Supplemental Table 1

References for Studies and Data Sets Included in Meta-analysis Models

	Source ID	Sample ID
Atzaba-Poria, N., & Pike, A. (2008). Correlates of parental differential treatment: Parental and contextual factors during middle childhood. *Child Development, 79*(1), 217-232. doi:10.1111/j.1467-8624.2007.01121.x	1	1
Buist, K. L. (2015). *Elementary children sample 1: Mixed-sex sibling pairs* [Unpublished raw data].	2	2
Buist, K. L. (2015). *Elementary children sample 1: Same-sex sibling pairs* [Unpublished raw data].	3	2
Buist, K. L. (2015). *Elementary children sample 2: Mixed-sex sibling pairs* [Unpublished raw data].	4	3
Buist, K. L. (2015). *Elementary children sample 2: Same-sex sibling pairs* [Unpublished raw data].	5	3
Buist, K. L. (2015). *High school children: Mixed-sex sibling pairs* [Unpublished raw data].	6	4
Buist, K. L. (2015). *High school children: Same-sex sibling pairs* [Unpublished raw data].	7	4
Buist, K. L. (2015). *Kindergarten children sample 1: Mixed-sex sibling pairs* [Unpublished raw data].	8	5
Buist, K. L. (2015). *Kindergarten children sample 1: Same-sex sibling pairs* [Unpublished raw data].	9	5
Buist, K. L. (2015). *Kindergarten children sample 2: Mixed-sex sibling pairs* [Unpublished raw data].	10	6
Buist, K. L. (2015). *Kindergarten children sample 2: Same-sex sibling pairs* [Unpublished raw data].	11	6
Campione-Barr, N. (2007–2008). *Parent and siblings relationship study: Mixed-sex sibling pairs* [Unpublished raw data].	12	7
Campione-Barr, N. (2007–2008). *Parent and siblings relationship study: Same-sex sibling pairs* [Unpublished raw data].	13	7
Day, R. D., Bean, R. A., Coyne, S. M., Dyer, W. M., Harper, J. M., Padilla-Walker, L. M. (2013). *Flourishing families: Mixed-sex sibling pairs—Time 6* [Unpublished raw data].	14	8

Day, R. D., Bean, R. A., Coyne, S. M., Dyer, W. M., Harper, J. M., Padilla-Walker, L. M. (2013). *Flourishing families: Same-sex sibling pairs—Time 6* [Unpublished raw data].	15	8
Day, R. D., Bean, R. A., Coyne, S. M., Dyer, W. M., Harper, J. M., Padilla-Walker, L. M. (2014). *Flourishing families: Mixed-sex sibling pairs—Time 7* [Unpublished raw data].	16	8
Day, R. D., Bean, R. A., Coyne, S. M., Dyer, W. M., Harper, J. M., Padilla-Walker, L. M. (2014). *Flourishing families: Same-sex sibling pairs—Time 7* [Unpublished raw data].	17	8
Day, R. D., Bean, R. A., Coyne, S. M., Dyer, W. M., Harper, J. M., Padilla-Walker, L. M. (2015). *Flourishing families: Mixed-sex sibling pairs—Time 8* [Unpublished raw data].	18	8
Day, R. D., Bean, R. A., Coyne, S. M., Dyer, W. M., Harper, J. M., Padilla-Walker, L. M. (2015). *Flourishing families: Same-sex sibling pairs—Time 8* [Unpublished raw data].	19	8
Jenkins, J. M., Rasbash, J., & O'Connor, T. G. (2003). The role of the shared family context in differential parenting. *Developmental Psychology, 39*(1), 99-113. doi:10.1037/0012-1649.39.1.99	20	9
Jensen, A. C. (2016). *Baby Boomers' Relationships with Siblings: Mixed-sex sibling pairs* [Unpublished raw data].	21	10
Jensen, A. C. (2016). Baby Boomers' Relationships with Siblings: Same-sex sibling pairs. Unpublished raw data.	22	10
Jensen, A. C. (2016). *Young adults' relationships with siblings: Mixed-sex sibling pairs* [Unpublished raw data].	23	11
Jensen, A. C. (2016). *Young adults' relationships with siblings: Same-sex sibling pairs* [Unpublished raw data].	24	11
Jensen, A. C. (2020) *Sibling influences on becoming adults study: Mixed-sex sibling pairs--Time 1* [Unpublished raw data].	25	12
Jensen, A. C. (2020) *Sibling influences on becoming adults study: Same-sex sibling pairs--Time 1* [Unpublished raw data].	26	12
Jensen, A. C., Whiteman, S. D., Rand, J. S., & Fingerman, K. L. (2017). You're just like your dad: Intergenerational patterns of differential treatment of siblings. *The Journals of Gerontology: Series B: Psychological Sciences and Social Sciences, 72(6), 1073–1083.* doi:10.1093/geronb/gbw033	27	13
McHale, S. M., & Crouter, A. C. (1998). *Adolescent project: Mixed-sex sibling pairs—Time 1* [Unpublished raw data].	28	14

McHale, S. M., & Crouter, A. C. (1998). *Adolescent project: Same-sex sibling pairs—Time 1* [Unpublished raw data].	29	14
McHale, S. M., & Crouter, A. C. (1999). *Adolescent project: Mixed-sex sibling pairs—Time 2* [Unpublished raw data].	30	14
McHale, S. M., & Crouter, A. C. (1999). *Adolescent project: Same-sex sibling pairs—Time 2* [Unpublished raw data].	31	14
McHale, S. M., & Crouter, A. C. (2000). *Adolescent project: Mixed-sex sibling pairs—Time 3* [Unpublished raw data].	32	14
McHale, S. M., & Crouter, A. C. (2000). *Adolescent project: Same-sex sibling pairs—Time 3* [Unpublished raw data].	33	14
McHale, S. M., & Crouter, A. C. (2004). *African American study: Mixed-sex sibling pairs—Time 1* [Unpublished raw data].	34	15
McHale, S. M., & Crouter, A. C. (2004). *African American study: Same-sex sibling pairs—Time 1* [Unpublished raw data].	35	15
McHale, S. M., & Crouter, A. C. (2005). *African American study: Mixed-sex sibling pairs—Time 2* [Unpublished raw data].	36	15
McHale, S. M., & Crouter, A. C. (2005). *African American study: Same-sex sibling pairs—Time 2* [Unpublished raw data].	37	15
McHale, S. M., & Crouter, A. C. (2006). *African American study: Mixed-sex sibling pairs—Time 3* [Unpublished raw data].	38	15
McHale, S. M., & Crouter, A. C. (2006). *African American study: Same-sex sibling pairs—Time 3* [Unpublished raw data].	39	15
McHale, S. M., & Crouter, A. C. (1996). *Middle childhood project: Mixed-sex sibling pairs—Time 1* [Unpublished raw data].	40	16
McHale, S. M., & Crouter, A. C. (1996). *Middle childhood project: Same-sex sibling pairs—Time 1* [Unpublished raw data].	41	16
McHale, S. M., & Crouter, A. C. (1997). *Middle childhood project: Mixed-sex sibling pairs—Time 2* [Unpublished raw data].	42	16
McHale, S. M., & Crouter, A. C. (1997). *Middle childhood project: Same-sex sibling pairs—Time 2* [Unpublished raw data].	43	16
McHale, S. M., & Crouter, A. C. (1998). *Middle childhood project: Mixed-sex sibling pairs—Time 3* [Unpublished raw data].	44	16
McHale, S. M., & Crouter, A. C. (1998). *Middle childhood project: Same-sex sibling pairs—Time 3* [Unpublished raw data].	45	16

McHale, S. M., & Crouter, A. C. (1999). Middle childhood project: Mixed-sex sibling pairs—Time 4 [Unpublished raw data].	46	16
McHale, S. M., & Crouter, A. C. (1999). Middle childhood project: Same-sex sibling pairs—Time 4 [Unpublished raw data].	47	16
McHale, S. M., & Crouter, A. C. (2000). Middle childhood project: Mixed-sex sibling pairs—Time 5 [Unpublished raw data].	48	16
McHale, S. M., & Crouter, A. C. (2000). Middle childhood project: Same-sex sibling pairs—Time 5 [Unpublished raw data].	49	16
McHale, S. M., & Crouter, A. C. (2001). Middle childhood project: Mixed-sex sibling pairs—Time 6 [Unpublished raw data].	50	16
McHale, S. M., & Crouter, A. C. (2001). Middle childhood project: Same-sex sibling pairs—Time 6 [Unpublished raw data].	51	16
McHale, S. M., & Crouter, A. C. (2002). Middle childhood project: Mixed-sex sibling pairs—Time 7 [Unpublished raw data].	52	16
McHale, S. M., & Crouter, A. C. (2002). Middle childhood project: Same-sex sibling pairs—Time 7 [Unpublished raw data].	53	16
McHale, S. M., & Crouter, A. C. (2003). Middle childhood project: Mixed-sex sibling pairs—Time 8 [Unpublished raw data].	54	16
McHale, S. M., & Crouter, A. C. (2003). Middle childhood project: Same-sex sibling pairs—Time 8 [Unpublished raw data].	55	16
McHale, S. M., & Crouter, A. C. (2004). Middle childhood project: Same-sex sibling pairs—Time 9 [Unpublished raw data].	56	16
McHale, S. M., & Crouter, A. C. (2006). Middle childhood project: Same-sex sibling pairs—Time 11 [Unpublished raw data].	57	16
McHale, S. M., & Crouter, A. C. (2006). Middle childhood project: Mixed-sex sibling pairs—Time 11 [Unpublished raw data].	58	16
Meunier, J. C., Roskam, I., Stievenart, M., Van De Moortele, G., Browne, D. T., & Wade, M. (2012). Parental differential treatment, child's externalizing behavior and sibling relationships: Bridging links with child's perception of favoritism and personality, and parents' self-efficacy. *Journal of Social and Personal Relationships, 29*(5), 612-638. doi:10.1177/0265407512443419	59	17
Padilla, J., McHale, S. M., De Jesús, S. A. R., Updegraff, K. A., & Umaña, T. A. J. (2018). Longitudinal course and correlates of parents' differential treatment of siblings in Mexican-origin families. *Family Process, 57*(4), 979–995. doi:10.1111/famp.12328	60	18

Rolan, E. & Marceau, K. (2018) Individual and sibling characteristics: Parental differential treatment and adolescent externalizing behaviors. *Journal of Youth and Adolescence, 47,* 2535-2553. doi:10.1007/s10964-018-0892-8	61	19
Solmeyer, A. R., Killoren, S. E., McHale, S. M., & Updegraff, K. A. (2011). Coparenting around siblings' differential treatment in Mexican-origin families. *Journal of Family Psychology, 25*(2), 251. doi:10.1037/a0023201	62	20
Suitor, J. J., Sechrist, J., & Pillemer, K. (2007). Within-family differences in mothers' support to adult children in Black and White families. *Research on Aging*, *29*(5), 410-435. doi:10.1177/0164027507303636	63	21
Volling, B. L. (1997). The family correlates of maternal and paternal perceptions of differential treatment in early childhood. *Family Relations*, 227-236.	64	22
Whiteman, S. D. (2009-2010). The Purdue, parent, and adolescent sibling study: Mixed-sex sibling pairs [Unpublished raw data].	65	23
Whiteman, S. D. (2009-2010). The Purdue, parent, and adolescent sibling study: Same-sex sibling pairs [Unpublished raw data].	66	23

Supplemental Table 2

Source and Effect-Size Characteristics for Models Examining Correlations Between Age Spacing and PDT

			Sample/Source Characteristics											Effect-size Characteristics				
Source ID	Sample ID	Peer Review	Sample Size	# of Effect Sizes	USA	Avg. Family Size	Avg. Age Dif.	Avg. Age of Child	White (%)	Same Sex Sib Pairs (%)	Male (%)	Younger Than Target Sib (%)	Used Beta	Variation or Perception	Concurrent or Recollected	Domains of PDT	Child or Parent Report	PDT from Mom, Dad, or Both
1	1	Y	172	8	N	2.43	2.2	6.3	92.00	46.51	49.71	50	N	V	C	P, N	C, P	M, D
2	2	N	167-169	4	N	2.1	2.9	11.09	90.11	0.00	45.67	60	N	P	C	P, C	C	M, D
3	2	N	194	4	N	2.58	3.12	11.16	90.11	100.00	49.09	55.1	N	P	C	P, C	C	M, D
4	3	N	190-191	4	N	2.67	3.13	11.28	90.11	0.00	50.67	52.02	N	P	C	P, C	C	M, D
5	3	N	143-144	4	N	2.78	3.17	11.13	90.11	100.00	52.35	52.35	N	P	C	P, C	C	M, D
6	4	N	407-408	2	N	2.71	3.17	12.49	90.11	0.00	47.63	57.35	N	P	C	P	C	M, D
7	4	N	347-353	2	N	2.81	3.07	12.45	90.11	100.00	47.3	53.54	N	P	C	P	C	M, D
8	5	N	305-306	2	N	2.6	2.7	5.19	90.11	0.00	48.32	49.22	N	P	C	P, C	P	M
9	5	N	293-294	2	N	2.59	2.71	5.28	90.11	100.00	50.65	53.25	N	P	C	P, C	P	M
10	6	N	108	2	N	2.61	2.75	5.3	90.11	0.00	52.78	50	N	P	C	P, C	P	D
11	6	N	115	2	N	2.64	3.81	5.4	90.11	100.00	45.22	50.43	N	P	C	P, C	P	D
12	7	N	68-132	12	Y	2.53	2.97	13.76	83.30	0.00	50	50	N	V	C	P, N, C	C	M, D
13	7	N	52-158	12	Y	2.62	2.61	13.44	88.00	100.00	45.57	50	N	V	C	P, N, C	C	M, D
14	8	N	200-205	2	Y	2.6	3.16	16.25	70.70	0.00	48.29	46.34	N	P	C	O	C	M, D
15	8	N	169-175	2	Y	2.65	3.08	16.32	64.00	100.00	48	48	N	P	C	O	C	M, D
16	8	N	195-206	2	Y	2.66	3.18	17.21	70.90	0.00	48.54	46.12	N	P	C	O	C	M, D
17	8	N	169-176	2	Y	2.74	3.03	17.35	64.80	100.00	48.3	48.86	N	P	C	O	C	M, D
18	8	N	185-203	2	Y	2.6	3.13	18.3	71.90	0.00	48.28	45.81	N	P	C	O	C	M, D
19	8	N	152-173	2	Y	2.62	2.98	18.42	64.70	100.00	49.71	48.55	N	P	C	O	C	M, D
21	10	N	67-234	20	Y	3.71	4.33	58.56	88.50	0.00	29.9	47.9	N	P	C, R	O, N	C	M, D

22	10	N	62-257	20	Y	3.77	4.71	59.32	86.00	100.00	33.9	47.9	N	P	C, R	O, N	C	M, D
23	11	N	128-140	16	Y	3.14	4	25.45	72.90	0.00	36.4	50.7	N	P	C	O, P, N, R	C	M, D
24	11	N	137-153	16	Y	3.15	3.79	25.31	70.10	100.00	44.2	49.4	N	P	C	O, P, N, R	C	M, D
25	12	N	781-855	20	Y	3.17	4	25.52	74.70	0.00	47.82	46.78	N	P	C	O, P, N, R	C	M, D
26	12	N	808-895	20	Y	3.21	4.11	25.36	74.60	100.00	52.07	44.8	N	P	C	O, P, N, R	C	M, D
27	13	Y	157	2	Y	3.34		23.42	66.00	33.00	53	0	N	V	C	P, R	P	B
28	14	N	194-196	18	Y	2.56	2.53	13.67	100.00	0.00	50	50	N	P, V	C	P, N, R	C	M, D, B
29	14	N	198	18	Y	2.6	2.41	13.77	100.00	100.00	54.55	50	N	P, V	C	P, N, R	C	M, D, B
30	14	N	192	18	Y	2.57	2.54	14.67	100.00	0.00	50	50	N	P, V	C	O, P, N, R	C	M, D, B
31	14	N	194-196	18	Y	2.6	2.41	14.79	100.00	100.00	54.08	50	N	P, V	C	O, P, N, R	C	M, D, B
32	14	N	190	18	Y	2.57	2.54	15.67	100.00	0.00	50	50	N	P, V	C	O, P, N, R	C	M, D, B
33	14	N	192-194	18	Y	2.61	2.41	15.8	100.00	100.00	54.08	50	N	P, V	C	O, P, N, R	C	M, D, B
34	15	N	194-206	14	Y	2.86	3.62	12.22	0.00	0.00	48.48	50	N	P, V	C	P, N, R	C	M, D, B
35	15	N	188-198	14	Y	2.74	3.8	12.24	0.00	100.00	50	50	N	P, V	C	P, N, R	C	M, D, B
36	15	N	240-278	6	Y	2.76	3.77	13.02	0.00	0.00	50	50	N	V	C	P, N	C	M, D
37	15	N	94-112	6	Y	2.74	3.59	13.96	0.00	100.00	50	50	N	V	C	P, N	C	M, D
38	15	N	170-196	4	Y	2.6	3.82	14.49	0.00	0.00	48.91	50	N	V	C	P	C	M, D
39	15	N	154-184	4	Y	2.62	3.68	14.37	0.00	100.00	51.05	50	N	V	C	P	C	M, D
40	16	N	190-210	18	Y	2.49	2.68	9.52	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
41	16	N	192-196	18	Y	2.59	2.54	9.61	100.00	100.00	48.98	50	N	P, V	C	P, N, R, C	C, P	M, D, B
42	16	N	205-206	24	Y	2.51	2.67	10.48	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
43	16	N	195-196	24	Y	2.58	2.54	10.57	100.00	100.00	48.98	50	N	P, V	C	P, N, R, C	C, P	M, D, B
44	16	N	204-206	24	Y	2.52	2.67	11.48	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
45	16	N	195-196	24	Y	2.6	2.54	11.57	100.00	100.00	48.98	50	N	P, V	C	P, N, R, C	C, P	M, D, B

46	16	N	172-176	8	Y	2.52	2.61	13.03	100.00	0.00	50	50	N	V	C	P, N	C, P	M, D
47	16	N	156-160	8	Y	2.6	2.53	13.09	100.00	100.00	48.98	50	N	V	C	P, N	C, P	M, D
48	16	N	160-166	8	Y	2.53	2.66	13.98	100.00	0.00	50	50	N	V	C	P, N	C, P	M, D
49	16	N	150-156	8	Y	2.61	2.55	14.09	100.00	100.00	48.98	50	N	V	C	P, N	C, P	M, D
50	16	N	186-196	22	Y	2.52	2.66	15.13	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
51	16	N	174-190	22	Y	2.57	2.51	15.22	100.00	100.00	48.96	50	N	P, V	C	P, N, R, C	C, P	M, D, B
52	16	N	182-192	18	Y	2.56	2.65	15.99	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
53	16	N	172-186	18	Y	2.56	2.5	16.12	100.00	100.00	48.94	50	N	P, V	C	P, N, R, C	C, P	M, D, B
54	16	N	104-196	36	Y	2.48	2.67	17.04	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
55	16	N	166-182	18	Y	2.34	2.53	17.11	100.00	100.00	47.83	50	N	P, V	C	P, N, R, C	C	M, D, B
56	16	N	102-115	18	Y	2.34	2.49	17.66	100.00	100.00	42.37	50	N	P, V	C	P, N, R, C	C, P	M, D, B
57	16	N	136-142	8	Y	1.97	2.5	25.12	100.00	100.00	43.06	50	N	P, V	C	P, R	C	M, D
58	16	N	134-140	8	Y	2	2.65	24.85	100.00	0.00	50.99	50	N	P, V	C	P, R	C	M, D
59	17	Y	117	4	N	2.6		5.27			61.54	47.1	Y	V	C	R, C	P	M, D
60	18	Y	246	4	Y	3.39	2.96		0.00	54.47	49.59	50	Y	V	C	P, N	C	M, D
61	19	Y	678-707	4	Y		1.54	12.8	94.00	100.00	51		N	P	C	O	C	M, D
62	20	Y	243	4	Y	3.72	2.93	14.24	0.00		50	50	N	P	C	P, N	P	M, D
64	22	Y	21	1	Y	2.1	2.67	4.67	100.00	53.33	56.67	50	Y	P	C	O	P	B
65	23	N	279-316	6	Y	2.78	2.59	15.92	74.20	0.00	50	50	N	P, V	C	O, P, N	C	M, D
66	23	N	300-334	6	Y	2.82	2.7	15.77	67.30	100.00	42.86	50	N	P, V	C	O, P, N	C	M, D

Note. Y = yes, N = no; Domains of PDT: P = positivity, N = negativity, R = resource, C = control, O = overall; Variation or Perception: P = perception, V = variation; Concurrent or Recollected: C = concurrent, R = recollected; Child or Parent Report: C = child, P = parent; PDT from Mom, Dad, or Both: M = mom, D = dad, B = mother and father; PDT = parental differential treatment.

Supplemental Table 3

Source and Effect-Size Characteristics for Models Examining Correlations Between Sex Composition and PDT

Sample/Source Characteristics														Effect-size Characteristics				
Source ID	Sample ID	Peer Review	Sample Size	# of Effect Sizes	USA	Avg. Family Size	Avg. Age Dif.	Avg. Age of Child	White (%)	Same Sex Sib Pairs (%)	Male (%)	Younger Than Target Sib (%)	Used Beta	Variation or Perception	Concurrent or Recollected	Domains of PDT	Child or Parent Report	PDT from Mom, Dad, or Both
1	1	Y	172	8	N	2.43	2.2	6.3	92.00	46.51	49.71	50	N	V	C	P, N	C, P	M, D
20	9	Y	3762	1	N	2.2		7.46					N	P	C	P	P	
27	13	Y	157	2	Y	3.34		23.42	66.00	33.00	53		N	V	C	P, R	P	B
59	17	Y	117	4	N	2.6		5.27			61.54	47.1	Y	V	C	R, C	P	M, D
60	18	Y	246	4	Y	3.39	2.96		0.00	54.47	49.59	50	Y	V	C	P, N	C	M, D

Note. Y = yes, N = no; Domains of PDT: P = positivity, N = negativity, R = resource, C = control, O = overall; Variation or Perception: P = perception, V = variation; Concurrent or Recollected: C = concurrent, R = recollected; Child or Parent Report: C = child, P = parent; PDT from Mom, Dad, or Both: M = mom, D = dad, B = mother and father; PDT = parental differential treatment.

Supplemental Table 4

Source and Effect-Size Characteristics for Models Examining Correlations Between Family Size and PDT

Sample/Source Characteristics													Effect-size Characteristics				
Source ID	Sample ID	Peer Review	Sample Size	# of Effect Sizes	USA	Avg. Family Size	Avg. Age Dif.	Avg. Age of Child	White (%)	Same Sex Sib Pairs (%)	Male (%)	Younger Than Target Sib	Used Beta	Variation or Perception	Concurrent or Recollected	Domains of PDT	Child or Parent Report	PDT from Mom, Dad, or Both
2	2	N	189-193	4	N	2.1	2.9	11.09	90.11	0.00	45.67	60	N	P	C	P, C	C	M, D
3	2	N	218	4	N	2.58	3.12	11.16	90.11	100.00	49.09	55.1	N	P	C	P, C	C	M, D
4	3	N	190-192	4	N	2.67	3.13	11.28	90.11	0.00	50.67	52.02	N	P	C	P, C	C	M, D
5	3	N	146-148	4	N	2.78	3.17	11.13	90.11	100.00	52.35	52.35	N	P	C	P, C	C	M, D
6	4	N	426-427	2	N	2.71	3.17	12.49	90.11	0.00	47.63	57.35	N	P	C	P	C	M, D
7	4	N	353-361	2	N	2.81	3.07	12.45	90.11	100.00	47.3	53.54	N	P	C	P	C	M, D
8	5	N	326-327	2	N	2.6	2.7	5.19	90.11	0.00	48.32	49.22	N	P	C	P, C	P	M
9	5	N	307-308	2	N	2.59	2.71	5.28	90.11	100.00	50.65	53.25	N	P	C	P, C	P	M
10	6	N	108	2	N	2.61	2.75	5.3	90.11	0.00	52.78	50	N	P	C	P, C	P	D
11	6	N	115	2	N	2.64	3.81	5.4	90.11	100.00	45.22	50.43	N	P	C	P, C	P	D
12	7	N	62-68	12	Y	2.53	2.97	13.76	83.30	0.00	50	50	N	V	C	P, N, C	C	M, D
13	7	N	48-58	12	Y	2.62	2.61	13.44	88.00	100.00	45.57	50	N	V	C	P, N, C	C	M, D
14	8	N	196-201	2	Y	2.6	3.16	16.25	70.70	0.00	48.29	46.34	N	P	C	O	C	M, D
15	8	N	164-170	2	Y	2.65	3.08	16.32	64.00	100.00	48	48	N	P	C	O	C	M, D
16	8	N	191-202	2	Y	2.66	3.18	17.21	70.90	0.00	48.54	46.12	N	P	C	O	C	M, D
17	8	N	163-170	2	Y	2.74	3.03	17.35	64.80	100.00	48.3	48.86	N	P	C	O	C	M, D
18	8	N	178-196	2	Y	2.6	3.13	18.3	71.90	0.00	48.28	45.81	N	P	C	O	C	M, D

19	8	N	142-162	2	Y	2.62	2.98	18.42	64.70	100.00	49.71	48.55	N	P	C	O	C	M, D
20	9	Y	3762	1	N	2.2		7.46					N	P	C	N	P	
21	10	N	67-234	20	Y	3.71	4.33	58.56	88.50	0.00	29.9	47.9	N	P	C, R	O, N	C	M, D
22	10	N	62-257	20	Y	3.77	4.71	59.32	86.00	100.00	33.9	47.9	N	P	C, R	O, N	C	M, D
23	11	N	128-140	16	Y	3.14	4	25.45	72.90	0.00	36.4	50.7	N	P	C	O, P, N, R	C	M, D
24	11	N	138-154	16	Y	3.15	3.79	25.31	70.10	100.00	44.2	49.4	N	P	C	O, P, N, R	C	M, D
25	12	N	781-855	20	Y	3.17	4	25.52	74.70	0.00	47.82	46.78	N	P	C	O, P, N, R	C	M, D
26	12	N	808-895	20	Y	3.21	4.11	25.36	74.60	100.00	52.07	44.8	N	P	C	O, P, N, R	C	M, D
27	13	Y	157	2	Y	3.34		23.42	66.00	33.00	53		N	V	C	P, R	P	B
28	14	N	194-196	18	Y	2.56	2.53	13.67	100.00	0.00	50	50	N	P, V	C	P, N, R	C	M, D, B
29	14	N	198	18	Y	2.6	2.41	13.77	100.00	100.00	54.55	50	N	P, V	C	P, N, R	C	M, D, B
30	14	N	192	18	Y	2.57	2.54	14.67	100.00	0.00	50	50	N	P, V	C	O, P, N, R	C	M, D, B
31	14	N	194-196	18	Y	2.6	2.41	14.79	100.00	100.00	54.08	50	N	P, V	C	O, P, N, R	C	M, D, B
32	14	N	190	18	Y	2.57	2.54	15.67	100.00	0.00	50	50	N	P, V	C	O, P, N, R	C	M, D, B
33	14	N	194-196	18	Y	2.61	2.41	15.8	100.00	100.00	54.08	50	N	P, V	C	O, P, N, R	C	M, D, B
34	15	N	194-206	14	Y	2.86	3.62	12.22	0.00	0.00	48.48	50	N	P, V	C	P, N, R	C	M, D, B
35	15	N	188-198	14	Y	2.74	3.8	12.24	0.00	100.00	50	50	N	P, V	C	P, N, R	C	M, D, B
36	15	N	240-278	6	Y	2.76	3.77	13.02	0.00	0.00	50	50	N	V	C	P, N	C	M, D
37	15	N	94-112	6	Y	2.74	3.59	13.96	0.00	100.00	50	50	N	V	C	P, N	C	M, D
38	15	N	170-196	4	Y	2.6	3.82	14.49	0.00	0.00	48.91	50	N	V	C	P	C	M, D
39	15	N	154-184	4	Y	2.62	3.68	14.37	0.00	100.00	51.05	50	N	V	C	P	C	M, D
40	16	N	190-210	18	Y	2.49	2.68	9.52	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
41	16	N	192-196	18	Y	2.59	2.54	9.61	100.00	100.00	48.98	50	N	P, V	C	P, N, R, C	C, P	M, D, B

42	16	N	205-206	24	Y	2.51	2.67	10.48	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
43	16	N	195-196	24	Y	2.58	2.54	10.57	100.00	100.00	48.98	50	N	P, V	C	P, N, R, C	C, P	M, D, B
44	16	N	204-206	24	Y	2.52	2.67	11.48	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
45	16	N	195-196	24	Y	2.6	2.54	11.57	100.00	100.00	48.98	50	N	P, V	C	P, N, R, C	C, P	M, D, B
46	16	N	172-176	8	Y	2.52	2.61	13.03	100.00	0.00	50	50	N	V	C	P, N	C, P	M, D
47	16	N	156-160	8	Y	2.6	2.53	13.09	100.00	100.00	48.98	50	N	V	C	N	C	M, D
48	16	N	160-166	8	Y	2.53	2.66	13.98	100.00	0.00	50	50	N	V	C	P, N	C, P	M, D
49	16	N	150-156	8	Y	2.61	2.55	14.09	100.00	100.00	48.98	50	N	V	C	P, N	C, P	M, D
50	16	N	186-196	22	Y	2.52	2.66	15.13	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
51	16	N	174-190	22	Y	2.57	2.51	15.22	100.00	100.00	48.96	50	N	P, V	C	P, N, R, C	C, P	M, D, B
52	16	N	182-192	18	Y	2.56	2.65	15.99	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
53	16	N	172-186	18	Y	2.56	2.5	16.12	100.00	100.00	48.94	50	N	P, V	C	P, N, R, C	C, P	M, D, B
54	16	N	104-196	36	Y	2.48	2.67	17.04	100.00	0.00	50	50	N	P, V	C	P, N, R, C	C, P	M, D, B
55	16	N	166-182	18	Y	2.34	2.53	17.11	100.00	100.00	47.83	50	N	P, V	C	P, N, R, C	C	M, D, B
56	16	N	102-115	18	Y	2.34	2.49	17.66	100.00	100.00	42.37	50	N	P, V	C	P, N, R, C	C, P	M, D, B
63	21	Y	542	3	Y	4.4		42.8	69.80		50.7		Y	P	C	P, R	P	M
65	23	N	279-316	6	Y	2.78	2.59	15.92	74.20	0.00	50	50	N	P, V	C	O, P, N	C	M, D
66	23	N	300-334	6	Y	2.82	2.7	15.77	67.30	100.00	42.86	50	N	P, V	C	O, P, N	C	M, D

Note. Y = yes, N = no; Domains of PDT: P = positivity, N = negativity, R = resource, C = control, O = overall; Variation or Perception: P = perception, V = variation; Concurrent or Recollected: C = concurrent, R = recollected; Child or Parent Report: C = child, P = parent; PDT from Mom, Dad, or Both: M = mom, D = dad, B = mother and father; PDT = parental differential treatment.

Analysis Code

NOTE: DataID is the variable for each unique sample ID. ArticleID is the variable for each unique source ID. ESNum is the variable for each unique effect size ID.

Functions used in the analysis

```
withold <- function (data, expr) {
  call <- match.call()
  analyses <- as.list(seq_len(data$m))
  for (i in seq_along(analyses)) {
    data.i <- complete(data, i)
    analyses[[i]] <- eval(expr = substitute(expr), envir = data.i, enclos = parent.frame())
    if (is.expression(analyses[[i]]))
      analyses[[i]] <- eval(expr = analyses[[i]], envir = data.i, enclos = parent.frame())
  }
  object <- list(call = call, call1 = data$call, nmis = data$nmis, analyses = analyses)
  oldClass(object) <- c("mira", "matrix")
  object
}

MetaModelsDescriptiveData <- function(x) {
  xData<-x %>% distinct(DataID, .keep_all = T) #keeping only one line per data set ID
  myvarsData <- c("ArticleID", "DataID", "AgeDif", "USA", "SampleSize") #creating vector
with needed variables
  xData <- xData[myvarsData] #keeping only needed variables
  xData$USA[xData$USA<0] <- NA
  xData$SampleSize[xData$DataID == 82] <- 761
  xData$SampleSize[xData$DataID == 83] <- 363
  xData$SampleSize[xData$DataID == 84] <- 335
  xData$SampleSize[xData$DataID == 85] <- 600
  xData$SampleSize[xData$DataID == 86] <- 223
  xData$SampleSize[xData$DataID == 87] <- 650
  xData$SampleSize[xData$DataID == 88] <- 491
  xData$SampleSize[xData$DataID == 89] <- 293
  xData$SampleSize[xData$DataID == 90] <- 1750
  xData$SampleSize[xData$DataID == 49] <- 392
  xData$SampleSize[xData$DataID == 65] <- 402
  xData$SampleSize[xData$DataID == 33] <- 380
  xData$SampleSize[xData$DataID == 48] <- 406
  xData$SampleSize[xData$DataID == 93] <- 274
  #retrieve stats for data set level variables
  TotalParticipants<-sum(xData$SampleSize)
  USAPec <- round(sum(xData$USA)/length(xData$USA),2)
  USATotal<- length(xData$USA[xData$USA==1])
```

```
  EuropeTotal<- length(xData$USA[xData$USA==0])
  #removing duplicates for Article level variables
  xArticle<-x %>% distinct(ArticleID, .keep_all = T) #keeping only one line per data set ID
  myvarsArticle <- c("ArticleID", "DataID", "SibAge",  "AgeDif", "FamSize", "WhitePerc",
"GCSamePerc", "MalePerc", "RelBirthYoungPerc", "SampleSize") #creating vector with needed
variables
  xArticle <- xArticle[myvarsArticle] #keeping only needed variables
  xArticle$SibAge[xArticle$SibAge<0] <- NA
  xArticle$AgeDif[xArticle$AgeDif<0] <- NA
  xArticle$FamSize[xArticle$FamSize<0] <- NA
  xArticle$WhitePerc[xArticle$WhitePerc<0] <- NA
  xArticle$GCSamePerc[xArticle$GCSamePerc<0] <- NA
  xArticle$MalePerc[xArticle$MalePerc<0] <- NA
  xArticle$RelBirthYoungPerc[xArticle$RelBirthYoungPerc<0] <- NA
  xArticle$WhitePerc[xArticle$WhitePerc>10] <- NA
  xArticle$GCSamePerc[xArticle$GCSamePerc>10] <- NA
  xArticle$MalePerc[xArticle$MalePerc>10] <- NA
  xArticle$RelBirthYoungPerc[xArticle$RelBirthYoungPerc>10] <- NA
  xArticle$WhitePerc <- xArticle$WhitePerc*10
  xArticle$GCSamePerc <- xArticle$GCSamePerc*10
  xArticle$MalePerc <- xArticle$MalePerc*10
  xArticle$RelBirthYoungPerc <- xArticle$RelBirthYoungPerc*10
  #retrieve stats for article level variables
  ChildAgeMean<-round(mean(xArticle$SibAge, na.rm = T),2)
  ChildAgeSD<-round(sd(xArticle$SibAge, na.rm = T),2)
  AgeDifMean<-round(mean(xArticle$AgeDif, na.rm = T),2)
  AgeDifSD<-round(sd(xArticle$AgeDif, na.rm = T),2)
  FamSizeMean<-round(mean(xArticle$FamSize, na.rm = T),2)
  FamSizeSD<-round(sd(xArticle$FamSize, na.rm = T),2)
  WhitePercMean<-round(mean(xArticle$WhitePerc, na.rm = T),2)
  WhitePercSD<-round(sd(xArticle$WhitePerc, na.rm = T),2)
  GCSamePercMean<-round(mean(xArticle$GCSamePerc, na.rm = T),2)
  GCSamePercSD<-round(sd(xArticle$GCSamePerc, na.rm = T),2)
  MalePercMean<-round(mean(xArticle$MalePerc, na.rm = T),2)
  MalePercSD<-round(sd(xArticle$MalePerc, na.rm = T),2)
  RelBirthYoungPercMean<-round(mean(xArticle$RelBirthYoungPerc, na.rm = T),2)
  RelBirthYoungPercSD<-round(sd(xArticle$RelBirthYoungPerc, na.rm = T),2)
  SampleSizeMean<-round(mean(xArticle$SampleSize, na.rm = T),2)
  SampleSizeSD<-round(sd(xArticle$SampleSize, na.rm = T),2)
  #tidy up data for effect size level variables
  myvarsALL <- c("ESNum", "PeerRev", "UsedBeta", "OverallPDT", "PosPDT", "NegPDT",
"ResourcePDT", "ControlPDT", "ChildPDTReport", "MomPDT", "DadPDT", "BothParsPDT",
"Variation", "Recollected") #creating vector with needed variables
  xALL <- x[myvarsALL] #keeping only needed variables
  xALL$PeerRev[xALL$PeerRev<0] <- NA
  xALL$UsedBeta[xALL$UsedBeta<0] <- NA
```

```
  xALL$OverallPDT[xALL$OverallPDT<0] <- NA
  xALL$PosPDT[xALL$PosPDT<0] <- NA
  xALL$NegPDT[xALL$NegPDT<0] <- NA
  xALL$ResourcePDT[xALL$ResourcePDT<0] <- NA
  xALL$ControlPDT[xALL$ControlPDT<0] <- NA
  xALL$ChildPDTReport[xALL$ChildPDTReport<0] <- NA
  xALL$MomPDT[xALL$MomPDT<0] <- NA
  xALL$DadPDT[xALL$DadPDT<0] <- NA
  xALL$BothParsPDT[xALL$BothParsPDT<0] <- NA
  xALL$Variation[xALL$Variation<0] <- NA
  xALL$Recollected[xALL$Recollected<0] <- NA
  xALL$PeerRev[xALL$PeerRev>1] <- NA
  xALL$UsedBeta[xALL$UsedBeta>1] <- NA
  xALL$OverallPDT[xALL$OverallPDT>1] <- NA
  xALL$PosPDT[xALL$PosPDT>1] <- NA
  xALL$NegPDT[xALL$NegPDT>1] <- NA
  xALL$ResourcePDT[xALL$ResourcePDT>1] <- NA
  xALL$ControlPDT[xALL$ControlPDT>1] <- NA
  xALL$ChildPDTReport[xALL$ChildPDTReport>1] <- NA
  xALL$MomPDT[xALL$MomPDT>1] <- NA
  xALL$DadPDT[xALL$DadPDT>1] <- NA
  xALL$BothParsPDT[xALL$BothParsPDT>1] <- NA
  xALL$Variation[xALL$Variation>1] <- NA
  xALL$Recollected[xALL$Recollected>1] <- NA
  #retrieve stats for effect size level variables
  PeerReviewedPec <- round(sum(xALL$PeerRev)/length(xALL$PeerRev),2)
  UsedBetaPec <- round(sum(xALL$UsedBeta)/length(xALL$UsedBeta),2)
  OverallPDTPec <- round(sum(xALL$OverallPDT)/length(xALL$OverallPDT),2)
  PosPDTPec <- round(sum(xALL$PosPDT)/length(xALL$PosPDT),2)
  NegPDTPec <- round(sum(xALL$NegPDT)/length(xALL$NegPDT),2)
  ResourcePDTPec <- round(sum(xALL$ResourcePDT)/length(xALL$ResourcePDT),2)
  ControlPDTPec <- round(sum(xALL$ControlPDT)/length(xALL$ControlPDT),2)
  ChildPDTReportPec <-
round(sum(xALL$ChildPDTReport)/length(xALL$ChildPDTReport),2)
  MomPDTPec <- round(sum(xALL$MomPDT)/length(xALL$MomPDT),2)
  DadPDTPec <- round(sum(xALL$DadPDT)/length(xALL$DadPDT),2)
  BothParsPDTPec <- round(sum(xALL$BothParsPDT)/length(xALL$BothParsPDT),2)
  VariationPec <- round(sum(xALL$Variation)/length(xALL$Variation),2)
  RecollectedPec <- round(sum(xALL$Recollected)/length(xALL$Recollected),2)
  PeerReviewedTotal<- length(xALL$PeerRev[xALL$PeerRev==1])
  NotPeerReviewedTotal<- length(xALL$PeerRev[xALL$PeerRev==0])
  VariationTotal<- length(xALL$Variation[xALL$Variation==1])
  PerceptionTotal<- length(xALL$Variation[xALL$Variation==0])
  DadPDTTotal<- length(xALL$DadPDT[xALL$DadPDT==1])
  MomPDTTotal<- length(xALL$MomPDT[xALL$MomPDT==1])
  BothParsPDTTotal<- length(xALL$BothParsPDT[xALL$BothParsPDT==1])
```

```
RecollectedTotal<- length(xALL$Recollected[xALL$Recollected==1])
ConcurrentTotal<- length(xALL$Recollected[xALL$Recollected==0])
ChildPDTReportTotal<- length(xALL$ChildPDTReport[xALL$ChildPDTReport==1])
ParentPDTReportTotal<- length(xALL$ChildPDTReport[xALL$ChildPDTReport==0])
UsedBetaTotal<- length(xALL$UsedBeta[xALL$UsedBeta==1])
UsedCorrTotal<- length(xALL$UsedBeta[xALL$UsedBeta==0])
OverallPDTTotal<- length(xALL$OverallPDT[xALL$OverallPDT==1])
PosPDTTotal<- length(xALL$PosPDT[xALL$PosPDT==1])
NegPDTTotal<- length(xALL$NegPDT[xALL$NegPDT==1])
ResourcePDTTotal<- length(xALL$ResourcePDT[xALL$ResourcePDT==1])
ControlPDTTotal<- length(xALL$ControlPDT[xALL$ControlPDT==1])
DO.NOT.INCLUDE.THE.REST.IN.THE.TABLE <- " "
LEVEL.FOUR.VARIABLES <- " "
LEVEL.THREE.VARIABLES <- " "
LEVEL.TWO.VARIABLES <- " "
#create factors for statistics and names and bind together
Numbers<- c(LEVEL.FOUR.VARIABLES,
        USAPec,
        LEVEL.THREE.VARIABLES,
        AgeDifMean, AgeDifSD,
        FamSizeMean, FamSizeSD,
        WhitePercMean, WhitePercSD,
        GCSamePercMean, GCSamePercSD,
        MalePercMean, MalePercSD,
        RelBirthYoungPercMean, RelBirthYoungPercSD,
        SampleSizeMean, SampleSizeSD,
        ChildAgeMean, ChildAgeSD,
        PeerReviewedPec,
        LEVEL.TWO.VARIABLES,
        UsedBetaPec,
        OverallPDTPec, PosPDTPec, NegPDTPec, ResourcePDTPec, ControlPDTPec,
        ChildPDTReportPec,
        MomPDTPec, DadPDTPec, BothParsPDTPec,
        VariationPec,
        RecollectedPec,
        TotalParticipants,
        DO.NOT.INCLUDE.THE.REST.IN.THE.TABLE,
        USATotal, EuropeTotal,
        PeerReviewedTotal, NotPeerReviewedTotal,
        VariationTotal, PerceptionTotal,
        DadPDTTotal, MomPDTTotal, BothParsPDTTotal,
        RecollectedTotal, ConcurrentTotal,
        ChildPDTReportTotal, ParentPDTReportTotal,
        UsedBetaTotal, UsedCorrTotal,
        OverallPDTTotal, PosPDTTotal, NegPDTTotal, ResourcePDTTotal, ControlPDTTotal)
Variables <- c("Level Four Variables",
```

```
            "Proportion USA",
            "Level Three Variables",
            "Age Spacing Mean", "Age Spacing SD",
            "Family Size Mean", "Family Size SD",
            "% White Mean", "% White SD",
            "% Same Sex Mean", "% Same Sex SD",
            "% Male Mean", "% Male SD",
            "% Younger Mean", "% Younger SD",
            "Sample Size Mean", "Sample Size SD",
            "Child Age Mean", "Child Age SD",
            "Proportion Peer Reviewed",
            "Level Two Variables",
            "Proportion Used Beta",
            "Proportion Overall PDT", "Proportion Positive PDT", "Proportion Negative PDT",
"Proportion Resource PDT", "Proportion Control PDT",
            "Proportion Child PDT Report",
            "Proportion Mom PDT", "Proportion Dad PDT", "Proportion Both Parents PDT",
            "Proportion Variation",
            "Proportion Recollected",
            "Total # of Participants",
            "DO NOT INCLUDE THE REST IN THE TABLE",
            "Number USA", "Number Europe",
            "Number Peer Reviewed", "Number Not Peer Reviewed",
            "Number Variation", "Number Perception",
            "Nuumber Dad PDT", "Number Mom PDT", "Number Both Pars PDT",
            "RecollectedTotal", "ConcurrentTotal",
            "Number Child Report", "Number Parent Report",
            "Number Used Beta", "Number Used Correlation",
            "Number Overall PDT", "Number Positive PDT", "Number Negative PDT", "Number
Resource PDT", "Number Control PDT")
  ToDisplay <- cbind (Variables, Numbers)
  #export to a csv file with the name of the original data frame in the file name
  fname <- sprintf("Findings/Descriptive Stats/%s Descriptive Stats.csv",deparse(substitute(x)))
  write.csv(ToDisplay, file = fname)
  #return the values and end the function
  return(ToDisplay)}

MetaVarCompsModel1 <- function(objectM1){

  #prepare the object for extraction
  objectM1 <- getfit(objectM1)

  #Retrieve the Level2 Sigma2
  M1L2Sigma2 <- objectM1[["sigma2"]][1]

  #Retrieve the Level3 Sigma2
```

```
  M1L3Sigma2 <- objectM1[["sigma2"]][2]

  #Retrieve the Level3 Sigma2
  M1L4Sigma2 <- objectM1[["sigma2"]][3]

  #calculate total variance
  M1TotsVar <- sum(M1L2Sigma2,M1L3Sigma2,M1L4Sigma2)

  #ICC for level 2
  M1L2ICC <- M1L2Sigma2/M1TotsVar

  #ICC for level 3
  M1L3ICC <- M1L3Sigma2/M1TotsVar

  #ICC for level 3
  M1L4ICC <- M1L4Sigma2/M1TotsVar

  #setting which values to report
  ValuesM1 <- c(M1L2ICC,
                M1L3ICC,
                M1L4ICC
  )
  NamesM1 <- c("Level 2 ICC - Model 1",
               "Level 3 ICC - Model 1",
               "Level 4 ICC - Model 1"
  )
  ToDisplayM1 <- rbind(NamesM1, ValuesM1)

  #return the values and end the function
  return(ToDisplayM1)
}

MetaVarCompsModel2 <- function(object){

  #prepare the object for extraction
  object <- getfit(object)

  #Retrieve the Q
  QList <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){QList[i] <-object[[i]]["QE"]} # a loop to retrieve the info from each model
  QListCreated <- print(unlist(QList)) #prints the info for each model as a vector instead of a list
  Q <- round(mean(QListCreated),5) #creates the mean for the info

  #Retrieve the Q's p-value
  QPList <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){QPList[i] <-object[[i]]["QEp"]} # a loop to retrieve the info from each model
```

```
  QpListCreated <- print(unlist(QPList)) #prints the info for each model as a vector instead of a
list
  Qp <- round(mean(QpListCreated),4) #creates the mean for the info

  #Retrieve the F for omnibus test of moderators
  FList <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){FList[i] <-object[[i]]["QM"]} # a loop to retrieve the info from each model
  FListCreated <- print(unlist(FList)) #prints the info for each model as a vector instead of a list
  FforMods <- round(mean(FListCreated),5) #creates the mean for the info

  #Retrieve the F's p-value
  FPList <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){FPList[i] <-object[[i]]["QMp"]} # a loop to retrieve the info from each model
  FpListCreated <- print(unlist(FPList)) #prints the info for each model as a vector instead of a
list
  FforModsp <- round(mean(FpListCreated),4) #creates the mean for the info

  #Retrieve the Level2 Sigma2
  L2Sig2List <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){L2Sig2List[i] <-object[[i]][["sigma2"]][1]} # a loop to retrieve the info from
each model
  L2Sig2ListCreated <- print(unlist(L2Sig2List)) #prints the info for each model as a vector
instead of a list
  L2Sigma2 <- mean(L2Sig2ListCreated) #creates the mean for the info

  #Retrieve the Level3 Sigma2
  L3Sig2List <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){L3Sig2List[i] <-object[[i]][["sigma2"]][2]} # a loop to retrieve the info from
each model
  L3Sig2ListCreated <- print(unlist(L3Sig2List)) #prints the info for each model as a vector
instead of a list
  L3Sigma2 <- mean(L3Sig2ListCreated) #creates the mean for the info

  #Retrieve the Level3 Sigma2
  L4Sig2List <- rep(NA,50) #an empty list to put the info in
  for (i in 1:50){L4Sig2List[i] <-object[[i]][["sigma2"]][3]} # a loop to retrieve the info from
each model
  L4Sig2ListCreated <- print(unlist(L4Sig2List)) #prints the info for each model as a vector
instead of a list
  L4Sigma2 <- mean(L4Sig2ListCreated) #creates the mean for the info

  #calculate total variance
  TotalVar <- sum(L2Sigma2, L3Sigma2, L4Sigma2)

  #ICC for level 2
  L2ICC <- L2Sigma2/TotalVar
```

```
  #ICC for level 3
  L3ICC <- L3Sigma2/TotalVar

  #ICC for level 3
  L4ICC <- L4Sigma2/TotalVar

  #setting which values to report
  ValuesM2 <- c(Q,
                Qp,
                FforMods,
                FforModsp,
                L2Sigma2,
                L2ICC,
                L3Sigma2,
                L3ICC,
                L4Sigma2,
                L4ICC
  )
  NamesM2 <- c("Q - Model 2",
               "p-value for Q - Model 2",
               "F value for omnibus test of moderators - Model 2",
               "p-value for F value - Model 2",
               "Level 2 Sigma2 - Model 2",
               "Level 2 ICC - Model 2",
               "Level 3 Sigma2 - Model 2",
               "Level 3 ICC - Model 2",
               "Level 4 Sigma2 - Model 2",
               "Level 4 ICC - Model 2"
  )
  ToDisplayM2 <- rbind(NamesM2, ValuesM2)

  #return the values and end the function
  return(ToDisplayM2)
}

MetaVarCompsModel2LevelVarianceTest <- function(FullModel, RestrictedModelLevel2,
RestrictedModelLevel3, RestrictedModelLevel4){

  #create an empty list for each level
  LRTListL2 <- as.list(rep(NA,50)) #an empty list to put the info in
  LRTPValueListL2 <- as.list(rep(NA,50)) #an empty list to put the info in

  LRTListL3 <- as.list(rep(NA,50)) #an empty list to put the info in
  LRTPValueListL3 <- as.list(rep(NA,50)) #an empty list to put the info in
```

```
  LRTListL4 <- as.list(rep(NA,50)) #an empty list to put the info in
  LRTPValueListL4 <- as.list(rep(NA,50)) #an empty list to put the info in

  #run the loop for each level
  for (i in 1:50){
    LRTListL2[[i]] <- anova(FullModel[["analyses"]][[i]],
RestrictedModelLevel2[["analyses"]][[i]])[["LRT"]] #this runs each test and saves the LRT value
in a list
    LRTPValueListL2[[i]] <- anova(FullModel[["analyses"]][[i]],
RestrictedModelLevel2[["analyses"]][[i]])[["pval"]] #this runs each test and saves the pvalue in a
list

    LRTListL3[[i]] <- anova(FullModel[["analyses"]][[i]],
RestrictedModelLevel3[["analyses"]][[i]])[["LRT"]] #this runs each test and saves the LRT value
in a list
    LRTPValueListL3[[i]] <- anova(FullModel[["analyses"]][[i]],
RestrictedModelLevel3[["analyses"]][[i]])[["pval"]] #this runs each test and saves the pvalue in a
list

    LRTListL4[[i]] <- anova(FullModel[["analyses"]][[i]],
RestrictedModelLevel4[["analyses"]][[i]])[["LRT"]] #this runs each test and saves the LRT value
in a list
    LRTPValueListL4[[i]] <- anova(FullModel[["analyses"]][[i]],
RestrictedModelLevel4[["analyses"]][[i]])[["pval"]] #this runs each test and saves the pvalue in a
list
  }

  #take each list and turn it into a vector so we can pool the LRT and p-value info
  LRTListCreatedL2 <- print(unlist(LRTListL2)) #prints the info for each model as a vector
instead of a list
  LRTModel2L2 <- round(mean(LRTListCreatedL2),5) #creates the mean for the info
  LRTPValueListCreatedL2 <- print(unlist(LRTPValueListL2)) #prints the info for each model
as a vector instead of a list
  LRTPValueModel2L2 <- round(mean(LRTPValueListCreatedL2),5) #creates the mean for the
info

  LRTListCreatedL3 <- print(unlist(LRTListL3)) #prints the info for each model as a vector
instead of a list
  LRTModel2L3 <- round(mean(LRTListCreatedL3),5) #creates the mean for the info
  LRTPValueListCreatedL3 <- print(unlist(LRTPValueListL3)) #prints the info for each model
as a vector instead of a list
  LRTPValueModel2L3 <- round(mean(LRTPValueListCreatedL3),5) #creates the mean for the
info

  LRTListCreatedL4 <- print(unlist(LRTListL4)) #prints the info for each model as a vector
instead of a list
```

```
  LRTModel2L4 <- round(mean(LRTListCreatedL4),5) #creates the mean for the info
  LRTPValueListCreatedL4 <- print(unlist(LRTPValueListL4)) #prints the info for each model
as a vector instead of a list
  LRTPValueModel2L4 <- round(mean(LRTPValueListCreatedL4),5) #creates the mean for the
info

  #setting which values to report
  ValuesM2VarComps <- c(
    LRTModel2L2,
    LRTPValueModel2L2,
    LRTModel2L3,
    LRTPValueModel2L3,
    LRTModel2L4,
    LRTPValueModel2L4
  )
  NamesM2VarComps <- c(
    "LRT for Model 2 - Level 2",
    "LRT P Value for Model 2 - Level 2",
    "LRT for Model 2 - Level 3",
    "LRT P Value for Model 2 - Level 3",
    "LRT for Model 2 - Level 4",
    "LRT P Value for Model 2 - Level 4"
  )
  ToDisplayM2VarComps <- rbind(NamesM2VarComps, ValuesM2VarComps)

  #return the values and end the function
  return(ToDisplayM2VarComps)
}
```

() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () ()
() () () () () ()
() () () () () ()
() () () Four-level multilevel meta models () () ()
() () () testing links between Age Spacing () () ()
() () () and absolute PDT () () ()
() () () () () ()
() () () () () ()
() () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()

#SET THE WORKING DIRCTORY
#SET THE WORKING DIRCTORY
#SET THE WORKING DIRCTORY

#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND THE MetaModelsDescriptiveData() FUNCTION
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND THE MetaModelsDescriptiveData() FUNCTION
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND THE MetaModelsDescriptiveData() FUNCTION
#for info on withold(), see https://www.metafor-project.org/doku.php/tips:multiple_imputation_with_mice_and_metafor for more details
#I created the MetaModelsDescriptiveData(), MetaVarCompsModel1(), MetaVarCompsModel2(), and MetaVarCompsModel2LevelVarianceTest()
#functions - more notes are available in the original file "Meta Analysis Functions.R"
library(metafor)
library(PcAux)
library(mice)
library(psych)
library(tidyverse)
library(car)

#IMPORT THE ENTIRE META DATA FRAME
#IMPORT THE ENTIRE META DATA FRAME
#IMPORT THE ENTIRE META DATA FRAME
WholeMeta <- read.csv(file = "Programs and Data/FinalCleanPDTMeta.csv", header = T)

```
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Age Spacing
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Age Spacing
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Age Spacing
Meta1 <- subset(x = WholeMeta, subset = MagnitudeDirectional == 0 & Outcome == 20)



#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
Meta2 <- Meta1[,

c("ESNum","SampleSize","Correlation","ArticleID","PeerRev","USA","DataID","FamSize","A
geDif",

"UsedBeta","OverallPDT","PosPDT","NegPDT","EvalPDT","ResourcePDT","FairPDT","Contr
olPDT",

"ChildPDTReport","ParentPDTReport","MomPDT","DadPDT","BothParsPDT","Variation","Pe
rception",
              "Recollected","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")]

MetaModelsDescriptiveData(Meta2) # this saves the descriptive stats for the non-imputed model
#we take descriptive stats from this data frame so that it includes missing data and the non-
centered versions of continous variables



#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#we will drop the non-centered versions of continuous variables, because we want the imputation
to be based on the centered versions
```

```
Meta3 <- Meta2
attach(Meta3)
Meta3$FamSizeC <- FamSize-mean(FamSize, na.rm = T)
Meta3$AgeDifC <- AgeDif-mean(AgeDif, na.rm = T)
Meta3$AgeC <- Age-mean(Age, na.rm = T)
Meta3$WhitePercC <- WhitePerc-mean(WhitePerc, na.rm = T)
Meta3$GCSamePercC <- GCSamePerc-mean(GCSamePerc, na.rm = T)
Meta3$MalePercC <- MalePerc-mean(MalePerc, na.rm = T)
Meta3$RelBirthYoungPercC <- RelBirthYoungPerc-mean(RelBirthYoungPerc, na.rm = T)
detach(Meta3)
DropVars <- names(Meta3) %in%
c("AgeDif","FamSize","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")
Meta3 <- Meta3[!DropVars]
DropVars <- Meta3 %in%
c("AgeDif","FamSize","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")



#SET THE EFFECT SIZES
#SET THE EFFECT SIZES
#SET THE EFFECT SIZES
#Do this before the PCAs, it doesn't like doing it on an imputed data frame, so do it first
Meta4 <- escalc(measure = "ZCOR", ri =Correlation,
                ni = SampleSize, data = Meta3, vtype = "LS", append = TRUE)




#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#re-do this step every time

MetaNOMISS <- Meta4[, which(colMeans(is.na(Meta4)) ==.00)]#creating data frame with only
variables with no missing values

names(MetaNOMISS)#I run this to get the names and then sort those into the type of variable
below

## Prepare the data by first sorting out what types of variables I have - continuous variables are
left off the code but listed below
```

```
myNoms   <- c("PeerRev",        "USA",        "UsedBeta",        "OverallPDT",      "PosPDT",
"NegPDT",         "EvalPDT",   "ResourcePDT",     "FairPDT",         "ControlPDT",
"ChildPDTReport",  "ParentPDTReport", "MomPDT",    "DadPDT",          "BothParsPDT",
"Variation",       "Perception",      "Recollected")
myIds    <- c("ESNum", "ArticleID", "DataID")
#myOrds   <- c() #none this time
#Continuous - "SampleSize",      "Correlation",    "MalePercC",     "yi",             "vi"



#Create data frame for creating PCAs - prep data step
ReadyForPCA <- prepData(rawData = MetaNOMISS,
                        nomVars = myNoms,
                        ordVars = NULL, # make sure to update this if there are ordinal variables
                        idVars = myIds,
                        moderators = NULL,
                        verbose = 2L,
                        nProcess = 4,
                        simMode = F)
y #the prepData code required a yes or no to continue, this tells it to continue

## Create principal component auxiliary variables:
PCAOut <- createPcAux(pcAuxData = ReadyForPCA,
                      nComps = c(0.5,0.5),
                      maxPolyPow = 3L,
                      doImputation = FALSE,
                      interactType = 1)
y #the createPcAux code required a yes or no to continue, this tells it to continue


## Merge the PC auxiliaries with the original data:
Meta5 <- mergePcAux(pcAuxData = PCAOut, rawData = Meta4) #use this data for imputation


#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
Meta5$FamSizeH <- Meta5$FamSizeC - sd(Meta5$FamSizeC, na.rm = T)
Meta5$FamSizeL <- Meta5$FamSizeC + sd(Meta5$FamSizeC, na.rm = T)
```

```
Meta5$AgeDifH <- Meta5$AgeDifC - sd(Meta5$AgeDifC, na.rm = T)
Meta5$AgeDifL <- Meta5$AgeDifC + sd(Meta5$AgeDifC, na.rm = T)
Meta5$AgeH <- Meta5$AgeC - sd(Meta5$AgeC, na.rm = T)
Meta5$AgeL <- Meta5$AgeC + sd(Meta5$AgeC, na.rm = T)
Meta5$WhitePercH <- Meta5$WhitePercC - sd(Meta5$WhitePercC, na.rm = T)
Meta5$WhitePercL <- Meta5$WhitePercC + sd(Meta5$WhitePercC, na.rm = T)
Meta5$GCSamePercH <- Meta5$GCSamePercC - sd(Meta5$GCSamePercC, na.rm = T)
Meta5$GCSamePercL <- Meta5$GCSamePercC + sd(Meta5$GCSamePercC, na.rm = T)
Meta5$MalePercH <- Meta5$MalePercC - sd(Meta5$MalePercC, na.rm = T)
Meta5$MalePercL <- Meta5$MalePercC + sd(Meta5$MalePercC, na.rm = T)
Meta5$RelBirthYoungPercH <- Meta5$RelBirthYoungPercC - sd(Meta5$RelBirthYoungPercC,
na.rm = T)
Meta5$RelBirthYoungPercL <- Meta5$RelBirthYoungPercC + sd(Meta5$RelBirthYoungPercC,
na.rm = T)
Meta5$NonPeerRev <- 1 #creating a variable where non-peer reviewed (mostly raw data) is
coded as 1, and peer reviewed as zero
Meta5$NonPeerRev[Meta5$PeerRev == 1] <- 0
Meta5$Concurrent <- 1 # creating a variable where concurrent PDT is 1, recollected is zero
Meta5$Concurrent[Meta5$Recollected == 1] <- 0
#measurement type X reporter interactions
Meta5$DSBMXCR <- Meta5$Variation*Meta5$ChildPDTReport
Meta5$DSBMXPR <- Meta5$Variation*Meta5$ParentPDTReport
Meta5$PBMXCR <- Meta5$Perception*Meta5$ChildPDTReport
Meta5$PBMXPR <- Meta5$Perception*Meta5$ParentPDTReport



#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#Only run this the first time, and then comment it out after the test run
#Meta5 <- Meta5[sample(nrow(Meta5), 100),]



#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
predMatrix <- make.predictorMatrix(Meta5)
#changing the matrix so that only the PCAs are used as auxiliary variables for imputing
######re-do this every time to make sure column numbers match
```

```
predMatrix[,c(1:32, 41:60)] <- 0  #A value of 1 in this matrix indicates that the corresponding column variable is used to predict the corresponding row variable
#only the PCAs should have 1s, and they should go down all the rows
predMatrix #verify - always verify this, it will mess everything up if it is wrong - only PCAs should have 1s
#checking the imputation method for each variable - it only really matters for the variables with missing data
#if a nominal variable is set to pmm, change it from numeric to factor so that it will use logistic reg
impMethod <- make.method(Meta5)
impMethod


#IMPUTE THE DATA
#IMPUTE THE DATA
#IMPUTE THE DATA
imp <- mice(Meta5, print=FALSE, m=50, predictorMatrix=predMatrix, method=impMethod, seed=1234)



#RUN AND EXPORT THE ANALYSIS - MODEL 1
#RUN AND EXPORT THE ANALYSIS - MODEL 1
#RUN AND EXPORT THE ANALYSIS - MODEL 1
#this runs the analysis
Model1 <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE, data=Meta5)
#this summarizes the analysis
summary(Model1, digits=3, transf=transf.ztor)
#this saves the analysis
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Model 1.txt")
RE_model <- Model1
RE_model
sink(NULL)

#this calculates the ICCs
M1ICCs<- MetaVarCompsModel1(Model1)
#this saves the ICCs
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Model 1 ICCs.txt")
RE_model <- M1ICCs
RE_model
sink(NULL)


#CHECK FAIL SAFE NUMBER FOR MODEL 1
#CHECK FAIL SAFE NUMBER FOR MODEL 1
```

```
#CHECK FAIL SAFE NUMBER FOR MODEL 1
#no need to use the imputed data set here because there are no imputed values on the effect size
FailSafe <- fsn(Meta5$yi, Meta5$vi,type="Rosenthal", alpha=.05, digits = 4)
FailSafe
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Fail Safe Number.txt")
RE_model <- FailSafe
RE_model
sink(NULL)

#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#Build a model without within-source variance
NoLevel2Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                      sigma2=c(0,NA, NA), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the within-source variance
WithinSource <- anova(Model1,NoLevel2Var)
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Within Source
Variance.txt")
RE_model <- WithinSource
RE_model
sink(NULL)


#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#Build a model without between-source variance
NoLevel3Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                      sigma2=c(NA,0, NA), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the between-source variance
BetweenSource<-anova(Model1,NoLevel3Var)
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Between Source
Variance.txt")
RE_model <- BetweenSource
RE_model
sink(NULL)


#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#Build a model without between-sample variance
NoLevel4Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                      sigma2=c(NA,NA, 0), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the between-sample variance
```

```
BetweenSample<-anova(Model1,NoLevel4Var)
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Between Sample
Variance.txt")
RE_model <- BetweenSample
RE_model
sink(NULL)


#RUNNING META REGRESSION - MODEL 2
#RUNNING META REGRESSION - MODEL 2
#RUNNING META REGRESSION - MODEL 2
#check frequencies of all nominal variables
sapply(Meta5[,c(5,6,8:23)], table)
table(Meta5$Variation, Meta5$ChildPDTReport)

#No Recollected
#No UsedBeta

#this runs the analysis by imputation (50 of them)
Model2 <- withold(imp, rma.mv(yi, vi,
mods = ~  USA +  # level four variables
FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +  GCSamePercC    +
AgeDifC  + AgeC + DadPDT + PeerRev + #level 3 variables
BothParsPDT   + ChildPDTReport + Variation     + NegPDT       + ResourcePDT + ControlPDT,
random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
Model2Pooled <- summary(pool(Model2))
Model2Pooled[-1] <- round(Model2Pooled[-1], digits=4)
Model2Pooled$LowerBound <- Model2Pooled$estimate - (Model2Pooled$std.error*1.96)
Model2Pooled$UpperBound <- Model2Pooled$estimate + (Model2Pooled$std.error*1.96)
Model2Pooled

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Model 2.txt")
RE_model <- Model2Pooled
RE_model
sink(NULL)


#retrieve the variance components for Model 2
DidWork <- MetaVarCompsModel2(Model2)
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Model 2 - variance
components.txt")
RE_model <- DidWork
RE_model
```

```
sink(NULL)

#test the significance of the variance components in Model 2 - this is done using the function below. The function takes the model for each imputation, and constrains the variance at that level to be zero and then tests model fit against the unrestricted(full) model, it then pools the needed stats across all 50 imputations
#for the MetaVarCompsModel2LevelVarianceTest function, the needed arguments are (unrestricted model, model with no level 2 variance, model with no level 3 variance, model with no level 4 variance)
Model2NoLevel2Var <- withold(imp, rma.mv(yi, vi,
                             mods = ~  USA +  # level four variables
                               FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC
+  GCSamePercC    + AgeDifC  + AgeC + DadPDT + PeerRev + #level 3 variables
                               BothParsPDT   + ChildPDTReport + Variation     + NegPDT       +
ResourcePDT + ControlPDT,
                             random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 |
DataID),sigma2=c(0,NA, NA), tdist=TRUE))
Model2NoLevel3Var <- withold(imp, rma.mv(yi, vi,
                             mods = ~  USA +  # level four variables
                               FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC
+  GCSamePercC    + AgeDifC  + AgeC + DadPDT + PeerRev + #level 3 variables
                               BothParsPDT   + ChildPDTReport + Variation     + NegPDT       +
ResourcePDT + ControlPDT,
                             random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 |
DataID),sigma2=c(NA,0, NA), tdist=TRUE))
Model2NoLevel4Var <- withold(imp, rma.mv(yi, vi,
                             mods = ~  USA +  # level four variables
                               FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC
+  GCSamePercC    + AgeDifC  + AgeC + DadPDT + PeerRev + #level 3 variables
                               BothParsPDT   + ChildPDTReport + Variation     + NegPDT       +
ResourcePDT + ControlPDT,
                             random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 |
DataID),sigma2=c(NA,NA, 0), tdist=TRUE))
M2VarTest <- MetaVarCompsModel2LevelVarianceTest(Model2, Model2NoLevel2Var,
Model2NoLevel3Var, Model2NoLevel4Var)
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Model 2 - variance components significance tests.txt")
RE_model <- M2VarTest
RE_model
sink(NULL)



#RUNNING META REGRESSION - MODEL 3
#RUNNING META REGRESSION - MODEL 3
#RUNNING META REGRESSION - MODEL 3
```

```
#Not enough variation


#TESTING SIMPLE SLOPES - DO NOT RE-RUN MODELS THAT THE INFO IS IN
MODEL 2 - ONLY THE INVERSE OF EACH VARIABLE
#TESTING SIMPLE SLOPES - DO NOT RE-RUN MODELS THAT THE INFO IS IN
MODEL 2 - ONLY THE INVERSE OF EACH VARIABLE
#TESTING SIMPLE SLOPES - DO NOT RE-RUN MODELS THAT THE INFO IS IN
MODEL 2 - ONLY THE INVERSE OF EACH VARIABLE

#PDT FROM MOM
#PDT FROM MOM
#PDT FROM MOM
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Simple Slopes - PDT
from Mom.txt")
RE_model <- Model2Pooled
RE_model
sink(NULL)

#PDT FROM BOTH PARENTS
#PDT FROM BOTH PARENTS
#PDT FROM BOTH PARENTS
PDTBoth <- withold(imp, rma.mv(yi, vi,
                       mods = ~  USA +  # level four variables
                         FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                         DadPDT + MomPDT   + ChildPDTReport + Variation     + NegPDT       +
ResourcePDT + ControlPDT,
                       random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
PDTBoth <- summary(pool(PDTBoth))
PDTBoth[-1] <- round(PDTBoth[-1], digits=4)
PDTBoth$LowerBound <- PDTBoth$estimate - (PDTBoth$std.error*1.96)
PDTBoth$UpperBound <- PDTBoth$estimate + (PDTBoth$std.error*1.96)
PDTBoth


sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Simple Slopes - PDT
from Both Parents.txt")
RE_model <- PDTBoth
RE_model
sink(NULL)
```

```
#POS PDT
#POS PDT
#POS PDT
sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Simple Slopes - Pos
PDT.txt")
RE_model <- Model2Pooled
RE_model
sink(NULL)



#NEG PDT
#NEG PDT
#NEG PDT
PDTNeg <- withold(imp, rma.mv(yi, vi,
                      mods = ~  USA +  # level four variables
                        FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                        DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT
+ ResourcePDT + ControlPDT,
                      random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
PDTNeg <- summary(pool(PDTNeg))
PDTNeg[-1] <- round(PDTNeg[-1], digits=4)
PDTNeg$LowerBound <- PDTNeg$estimate - (PDTNeg$std.error*1.96)
PDTNeg$UpperBound <- PDTNeg$estimate + (PDTNeg$std.error*1.96)
PDTNeg

sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Simple Slopes - Neg
PDT.txt")
RE_model <- PDTNeg
RE_model
sink(NULL)




#CONTROL PDT
#CONTROL PDT
#CONTROL PDT
PDTControl <- withold(imp, rma.mv(yi, vi,
```

```
                        mods = ~  USA +  # level four variables
                          FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                          DadPDT + BothParsPDT   + ChildPDTReport + Variation     + NegPDT
+ ResourcePDT + PosPDT,
                        random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
PDTControl <- summary(pool(PDTControl))
PDTControl[-1] <- round(PDTControl[-1], digits=4)
PDTControl$LowerBound <- PDTControl$estimate - (PDTControl$std.error*1.96)
PDTControl$UpperBound <- PDTControl$estimate + (PDTControl$std.error*1.96)
PDTControl

sink(file = "Findings/Imputed Data Findings/Age Spacing/Age Spacing - Simple Slopes -
Control PDT.txt")
RE_model <- PDTControl
RE_model
sink(NULL)
```

() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () ()
() () () () () ()
() () () () () ()
() () () Four-level multilevel meta models () () ()
() () () testing links between Sex Composition () () ()
() () () and absolute PDT () () ()
() () () () () ()
() () () () () ()
() () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()
() () () () () () () () () () () () () () () () () ()

#SET THE WORKING DIRCTORY
#SET THE WORKING DIRCTORY
#SET THE WORKING DIRCTORY

#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND THE MetaModelsDescriptiveData() FUNCTION
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND THE MetaModelsDescriptiveData() FUNCTION
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND THE MetaModelsDescriptiveData() FUNCTION
#for info on withold(), see https://www.metafor-project.org/doku.php/tips:multiple_imputation_with_mice_and_metafor for more details
#I created the MetaModelsDescriptiveData(), MetaVarCompsModel1(), MetaVarCompsModel2(), and MetaVarCompsModel2LevelVarianceTest()
#functions - more notes are available in the original file "Meta Analysis Functions.R"
library(metafor)
library(PcAux)
library(mice)
library(psych)
library(tidyverse)
library(car)

#IMPORT THE ENTIRE META DATA FRAME
#IMPORT THE ENTIRE META DATA FRAME
#IMPORT THE ENTIRE META DATA FRAME
WholeMeta <- read.csv(file = "Programs and Data/FinalCleanPDTMeta.csv", header = T)

```
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Sex Composition
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Sex Composition
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Sex Composition
Meta1 <- subset(x = WholeMeta, subset = MagnitudeDirectional == 0 & Outcome == 21)



#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
Meta2 <- Meta1[,

c("ESNum","SampleSize","Correlation","ArticleID","PeerRev","USA","DataID","FamSize","A
geDif",

"UsedBeta","OverallPDT","PosPDT","NegPDT","EvalPDT","ResourcePDT","FairPDT","Contr
olPDT",

"ChildPDTReport","ParentPDTReport","MomPDT","DadPDT","BothParsPDT","Variation","Pe
rception",
            "Recollected","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")]

MetaModelsDescriptiveData(Meta2) # this saves the descriptive stats for the non-imputed model
#we take descriptive stats from this data frame so that it includes missing data and the non-
centered versions of continous variables



#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#we will drop the non-centered versions of continuous variables, because we want the imputation
to be based on the centered versions
Meta3 <- Meta2
attach(Meta3)
```

```
Meta3$FamSizeC <- FamSize-mean(FamSize, na.rm = T)
Meta3$AgeDifC <- AgeDif-mean(AgeDif, na.rm = T)
Meta3$AgeC <- Age-mean(Age, na.rm = T)
Meta3$WhitePercC <- WhitePerc-mean(WhitePerc, na.rm = T)
Meta3$GCSamePercC <- GCSamePerc-mean(GCSamePerc, na.rm = T)
Meta3$MalePercC <- MalePerc-mean(MalePerc, na.rm = T)
Meta3$RelBirthYoungPercC <- RelBirthYoungPerc-mean(RelBirthYoungPerc, na.rm = T)
detach(Meta3)
DropVars <- names(Meta3) %in%
c("AgeDif","FamSize","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")
Meta3 <- Meta3[!DropVars]
DropVars <- Meta3 %in%
c("AgeDif","FamSize","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")



#SET THE EFFECT SIZES
#SET THE EFFECT SIZES
#SET THE EFFECT SIZES
#Do this before the PCAs, it doesn't like doing it on an imputed data frame, so do it first
Meta4 <- escalc(measure = "ZCOR", ri =Correlation,
            ni = SampleSize, data = Meta3, vtype = "LS", append = TRUE)




#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#re-do this step every time

MetaNOMISS <- Meta4[, which(colMeans(is.na(Meta4)) ==.00)]#creating data frame with only
variables with no missing values

names(MetaNOMISS)#I run this to get the names and then sort those into the type of variable
below

## Prepare the data by first sorting out what types of variables I have - continuous variables are
left off the code but listed below
myNoms  <- c("PeerRev"    ,    "USA", "UsedBeta"    ,    "OverallPDT",      "PosPDT",
"NegPDT",          "EvalPDT",   "ResourcePDT",     "FairPDT",         "ControlPDT",
```

```
"ChildPDTReport",  "ParentPDTReport", "MomPDT",    "DadPDT"   ,      "BothParsPDT",
"Variation",      "Perception",     "Recollected")
myIds   <- c("ESNum", "ArticleID", "DataID")
#myOrds   <- c() #none this time
#Continuous -  "SampleSize" ,    "Correlation","yi"      ,      "vi"  ,    "FamSizeC",




#Create data frame for creating PCAs - prep data step
ReadyForPCA <- prepData(rawData = MetaNOMISS,
                        nomVars = myNoms,
                        ordVars = NULL, # make sure to update this if there are ordinal variables
                        idVars = myIds,
                        moderators = NULL,
                        verbose = 2L,
                        nProcess = 4,
                        simMode = F)
y #the prepData code required a yes or no to continue, this tells it to continue

## Create principal component auxiliary variables:
PCAOut <- createPcAux(pcAuxData = ReadyForPCA,
                      nComps = c(0.5,0.5),
                      maxPolyPow = 3L,
                      doImputation = FALSE,
                      interactType = 1)
y #the createPcAux code required a yes or no to continue, this tells it to continue


## Merge the PC auxiliaries with the original data:
Meta5 <- mergePcAux(pcAuxData = PCAOut, rawData = Meta4) #use this data for imputation


#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
Meta5$FamSizeH <- Meta5$FamSizeC - sd(Meta5$FamSizeC, na.rm = T)
Meta5$FamSizeL <- Meta5$FamSizeC + sd(Meta5$FamSizeC, na.rm = T)
Meta5$AgeDifH <- Meta5$AgeDifC - sd(Meta5$AgeDifC, na.rm = T)
```

```
Meta5$AgeDifL <- Meta5$AgeDifC + sd(Meta5$AgeDifC, na.rm = T)
Meta5$AgeH <- Meta5$AgeC - sd(Meta5$AgeC, na.rm = T)
Meta5$AgeL <- Meta5$AgeC + sd(Meta5$AgeC, na.rm = T)
Meta5$WhitePercH <- Meta5$WhitePercC - sd(Meta5$WhitePercC, na.rm = T)
Meta5$WhitePercL <- Meta5$WhitePercC + sd(Meta5$WhitePercC, na.rm = T)
Meta5$GCSamePercH <- Meta5$GCSamePercC - sd(Meta5$GCSamePercC, na.rm = T)
Meta5$GCSamePercL <- Meta5$GCSamePercC + sd(Meta5$GCSamePercC, na.rm = T)
Meta5$MalePercH <- Meta5$MalePercC - sd(Meta5$MalePercC, na.rm = T)
Meta5$MalePercL <- Meta5$MalePercC + sd(Meta5$MalePercC, na.rm = T)
Meta5$RelBirthYoungPercH <- Meta5$RelBirthYoungPercC - sd(Meta5$RelBirthYoungPercC,
na.rm = T)
Meta5$RelBirthYoungPercL <- Meta5$RelBirthYoungPercC + sd(Meta5$RelBirthYoungPercC,
na.rm = T)
Meta5$NonPeerRev <- 1 #creating a variable where non-peer reviewed (mostly raw data) is
coded as 1, and peer reviewed as zero
Meta5$NonPeerRev[Meta5$PeerRev == 1] <- 0
Meta5$Concurrent <- 1 # creating a variable where concurrent PDT is 1, recollected is zero
Meta5$Concurrent[Meta5$Recollected == 1] <- 0
#measurement type X reporter interactions
Meta5$DSBMXCR <- Meta5$Variation*Meta5$ChildPDTReport
Meta5$DSBMXPR <- Meta5$Variation*Meta5$ParentPDTReport
Meta5$PBMXCR <- Meta5$Perception*Meta5$ChildPDTReport
Meta5$PBMXPR <- Meta5$Perception*Meta5$ParentPDTReport



#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#Only run this the first time, and then comment it out after the test run
#Meta5 <- Meta5[sample(nrow(Meta5), 100),]



#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
predMatrix <- make.predictorMatrix(Meta5)
#changing the matrix so that only the PCAs are used as auxiliary variables for imputing
######re-do this every time to make sure column numbers match
```

```
predMatrix[,c(1:32, 37:56)] <- 0  #A value of 1 in this matrix indicates that the corresponding column variable is used to predict the corresponding row variable
#only the PCAs should have 1s, and they should go down all the rows
predMatrix #verify - always verify this, it will mess everything up if it is wrong - only PCAs should have 1s
#checking the imputation method for each variable - it only really matters for the variables with missing data
#if a nominal variable is set to pmm, change it from numeric to factor so that it will use logistic reg
impMethod <- make.method(Meta5)
impMethod


#IMPUTE THE DATA
#IMPUTE THE DATA
#IMPUTE THE DATA
imp <- mice(Meta5, print=FALSE, m=50, predictorMatrix=predMatrix, method=impMethod, seed=1234)



#RUN AND EXPORT THE ANALYSIS - MODEL 1
#RUN AND EXPORT THE ANALYSIS - MODEL 1
#RUN AND EXPORT THE ANALYSIS - MODEL 1
#this runs the analysis
Model1 <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE, data=Meta5)
#this summarizes the analysis
summary(Model1, digits=3, transf=transf.ztor)
#this saves the analysis
sink(file = "Findings/Imputed Data Findings/Sex Composition/Sex Composition - Model 1.txt")
RE_model <- Model1
RE_model
sink(NULL)

#this calculates the ICCs
M1ICCs<- MetaVarCompsModel1(Model1)
#this saves the ICCs
sink(file = "Findings/Imputed Data Findings/Sex Composition/Sex Composition - Model 1 ICCs.txt")
RE_model <- M1ICCs
RE_model
sink(NULL)


#CHECK FAIL SAFE NUMBER FOR MODEL 1
```

```
#CHECK FAIL SAFE NUMBER FOR MODEL 1
#CHECK FAIL SAFE NUMBER FOR MODEL 1
#no need to use the imputed data set here because there are no imputed values on the effect size
FailSafe <- fsn(Meta5$yi, Meta5$vi,type="Rosenthal", alpha=.05, digits = 4)
FailSafe
sink(file = "Findings/Imputed Data Findings/Sex Composition/Sex Composition - Fail Safe
Number.txt")
RE_model <- FailSafe
RE_model
sink(NULL)

#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#Build a model without within-source variance
NoLevel2Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                sigma2=c(0,NA, NA), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the within-source variance
WithinSource <- anova(Model1,NoLevel2Var)
sink(file = "Findings/Imputed Data Findings/Sex Composition/Sex Composition - Within Source
Variance.txt")
RE_model <- WithinSource
RE_model
sink(NULL)


#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#Build a model without between-source variance
NoLevel3Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                sigma2=c(NA,0, NA), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the between-source variance
BetweenSource<-anova(Model1,NoLevel3Var)
sink(file = "Findings/Imputed Data Findings/Sex Composition/Sex Composition - Between
Source Variance.txt")
RE_model <- BetweenSource
RE_model
sink(NULL)


#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#Build a model without between-sample variance
NoLevel4Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
```

```
                sigma2=c(NA,NA, 0), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the between-sample variance
BetweenSample<-anova(Model1,NoLevel4Var)
sink(file = "Findings/Imputed Data Findings/Sex Composition/Sex Composition - Between
Sample Variance.txt")
RE_model <- BetweenSample
RE_model
sink(NULL)


#RUNNING META REGRESSION - MODEL 2
#RUNNING META REGRESSION - MODEL 2
#RUNNING META REGRESSION - MODEL 2


#NO SIGNIFICANT VARIATION AT ANY LEVEL
```

```
# () () () () () () () () () () () () () () () () () ()
# () () () () () () () () () () () () () () () () () ()
# () () () () () () () () () () () () () () () () () ()
# () () ()                                    () () ()
# () () ()                                    () () ()
# () () ()                                    () () ()
# () () ()      Four-level multilevel meta models        () () ()
# () () ()      testing links between Family Size        () () ()
# () () ()      and absolute PDT                   () () ()
# () () ()                                    () () ()
# () () ()                                    () () ()
# () () ()                                    () () ()
# () () () () () () () () () () () () () () () () () ()
# () () () () () () () () () () () () () () () () () ()
# () () () () () () () () () () () () () () () () () ()
```

```
#SET THE WORKING DIRCTORY
#SET THE WORKING DIRCTORY
#SET THE WORKING DIRCTORY
```

```
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND
THE MetaModelsDescriptiveData() FUNCTION
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND
THE MetaModelsDescriptiveData() FUNCTION
#ACTIAVTE ALL NEEDED PACKAGES AND CREATE THE withold() FUNCTION AND
THE MetaModelsDescriptiveData() FUNCTION
#for info on withold(), see https://www.metafor-
project.org/doku.php/tips:multiple_imputation_with_mice_and_metafor for more details
#I created the MetaModelsDescriptiveData(), MetaVarCompsModel1(),
MetaVarCompsModel2(), and MetaVarCompsModel2LevelVarianceTest()
#functions - more notes are available in the original file "Meta Analysis Functions.R"
library(metafor)
library(PcAux)
library(mice)
library(psych)
library(tidyverse)
library(car)
```

```
#IMPORT THE ENTIRE META DATA FRAME
#IMPORT THE ENTIRE META DATA FRAME
#IMPORT THE ENTIRE META DATA FRAME
WholeMeta <- read.csv(file = "Programs and Data/FinalCleanPDTMeta.csv", header = T)
```

```
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Family Size
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Family Size
#SUBSET A NEW DATA FRAME THAT HAS ONLY CASES WITH ABSOLUTE PDT AND
Family Size
Meta1 <- subset(x = WholeMeta, subset = MagnitudeDirectional == 0 & Outcome == 16)



#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
#CREATE A SUBSET WITH ONLY THE NEEDED VARIABLES AND EXPORT THE
DESCRIPTIVE STATS
Meta2 <- Meta1[,

c("ESNum","SampleSize","Correlation","ArticleID","PeerRev","USA","DataID","FamSize","A
geDif",

"UsedBeta","OverallPDT","PosPDT","NegPDT","EvalPDT","ResourcePDT","FairPDT","Contr
olPDT",

"ChildPDTReport","ParentPDTReport","MomPDT","DadPDT","BothParsPDT","Variation","Pe
rception",
            "Recollected","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")]

MetaModelsDescriptiveData(Meta2) # this saves the descriptive stats for the non-imputed model
#we take descriptive stats from this data frame so that it includes missing data and the non-
centered versions of continous variables



#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#CREATE A NEW DATA FRAME WITH CENTERED VERSIONS OF ALL CONTINOUS
VARIABLES
#we will drop the non-centered versions of continuous variables, because we want the imputation
to be based on the centered versions
```

```
Meta3 <- Meta2
attach(Meta3)
Meta3$FamSizeC <- FamSize-mean(FamSize, na.rm = T)
Meta3$AgeDifC <- AgeDif-mean(AgeDif, na.rm = T)
Meta3$AgeC <- Age-mean(Age, na.rm = T)
Meta3$WhitePercC <- WhitePerc-mean(WhitePerc, na.rm = T)
Meta3$GCSamePercC <- GCSamePerc-mean(GCSamePerc, na.rm = T)
Meta3$MalePercC <- MalePerc-mean(MalePerc, na.rm = T)
Meta3$RelBirthYoungPercC <- RelBirthYoungPerc-mean(RelBirthYoungPerc, na.rm = T)
detach(Meta3)
DropVars <- names(Meta3) %in%
c("AgeDif","FamSize","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")
Meta3 <- Meta3[!DropVars]
DropVars <- Meta3 %in%
c("AgeDif","FamSize","Age","WhitePerc","GCSamePerc","MalePerc","RelBirthYoungPerc")



#SET THE EFFECT SIZES
#SET THE EFFECT SIZES
#SET THE EFFECT SIZES
#Do this before the PCAs, it doesn't like doing it on an imputed data frame, so do it first
Meta4 <- escalc(measure = "ZCOR", ri =Correlation,
                ni = SampleSize, data = Meta3, vtype = "LS", append = TRUE)




#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#CREATE THE PCAS BASED ON VARIABLES IN Meta3 THAT HAVE NO MISSING
VALUES
#re-do this step every time

MetaNOMISS <- Meta4[, which(colMeans(is.na(Meta4)) ==.00)]#creating data frame with only
variables with no missing values

names(MetaNOMISS)#I run this to get the names and then sort those into the type of variable
below

## Prepare the data by first sorting out what types of variables I have - continuous variables are
left off the code but listed below
```

```
myNoms   <- c("PeerRev",        "USA",        "UsedBeta",        "OverallPDT",      "PosPDT",
"NegPDT",         "EvalPDT",   "ResourcePDT",     "FairPDT",         "ControlPDT",
"ChildPDTReport",  "ParentPDTReport", "MomPDT",    "DadPDT",          "BothParsPDT",
"Variation",       "Perception",      "Recollected")
myIds    <- c("ESNum", "ArticleID", "DataID")
#myOrds   <- c() #none this time
#Continuous - "SampleSize",      "Correlation",    "MalePercC",     "yi",             "vi"




#Create data frame for creating PCAs - prep data step
ReadyForPCA <- prepData(rawData = MetaNOMISS,
                        nomVars = myNoms,
                        ordVars = NULL, # make sure to update this if there are ordinal variables
                        idVars = myIds,
                        moderators = NULL,
                        verbose = 2L,
                        nProcess = 4,
                        simMode = F)
y #the prepData code required a yes or no to continue, this tells it to continue

## Create principal component auxiliary variables:
PCAOut <- createPcAux(pcAuxData = ReadyForPCA,
                      nComps = c(0.5,0.5),
                      maxPolyPow = 3L,
                      doImputation = FALSE,
                      interactType = 1)
y #the createPcAux code required a yes or no to continue, this tells it to continue


## Merge the PC auxiliaries with the original data:
Meta5 <- mergePcAux(pcAuxData = PCAOut, rawData = Meta4) #use this data for imputation


#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
#CREATING INTERACTION TERMS, HIGH AND LOW OF MODERATORS AND
REVERSING DUMMY CODES AS NEEDED - ALL FOR INTERACTIONS AND SIMPLE
SLOPES
Meta5$FamSizeH <- Meta5$FamSizeC - sd(Meta5$FamSizeC, na.rm = T)
```

```
Meta5$FamSizeL <- Meta5$FamSizeC + sd(Meta5$FamSizeC, na.rm = T)
Meta5$AgeDifH <- Meta5$AgeDifC - sd(Meta5$AgeDifC, na.rm = T)
Meta5$AgeDifL <- Meta5$AgeDifC + sd(Meta5$AgeDifC, na.rm = T)
Meta5$AgeH <- Meta5$AgeC - sd(Meta5$AgeC, na.rm = T)
Meta5$AgeL <- Meta5$AgeC + sd(Meta5$AgeC, na.rm = T)
Meta5$WhitePercH <- Meta5$WhitePercC - sd(Meta5$WhitePercC, na.rm = T)
Meta5$WhitePercL <- Meta5$WhitePercC + sd(Meta5$WhitePercC, na.rm = T)
Meta5$GCSamePercH <- Meta5$GCSamePercC - sd(Meta5$GCSamePercC, na.rm = T)
Meta5$GCSamePercL <- Meta5$GCSamePercC + sd(Meta5$GCSamePercC, na.rm = T)
Meta5$MalePercH <- Meta5$MalePercC - sd(Meta5$MalePercC, na.rm = T)
Meta5$MalePercL <- Meta5$MalePercC + sd(Meta5$MalePercC, na.rm = T)
Meta5$RelBirthYoungPercH <- Meta5$RelBirthYoungPercC - sd(Meta5$RelBirthYoungPercC,
na.rm = T)
Meta5$RelBirthYoungPercL <- Meta5$RelBirthYoungPercC + sd(Meta5$RelBirthYoungPercC,
na.rm = T)
Meta5$NonPeerRev <- 1 #creating a variable where non-peer reviewed (mostly raw data) is
coded as 1, and peer reviewed as zero
Meta5$NonPeerRev[Meta5$PeerRev == 1] <- 0
Meta5$Concurrent <- 1 # creating a variable where concurrent PDT is 1, recollected is zero
Meta5$Concurrent[Meta5$Recollected == 1] <- 0
#measurement type X reporter interactions
Meta5$DSBMXCR <- Meta5$Variation*Meta5$ChildPDTReport
Meta5$DSBMXPR <- Meta5$Variation*Meta5$ParentPDTReport
Meta5$PBMXCR <- Meta5$Perception*Meta5$ChildPDTReport
Meta5$PBMXPR <- Meta5$Perception*Meta5$ParentPDTReport



#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#DRAW SMALL RANDOM SAMPLE OF THE DATA TO TEST RUN THE ANALYSIS
AND MAKE SURE IT WORKS
#Only run this the first time, and then comment it out after the test run
#Meta5 <- Meta5[sample(nrow(Meta5), 100),]



#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
#PREPPING THE PREDICTION MATRIX FOR THE MICE MODEL - THIS TELLS IT
WHICH VARIABLES TO USE AS PREDICTORS AND WHICH TO IMPUTE
predMatrix <- make.predictorMatrix(Meta5)
#changing the matrix so that only the PCAs are used as auxiliary variables for imputing
```

```
######re-do this every time to make sure column numbers match
predMatrix[,c(1:32, 40:59)] <- 0  #A value of 1 in this matrix indicates that the corresponding column variable is used to predict the corresponding row variable
#only the PCAs should have 1s, and they should go down all the rows
predMatrix #verify - always verify this, it will mess everything up if it is wrong - only PCAs should have 1s
#checking the imputation method for each variable - it only really matters for the variables with missing data
#if a nominal variable is set to pmm, change it from numeric to factor so that it will use logistic reg
impMethod <- make.method(Meta5)
impMethod


#IMPUTE THE DATA
#IMPUTE THE DATA
#IMPUTE THE DATA
imp <- mice(Meta5, print=FALSE, m=50, predictorMatrix=predMatrix, method=impMethod, seed=1234)



#RUN AND EXPORT THE ANALYSIS - MODEL 1
#RUN AND EXPORT THE ANALYSIS - MODEL 1
#RUN AND EXPORT THE ANALYSIS - MODEL 1
#this runs the analysis
Model1 <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE, data=Meta5)
#this summarizes the analysis
summary(Model1, digits=3, transf=transf.ztor)
#this saves the analysis
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Model 1.txt")
RE_model <- Model1
RE_model
sink(NULL)

#this calculates the ICCs
M1ICCs<- MetaVarCompsModel1(Model1)
#this saves the ICCs
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Model 1 ICCs.txt")
RE_model <- M1ICCs
RE_model
sink(NULL)


#CHECK FAIL SAFE NUMBER FOR MODEL 1
```

```
#CHECK FAIL SAFE NUMBER FOR MODEL 1
#CHECK FAIL SAFE NUMBER FOR MODEL 1
#no need to use the imputed data set here because there are no imputed values on the effect size
FailSafe <- fsn(Meta5$yi, Meta5$vi,type="Rosenthal", alpha=.05, digits = 4)
FailSafe
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Fail Safe Number.txt")
RE_model <- FailSafe
RE_model
sink(NULL)

#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#CHECK FOR SIGNIFICANT WITHIN SOURCE VARIATION - LEVEL 2
#Build a model without within-source variance
NoLevel2Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                sigma2=c(0,NA, NA), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the within-source variance
WithinSource <- anova(Model1,NoLevel2Var)
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Within Source
Variance.txt")
RE_model <- WithinSource
RE_model
sink(NULL)


#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 3
#Build a model without between-source variance
NoLevel3Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                sigma2=c(NA,0, NA), tdist=TRUE, data=Meta5)
# Perform a likelihood-ratio-test to determine the significance of the between-source variance
BetweenSource<-anova(Model1,NoLevel3Var)
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Between Source
Variance.txt")
RE_model <- BetweenSource
RE_model
sink(NULL)


#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#CHECK FOR SIGNIFICANT BETWEEN SOURCE VARIATION - LEVEL 4
#Build a model without between-sample variance
NoLevel4Var <- rma.mv(yi, vi, random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID),
                sigma2=c(NA,NA, 0), tdist=TRUE, data=Meta5)
```

```
# Perform a likelihood-ratio-test to determine the significance of the between-sample variance
BetweenSample<-anova(Model1,NoLevel4Var)
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Between Sample
Variance.txt")
RE_model <- BetweenSample
RE_model
sink(NULL)


#RUNNING META REGRESSION - MODEL 2
#RUNNING META REGRESSION - MODEL 2
#RUNNING META REGRESSION - MODEL 2
#check frequencies of all nominal variables
sapply(Meta5[,c(5,6,8:23)], table)
table(Meta5$Variation, Meta5$ChildPDTReport)

#No Recollected
#No UsedBeta

#this runs the analysis by imputation (50 of them)
Model2 <- withold(imp, rma.mv(yi, vi,
                      mods = ~    # level four variables
                        FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                        DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                      random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
Model2Pooled <- summary(pool(Model2))
Model2Pooled[-1] <- round(Model2Pooled[-1], digits=4)
Model2Pooled$LowerBound <- Model2Pooled$estimate - (Model2Pooled$std.error*1.96)
Model2Pooled$UpperBound <- Model2Pooled$estimate + (Model2Pooled$std.error*1.96)
Model2Pooled

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Model 2.txt")
RE_model <- Model2Pooled
RE_model
sink(NULL)


#retrieve the variance components for Model 2
DidWork <- MetaVarCompsModel2(Model2)
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Model 2 - variance
components.txt")
```

```
RE_model <- DidWork
RE_model
sink(NULL)

#test the significance of the variance components in Model 2 - this is done using the function below. The function takes the model for each imputation, and constrains the variance at that level to be zero and then tests model fit against the unrestricted(full) model, it then pools the needed stats across all 50 imputations
#for the MetaVarCompsModel2LevelVarianceTest function, the needed arguments are (unrestricted model, model with no level 2 variance, model with no level 3 variance, model with no level 4 variance)
Model2NoLevel2Var <- withold(imp, rma.mv(yi, vi,
                             mods = ~    # level four variables
                               FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC
+  GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                               DadPDT + BothParsPDT   + ChildPDTReport + Variation     +
PosPDT + NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                             random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 |
DataID),sigma2=c(0,NA, NA), tdist=TRUE))
Model2NoLevel3Var <- withold(imp, rma.mv(yi, vi,
                             mods = ~    # level four variables
                               FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC
+  GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                               DadPDT + BothParsPDT   + ChildPDTReport + Variation     +
PosPDT + NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                             random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 |
DataID),sigma2=c(NA,0, NA), tdist=TRUE))
Model2NoLevel4Var <- withold(imp, rma.mv(yi, vi,
                             mods = ~    # level four variables
                               FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC
+  GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                               DadPDT + BothParsPDT   + ChildPDTReport + Variation     +
PosPDT + NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                             random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 |
DataID),sigma2=c(NA,NA, 0), tdist=TRUE))
M2VarTest <- MetaVarCompsModel2LevelVarianceTest(Model2, Model2NoLevel2Var,
Model2NoLevel3Var, Model2NoLevel4Var)
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Model 2 - variance components significance tests.txt")
RE_model <- M2VarTest
RE_model
sink(NULL)
```

```
#TESTING SIMPLE SLOPES - DO NOT RE-RUN MODELS THAT THE INFO IS IN
MODEL 2 - ONLY THE INVERSE OF EACH VARIABLE
#TESTING SIMPLE SLOPES - DO NOT RE-RUN MODELS THAT THE INFO IS IN
MODEL 2 - ONLY THE INVERSE OF EACH VARIABLE
#TESTING SIMPLE SLOPES - DO NOT RE-RUN MODELS THAT THE INFO IS IN
MODEL 2 - ONLY THE INVERSE OF EACH VARIABLE


#SMALLER FAMILIES
#SMALLER FAMILIES
#SMALLER FAMILIES
SmallFams <- withold(imp, rma.mv(yi, vi,
                         mods = ~    # level four variables
                           FamSizeL       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                           DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                         random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
SmallFams <- summary(pool(SmallFams))
SmallFams[-1] <- round(SmallFams[-1], digits=4)
SmallFams

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Family Size
Small.txt")
RE_model <- SmallFams
RE_model
sink(NULL)


#LARGER FAMILIES
#LARGER FAMILIES
#LARGER FAMILIES
LargeFams <- withold(imp, rma.mv(yi, vi,
                         mods = ~    # level four variables
                           FamSizeH       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                           DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                         random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
LargeFams <- summary(pool(LargeFams))
LargeFams[-1] <- round(LargeFams[-1], digits=4)
```

```
LargeFams

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Family Size
Large.txt")
RE_model <- LargeFams
RE_model
sink(NULL)



#AGE LOW
#AGE LOW
#AGE LOW
AgeLow <- withold(imp, rma.mv(yi, vi,
                      mods = ~    # level four variables
                        FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeL +  PeerRev + #level 3 variables
                        DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                      random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
AgeLow <- summary(pool(AgeLow))
AgeLow[-1] <- round(AgeLow[-1], digits=4)
AgeLow

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Age Low.txt")
RE_model <- AgeLow
RE_model
sink(NULL)


#AGE High
#AGE High
#AGE High
AgeHigh <- withold(imp, rma.mv(yi, vi,
                      mods = ~    # level four variables
                        FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeH +  PeerRev + #level 3 variables
                        DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                      random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
```

```
AgeHigh <- summary(pool(AgeHigh))
AgeHigh[-1] <- round(AgeHigh[-1], digits=4)
AgeHigh

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Age High.txt")
RE_model <- AgeHigh
RE_model
sink(NULL)



#RAW DATA
#RAW DATA
#RAW DATA
sink(file = "Findings/Imputed Data Findings/Family Size/Family Size - Simple Slopes - Raw
data.txt")
RE_model <- Model2Pooled
RE_model
sink(NULL)


#PEER REVIEWED
#PEER REVIEWED
#PEER REVIEWED
PeerRev <- withold(imp, rma.mv(yi, vi,
                        mods = ~    # level four variables
                          FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  NonPeerRev + #level 3 variables
                          DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + ControlPDT, #level 2 variables
                        random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
PeerRev <- summary(pool(PeerRev))
PeerRev[-1] <- round(PeerRev[-1], digits=4)
PeerRev

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Peer
Reviewed.txt")
RE_model <- PeerRev
RE_model
sink(NULL)
```

```
#RESOURCE PDT
#RESOURCE PDT
#RESOURCE PDT
ResourcePDT <- withold(imp, rma.mv(yi, vi,
                        mods = ~    # level four variables
                          FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                          DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + OverallPDT + ControlPDT, #level 2 variables
                        random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
ResourcePDT <- summary(pool(ResourcePDT))
ResourcePDT[-1] <- round(ResourcePDT[-1], digits=4)
ResourcePDT

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Resource
PDT.txt")
RE_model <- ResourcePDT
RE_model
sink(NULL)

#CONTROL PDT
#CONTROL PDT
#CONTROL PDT
ControlPDT <- withold(imp, rma.mv(yi, vi,
                       mods = ~    # level four variables
                         FamSizeC       + WhitePercC + MalePercC    + RelBirthYoungPercC +
GCSamePercC    + AgeDifC  + AgeC +  PeerRev + #level 3 variables
                         DadPDT + BothParsPDT   + ChildPDTReport + Variation     + PosPDT +
NegPDT       + ResourcePDT + OverallPDT, #level 2 variables
                       random = list(~ 1 | ESNum, ~ 1 | ArticleID, ~ 1 | DataID), tdist=TRUE))

#this summarizes and pools the analysis
ControlPDT <- summary(pool(ControlPDT))
ControlPDT[-1] <- round(ControlPDT[-1], digits=4)
ControlPDT

#this exports the findings to a text file
sink(file = "Findings/Imputed Data Findings/Family Size/Testing Simple Slopes - Control
PDT.txt")
RE_model <- ControlPDT
RE_model
sink(NULL)
```