2marginal $R^2 = 2conditional R^2 =$

Field Experiment

Jacob Taylor   Evolutionary Anthropology, University of Oxford
Emma Cohen
Institute of Cognitive

```
## -- Attaching packages --------------------------------------------------------------------
tidyverse 1.2.1 --
## ggplot2 2.2.1     purrr  0.2.5
## tibble  1.4.2     dplyr  0.7.6
## tidyr   0.8.1     stringr 1.3.1
## readr   1.1.1     forcats 0.3.0
## Warning:  package 'dplyr' was built under R version 3.5.1
## -- Conflicts -----------------------------------------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## Loading required package:  Matrix
##
## Attaching package:  'Matrix'
## The following object is masked from 'package:tidyr':
##
##     expand
##
## Attaching package:  'lmerTest'
## The following object is masked from 'package:lme4':
##
##     lmer
## The following object is masked from 'package:stats':
##
##     step
## Loading required package:  MASS
##
## Attaching package:  'MASS'
## The following object is masked from 'package:dplyr':
##
##     select
## Loading required package:  mvtnorm
## Loading required package:  sandwich
## mediation:  Causal Mediation Analysis
## Version:  4.4.6
## Loading required package:  carData
##
## Attaching package:  'car'
## The following object is masked from 'package:dplyr':
##
##     recode
## The following object is masked from 'package:purrr':
##
##     some
##
## Attaching package:  'HLMdiag'
## The following object is masked from 'package:stats':
##
##     covratio
```

```
##
## Attaching package:  'influence.ME'
## The following object is masked from 'package:stats':
##
##     influence
## Error in contrib.url(repos, type):  trying to use CRAN without setting a mirror
##
##  This is piecewiseSEM version 2.0.2
##
##  If you have used the package before, it is strongly recommended you read Section 3 of the vignette('
to familiarize yourself with the new syntax
##
##  Questions or bugs can be addressed to <jlefcheck@bigelow.org>
## Downloading GitHub repo crsh/papaja@master
## from URL https://api.github.com/repos/crsh/papaja/zipball/master
## Installing papaja
## '/Library/Frameworks/R.framework/Resources/bin/R' --no-site-file  \
##   --no-environ --no-save --no-restore --quiet CMD INSTALL  \
##   '/private/var/folders/0p/39cmn5sj3355scsq_n6nq3440000gq/T/RtmpE9D3rB/devtools45ee119e7a62/crsh-papaj
\
##   --library='/Library/Frameworks/R.framework/Versions/3.5/Resources/library'  \
##   --install-tests
##
```

1   Abstract

This study further tested the hypothesised relationship between joint action, team click, and social bonding in a controlled experimental environment. An experiment was designed to test the prediction, partly confirmed in an *in situ* survey study (Chapter 8 HYPERLINK), that athletes who perceived higher quality team performance in joint action would also experience higher levels of team click and social bonding. 58 Professional Chinese rugby players (Men = 31) participated in a between-subjects design in which athletes' expectations for joint action training drill (**Passos2011**)) were manipulated in one of two conditions. Athletes in the "low difficulty" condition were primed with information to suggest that the training drill would require minimal rugby-sepcific individual and joint-action competence (i.e., 2/10 difficulty rating). By contrast, in the "high difficulty" condition, athletes were primed to expect the training drill to be relatively difficult (8/10 difficulty rating). Pre- and post-experiment surveys were conducted, measuring athletes' perceptions of main variables of interest (individual and group performance, team click, and social bonding (both to the training group and the team more generally)), as well as athlete personality type and states of athlete states of arousal, fatigue, and injury. Video footage of each experimental session was also recorded to derive an objective measure of interpersonal coordination not reliant on self-report. It was predicted that athletes in the "high-difficulty" condition would experience higher levels of team click and social bonding due to higher positive violation of expectations around group performance. Athletes in the "low difficulty" condition would on average experience less strongly the "click" of joint action and social bonding because they would not experience the same level of positive violations of expectations around group performance. Results provided some confirmation of these predictions.

2   Introduction

The previous chapter presented empirical evidence of a relationship between perceptions of joint action success and social bonding, fully mediated by the phenomenon of "team click." A relationship between violation of expectations surrounding team performance and feelings of team click was also observed, but a model of the relationship between team performance expectation vioaltion and social bonding did not reliably fit the data. The results of this naturalistic *in situ* study are noteworthy as they provide evidence for the prediction that perceptions of joint action influence psychological processes of affiliation and connection with teammates, possibly via the mediating construct of "team click." Controlled experimental research is needed in order to more directly access causal mechanisms that underpin the relationship between joint action and social bonding. If perceived joint action success and positive violation of expectation around team performance are the two candidate predictors of team click and social bonding, how is it possible to experimentally manipulate conditions conducive to maximising these two factors in joint action? In addition, how is it possible to isolate joint action from other potential sources of information regarding the team and its performance such as explicit feedback from coaches and other

players surrounding performance and joint action outcome? This experimental study was designed with these considerations in mind.

The Lit Review: 1. Re-cap literature (Synchrony, joint action, pink noise, team click) 2. Extend to expectation violation: 3.Priming literature: prime difficulty and uncertainty: - Challenge to skill ratio - Flow literature. By manipulating the level of uncertainty of information regarding the expectation

The Closing: These considerations lead to the hypothesis that expectations of higher levels of technical challenge in joint action will lead to higher levels of social bonding, due to the likelihood that athletes will perceive team performance more positively relative to prior expectations in the high-difficulty prime than in a low difficulty prime. As predicted in the previous study, it was predicted that the feeling of "team click" would mediate the relationship between positive violations of expectations and social bonding. The predictions are outlined as follows:

Prediction 1: More positive violations of expectations around team performance will correlate with higer feelings of team click with the training group

Prediction 1.a: Violations of expectations around team performance moderate the relationship between perceptions of joint action success and team click training group

Prediction 2: Feelings of group click will positively correlate with feelings of social bonding to the group

Prediction 3: More positive violations of team performance expectations will predict higher levels of Social Bonding to the training group

Prediction 4: Feelings of group click will mediate a relationship between more positive violations of expectations around group performance and social bonding to the group

A between-subjects experimental design was used, in which expectations of technical difficulty was manipulated in one of two conditions, "high difficulty" and "low difficulty" condition. Surveys measuring athletes' perceptions of joint action success relative to prior expectations, team click, and feelings of social bonding to the specific training group were recorded both before and after the experiment.

3 Method

3.1 Participants

This study was approved by the University of Oxford's Central University Research Ethics Committee (SAME/CUREC1A/15-059).

3.2 Materials

3.2.1 Experimental Paradigm: Invasion Drill

This study required athletes to participate in a common rugby training drill, which was preceeded by one of two experimental primes——a "high difficulty" condition, or a "low difficulty" condition. Replicating **Passos2011** a common training drill known as "invasion drill" (**Biscombe1998**) was selected as it was representative of a typical subphase of joint action in rugby union. In this drill, a group of four rugby players form an attacking group and two pairs of opponents form a first and second defensive line. There were two performance aims of this drill: (a) attackers were asked carry the ball into the try area, and (b) defenders were asked to stop the attacker's progression toward the try line. One trial consisted of one attempt by the attacking sub-group to penetrate two defending sub-groups, and carry the ball over the try line. The drill was conducted on a regulation multipurpose 110m x 70m grass or artificial turf training field within a 22m x 15m rectangle area marked by plastic cones (see Figure **??**). The ball used was size 5, as recommended by World Rugby for this age group of athletes.

3.2.2 Experimental Conditions: high and low difficulty

Two different pre-experiment primes were developed for the same training drill in order to manipulate athletes' expectations around the technical difficulty of the impending joint-action task. Athletes in the "high difficulty" condition learned that the overall average difficulty rating for the training drill, provided by World Rugby coaches and athletes, was 77.5/100, or approximately 8 out of 10. Athletes were told that the drill would require an extension of their abilities as individuals and as a training group (see appendix for full script). Athletes in the "low difficulty" condition, by contrast, learned that they would be participating in an ostensibly different drill with an overall average difficulty rating (provided by World Rugby coaches and athletes) of 22.5/100, or approximately 2 out of 10. In reality, the training drill was exactly the same for each experimental session, and the relative difficulty of this drill was estimated in a pilot study to be approximately 5/10, see APPENDIX). By either over- or under-rating the difficulty of the impending training drill, athletes' expectations around certainty of performance

15 m

7 m

Try line

15 m

Attack direction

5 m

1st line of defence

2nd line of defence
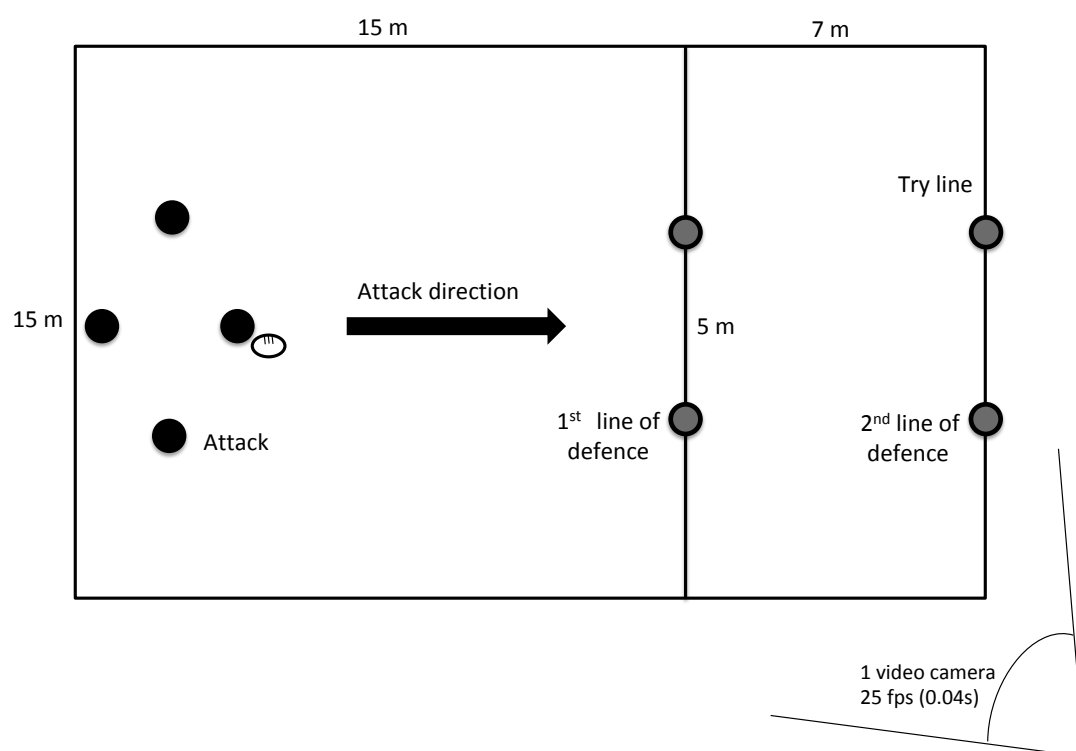
Attack

1 video camera
25 fps (0.04s)

Figure 1: The Invasion Drill, adapted from Passos (2011)

would be altered, such that athletes in the high difficulty condition would experience on average higher (positive) violation of expectations around team performance, whereas athletes in the low difficulty condition would on average perceive team performance less positively relative to prior of expectations. Surveys were designed and administered using Qualtrics software (Qualtrics version 9, Provo, UT). Survey data were processed and analysed using the R environment (R Development Core Team, 2006).

3.2.3 Measures

Surveys

Surveys were administered at three time points: 1) one day before the experiment (Baseline), 2) immediately before the experiment after receiving the experimental prime (Pre-Experiment), and 3) once immediately following the completion of the experiment (Post-Experiment)(see **??**). Consistent with the previous study (Chapter 8), athletes responded to questions about 1) perceptions of components and overall individual and team performance in joint action, 2) feelings associated with "team click" in joint action, and feelings of social bonding and group membership. In addition, athletes responded to survey items designed to measure possible moderators of the relationship between joint action and social bonding, namely: perceptions of arousal, fatigue, injury status, and personality type.

At Baseline, athletes were asked about their impressions of recent individual performance and the performance of their province team as a whole. Survey items included questions relating to specific components of on-field performance (e.g., team components: Attack, defence, onfield communication, and support play; individual components: Passing technique, one-on-one tackling, effectiveness in contact areas) as well as items relating to overall impressions of performance, for example: "How well do you feel your team has been performing in training and competition over the past month?" (100 point scale, 0 = "Extremely bad", 100 = "Extremely good"). Athletes also responded to questions about feelings of team click and social bonding with their team as a whole. In addition, the Baseline survey also included items concerning basic personal information, current injury status, subjective and objective measures of tehcnical competence, and personality type. For a full explanation of survey items measured at Baseline, see Chapter 6 (Introduction to Part B) and Appendix **??** Section **??**.

At time 2, immediately prior to the experiment (Pre-Experiment), athletes were asked questions about feelings and expectations concerning their individual performance and the performance of their specific training group in the impending training drill. For example, athletes were asked "How do you feel *right now* about your individual performance (100 point scale, 0 = "Extremely bad", 100 = "Extremely good")". Athletes were also asked a series of questions about feelings of team click and social bonding to their specific training group. For team click, for example, athletes were asked questions like: "How do you feel the tacit understanding is between the training group today?" (100 point scale, 0 = "Extremely bad", 100 = "Extremely good"). For social bonding, athletes were asked questions like: "How emotionally supportive does your training group feel right now?" (100 point scale, 0 = "Extremely weak", 100 = "Extremely strong"). For a full explanation of survey items measured at Pre-Experiment, see Appendix **??** Section **??**.

At time 3, immediately after the completion of the experiment (Post-Experiment), athletes completed questions concerning their impression of their own performance and the performance of their training group in the training drill, relative to their prior expectations. Athletes were also asked about their feelings of team click and social bonding to the training group as well as the provincial team as a whole (team-focussed questons mirrored the survey items that were asked one day earlier at Baseline). For a full explanation of Post-Experiment survey items, see Appendix **??** Section **??**.

Surveys were generated in English by the researcher using Qualtrics software (Qualtrics version 9, Provo, UT). Surveys were translated into modern Chinese and back translated by two independent native Chinese speaking translators from Beijing Sports University to verify accuracy. Athletes completed the modern Chinese version of each survey using WeChat on their personal mobile devices connected remotely to the internet.

istogram of tlc$age[tlc$condition == "l

tlc$age[tlc$condition == "high"]

listogram of tlc$age[tlc$condition == "
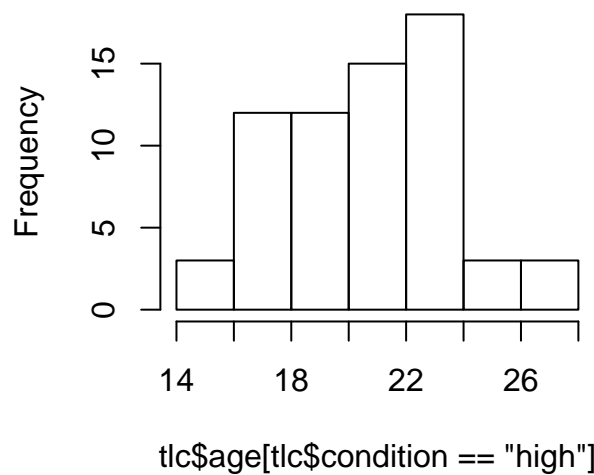
tlc$age[tlc$condition == "low"]
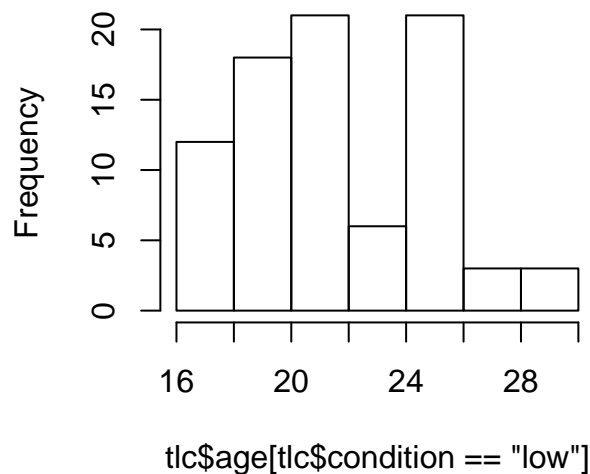
3.2.4 Video analysis of training group performance in training drill

In order to produce quantitative measures of interpersonal coordination between co-actors, athletes' motion was captured by a single digital video camera (Sony FDR-AX700 4K HDR Camcorder) mounted on a 1.2m high tripod. Digital video images of action were acquired by a computer, using a USB2.0 cable, and files were saved on an encrypted external hard disk in .AVI format. For image treatment, TACTO 8.0 software was used for digitizing at 25 frames per second.

Video footage from all 8 experiment sessions was analysed by an hypothesis blind research assistant. Each experimental trial of the Invasion Drill (16 in total) was coded according to the quality of performance relative to the primary (attacking) goal of the drill, i.e., scoring a try by carrying the ball over the try line. A maximum value of 6 was awarded to a "clear try" (trials in which a clear unobstructed try was scored by the attacking sub-unit); 5 to a "rough try" (the ball-carrier crossed the try-line following some minimal level of physical contact or obstruction from the defence that did not halt the momentum of the attacking phase); 4 to an "Obstructed try" (a trial in which a try is scored, but the momentum of the attacking sub-unit was clearly interrupted); 3 to a "no try - defence" (a trial in which a ball carrier is completely obstructed by a defender, and the trial is unsuccessful largely due to the effecticeness of the defence); 2 to a "no try - defence forced error" (unsuccessful trial due to a combination of the defence and error in attack); 1 to a "no try - unforced error" (a failed trial due to an unforced error in attack). For each experiment session, an average performance outcome ("Trial Outcome

Average") and the standard deviation of this outcome ("Trial Outcome SD") was calculated in order to capture the central tendency and variability of performance for each training group.

3.3   Procedure

Permission to run the study was sought from the head coach of each of the four teams (Beijing men's, Beijing women's, Shandong men's, Shandong women's). These coaches nominated athletes who were fit and able to complete the session without compromising their existing training schedules. Athletes were randomly assigned to one of two conditions, and then athletes in each experimental group were subsequently adjusted subsequently so that each condition was matched as much as possible according to average training age. Once athletes were assigned to an experimental group, they were then added to a WeChat group populated by other training group members and the researcher.

3.3.1   Cover Story

Athletes were notified (first via WeChat and then in a team meeting) that they were participating in a trial of a number of different rugby training drills selected from a recent report by World Rugby concerning training methods for rugby sevens. Athletes were told that the training drills had been previously rated by a selection of international level coaches and players from all over the world (including Asia and China). Athletes were informed that the purpose of the exercise was for me to assess the ratings provided by World Rugby by replicating these drills with more rugby athletes in China. Athletes were told that survey measures and video footage would be collected, which would be later analysed to assess individual performance of athletes in each training drill. It was also explained to athletes that there would be a second round of drills also requiring groups of eight athletes, but the makeup of these groups may reshuffle depending on athletes preferences. This detail allowed for the includion of a post-experiment bonding measure, in which athletes were asked to what extent they wished to continue to train with the same 8-athlete team in a subsequent round of drills.

Approximately 24 hours before the experiment session, athletes were instructed to complete the baseline survey by opening a link provided in the WeChat group. This survey included written consent for the study. Approximately 1 hour before the experiment was due to take place, athletes were administered with an experimental prime via WeChat. In the "high difficulty" condition, athletes were primed to believe that they were about to participate in a very difficult training drill (on the upper end of their individual and group technical abilities), whereas athletes in the "low difficulty" condition were primed to believe that they were about to participate in a relatively easy training drill well within their technical abilities (see APPENDIX for full script). All athletes were lead to believe that athletes in the other experiemntal groups were performing different drills to their assigned drill. In reality, the training drill for each condition was identical, the only thing that varied was the pre-experiment prime.

Once athletes were assembled at the designated training field, I verbally re-administered the same prime that had been sent to athletes via WeChat an hour earlier. In addition, athletes were told in more detail about the requirements of the Invasion Drill (see APPENDIX for full details). Specifically, athletes were told that their performance in the experiment would be assessed based on subsequent video analysis. I told athletes that they would be assessed based on their performance in attack, defense, and their ability to coordinate attack and defense with others. Athletes were provided with no other explicit information regarding performance goals, besides completing the experimental drill to the best of their ability within the rules of rugby.

Athletes were then administered with the time 2 survey (Pre-experiment), which took approximately 3-5 minutes to complete. After every athlete had finished the survey, athletes participated in a standard warm up routine lasting approximately 10 minutes, including slow jogging and dynamic stretching. Athletes were instructed not to use the rugby ball during this period, which served to reduce the amount of incidental coordination and interaction between participants prior to the start of the experiment trials. Once athletes had completed the warm up, the researcher assembled the group within the training drill area. One athlete was randomly assigned to stand at each of 8 available plastic markers (6 in the case of the modified drills). In this position, athletes were told once more about the structure and procedure for the training drill, in particular the way in which athletes were expected to rotate clockwise after every trial of the drill so that athletes did not habituate to certain positions or sub-units in the drill.

To begin the drill, the ball-carrier at the front and centre of the attack sub-group was instructed to tap the ball with his / her foot, before initiating the attacking sub-phase by advancing forward towards the defence sub-units. In the case that the ball was immediately fumbled during the initiation of the trial, the training group was instructed to restart that trial and the trial in which the mistake was made was not counted. Following a block of 4 practice trials, athletes were told by the researcher that the formal test was beginning. The group of athletes then completed 16 trials of the drill, which allowed each athlete to complete four trials of attack and four trials of defence, in different positions. Following completion of all 16 test trials, athletes were assembled by the researcher and thanked for their participation, before being sent to the sideline of the training field to complete the final post-experiment survey using their mobile devices. Following the completion of this final survey, athletes were

told that they would be informed within two days about the next experiment trial (in fact, no more experiments were taking place).

Video Analysis Procedure

The digital camera and tripod were positioned on a platform 2m above the level of the playing field, approximately 10-15m from the bottom try-line corner of the Invasion Drill perimeter (see figure **??**). The experimenter began video recording before the athletes arrived, and ceased recording after all athletes had left the training field following the experiment. Digital video images of action were acquired by a computer, using a USB2.0 cable, and files were saved on an encrypted external hard disk in .AVI format.

4 Results

A total of 52 athletes participated in 7 experimental sessions. In one session (Shandong men's high difficulty condition) a research assistant stood in as a dummy participant for an athlete could not participate due to injury. In the two sessions with Beijing women's team, due to failure of athletes to take part due to injury (3) or illness (2), the training drill was modified to a "3 on 2 + 1" drill, requiring only six athletes to complete instead of eight. In one of these experimental sessions, a female dummy participant was also required to fill in to make up a total of 6 athletes for the modified drill. In both cases, dummy participants were competent ex-athletes who were naive to the predictions of the study. Dummy participants did not participate in the survey responses before or after the session. Survey and video data of the remaining 58 participants was analysed.

4.1 Descriptive Statistics

4.1.1 Participants

```
## Warning in bind_rows_(x, .id):  Unequal factor levels:  coercing to character
## Warning in bind_rows_(x, .id):  binding character and factor vector, coercing into character
vector
## Warning in bind_rows_(x, .id):  binding character and factor vector, coercing into character
vector
## Error in library(table):  there is no package called 'table'
```

	Overall	high	low
n	52	23	29
Age (mean (sd))	21.70 (3.21)	21.14 (2.95)	22.14 (3.39)
Sex = male (%)	31 (59.6)	15 (65.2)	16 (55.2)
Team (%)			
Beijing Men	16 (30.8)	8 (34.8)	8 (27.6)
Beijing Women	5 (9.6)	0 (0.0)	5 (17.2)
Shandong Men	15 (28.8)	7 (30.4)	8 (27.6)
Shandong Women	16 (30.8)	8 (34.8)	8 (27.6)
Position = Starting Team (%)	16 (32.0)	7 (31.8)	9 (32.1)
TrainingAge (mean (sd))	4.38 (2.09)	4.00 (1.85)	4.68 (2.25)
YearsInTeam (mean (sd))	5.84 (2.00)	5.77 (1.77)	5.89 (2.20)
AthleteLevel (%)			
1st Level	11 (22.4)	5 (22.7)	6 (22.2)
2nd Level	4 (8.2)	3 (13.6)	1 (3.7)
Master	34 (69.4)	14 (63.6)	20 (74.1)
ContractStatus (%)			
Full time contract	7 (14.0)	2 (9.1)	5 (17.9)
Full time employee	21 (42.0)	10 (45.5)	11 (39.3)
Student contract	5 (10.0)	2 (9.1)	3 (10.7)
Training contract	15 (30.0)	7 (31.8)	8 (28.6)
Trial	2 (4.0)	1 (4.5)	1 (3.6)

Table 1: Overview of experiment sample, overall and by condition (low and high difficulty).

The attributes of the overall sample of athletes are displayed in Table **??**. The average training age of athletes was 4.38 years (2.0887454), with athletes having spent on average over 5.84 years in the team. 40% of the sample were full-time employees of their provincial team, with the rest either being employed on a full-time (but fixed

term) contract (13.46%)), a "student contract" ((28.85%))), a short term training contract ((9.62%))), or on a short-term trial basis ((3.85%))). 16 of 52 (31%) of athletes declared that there were in the starting team of their respective provincial teams. Attributes of athletes in the high and low difficulty conditions were evenly matched (see Table **??**), with only a marginally significant difference in average age between conditions (high difficulty condition = 21.1363636(2.9486955), low difficulty condition = 22.1428571(3.3853091).

4.1.2 Survey Responses

Athlete responses to survey items were collated according to main variables of interest, particularly perceptions of individual and team performance in joint action, social variables relating to team click, social bonding, and moderator variables related to technical competence, arousal, exertion, and personality. In general, the central tendency of all survey items was between .5 to 1.5 standard deviations above the mid-point of the scale, with almost all survey item distributions exhibiting low to moderate negative skew. Average athlete perceptions of individual (Appendix Tables **??** and **??**), group (Appendix Tables **??** and **??**), and team (measured at Baseline only, see Appendix Table **??**) performance ranged from a low of 58.31(17.71) for perceptions of individual performance relative to prior expectations (measured post-Experiment) to a high of 78.62 (12.75) for athlete confidence in group ability to meet the technical challenges of the drill, measured pre-Experiment. Athletes were on average more critical in regards to perceptions of individual performance than they were of group and team performance, and were generally more critical of individual and group performance in the post-Experiment survey than they were in the Baseline and pre-Experiment surveys (see Appendix Tables **??** **??**).

Central tendencies of variables measuring athletes' perceptions of group click and social bonding were also well above the mid-point of their respective scales, ranging from a low of 68.34 (SD = 14.69) for feelings of unspoken understanding measured post-Experiment in the low difficulty condition, to a high of 90.21 (SD = 9.33) for feelings of shared goal with the training group measured pre-Experiemnt in the low difficulty condition (see Appendix Tables **??**–**??**). Measures relating to Team Click and Social bonding also appeared to decrease in the post-Experiment survey relative to the pre-Experiment and Baseline surveys. Objective measures of training group performance derived from video footage appeared to be consistent across experiment conditions (see Appendix Table **??**). For a description of the remaining moderator variables, see Appendix section **??**.

```
##
## Attaching package:  'scales'
## The following object is masked from 'package:purrr':
##
##     discard
## The following object is masked from 'package:readr':
##
##     col_factor
## Warning in ModuleReturnVarsExist(vars, data):  The data frame does not have:  GroupPerformVsExpected
Dropped
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
```

```
## Warning in ModuleReturnVarsExist(vars, data):  The data frame does not have:  GroupPerformVsExpected
Dropped
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.


## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.


## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
```

```
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.


## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
```
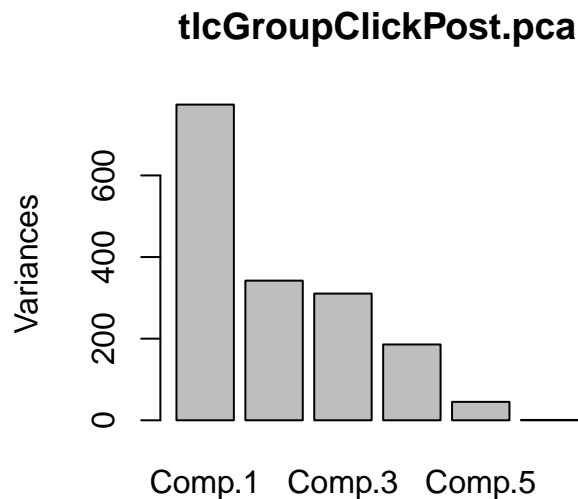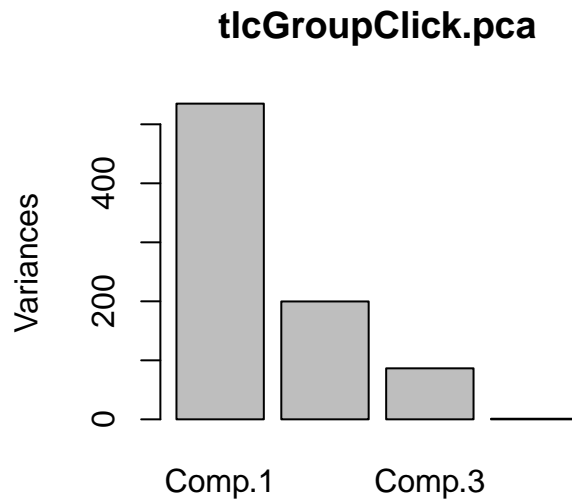
```
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
```

```
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
```

```
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
```

```
## Error in .f(.x[[i]], ...):  object 'fullOutcomeAvg' not found
## Warning in ModuleReturnVarsExist(vars, data):  The data frame does not have:  TrialOutcomeAvg
Dropped
## Error in ModuleStopIfNoVarsLeft(vars):  No valid variables.
```

```
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
```

```
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
```

```
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
```

```
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in min(x, na.rm = TRUE): no non-missing arguments to min; returning Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in max(x, na.rm = TRUE): no non-missing arguments to max; returning -Inf
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
## Warning in StdDiff(variable = var, group = strataVar):  Variable has only NA's in at least one
stratum.  na.rm turned off.
```

4.2 Data Reduction

Exploratory factor analysis was used to reduce multicolinearity between variables while retaining as much variance as possible in the observed data (**Yong2013**). Data reduction was performed on the survey items related to the main outcome variables of interest, namely team click and social bonding directed at the training group. The procedure for data reduction followed the method outlined in the previous empirical chapter (Chapter 8). Moderator variables—technical competence (objective and subjective), components of individual and training group performance, arousal, exertion, and athlete perception of team discipline—were also reduced to factors (see APPENDIX **??** HYPERLINK for full description of method). In addition, items relating to team click and social bonding directed at the provincial team as a whole (and not just the specific training group) were reduced to factors, in order to assess pre- to post-experiment variation in generalised bonding to the team.

4.2.1 Group Performance relative to prior expectations

The main predictor variable of interest, perceptions of training group performance relative to prior expectations, was a single item measure, and did not require data reduction. Considering that the outcome variables of interest (factors) were transformed to standardized z-scores (mean = 0, SD = 1),perceptions of team performance relative to prior expectation was also standardised (mean = 0, SD = 1) for the purposes of equivalence of predictor and outcome variables in subsequent linear mixed effects modelling.

4.2.2 Feelings of team click within the training group

tlcGroupClick.pca

Variances

```
   Comp.1        Comp.3
```

tlcGroupClickPost.pca

Variances

```
 Comp.1   Comp.3   Comp.5
```

```
## Loading required package:  Hmisc
## Loading required package:  lattice
## Loading required package:  survival
## Loading required package:  Formula
##
## Attaching package:  'Hmisc'
## The following objects are masked from 'package:xtable':
##
##     label, label<-
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

Items relating to training group team click (unspoken understanding, general atmosphere, click pictorial, and ability extended by the group) were subjected to EFA. Correlations between the remaining were very high (all r's ¿ 0.2368954), which suggested that one factor was appropriate (see Table ee Table **??**). The KMO index and

Bartlett's test both suggested high sampling adequacy, (KMO $= 0.6763438$, $\chi^2(6, N = 104) = 104.2571531, p = 3.2406122 \times 10^{-20}$). $One factor, labelled "TeamClick" was imposed on the data, which explained 43.2\% of the overall variance (SSL 2). Guttmans \lambda = 0.6916979 and Cronbachs \alpha = 0.7232482 indicated that the data reduction was appropriate and reliable.

In addition to the four variables common to both pre- and post-experiment surveys, athletes were also asked in the post-experiment survey about their own reliability and the reliability of other athletes to perform their role on the field specifically. These two variables, "Reliability For Others" and "Reliability Of Others" were included in an EFA involving only the post-experiment measures. The variable "reliability for others" was removed from the matrix because it did not significantly correlate with other variables (all r's ¡= .16).

4.2.3 Feelings of Social Bonding to the training group

tlcGroupBonding.pca
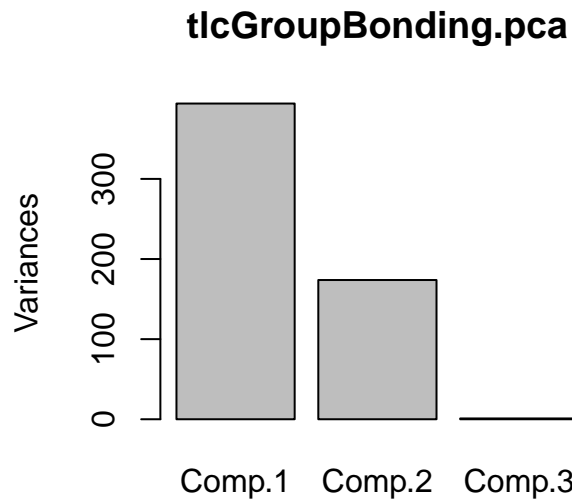
Items concerning bonding to the training group (emotional support, shared goal, fusion pictorial) were subjected to EFA. Correlations between items were high (all r's ¿ 0.3248347), which suggested that one factor would be appropriate (see Table ee Table **??**). The KMO index and Bartlett's test both suggested high sampling adequacy, (KMO $= 0.6401899$, $\chi^2(3, N = 104) = 35.4371015, p = 9.8486025 \times 10^{-8}$). $One factor, labelled "GroupSocialBonding" was imposed on th 1.02). Guttmans \lambda = 0.5074998 and Cronbachs \alpha = 0.6065765 indicated that the data reduction was appropriate and reliable.

4.2.4 Additional data reduction

In addition to athlete expectation violations concerning group and individual performance (predictor variable), group click (moderator and outcome variable), and group bonding (outcome variable), measures relating to athlete perceptions of components of group and individual performance, social bonding to the team as a whole, athlete arousal, technical competence (subjective and objective measures), and athlete perceptions of team discipline were subjected to data reduction (see Appendix Section **??** for full details).

tlcGroupPerformance.pca
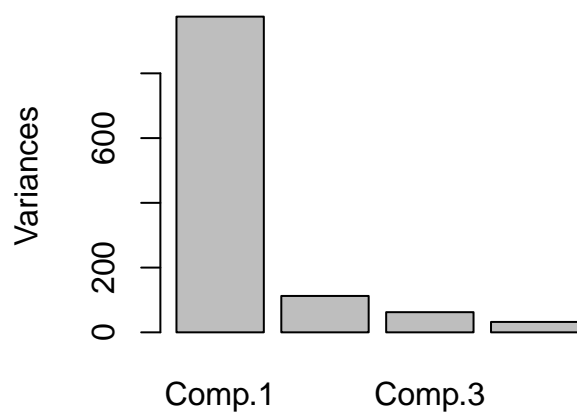
tlcIndPerformance.pca
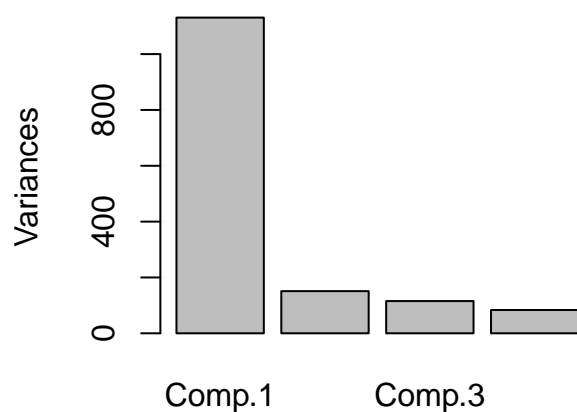

```
## 
## Attaching package:  'psych'
## The following object is masked from 'package:Hmisc':
## 
##     describe
## The following objects are masked from 'package:scales':
## 
##     alpha, rescale
## The following object is masked from 'package:car':
## 
##     logit
## The following object is masked from 'package:mediation':
## 
##     mediate
## The following objects are masked from 'package:ggplot2':
## 
##     %+%, alpha
```

4.3 Manipulation Checks

The effectiveness of the experimental manipulation was assesed based on a comparison of athletes responses to three survey items immediately following the high or low difficulty prime and immediately before the training drill. Athletes were asked about their confidence in their individual and training group's ability to handle the technical challenges of the training drill. Due to the fact that athletes had been primed to expect either a difficult or simple training drill, it was expected that athletes in the high difficulty condition would be less confident about their own and their group's ability to meet the technical challenges of the training session. In addition to self-reported confidence, it was also predicted that athlete self-reports of arousal, measured measured at the same time as the previous questions regarding confidence, would also vary according to condition. It was predicted that that athletes in the high difficulty condition would be more aroused than athtletes in the low difficulty condition. Meaningful variation in these variables could be used to check that the experimental manipulation was successful.
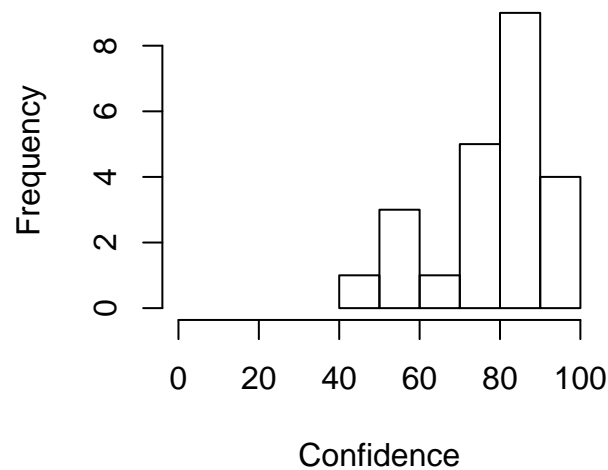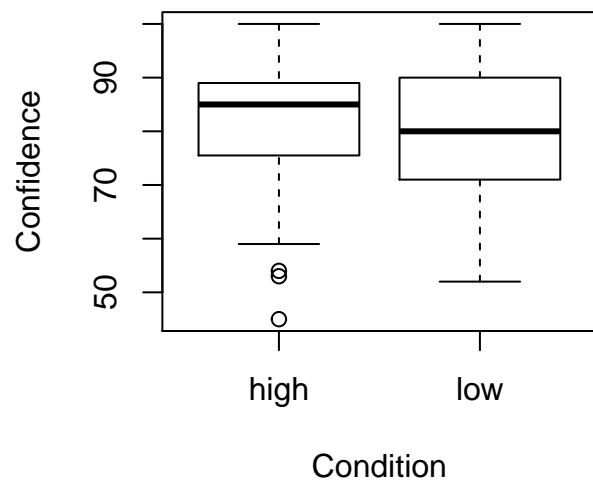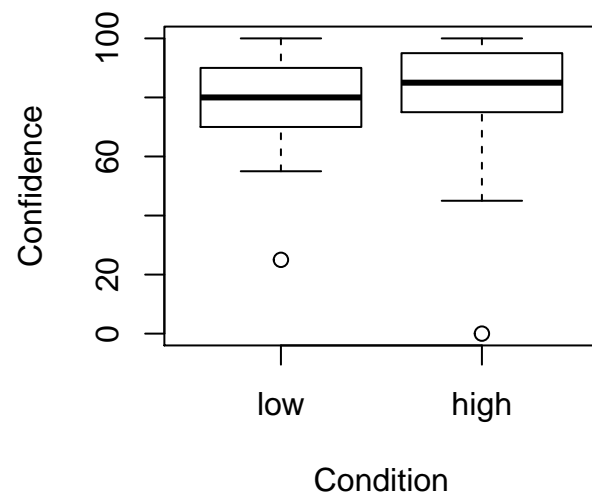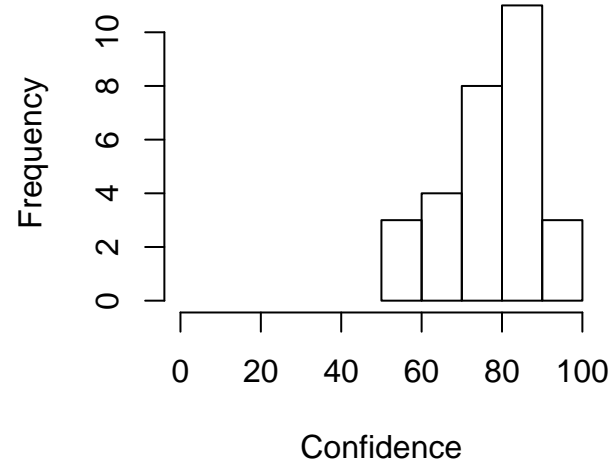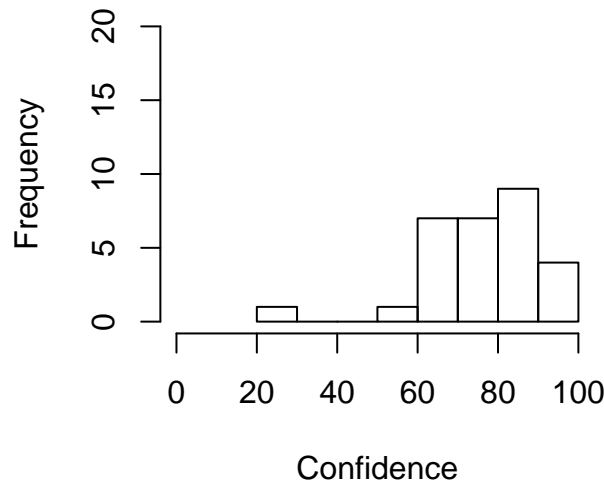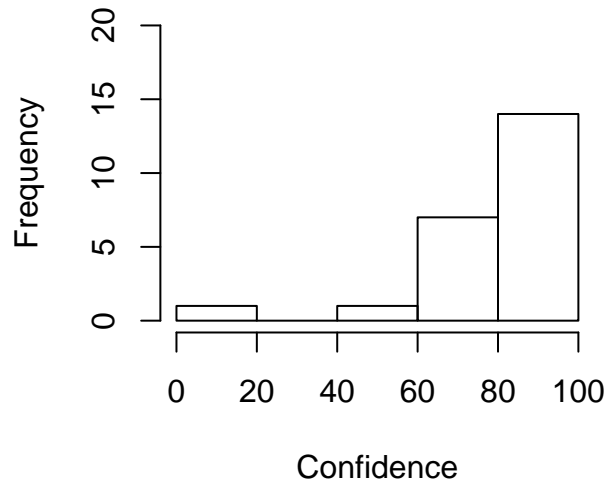
Results of manipulation checks indicate that there were no overall condition-wise differences in athlete confidence in group or individual ability to meet the technical challenges of the drill (see Table **??** and Figures **??** and **??**). Although the central tendency of these variables was not significantly different, the box plots (Figures **??**) and histograms of both group and individual confidence variables by condition show that the variance of responses in the high difficulty condition appears to be larger than the low difficulty condition (see Figures **?? ??**). A Bartlett's test of homogeneity of variance revealed that there was no meaningful difference in variance of confidence in group ability (c('Bartlett's K-squared' = 0.693804937289204), c(df = 1), 0.404873187953028, $groupConfidentTechnicalChallenges_2bycondition, Bartletttestofhomogeneityofvariances).Thesametestappliedtoathleteconf$ $squared' = 3.75907581131734), c(df = 1), 0.0525216007345362, indConfidenceChallenges_2bycondition, Bartletttestofhomoge$

```
arousalFactorPreBoxPlot <- boxplot(arousalFactorPre ~ condition,
                                   varwidth = TRUE,
                                   width = c(.5,.5),
                                   notch = F,
                                   main = NA,
                                   xlab = "Condition",
                                   ylab = "Arousal",
                                   data = twc)
```
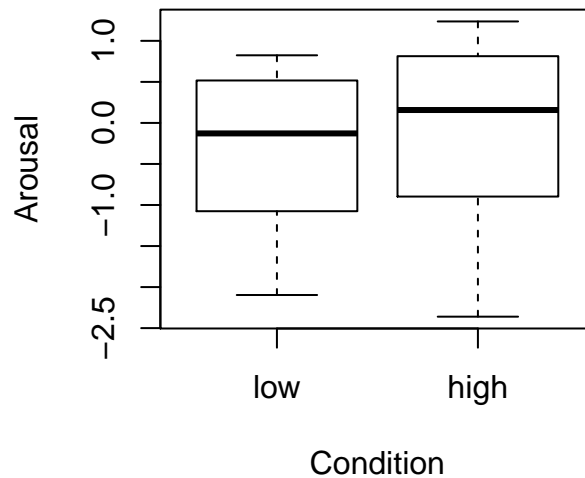
```
histArousalFactorPreHigh <- hist(high$arousalFactorPre,
                                 main = NA,
                                 xlab = "Arousal",
                                 xlim = c(-3, 2),
                                 ylim = c(0,10))
```

```
hist(low$arousalFactorPre,
                          main = NA,
                          xlab = "Arousal",
                          xlim = c(-3, 2),
                          ylim = c(0,10))
```

An analysis of arousal by condition also failed to reveal meaningful variation by condition, (see Table **??** and Figure **??**). Judging by the boxplot and histogram of each condition (see Figures **??** and **??**), the variance of arousal in the high difficulty condition did appear to be higher than the low difficulty condition. A Bartlett's test for homogeneity of variance, however, revealed that the difference was not significant, c('Bartlett's K-squared' = 1.03553340114123), c(df = 1), 0.308862564823246, arousalFactorPre by condition, Bartlett test of homogeneity of variances. These results suggested that the experimental prime was explicitly successful in significantly manipulating athlete self-reported confidence in individuala and group ability to meet the technical challenges associated with the training drill. There was also no evidence that self-reported arousal differed significantly according to condition. It was still possible, however, that the experimental primes influenced athletes implicitly, in ways that were not measured by surveys, which record explicit self-report.

4.4 Results by condition

Predictions of this study were first tested according to condition. It was predicted that the high difficulty prime would generate closer attention to the details of joint action between participants, and more positive violation of expectations around group performance.

```
groupJointActionSuccessBoxPlot <- boxplot(jointActionSuccessPost ~ condition,
                                          varwidth = TRUE,
                                          width = c(.5,.5),
                                          notch = F,
                                          main = NA,
                                          xlab = "Condition",
                                          ylab = "Joint Action Success",
                                          data = twc)
```
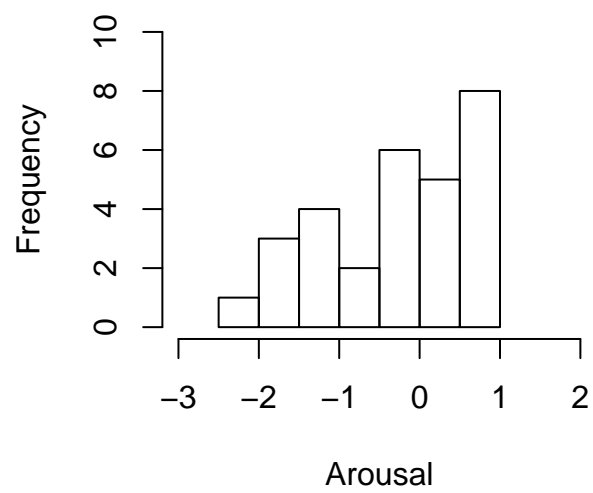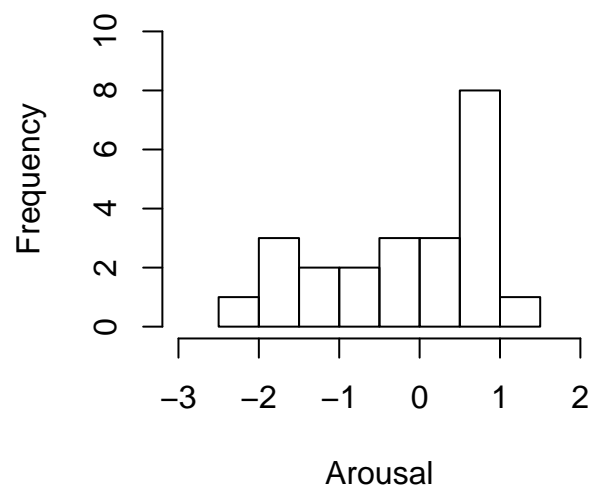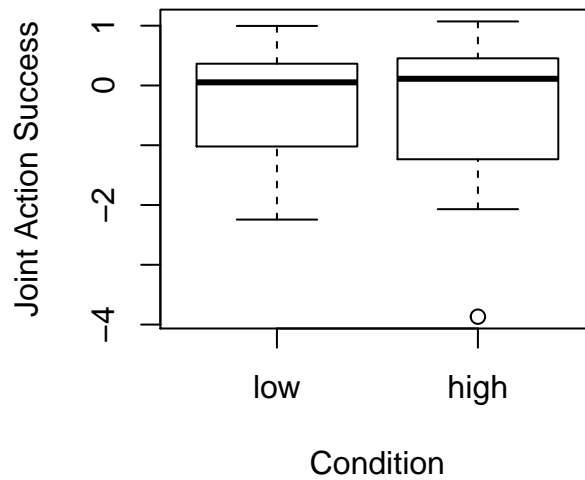
```
indComponentPerformancePostBoxPlot <- boxplot(indPerformancePost ~ condition,
                                   varwidth = TRUE,
                                   width = c(.5,.5),
                                   notch = F,
                                   main = NA,
                                   xlab = "Condition",
                                   ylab = "Individual Performance",
                                   data = twc)
```

```
groupPerfExpPostBoxPlot <- boxplot(groupPerformance_3Norm ~ condition,
                                   varwidth = TRUE,
                                   width = c(.5,.5),
                                   notch = F,
                                   xlab = "Condition",
                                   ylab = "Group Performance vs. Expectations",
                                   data = twc)
```

```
indPerfExpPostBoxPlot <- boxplot(indPerformance_3Norm ~ condition,
                                   varwidth = TRUE,
                                   width = c(.5,.5),
                                   notch = F,
                                   main = NA,
                                   xlab = "Condition",
                                   ylab = "Individual Performance vs. Expectations",
                                   data = twc)
```

```
# t.test(high£indPerformance_3, low£indPerformance_3)
```

As Table **??** shows, there were no observable significant differences between conditions in athlete perceptions of joint action success, individual component performance, group performance relative to prior expectations, or individual performance relative to prior expectations (see also box plots comparing these factors by condition, Figures **?? ??**). These results hold when controlling for an athlete's confidence in individual and group performance reported at time 2 prior to the experiment (see APPENDIX for detailed results).
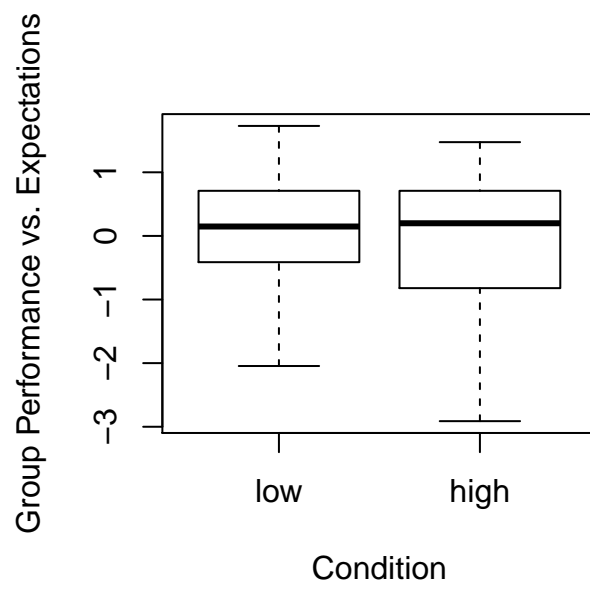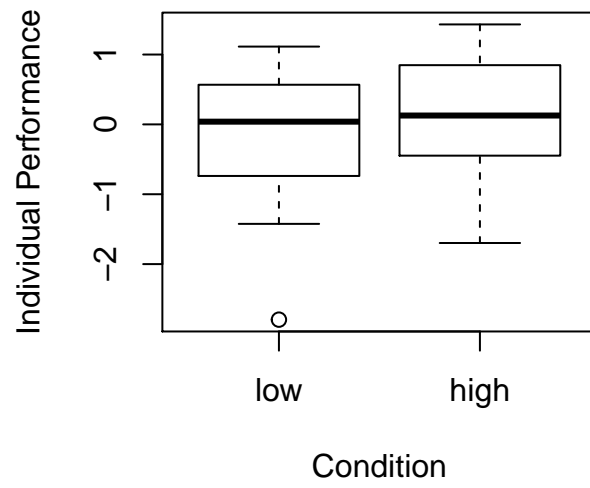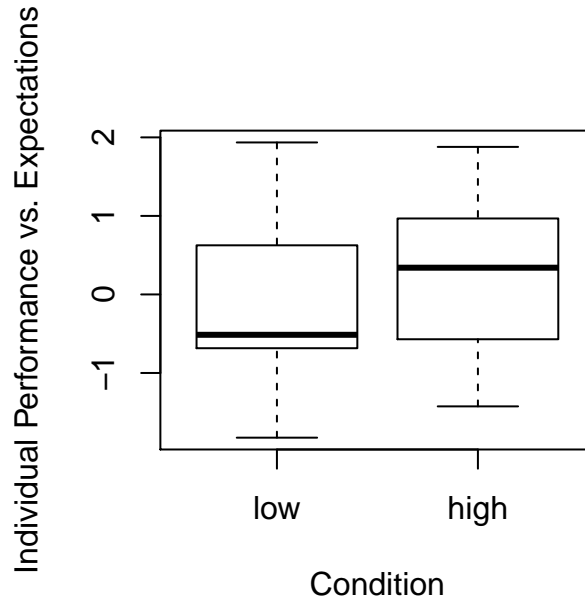
```
## Error:  <text>:33:82:  unexpected ')'
## 32:  #tlc$indPerformanceExpectations[tlc$time == 3] <- scales::rescale(tlc$indPerformanceExpectations
== 3],
## 33:                                                          to = c(0, 100))
##                                                                                  ^
```

There were also no observable differences in group click, group bonding, or bonding to team following the experiment (see Tables **??** **??**). These results suggest that there was no observable variaiton in athlete self-report according to condition. Given these results, the entire experiment sample was collapsed into one group and study predictions were tested in this group as a whole.

```
## Error in is.data.frame(data):  object 'tlcFactors' not found
## Error in print(tableFactors, quote = FALSE, noSpaces = FALSE, printToggle = FALSE): object 'tableFact
not found
## Error in xtable(table_overall, type = "latex", caption = "Variables of interest over time \n
for entire sample", :  object 'table_overall' not found
```

	Baseline	Pre	Post
n	58	58	58
JointActionSuccess (mean (sd))	NaN (NA)	0.27 (0.86)	-0.28 (1.01)
IndividualComponentPerformance (mean (sd))	-0.05 (0.98)	0.10 (0.93)	-0.06 (0.97)
GroupPerformVsExpected (mean (sd))	NaN (NA)	NaN (NA)	63.37 (21.23)
IndividualPerformanceExpectations (mean (sd))	NaN (NA)	NaN (NA)	61.49 (17.44)
GroupClick (mean (sd))	NaN (NA)	0.19 (0.85)	-0.19 (0.97)
GroupBonding (mean (sd))	NaN (NA)	0.12 (0.81)	-0.12 (0.85)
GroupClickPost (mean (sd))	NaN (NA)	NaN (NA)	-0.00 (0.93)
TeamBondingFactor (mean (sd))	0.02 (0.93)	NaN (NA)	-0.02 (0.84)
Arousal (mean (sd))	NaN (NA)	-0.23 (0.94)	0.23 (0.92)

Table 2: Variables of interest over time for entire sample

Table **??** shows that the post-Experiment measures of Joint Action Success and Team Click were significantly lower than pre-Experiment measurements. Group Bonding also appeared to reduce, between pre- and post-Experiment measures, but this change was not significant. Athlete reports of arousal significantly increased following the experiment, as would be expected due to the moderate to high levels of physiological exertion associated with the training drill. Athlete perceptions of components of individual performance and bonding to the team as a whole did not vary significantly acrross time points.

4.5 Analysis of Study Predictions

4.5.1 Post-Experiment Survey Data

Study predictions were tested using the post-Experiment survey data. Intra-class correlation estimates were calculated in order to account for the possibility of clustering of model residuals according to groupings of athlete sex, team, experiment location and specific experiment session (see Table **??**). ICC values indicated that between-group variance was low to moderate, with highest values for feelings of group click and social bonding between teams and experimental sessions. Given the constraints on model complexity imposed by the relatively small sample size of the study, experimental session was chosen as a level 2 grouping variable. High variation in the mean and standard deviation of objective performance outcome (see Figure **??**) further supported the decision to account for variation attributable to experiment session. Consistent with the previous study, a linear mixed effects regression (LMER) model (package "lme4" in the R environment) was used to test relationships between variables of interest (see Ch5 Section X).

```
## Warning in ICCest(sessionName, groupConfidentTechnicalChallenges_2, data = twc, :  Missing levels
of 'x' have been removed
## Warning in ICCest(sessionName, indConfidenceChallenges_2, data = twc, CI.type = "THD"):  Missing
levels of 'x' have been removed
## Warning in ICCest(sessionName, arousalFactorPre, data = twc, CI.type = "THD"):  Missing levels
of 'x' have been removed
## Warning in ICCest(sessionName, groupPerformance_3Norm, data = twc, CI.type = "THD"):
## Warning in ICCest(sessionName, groupPerformance_3Norm, data = twc, CI.type = "THD"):  Missing
levels of 'x' have been removed
## Warning in ICCest(sessionName, groupClickPostFactor, data = twc, CI.type = "THD"):
## Warning in ICCest(sessionName, groupClickPostFactor, data = twc, CI.type = "THD"):  Missing levels
of 'x' have been removed
## Warning in ICCest(sessionName, groupBondingFactorPost, data = twc, CI.type = "THD"):
## Warning in ICCest(sessionName, groupBondingFactorPost, data = twc, CI.type = "THD"):  Missing
levels of 'x' have been removed
## Warning in ICCest(sessionName, fullOutcomeAvg, data = twc, CI.type = "THD"):  Missing levels
of 'x' have been removed
## Warning in anova.lm(aov(y ~ x, data = tdata)):  ANOVA F-tests on an essentially perfect fit are
unreliable
```

```
## Warning in ICCest(sex, groupPerformance_3Norm, data = twc, CI.type = "THD"):
## Warning in ICCest(sex, groupClickPostFactor, data = twc, CI.type = "THD"):
## Warning in ICCest(sex, groupBondingFactorPost, data = twc, CI.type = "THD"):
```

```
## Warning in ICCest(team, groupPerformance_3Norm, data = twc, CI.type = "THD"):
## Warning in ICCest(team, groupClickPostFactor, data = twc, CI.type = "THD"):
## Warning in ICCest(team, groupBondingFactorPost, data = twc, CI.type = "THD"):
```

```
## Warning in ICCest(sessionLocation, groupPerformance_3Norm, data = twc, CI.type = "THD"):
## Warning in ICCest(sessionLocation, groupClickPostFactor, data = twc, CI.type = "THD"):
## Warning in ICCest(sessionLocation, groupBondingFactorPost, data = twc, CI.type = "THD"):
```

```
## Error:  <text>:46:53: unexpected ','
## 45:                                 + tipiExtraverted
## 46:                                 + ( 1|sessionName),
##                                                      ^
```

4.5.2 Analysis of pre- and post-Experiment survey data

PRE-POST ANALYSIS: 1. pre-post changes: i. performance expectations? ii. click? iii. bonding?

```
## Saving 3.5 x 3.5 in image
## Warning:  Removed 53 rows containing non-finite values (stat_boxplot).
## Warning:  Removed 53 rows containing non-finite values (stat_boxplot).
```

```
## Saving 3.5 x 3.5 in image
## Warning:  Removed 53 rows containing non-finite values (stat_boxplot).
## Warning:  Removed 53 rows containing non-finite values (stat_boxplot).
```
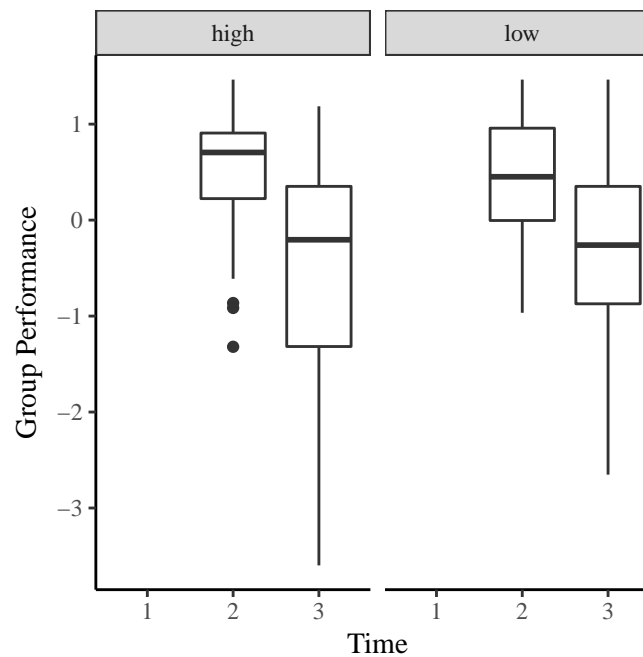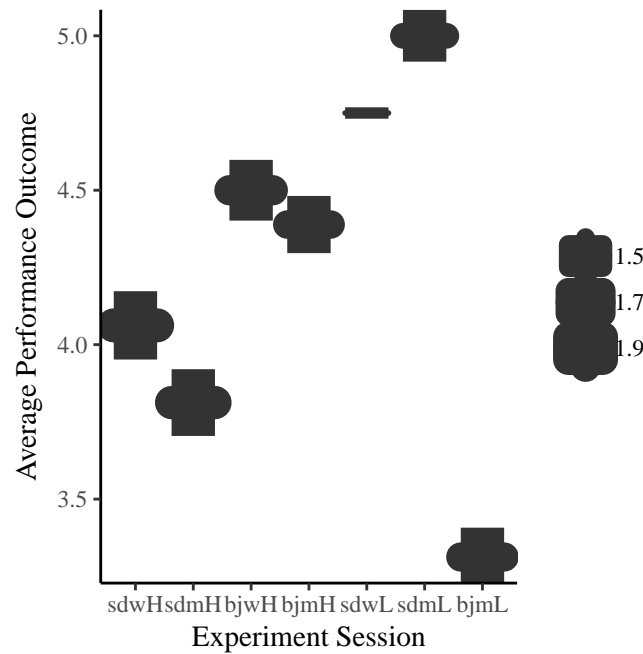
```
## Saving 3.5 x 3.5 in image
## Warning:  Removed 53 rows containing non-finite values (stat_boxplot).
## Warning:  Removed 53 rows containing non-finite values (stat_boxplot).
```

```
## Error in '$<-.data.frame'('*tmp*', groupPerfChangePrePost, value = integer(0)):  replacement
has 0 rows, data has 156
## Error in hist.default(tlc$groupPerfChangePrePost[tlc$time == 3], main = NA, :  'x' must be numeric
## Error in hist.default(tlc$indPerfChangePrePost[tlc$time == 3], main = NA, :  invalid number of
'breaks'
```
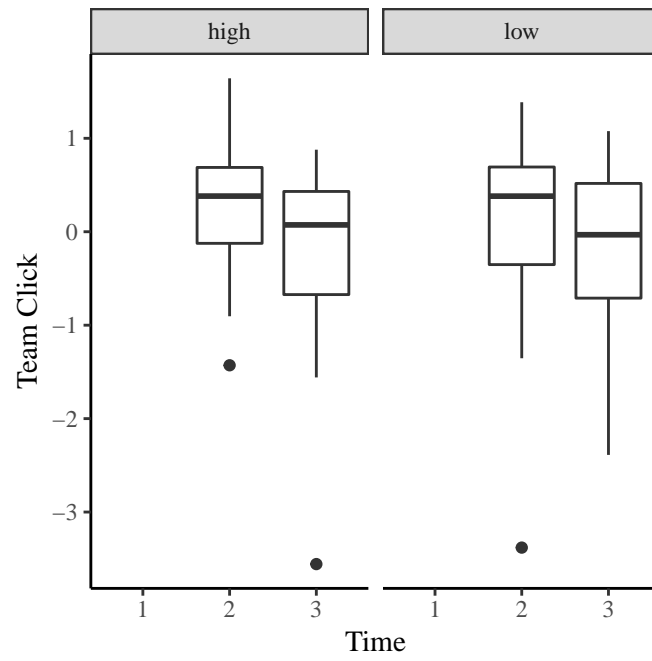
Figure 2: Pre- and post-Experiment measures of group click
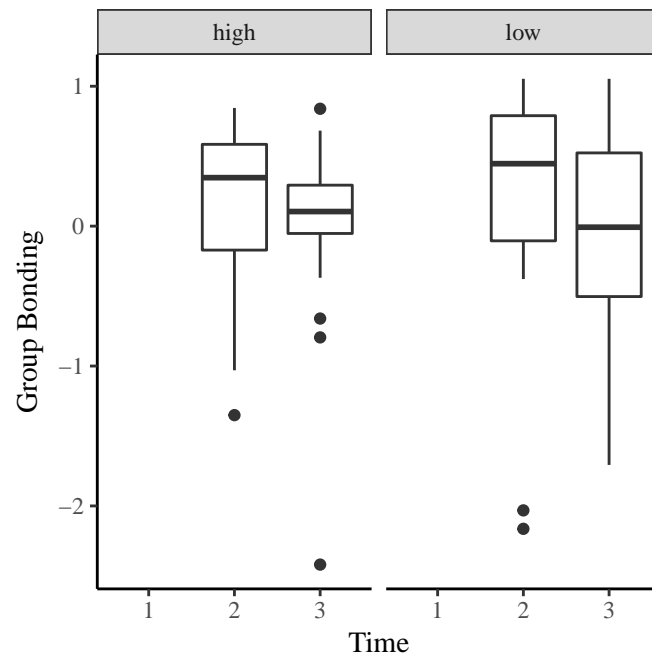
Figure 3: Pre- and post-Experiment measures of group Bonding

Change in team bonding
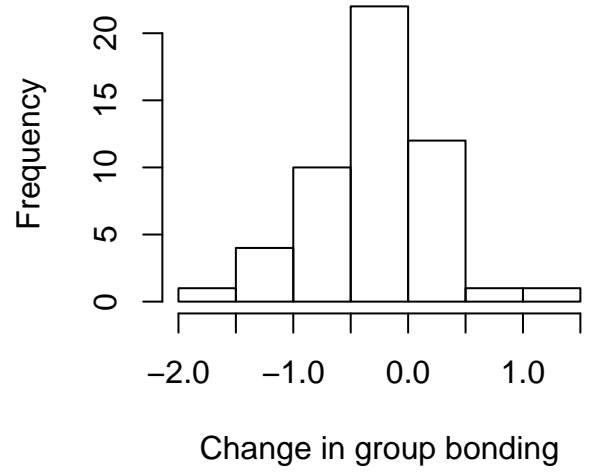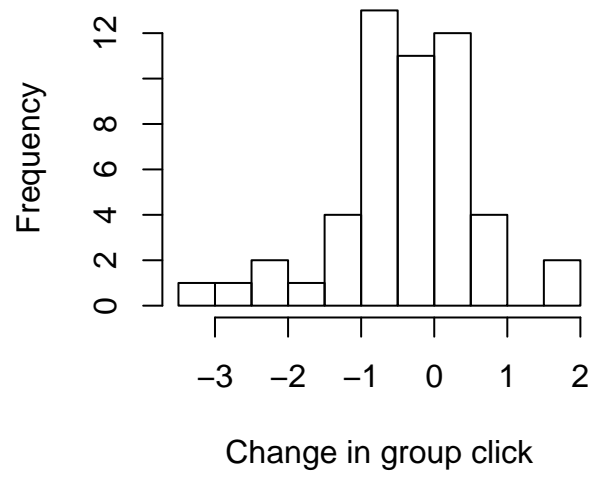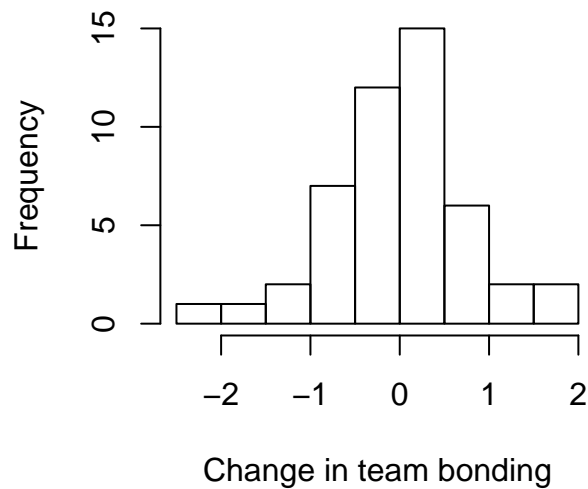
Error in xy.coords(x, y, xlabel, ylabel, log): 'x' and 'y' lengths differ
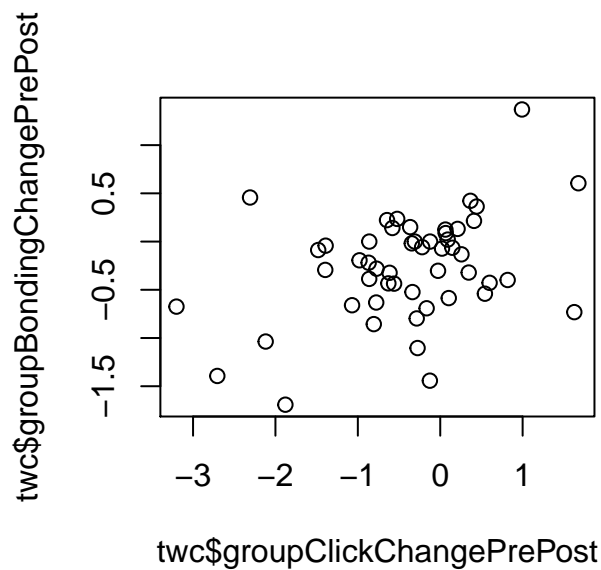
Error in xy.coords(x, y, xlabel, ylabel, log): 'x' and 'y' lengths differ

Error in array(x, c(length(x), 1L), if (!is.null(names(x))) list(names(x), : 'data' must be
of a vector type, was 'NULL'

Error in eval(icall$y, data, parent.frame()): object 'groupPerfChangePrePostNorm' not found
Warning in ICCest(sessionName, groupClickChangePrePostNorm, data = twc, :
Warning in ICCest(sessionName, groupClickChangePrePostNorm, data = twc, : Missing levels of
'x' have been removed
Warning in ICCest(sessionName, groupBondingChangePrePostNorm, data = twc, :
Warning in ICCest(sessionName, groupBondingChangePrePostNorm, data = twc, : Missing levels of
'x' have been removed
Warning in ICCest(sessionName, teamBondingChangePrePostNorm, data = twc, :

```
## Warning in ICCest(sessionName, teamBondingChangePrePostNorm, data = twc, :  Missing levels of
'x' have been removed
```

```
## Error in eval(icall$y, data, parent.frame()):  object 'groupPerfChangePrePostNorm' not found
## Warning in ICCest(sex, groupClickChangePrePostNorm, data = twc, CI.type = "THD"):
## Warning in ICCest(sex, groupBondingChangePrePostNorm, data = twc, CI.type = "THD"):
## Warning in ICCest(sex, teamBondingChangePrePostNorm, data = twc, CI.type = "THD"):
```

```
## Error in eval(icall$y, data, parent.frame()):  object 'groupPerfChangePrePostNorm' not found
## Warning in ICCest(team, groupClickChangePrePostNorm, data = twc, CI.type = "THD"):
## Warning in ICCest(team, groupBondingChangePrePostNorm, data = twc, CI.type = "THD"):
## Warning in ICCest(team, teamBondingChangePrePostNorm, data = twc, CI.type = "THD"):
```

```
## Error in eval(icall$y, data, parent.frame()):  object 'groupPerfChangePrePostNorm' not found
## Warning in ICCest(sessionLocation, groupClickChangePrePostNorm, data = twc, :
## Warning in ICCest(sessionLocation, groupBondingChangePrePostNorm, data = twc, :
## Warning in ICCest(sessionLocation, teamBondingChangePrePostNorm, data = twc, :
```

```
## Error in eval(expr, envir, enclos):  object 'performanceSessionPrePostICC' not found
## Error in eval(expr, envir, enclos):  object 'performanceSexPrePostICC' not found
## Error in eval(expr, envir, enclos):  object 'performanceTeamPrePostICC' not found
## Error in eval(expr, envir, enclos):  object 'performanceLocationPrePostICC' not found
## Error in data.frame(variable, session, sex, team, location):  arguments imply differing number
of rows:  4, 3
## Error in eval(expr, envir, enclos):  object 'ICCTablePrePost' not found
## Error in xtable(ICCTablePrePost, caption = "Pre-Post Tournament intra class correlations by group",
:  object 'ICCTablePrePost' not found
```

```
## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in confint.merMod(conditionGroupPerfPrePost, method = "Wald", level = 0.95):  object 'condition
not found
## Error:  Invalid grouping factor specification, sessionName
```

**Prediction 1: More positive perceptions of team performance relative to prior expectations will
correlate with higer feelings of team click with the training group**

```
library(tidyverse)
  groupPerfExpClickScatter <- ggplot(twc, aes(x=groupPerformance_3Norm, y=groupClickPostFactor,
                        color=condition)) + geom_point(shape = 1) +
                        geom_smooth(method=lm, se=FALSE)   # Don't add shaded confidence region
  groupPerfExpClickScatter +
            xlab("Group Performance vs Expectations") +
            ylab("Team Click") +
            apatheme

## Warning:  Removed 1 rows containing non-finite values (stat_smooth).
## Warning:  Removed 1 rows containing missing values (geom_point).
```

Figure **??** shows a strong positive correlation between violations of prior expectations concerning team per-
formance and feelings of "click" with the training group post-Experiment. The slope of this relationship appears
to vary according to condition, as indicated on the scatterplot by the two non-parallel lines of best fit.

```
## Error in confint.merMod(groupPerfExpClick.controls, method = "Wald", level = 0.95):  object 'groupPer
not found
```

Distribution of residuals
(standardised)
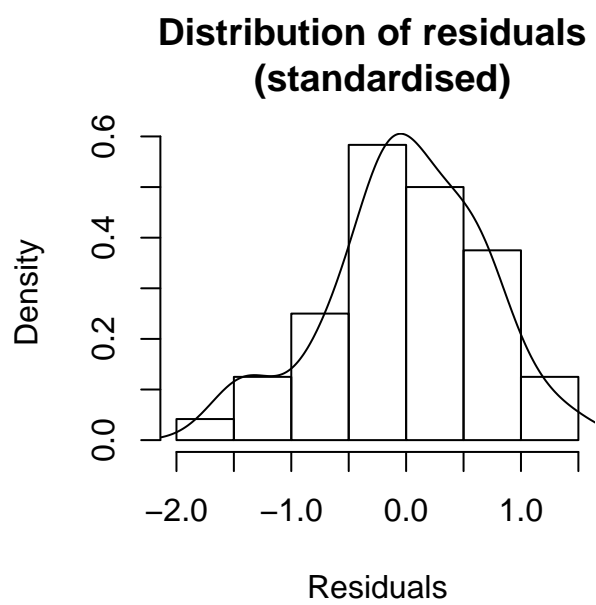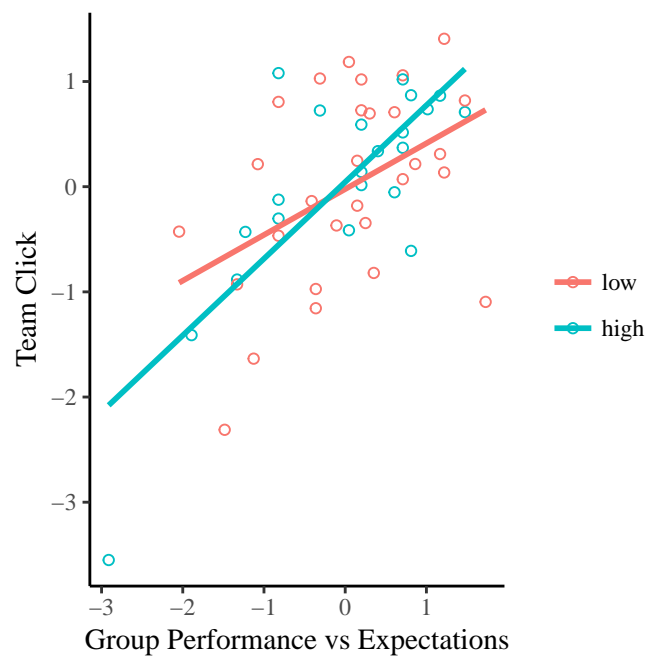
Scatterplot of Residuals (standardised)

Norm QQ Plot of Residuals
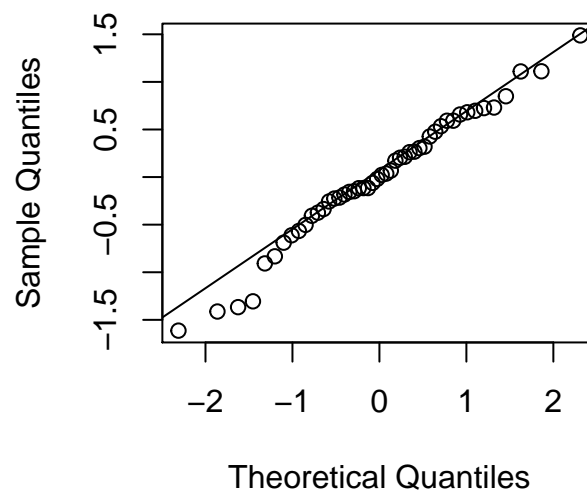
**Influential cases
(Cook's Distance)**
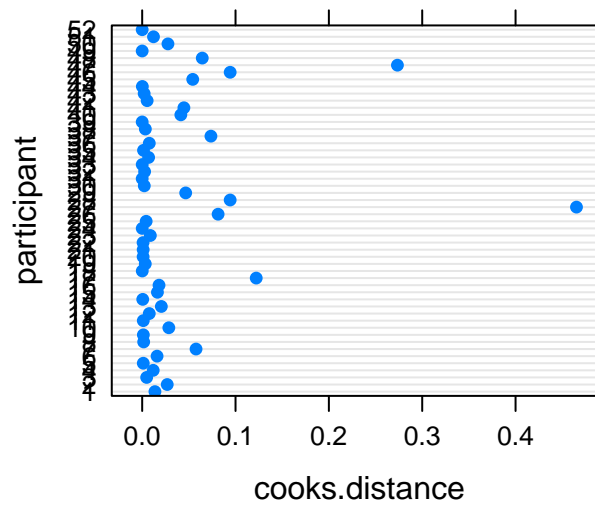
Model residuals were normally distributed around zero (0.9796322 0.98, p =0.563476) and an analysis of Cook's distances suggests that the model did not contain any observations that unjustifiably influenced parameter estimates (all 0.47 , for full presentation of model assumptions, see Appendix Figure **??**).

Based on these assessments, the model provided a robust basis for the inference that—in line with predictions—more positive violation of expectations concerning group performance predicted stronger feelings of team click. Also as predicted, this result was moderated by experiment condition, such that the effect of group performance expectation violation on group click was stronger in the high difficulty condition.

Prediction 1.a: Violations of expectations around team performance moderate the relationship between perceptions of joint action success and team click training group

```
## Saving 3.5 x 3.5 in image
## Warning:  Removed 1 rows containing missing values (geom_point).
## Warning:  Removed 1 rows containing missing values (geom_point).
```
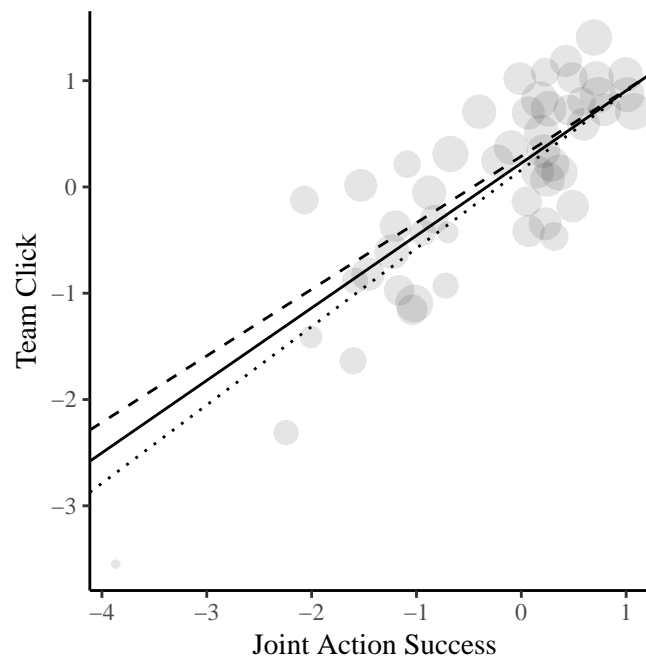

MODELS

```
## Saving 3.5 x 3.5 in image
## Warning: Removed 1 rows containing missing values (geom_point).
## Warning: Removed 1 rows containing missing values (geom_point).
```
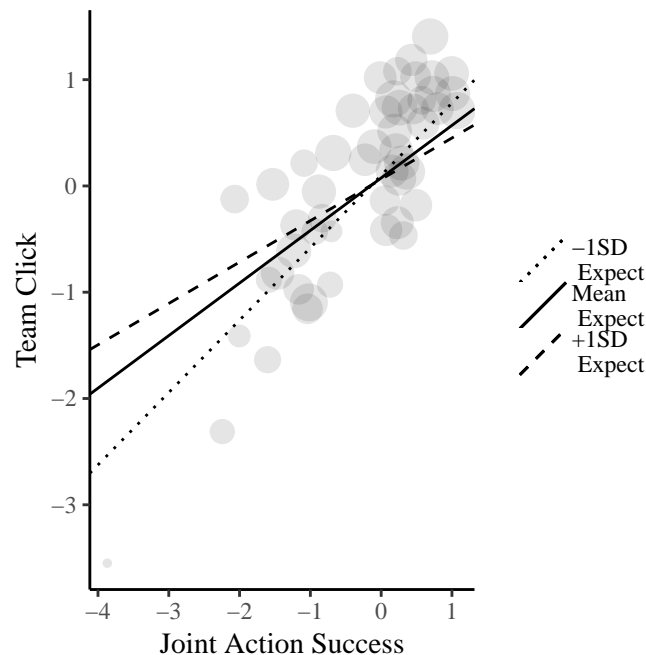

Figure **??** shows that the slope of the relationship between between perceptions of Joint Action Success and Team Click appears to be moderated by perception of team performance relative to prior expectations, such that higher levels (+1 standard deviation) of group performance expectation violation results in an increase of the steepness of the slope.

A LMER model, with Joint Action Success, Group Performance Expectations, and their interaction included as fixed effects, and perceptions of individual component performance, post-Experiment arousal, and average performance outcome included as controls, confirmed that the negative interation effect of Group Performance Expectations and Joint Action Success on Team Click was significant , suggesting that the relationship between joint action success an group click is more pronounced for *less* positive violations of expectations around group performance.

Distribution of M1a Residuals
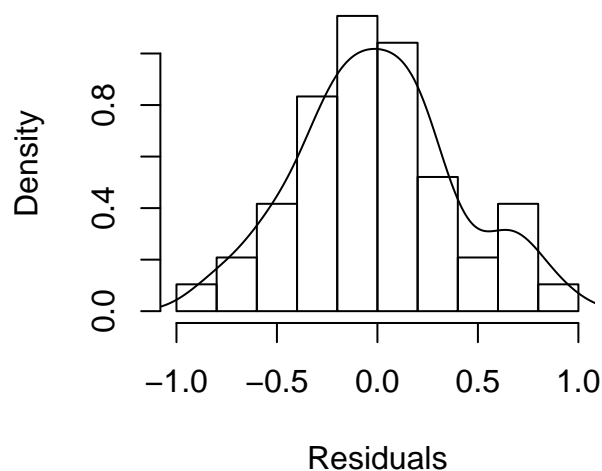
Scatterplot of M1a Residuals
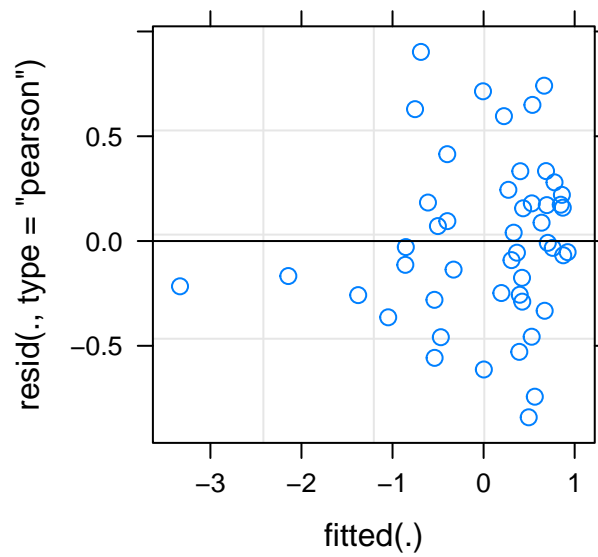
Norm QQ Plot of M1a Residuals


```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :  unable to evaluate
scaled gradient
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :  Model failed
to converge:  degenerate  Hessian with 1 negative eigenvalues
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :  unable to evaluate
scaled gradient
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :  Model failed
to converge:  degenerate  Hessian with 1 negative eigenvalues
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :  unable to evaluate
scaled gradient
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :  Model failed
to converge:  degenerate  Hessian with 1 negative eigenvalues
```

M1a Influential Cases
(Cook's Distance)
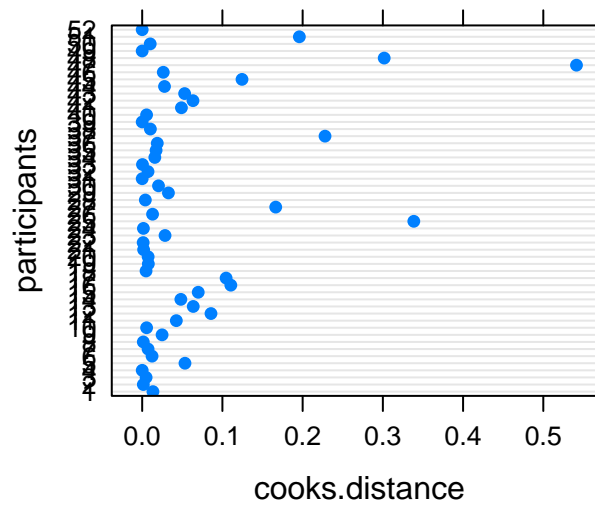
Prediction 2: Feelings of group click predict feelings of social bonding to the training group

```
groupClickBondScatter <- ggplot(twc, aes(x=groupClickPostFactor,
                         y=groupBondingFactorPost, color=condition)) + geom_point(shape = 1) +
                         geom_smooth(method=lm, se=F)   # add shaded confidence region
groupClickBondScatter +
#             guides(color = guide_legend(title="Condition"))+
            theme(plot.title = element_text(size=12)) +
            labs(x = "Team Click", y = "Social Bonding", color = "Condition")+
            apatheme +
            ggsave("groupClickBondScatter.pdf")
            #             scale_fill_
            #"Perceptions of Team Click \n predict feelings of Social Bonding to training group"
            #guides(fill=guide_legend("my awesome title"))
```

Figure **??** shows a clear relationship between Team Click and Social Bonding to the training group following the training group, with the slope for the high difficulty condition appearing slightly steeper than the low difficulty condition.

A LMER model (controlling for arousal, average trial performance outcome, and technical competence (subjective and objective)) revealed a strong positive relationship between feelings of group click and feelings of social bonding to the training group,

Distribution of residuals
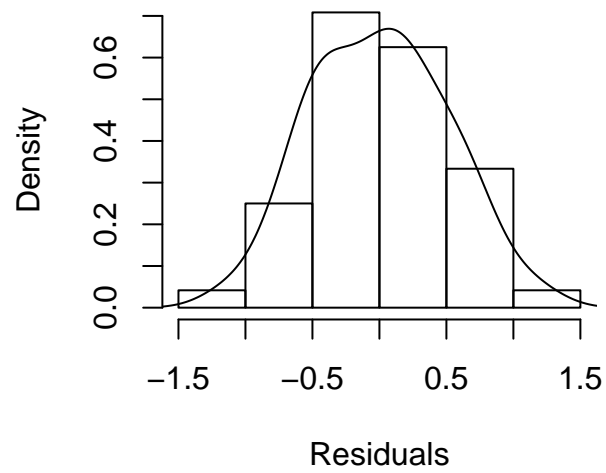
Scatterplot of residuals
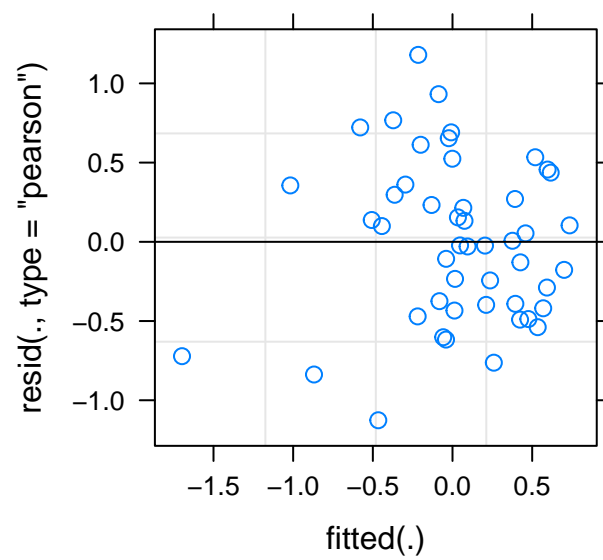
Norm QQ Plot of residuals
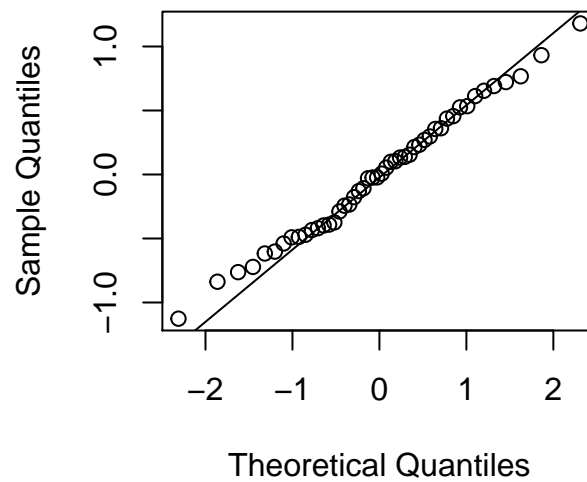
Influential Cases
(Cook's Distance)
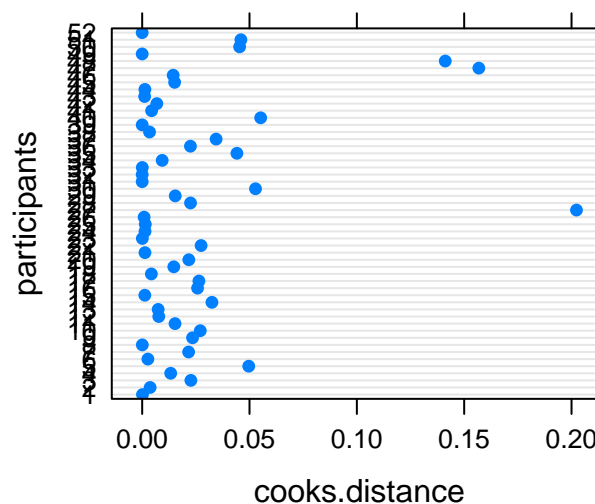
Prediction 3: More positive perceptions of team performance relative to expectations will predict higher levels of Social Bonding to the training group

```
## Saving 3.5 x 3.5 in image
## Warning:  Removed 1 rows containing non-finite values (stat_smooth).
## Warning:  Removed 1 rows containing missing values (geom_point).
## Warning:  Removed 1 rows containing non-finite values (stat_smooth).
## Warning:  Removed 1 rows containing missing values (geom_point).
```

@ Figure **??** shows a positive relationship between violation of group performance expectations and Social Bonding to the training group. The plot shows a pronounced difference in the slope of each experimental condition, suggesting that the relationship was stronger in the high difficulty condition.

A LMER model with group performance expectation violation, condition, and their interaction as fixed effects, experiment session as random effect (correlated slope and intercept), and controls, revealed a significant positive relationship between the interaction of group performance expectation violation and condition on social bonding
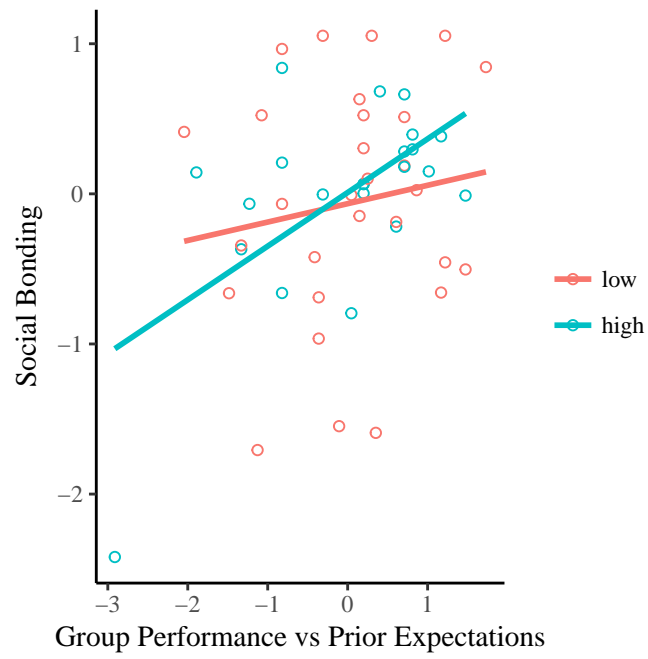
Figure 4: Positive violation of group performance expectations predict feelings of Social Bonding to training group
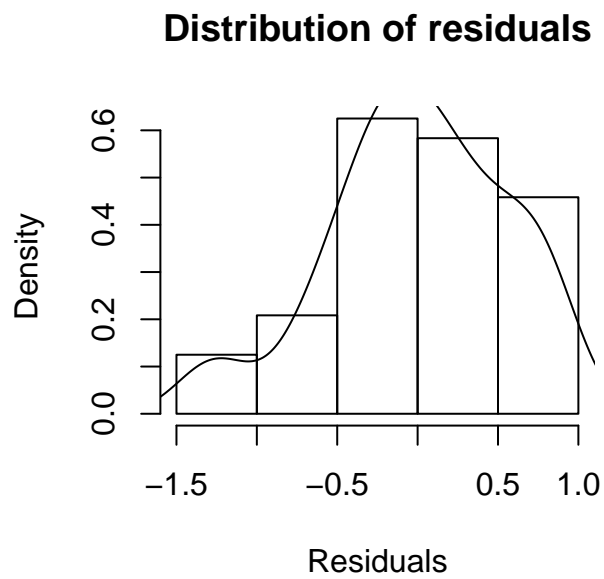
Scatterplot of residuals
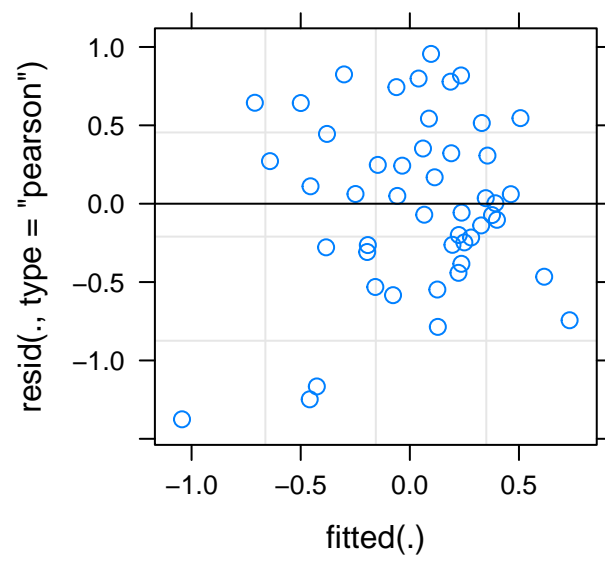
Norm QQ Plot of residuals
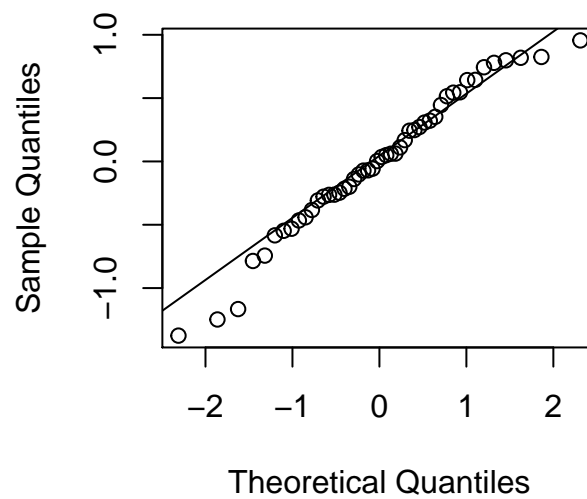

```
## Warning in optwrap(optimizer, devfun, getStart(start, rho$lower, rho$pp), :  convergence code
1 from optimx
```

Influential Cases
(Cook's Distance)
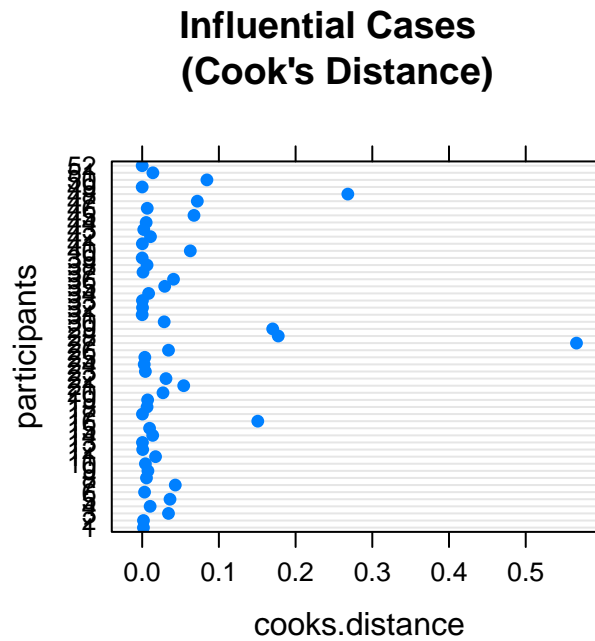
Model residuals were normally distributed around zero (0.9869075, p =0.8643267) and all were less than 0.57 (see model assumptions in Appendix **??**) .

These results suggested that the relationship between positive violation of group performance expectations and social bonding was significant, only in the high difficulty condition, and not overall accoss conditions.

Prediction 4: Feelings of group click will mediate a relationship between more positive perceptions of team performance relative to prior expectations around group performance and social bonding to the group

```
## Error in mediate(m2, m3, treat = "groupPerformance_3Norm", mediator = "groupClickPostFactor"):
unused arguments (treat = "groupPerformance_3Norm", mediator = "groupClickPostFactor")
## Error in summary(med.out.postExperiment):  object 'med.out.postExperiment' not found
## Error in mediate(m2, m3, treat = "groupPerformance_3Norm", mediator = "groupClickPostFactor",
:  unused arguments (treat = "groupPerformance_3Norm", mediator = "groupClickPostFactor", covariates
= list(condition = "high"))
## Error in summary(med.out.postExperiment.High):  object 'med.out.postExperiment.High' not found
## Error in mediate(m2, m3, treat = "groupPerformance_3Norm", mediator = "groupClickPostFactor",
:  unused arguments (treat = "groupPerformance_3Norm", mediator = "groupClickPostFactor", covariates
= list(condition = "low"))
## Error in summary(med.out.postExperiment.Low):  object 'med.out.postExperiment.Low' not found
```

```
## Error in plot(med.out.postExperiment, main = "Post-Drill mediation model"):  object 'med.out.postExpe
not found
```

```
## Error in plot(med.out.postExperiment.High, main = "High difficulty"):  object 'med.out.postExperiment
not found
```

```
## Error in plot(med.out.postExperiment.Low, main = "Low difficulty"):  object 'med.out.postExperiment.L
not found
```

Mediation analyses were conducted using linear mixed effects regressions in the Causal Mediation Analysis package in R (Version 4.4.5). To make inferences concerning the average indirect and total effects, quasi-Bayesian Markov Chain Monte Carlo (MCMC) method based on normal approximation and 1000 simulations was used to estimate the 95% Confidence Intervals (**Tofighi2016a**; **Imai2010**). MCMC estimation is a form of non-parametric bootstrapping whereby the sampling distribution for the effect of interest is not assumed to be normal but is instead simulated from the model estimates and their asymptotic variances and covariances **Preacher2008**

Results of the mediation analysis revealed significant average indirect effect of Group Performance Expectation Violation on Social Bonding attributable to Team Click,

Discussion of post-Experiment results

```
## Saving 3.5 x 3.5 in image
## Error in FUN(X[[i]], ...):  object 'groupPerfChangePrePostNorm' not found
```

```
##
## Attaching package:  'lmerTest'
## The following object is masked from 'package:lme4':
##
##     lmer
## The following object is masked from 'package:stats':
##
##     step
## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in confint.merMod(perfClickChange.lmer, method = "Wald", level = 0.95):  object 'perfClickChang
not found
```

```
## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in confint.merMod(perfClickChangeCondition.lmer, method = "Wald", :  object 'perfClickChangeCon
not found
## Error in anova(perfClickChange.lmer, perfClickChangeCondition.lmer):  object 'perfClickChange.lmer'
not found
```

```
## Error in resid(perfClickChangeCondition.lmer, type = "pearson"):  object 'perfClickChangeCondition.lm
not found
## Error in hist(groupPerfClickChangeResid, freq = FALSE, main = "Distribution of residuals \n (standard
:  object 'groupPerfClickChangeResid' not found
## Error in density(groupPerfClickChangeResid, na.rm = TRUE): object 'groupPerfClickChangeResid'
not found
## Error in is.numeric(x):  object 'groupPerfClickChangeResid' not found
## Error in psych::describe(groupPerfClickChangeResid):  object 'groupPerfClickChangeResid' not
found
```

```
## Error in plot(perfClickChangeCondition.lmer, main = "Scatterplot of residuals \n (standardised)"):
object 'perfClickChangeCondition.lmer' not found
```

```
## Error in residuals(perfClickChangeCondition.lmer):  object 'perfClickChangeCondition.lmer' not
found
## Error in residuals(perfClickChangeCondition.lmer):  object 'perfClickChangeCondition.lmer' not
found
```

```
## Error in ifelse(as.character(model@call)[3] == "data.update", data.adapted <- model.frame(model),
:  object 'perfClickChangeCondition.lmer' not found
## Error in cooks.distance(groupPerfChangeClickInfl):  object 'groupPerfChangeClickInfl' not found
## Error in eval(expr, envir, enclos):  object 'groupPerfChangeClickCooksD' not found
## Error in plot(groupPerfChangeClickInfl, which = "cook", main = "Influential cases \n (Cook's
Distance)", :  object 'groupPerfChangeClickInfl' not found
```
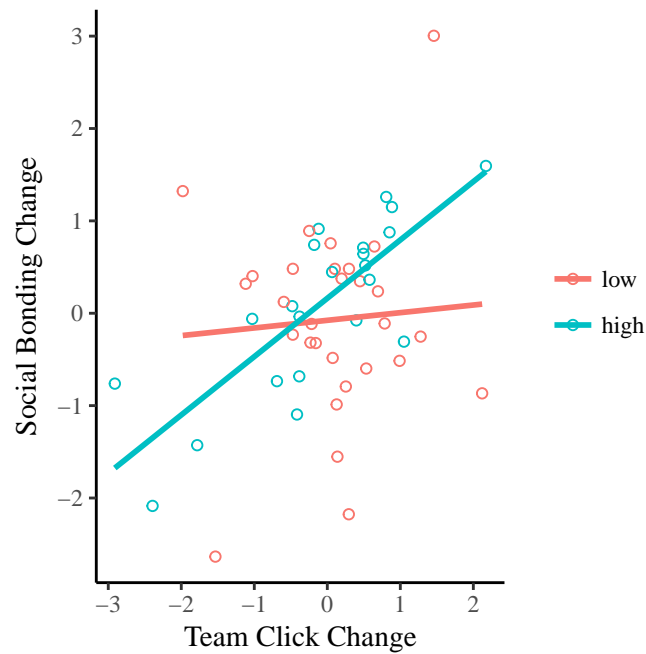
```
## Saving 3.5 x 3.5 in image
## Warning:  Removed 1 rows containing non-finite values (stat_smooth).
## Warning:  Removed 1 rows containing missing values (geom_point).
## Warning:  Removed 1 rows containing non-finite values (stat_smooth).
## Warning:  Removed 1 rows containing missing values (geom_point).
```
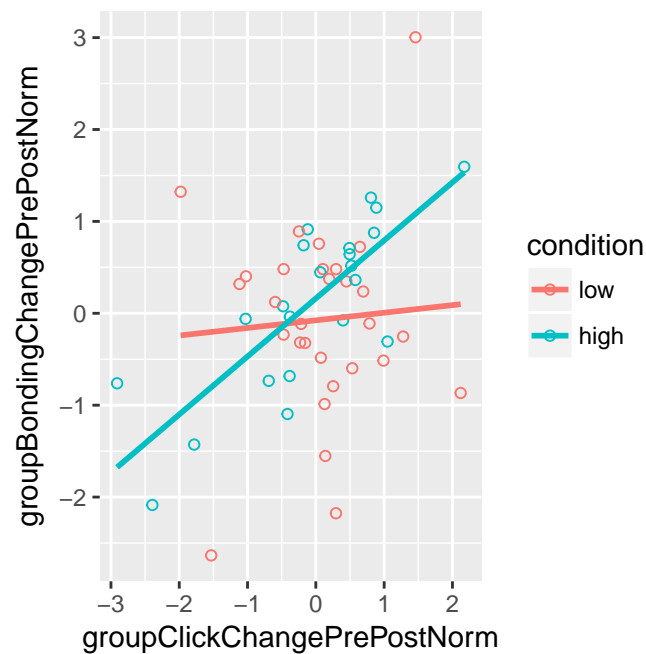
```
## Error in hist(clickBondChangeResid, freq = FALSE, main = "Distribution of residuals \n (standardised)
:  object 'clickBondChangeResid' not found
## Error in density(clickBondChangeResid, na.rm = TRUE): object 'clickBondChangeResid' not found
## Error in is.numeric(x):  object 'clickBondChangeResid' not found
## Error in psych::describe(clickBondChangeResid):  object 'clickBondChangeResid' not found


## Error in plot(clickBondingChangeCondition.lmer, main = "Scatterplot of residuals \n (standardised)"):
object 'clickBondingChangeCondition.lmer' not found


## Error in residuals(clickBondingChangeCondition.lmer):  object 'clickBondingChangeCondition.lmer'
not found
## Error in residuals(clickBondingChangeCondition.lmer):  object 'clickBondingChangeCondition.lmer'
not found


## Error in ifelse(as.character(model@call)[3] == "data.update", data.adapted <- model.frame(model),
:  object 'clickBondingChangeCondition.lmer' not found
## Error in cooks.distance(clickBondingChangeInfl):  object 'clickBondingChangeInfl' not found
## Error in eval(expr, envir, enclos):  object 'clickBondingChangeClickCooksD' not found
## Error in plot(clickBondingChangeInfl, which = "cook", main = "Influential cases \n (Cook's Distance)"
:  object 'clickBondingChangeInfl' not found


## Saving 3.5 x 3.5 in image
## Error in FUN(X[[i]], ...):  object 'groupPerfChangePrePostNorm' not found


## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in confint.merMod(perfBondingChange.lmer, method = "Wald", level = 0.95):  object 'perfBondingC
not found


## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in confint.merMod(perfBondingChangeCondition.lmer, method = "Wald", :  object 'perfBondingChang
not found
## Error in anova(perfBondingChange.lmer, perfBondingChangeCondition.lmer):  object 'perfBondingChange.l
not found


## Error in resid(perfBondingChangeCondition.lmer, type = "pearson"):  object 'perfBondingChangeConditio
not found
## Error in hist(perfBondChangeResid, freq = FALSE, main = "Distribution of residuals \n (standardised)"
:  object 'perfBondChangeResid' not found
## Error in density(perfBondChangeResid, na.rm = TRUE): object 'perfBondChangeResid' not found
## Error in is.numeric(x):  object 'perfBondChangeResid' not found
## Error in psych::describe(perfBondChangeResid):  object 'perfBondChangeResid' not found


## Error in plot(perfBondingChangeCondition.lmer, main = "Scatterplot of residuals \n (standardised)"):
object 'perfBondingChangeCondition.lmer' not found


## Error in residuals(perfBondingChangeCondition.lmer):  object 'perfBondingChangeCondition.lmer'
not found
## Error in residuals(perfBondingChangeCondition.lmer):  object 'perfBondingChangeCondition.lmer'
not found
```

```
## Error in ifelse(as.character(model@call)[3] == "data.update", data.adapted <- model.frame(model),
:  object 'perfBondingChangeCondition.lmer' not found
## Error in cooks.distance(perfBondingChangeInfl):  object 'perfBondingChangeInfl' not found
## Error in eval(expr, envir, enclos):  object 'perfBondingChangeClickCooksD' not found
## Error in plot(perfBondingChangeInfl, which = "cook", main = "Influential cases \n (Cook's Distance)",
:  object 'perfBondingChangeInfl' not found
```

```
## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in eval(predvars, data, env):  object 'groupPerfChangePrePostNorm' not found
## Error in mediate(m2Change, m3Change, treat = "groupPerfChangePrePostNorm", :  unused arguments
(treat = "groupPerfChangePrePostNorm", mediator = "groupClickChangePrePost")
## Error in summary(med.out.groupPerfClickChange):  object 'med.out.groupPerfClickChange' not found
## Error in mediate(m2Change, m3Change, treat = "groupPerfChangePrePostNorm", :  unused arguments
(treat = "groupPerfChangePrePostNorm", mediator = "groupClickChangePrePost", covariates = list(condition
= "high"))
## Error in summary(med.out.groupPerfClickChange.High):  object 'med.out.groupPerfClickChange.High'
not found
## Error in mediate(m2Change, m3Change, treat = "groupPerfChangePrePostNorm", :  unused arguments
(treat = "groupPerfChangePrePostNorm", mediator = "groupClickChangePrePost", covariates = list(condition
= "low"))
## Error in summary(med.out.groupPerfClickChange.Low):  object 'med.out.groupPerfClickChange.Low'
not found
```

```
## Error in plot(med.out.groupPerfClickChange, main = "Pre- to Post-Drill \n mediation model"):
object 'med.out.groupPerfClickChange' not found
```

```r
plot(med.out.groupPerfClickChange.High, main = "High difficulty")
```

```
## Error in plot(med.out.groupPerfClickChange.High, main = "High difficulty"):  object 'med.out.groupPer
not found
```

```r
dev.copy(pdf, "groupPerfExpClickChangeMedPlotHigh.pdf")
dev.off()
```

```
## Error in plot(med.out.groupPerfClickChange.Low, main = "Low difficulty"):  object 'med.out.groupPerfC
not found
```

5 Discussion