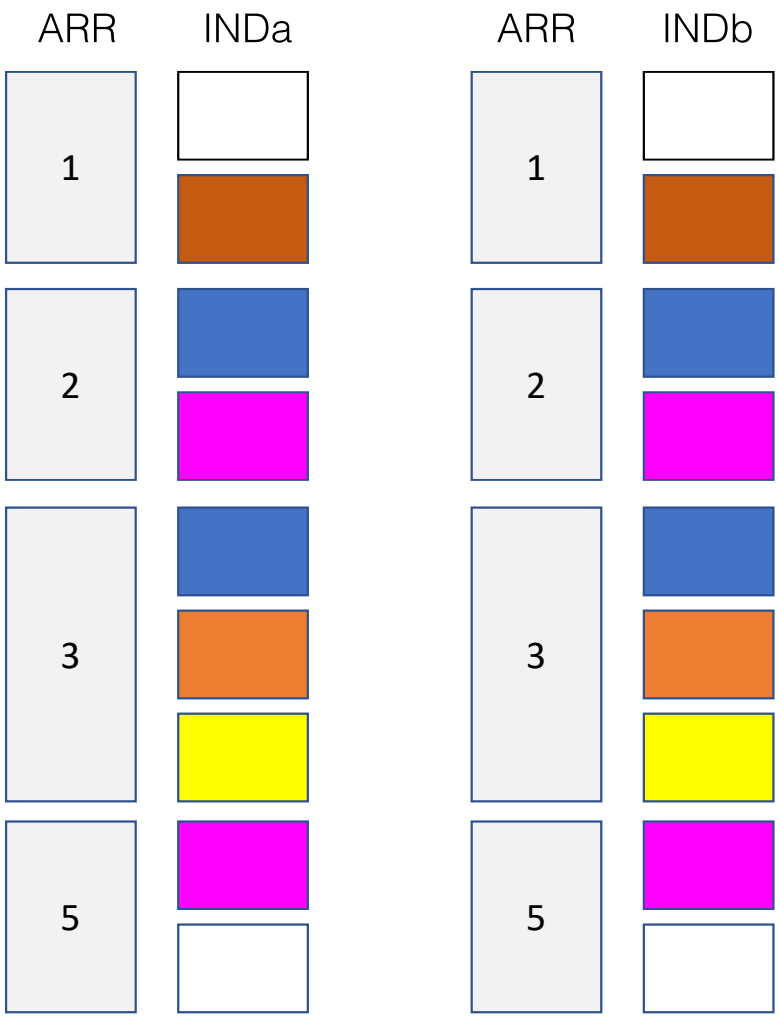
```
library(reshape2)
# Créer un numéro séquentiel pour cast
data.w <- data[order(data$DATE, data$ARR),]
data.w <- transform(data.w, Obs = seq_along(ARR) - match(ARR, ARR) + 1)
# Format large
data.w <- dcast(data.w,  ARR + DELIT + DATE ~ Obs, value.var="IND")

dyads <- merge(data, data.w, by=c("ARR", "DELIT", "DATE")) # Fusionner le large
dyads <- melt(dyads, id.vars=1:4, na.rm=TRUE) # Dyades format long
dyads$variable <- NULL

dyads <- dyads[dyads$IND != dyads$value] # Effacer loops
```

IND	ARR	DATE	DÉLIT
WHI	0	01/01/2001	A
BLU	0	01/01/2001	A
WHI	1	02/01/2001	A
BRU	1	02/01/2001	A
BLU	2	03/01/2001	A
PIN	2	03/01/2001	A
BLU	3	04/01/2001	B
ORA	3	04/01/2001	B
YEL	3	04/01/2001	B
WHI	5	05/01/2001	A
PIN	5	05/01/2001	A
WHI	6	06/01/2001	A

ARR	DATE	DÉLIT	INDa	INDb
0	01/01/2001	A	WHI	BLU
1	02/01/2001	A	WHI	BRU
2	03/01/2001	A	BLU	PIN
3	04/01/2001	B	BLU	ORA
3	04/01/2001	B	BLU	YEL
3	04/01/2001	B	ORA	BLU
5	05/01/2001	A	WHI	PIN

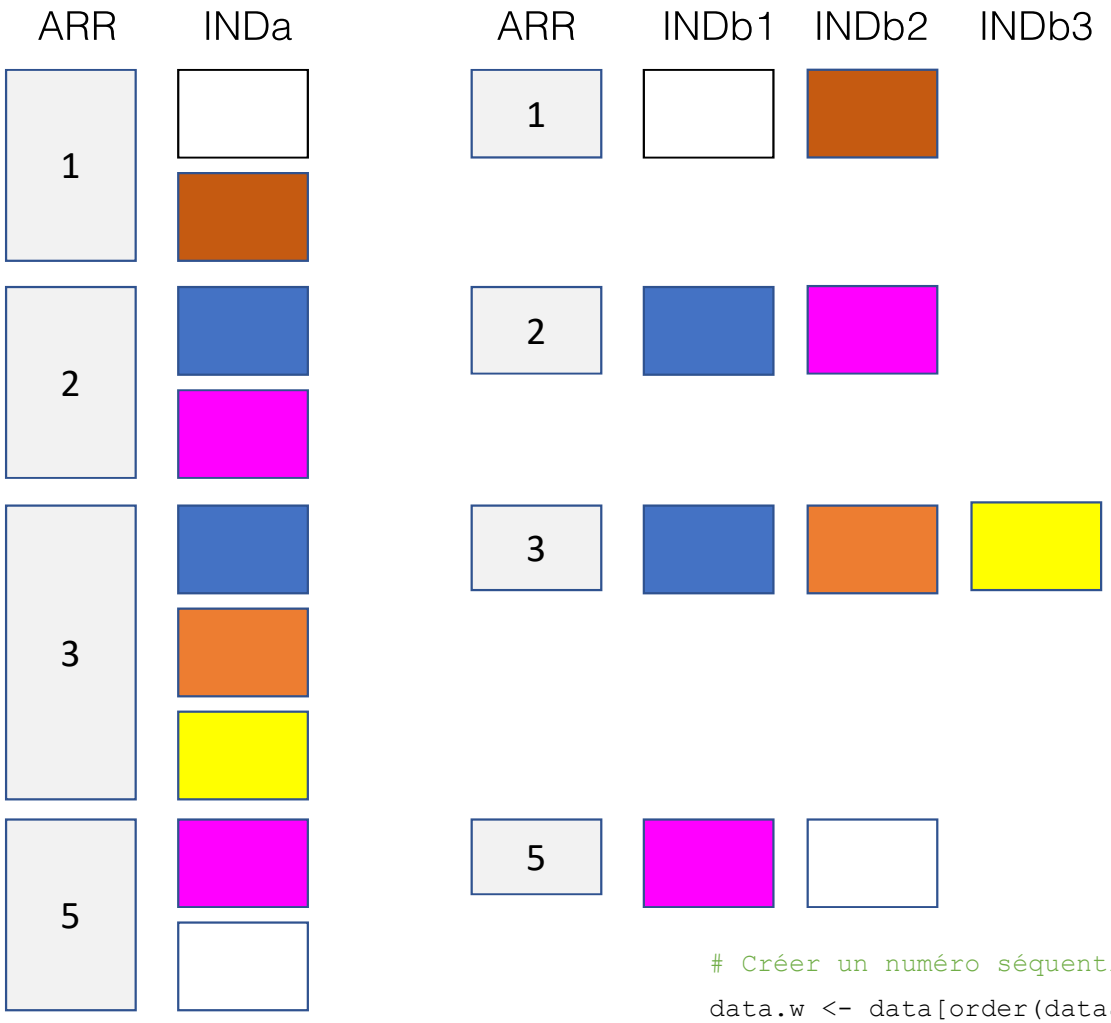


DONNÉES INDIVIDUELLES

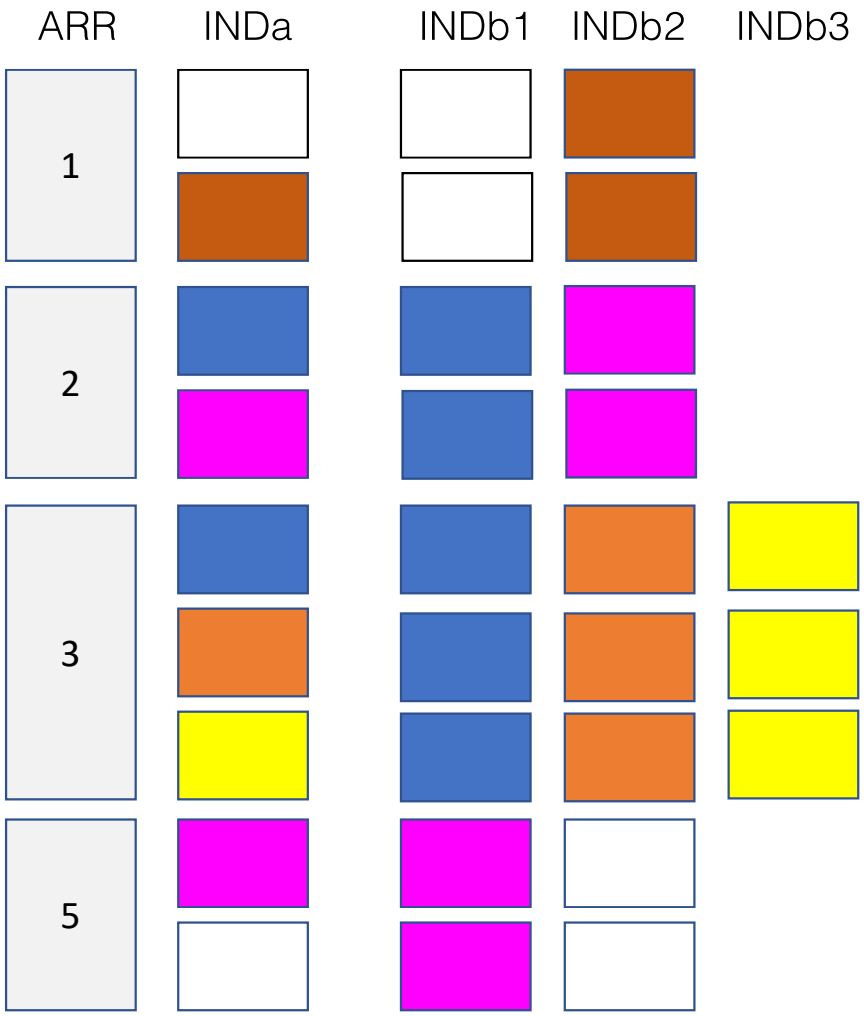


DCAST

DCAST

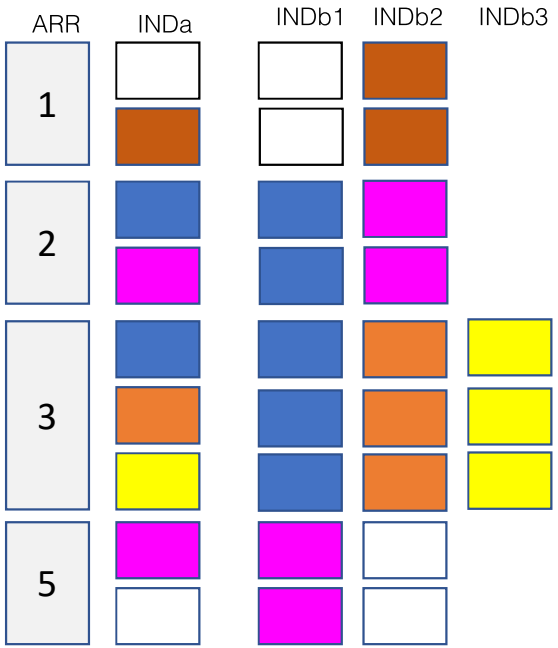



```
# Créer un numéro séquentiel pour cast
data.w <- data[order(data$DATE, data$ARR),]
data.w <- transform(data.w, Obs = seq_along(ARR) - match(ARR, ARR) + 1)
# Format large
data.w <- dcast(data.w,  ARR + DELIT + DATE ~ Obs, value.var="IND")
```



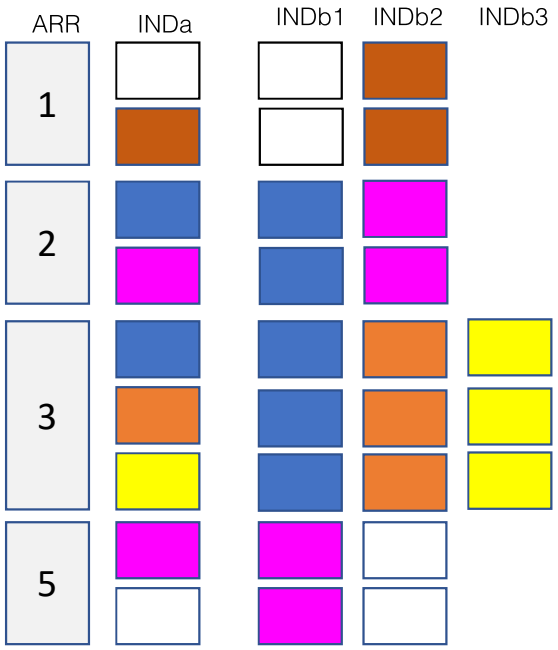
MERGE

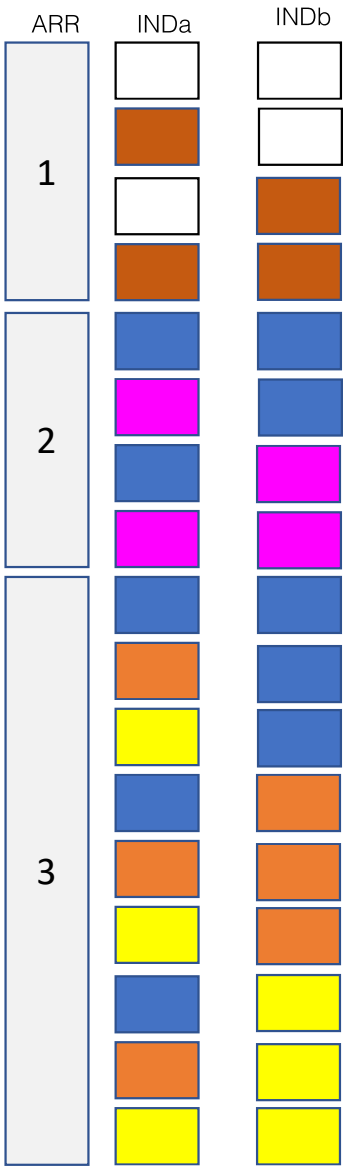
```
dyads <- merge(data, data.w, by=c("ARR", "DELIT", "DATE")) # Fusionner le large
```



MELT

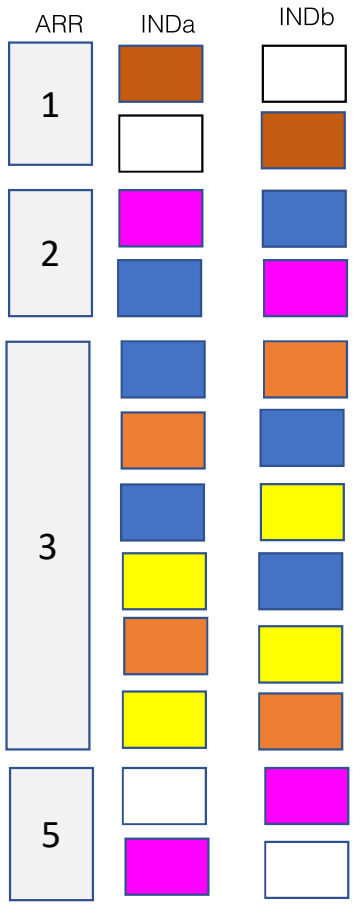
```
dyads <- melt(dyads, id.vars=1:4, na.rm=TRUE) # Dyades format long
```

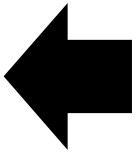


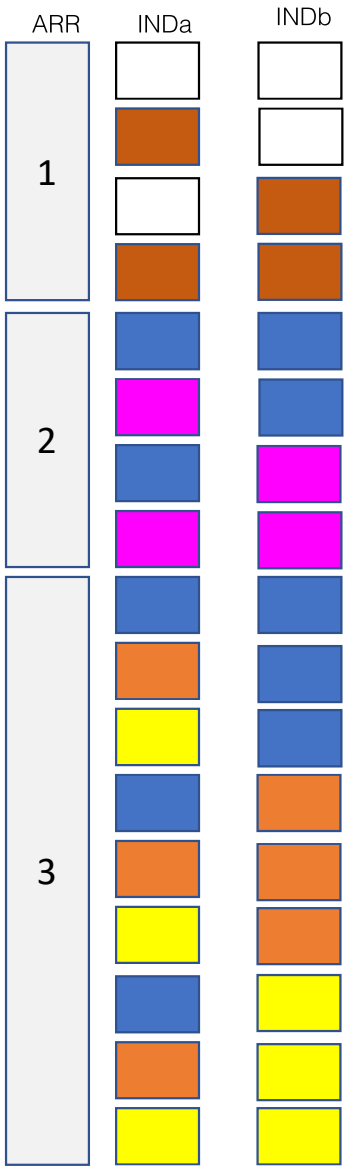


MELT

```
dyads <- melt(dyads, id.vars=1:4, na.rm=TRUE) # Dyades format long
```





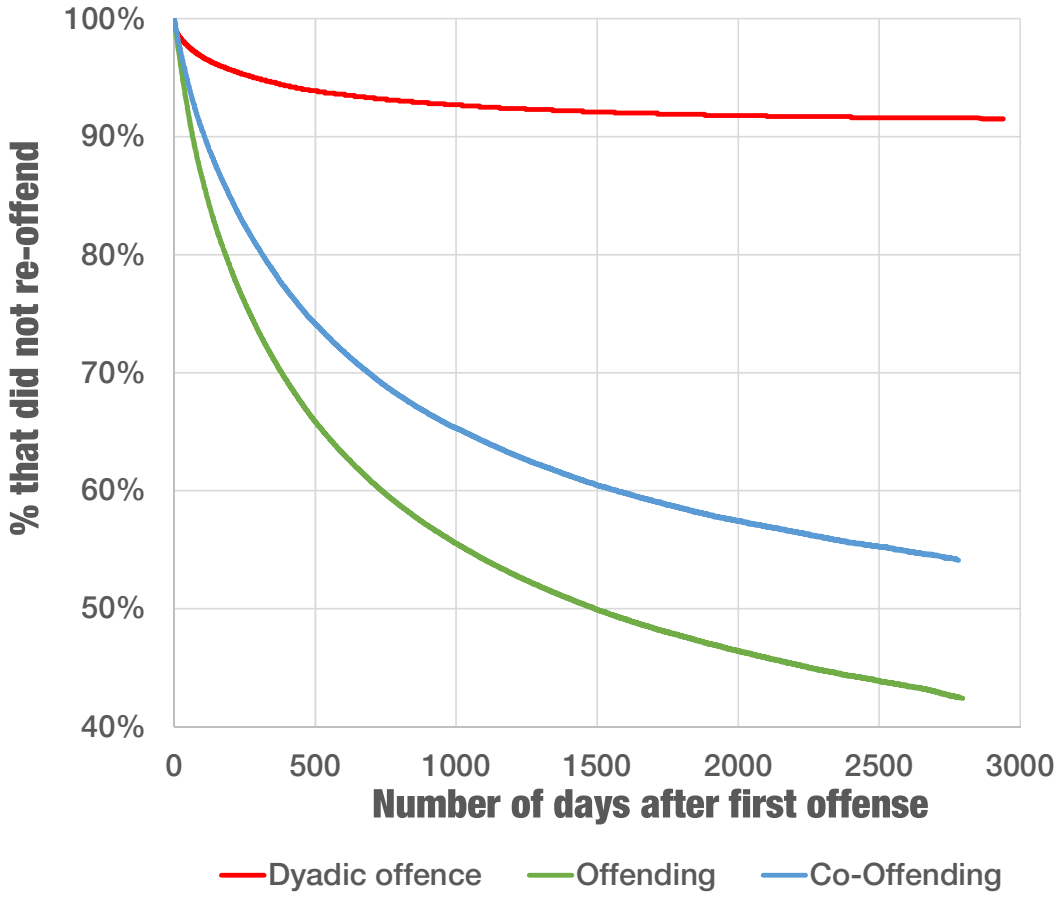


CLEAN

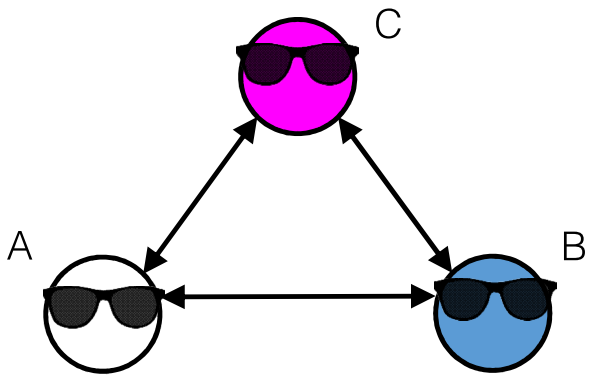
```
dyads <- dyads[dyads$IND != dyads$value] # Effacer loops
```

RESULTS

TAUX DE RÉCIDIVE

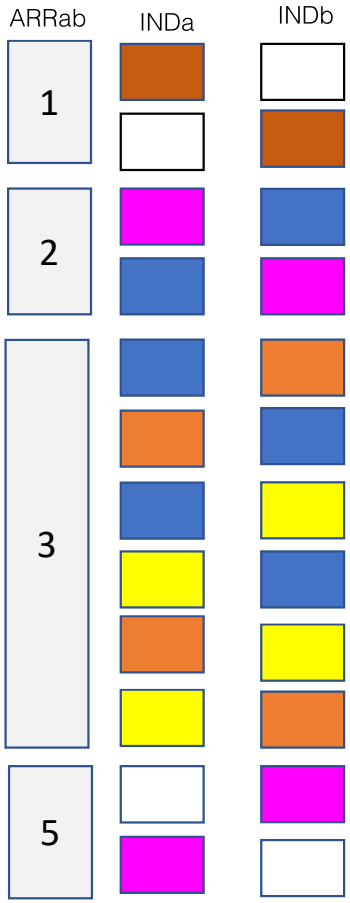


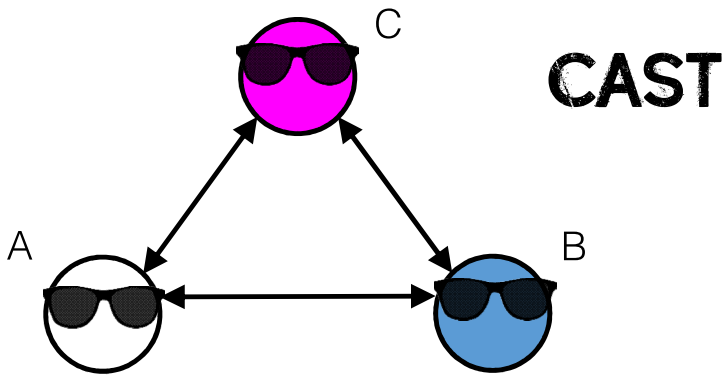
STRUCTURE DES DONNÉES



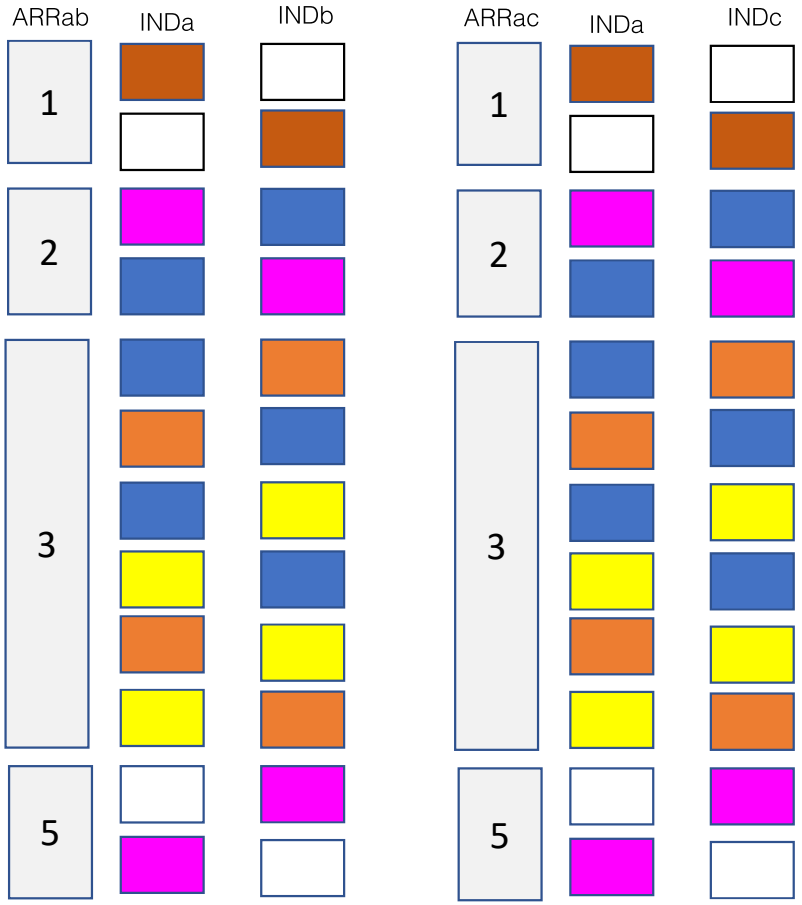
Création de triades :

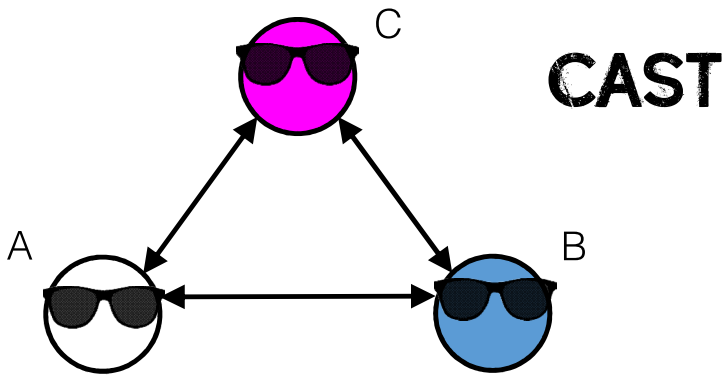
```
# Créer un numéro séquentiel pour les dyades
dyads.ind <- dyads[order(dyads$IND, dyads$DATE),]
dyads.ind <- transform(dyads.ind, Obs = seq_along(IND) - match(IND, IND) + 1)

# Créer une banque diadique par individus uniques
dyads.ind <- melt(dyads.ind, id.vars=c("IND", "Obs"))
dyads.ind <- dcast(dyads.ind, IND ~ variable + Obs, value.var="value")

# Trouver les liens entre A et C
triads <- merge(dyads, dyads.ind, by="IND")
triads <- melt(triads, id.var=1:5, measure.vars=patterns("ARR_", "DELIT_", "DATE_", "value_"), value.name=c("ARRac", "DELITac",
"DATEac", "INDc"), na.rm=TRUE)
triads$variable <- NULL

# Trouver les liens entre C et D
triads <- merge(triads, dyads.ind, by.x="INDc", by.y="IND")
triads <- melt(triads, id.var=1:9, measure.vars=patterns("ARR_", "DELIT_", "DATE_", "value_"), value.name=c("ARRcd", "DELITcd",
"DATEcd", "INDd"), na.rm=TRUE)
triads$variable <- NULL

# Garder les loops (D == C)
triads <- triads[triads$INDd == triads$value]
```



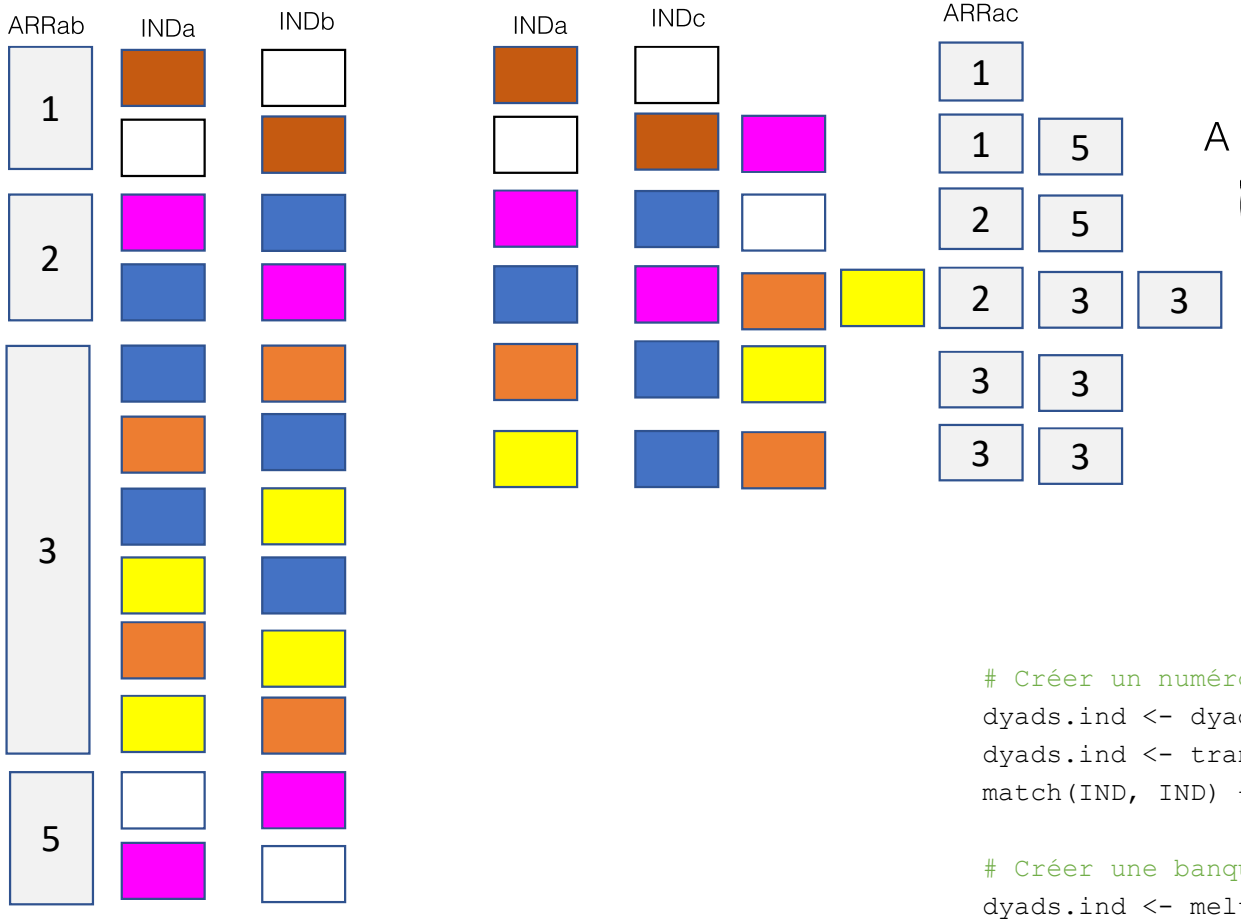


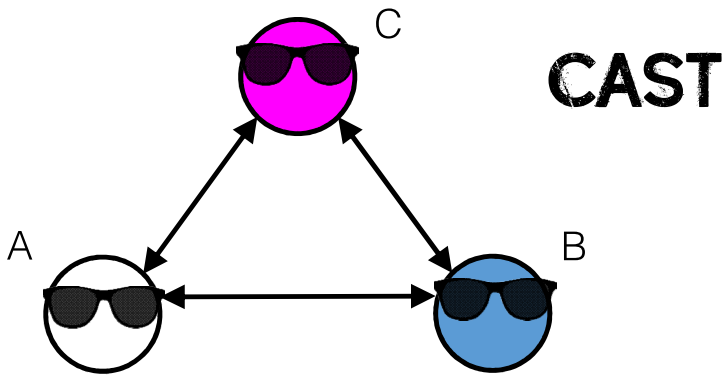

```
# Créer un numéro séquentiel pour les dyades
dyads.ind <- dyads[order(dyads$IND, dyads$DATE),]
dyads.ind <- transform(dyads.ind, Obs = seq_along(IND) -
match(IND, IND) + 1)

# Créer une banque diadique par individus uniques
dyads.ind <- melt(dyads.ind, id.vars=c("IND", "Obs"))
dyads.ind <- dcast(dyads.ind, IND ~ variable + Obs,
value.var="value")
```



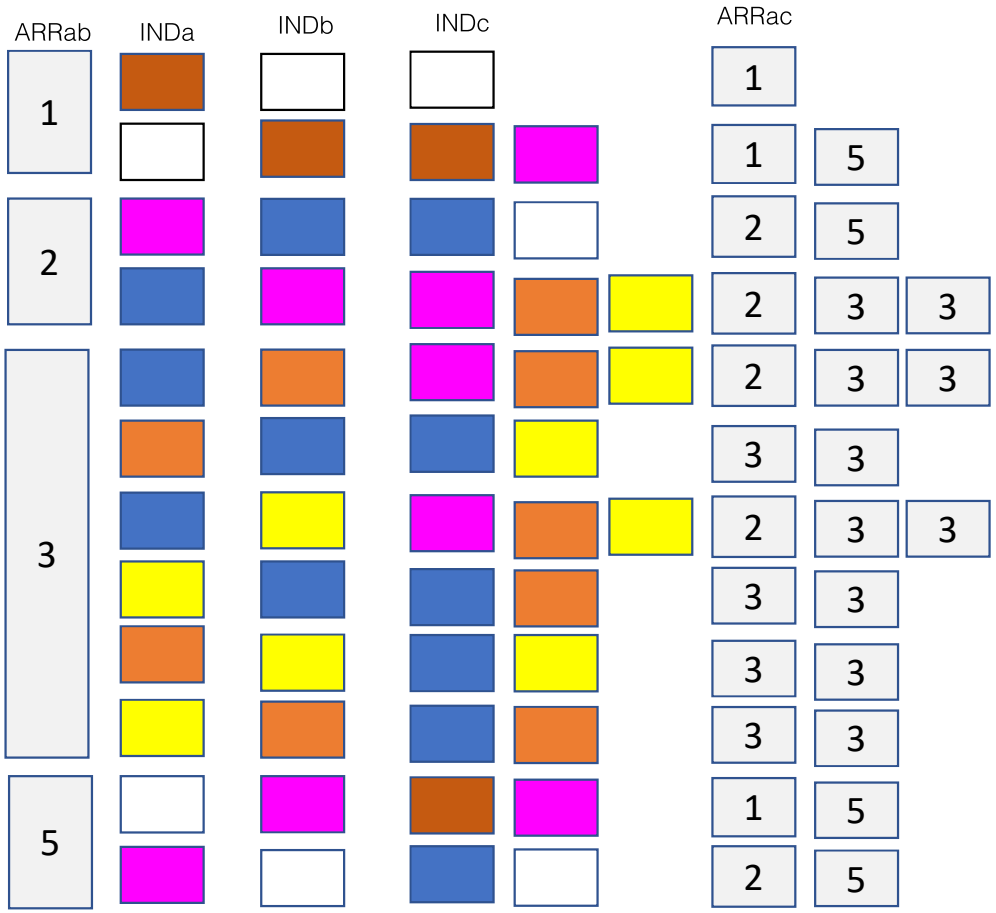


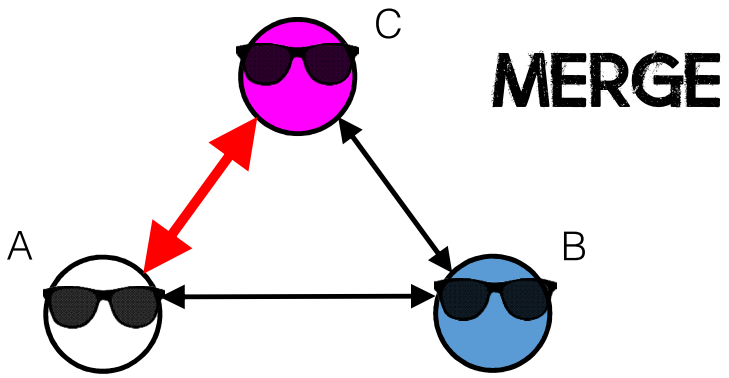

```
# Créer un numéro séquentiel pour les dyades
dyads.ind <- dyads[order(dyads$IND, dyads$DATE),]
dyads.ind <- transform(dyads.ind, Obs = seq_along(IND) -
match(IND, IND) + 1)

# Créer une banque diadique par individus uniques
dyads.ind <- melt(dyads.ind, id.vars=c("IND", "Obs"))
dyads.ind <- dcast(dyads.ind, IND ~ variable + Obs,
value.var="value")
```



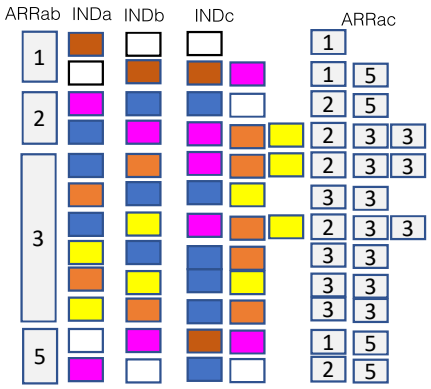


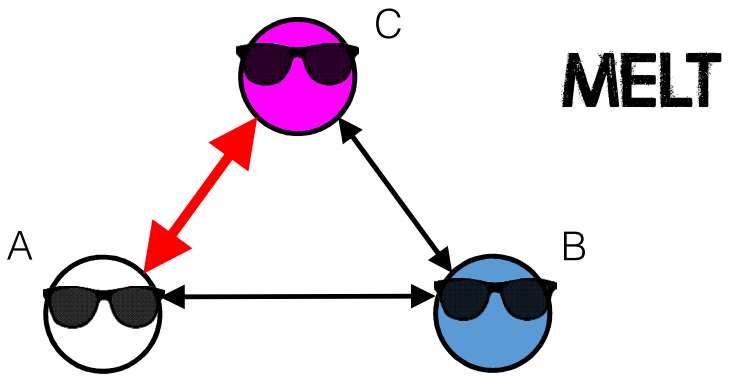

```
# Créer un numéro séquentiel pour les dyades
dyads.ind <- dyads[order(dyads$IND, dyads$DATE),]
dyads.ind <- transform(dyads.ind, Obs = seq_along(IND) -
match(IND, IND) + 1)

# Créer une banque diadique par individus uniques
dyads.ind <- melt(dyads.ind, id.vars=c("IND", "Obs"))
dyads.ind <- dcast(dyads.ind, IND ~ variable + Obs,
value.var="value")
```



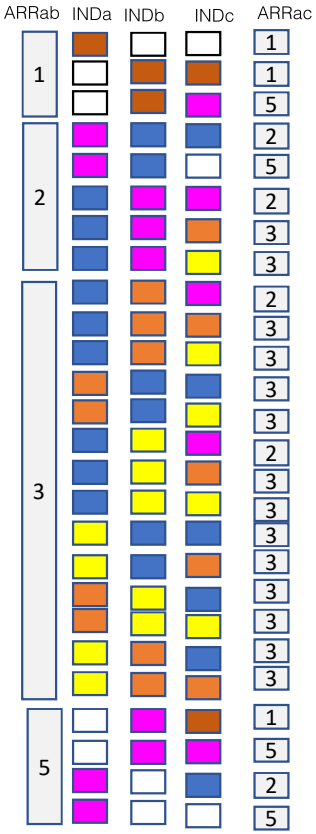


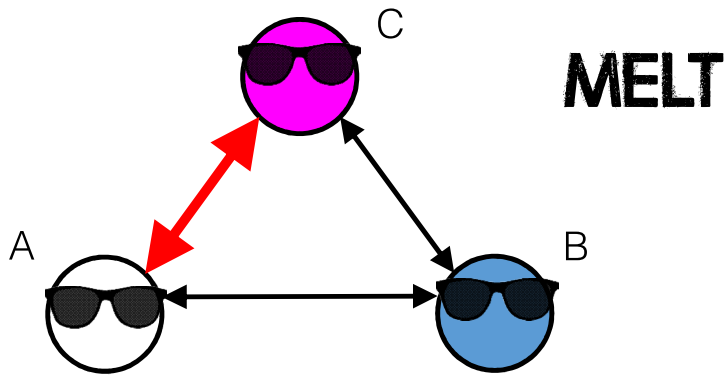

```
# Trouver les liens entre A et C
triads <- merge(dyads, dyads.ind, by="IND")
```



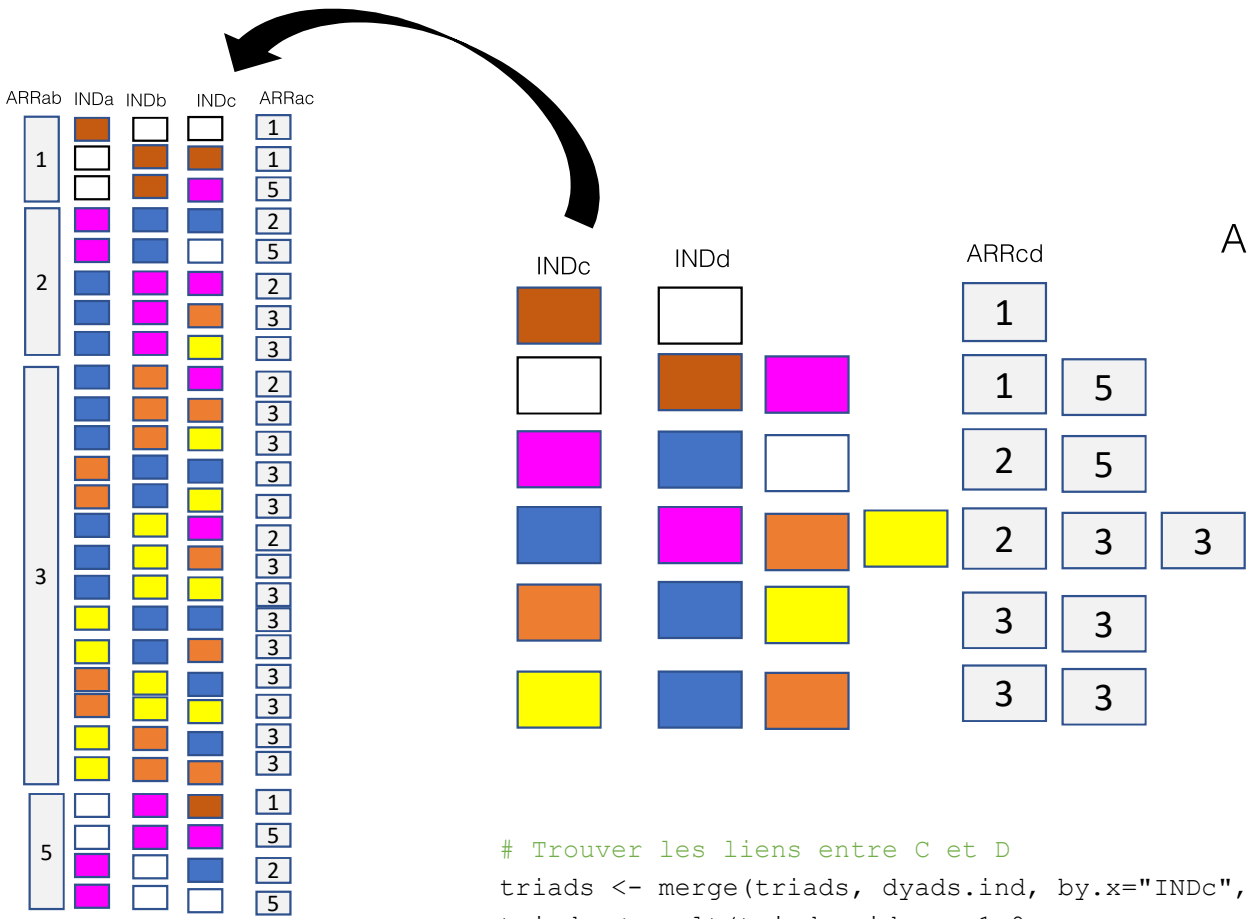


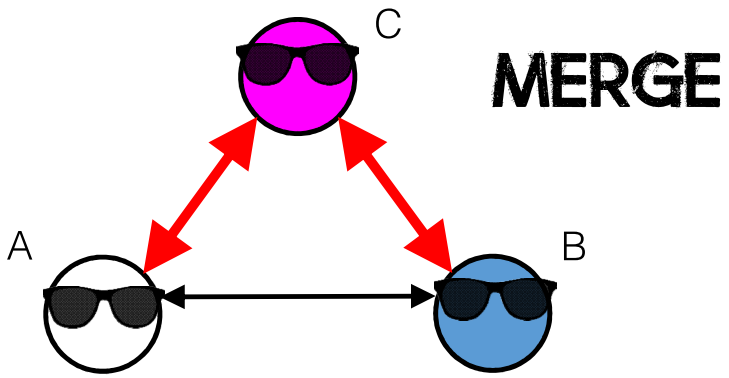

```
triads <- melt(triads, id.var=1:5, measure.vars=patterns("ARR_", "DELIT_", "DATE_", "value_"),
value.name=c("ARRac", "DELITac", "DATEac", "INDc"), na.rm=TRUE)
```



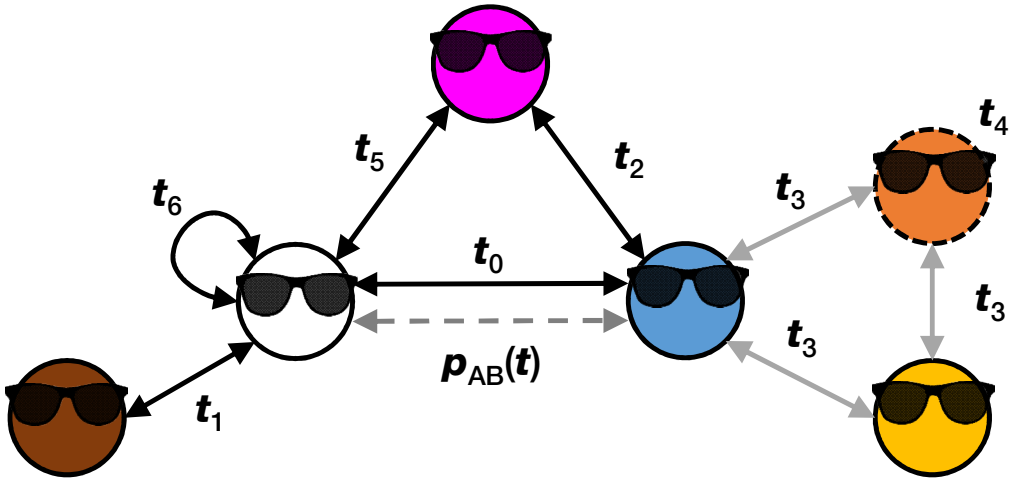



```
triads <- melt(triads, id.var=1:5, measure.vars=patterns("ARR_", "DELIT_", "DATE_", "value_"),
value.name=c("ARRac", "DELITac", "DATEac", "INDc"), na.rm=TRUE)
```






```
# Trouver les liens entre C et D
triads <- merge(triads, dyads.ind, by.x="INDc", by.y="IND")
triads <- melt(triads, id.var=1:9, measure.vars=patterns("ARR_", "DELIT_", "DATE_",
"value_"), value.name=c("ARRcd", "DELITcd", "DATEcd", "INDd"), na.rm=TRUE)
triads$variable <- NULL

# Garder les loops (D == C)
triads <- triads[triads$INDd == triads$value]
```



INDa	INDc	INDb	DELITab	DELITac	DELITcb	DATEab	DATEac	DATEcb
PIN	BLU	WHI	A	A	A	2001-01-05	2001-01-02	2001-01-01
YEL	ORA	BLU	B	B	B	2001-01-04	2001-01-03	2001-01-03
WHI	PIN	BLU	A	A	A	2001-01-01	2001-01-04	2001-01-02
PIN	WHI	BLU	A	A	A	2001-01-03	2001-01-04	2001-01-01
ORA	YEL	BLU	B	B	B	2001-01-04	2001-01-03	2001-01-03
WHI	BLU	PIN	A	A	A	2001-01-05	2001-01-01	2001-01-02
BLU	ORA	YEL	B	B	B	2001-01-04	2001-01-03	2001-01-03
BLU	PIN	WHI	A	A	A	2001-01-01	2001-01-02	2001-01-04
BLU	YEL	ORA	B	B	B	2001-01-04	2001-01-03	2001-01-03
YEL	BLU	ORA	B	B	B	2001-01-04	2001-01-03	2001-01-03
BLU	WHI	PIN	A	A	A	2001-01-03	2001-01-01	2001-01-04
ORA	BLU	YEL	B	B	B	2001-01-04	2001-01-03	2001-01-03

UN PROBLÈME DE TAILLE...

- Des données à un niveau de précision de la journée.
- Des données nichées, donc dépendantes.
- Les interactions de l'entourage des individus.

- n = 181 615 individus; t = 8 ans

- 10 809 140 points de mesures…

UN PROBLÈME DE TAILLE…

- Modèles à risques proportionnels avec covariées dépendantes temporellement (*survival*)
- Dépendance : Effet aléatoire… (*frailty*)

```
Model1 <- coxph(Surv(time=Sample.dta$TimeA, time2=Sample.dta$TimeB,
        event=Sample.dta$Fail) ~ GenderA + GenderB + RaceABlk + RaceAHis + RaceAOth +
        RaceBBlk + RaceBHis + RaceBOth + log(age.B + 1) + log(age.A + 1) + Gang_A +
        Gang_B + log(Victimisation.A + 1) + log(SecondVictimisation.A + 1) +
        log(Victimisation.B + 1) + log(SecondVictimisation.B + 1) + log(CoArrest.A + 1) +
        log(SoloArrest.A + 1) + log(CoArrest.B + 1) + log(SoloArrest.B + 1) + OccurenceLN +
        OccurenceLNsq + D_Violent + D_Market + D_Justice + GenderHomo + RaceHomo +
        log(AgeHomo + 1) + DiffGang + SameGang + LnDistanceNoMiss + log(NewTie.A + 1) +
        log(NewTie.B + 1) + log(Tryad + 1) + cluster(DyadA) +  frailty(DyadB), data=Sample.dta)
```

UN PROBLÈME DE TAILLE...

- Compromis
  - Sous échantillon avec mesures populationnelles
  - *Cluster*
  - 100 itérations avec résultats groupés (*pooled*)

```
library(foreach)
results <- foreach(i=1:100, .combine='rbind')%do% {
  Sample.R <- sample(Dyads.R, 2435)
  Sample.NR <- sample(Dyads.NR, 4896)
  Sample <- as.data.table(c(Sample.R, Sample.NR))
  Sample.dta <- merge(Dyads.dta, Sample, all.y=TRUE, all.x=FALSE, by.x="DyadID2", by.y="V1")
  Model1 <- coxph(Surv(time=Sample.dta$TimeA, time2=Sample.dta$TimeB, event=Sample.dta$Fail) ~
           GenderA + GenderB + RaceABlk + RaceAHis + RaceAOth + RaceBBlk + RaceBHis + RaceBOth +
           log(age.B + 1) + log(age.A + 1) + Gang_A + Gang_B + log(Victimisation.A + 1) +
           log(SecondVictimisation.A + 1) + log(Victimisation.B + 1) +
           log(SecondVictimisation.B + 1) + log(CoArrest.A + 1) + log(SoloArrest.A + 1) +
           log(CoArrest.B + 1) + log(SoloArrest.B + 1) + OccurenceLN + OccurenceLNsq + D_Violent + D_Market + D_Justice +
           GenderHomo + RaceHomo + log(AgeHomo + 1) + DiffGang + SameGang + LnDistanceNoMiss + log(NewTie.A + 1) +
           log(NewTie.B + 1) + log(Tryad + 1) + cluster(DyadA) +  + cluster(DyadB), data=Sample.dta)
  summary(Model1)$coefficients
}
```

POUR LES RÉSULTATS

Charette, Y., & Papachristos, A. V. (2017). The network dynamics of co-offending careers. *Social Networks*.

YANICK CHARETTE, PH.D.

YANICK.CHARETTE.1@ULAVAL.CA

R À QUÉBEC, 26 MAI 2017

UNIVERSITÉ LAVAL

MERCI!

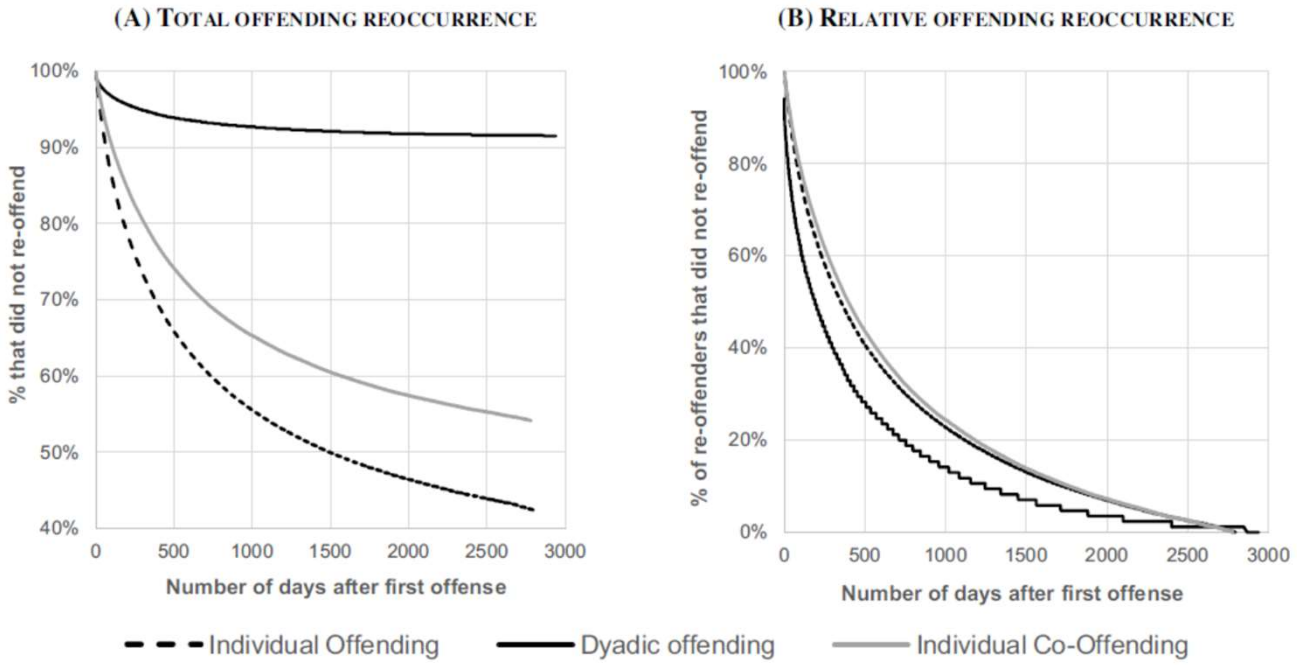


Fig. 1. Total (a) and relative (b) survival curves before the reoccurrence of individual offending, individual co-offending, and dyadic offending.

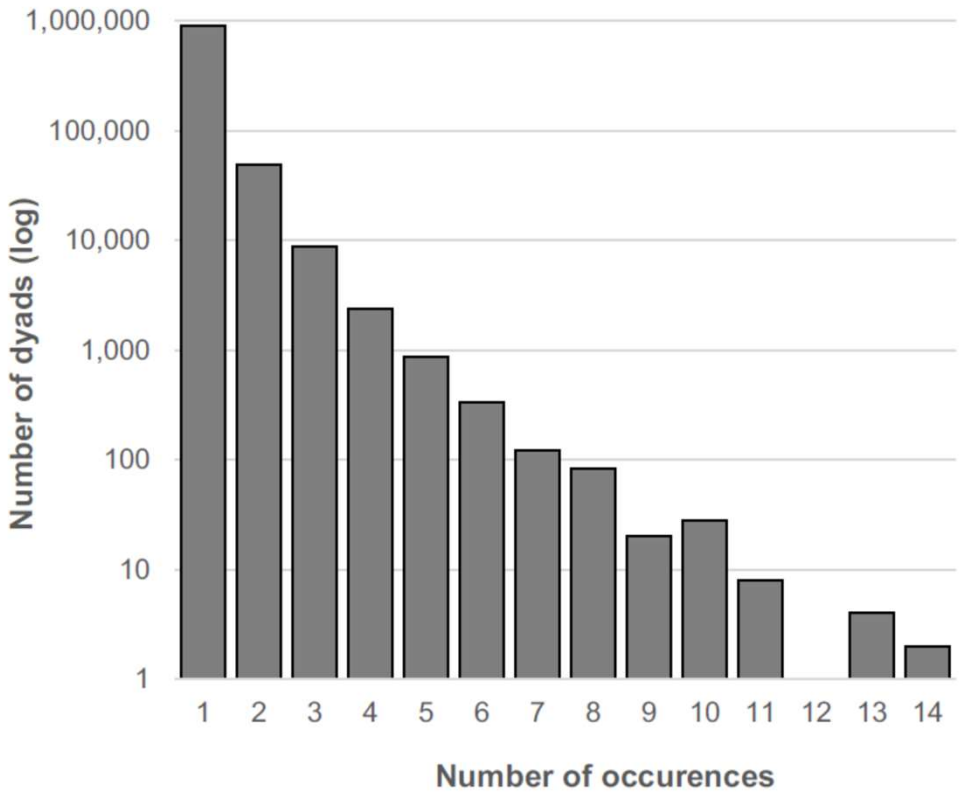


Fig. 3. Number of co-offenses per dyads.

Table 2
Survival models predicting reoccurrence of a dyad (n = 8,621offenders; n = 14,640 dyads).

	Model I			Model II		
	β	Exp(b)	(95% CI)	β	Exp(b)	(95% CI)
Nodal characteristics						
Male	0.04 *	1.14	(1.02–1.29)	0.16 **	1.18	(1.05–1.32)
Race (reference = White)						
Black	0.04	1.09	(0.92–1.29)	0.10	1.12	(0.95–1.33)
Hispanic	0.04	1.12	(0.96–1.31)	0.12	1.14	(0.97–1.33)
Others	0.00	1.05	(0.70–1.57)	0.01	1.04	(0.69–1.56)
Age (log)	−0.05 ***	0.82	(0.73–0.92)	−0.17 **	0.71	(0.69–0.73)
Gang member	0.22 ***	1.56	(1.44–1.69)	0.47 ***	1.60	(1.48–1.73)
Victimization (log)	0.04 *	1.15	(1.03–1.29)	0.17 ***	1.19	(1.06–1.33)
2nd order victimization (log)	−0.14 ***	0.92	(0.89–0.94)	−0.09 ***	0.91	(0.89–0.94)
Number co-arrests	0.44 ***	1.98	(1.81–2.16)	–	–	–
Same type	–	–	–	0.57 ***	1.78	(1.63–1.94)
Different type	–	–	–	0.11 **	1.12	(1.04–1.21)
Number of solo arrests	0.06 ***	1.06	(1.03–1.10)	–	–	–
Same type	–	–	–	0.06 **	1.06	(1.02–1.11)
Different type	–	–	–	0.01	1.01	(0.97–1.05)
Dyadic characteristics						
Past of dyadic co-arrests (log)	0.95 ***	10.40	(8.73–12.38)	2.17 ***	8.54	(7.14–10.21)
Past of dyadic co-arrests (log; sq)	−0.29 ***	0.52	(0.46–0.60)	−0.58 ***	0.57	(0.49–0.66)
Type of offense (reference = Predatory)						
Violent	−0.05 ***	0.85	(0.79–0.91)	−0.12	0.93	(0.87–1.00)
Market	−0.10 ***	0.80	(0.76–0.85)	−0.07 *	0.92	(0.86–0.98)
Justice	−0.05 ***	0.91	(0.87–0.96)	−0.04	1.04	(0.97–1.11)
Homophily						
Gender	0.12 ***	1.55	(1.39–1.73)	0.43 ***	1.52	(1.36–1.70)
Race	0.05 ***	1.23	(1.12–1.35)	0.20 ***	1.21	(1.10–1.33)
Age differential (log)	−0.27 ***	0.71	(0.69–0.73)	−0.35 ***	0.71	(0.69–0.73)
Same gang	0.16 ***	1.41	(1.27–1.55)	0.31 ***	1.37	(1.24–1.52)
Different gang	−0.11 ***	0.67	(0.59–0.76)	−0.41 ***	0.67	(0.59–0.76)
Geographic distance (log)	−0.16 ***	0.96	(0.96–0.97)	−0.04 ***	0.96	(0.96–0.97)
Network structure						
Centrality (log)	−0.28 ***	0.69	(0.64–0.74)	–	–	–
Same Type (log)	–	–	–	−0.34 ***	0.71	(0.66–0.76)
Different Type (log)	–	–	–	0.04	1.03	(0.98–1.09)
Transitivity (log)	0.07 **	1.06	(1.02–1.10)	–	–	–
Same Type (log)	–	–	–	0.08 ***	1.08	(1.04–1.13)
Different Type (log)	–	–	–	0.01	1.02	(0.98–1.06)
Likelihood ratio test	3262.2; df = 34; p < 0.001			3218.9; df = 41; p < 0.001		
AIC	41370.4			41350.8		

* p < 0.05.
** p < 0.01.
*** p < 0.001.

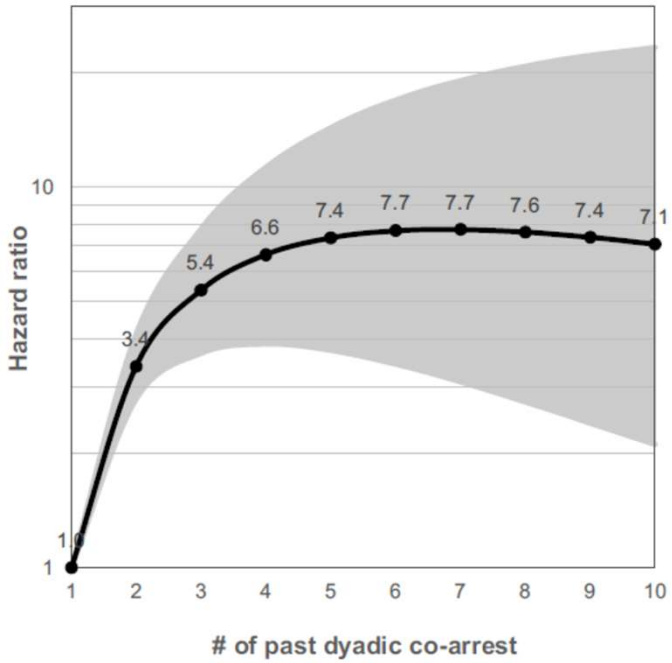


Fig. 4. Hazard ratio in function of the number of past experiences of cooperation (95% confidence interval).