Exact Regression for Small and Wide Data: Analyzing Transgender Participation in a Game-Based Intervention

by

Violet Hecmanczuk

BA, The University of Alabama, 2019

Submitted to the Graduate Faculty of the

School of Public Health in partial fulfillment

of the requirements for the degree of

Master of Science

University of Pittsburgh

2023

UNIVERSITY OF PITTSBURGH

SCHOOL OF PUBLIC HEALTH

This thesis was presented

by

Violet Hecmanczuk

It was defended on

April 24, 2023

and approved by

Thesis Advisor: Jenna C. Carlson, PhD, Assistant Professor, Biostatistics
School of Public Health, University of Pittsburgh

Ada Youk, PhD, Research Associate Professor, Biostatistics
School of Public Health, University of Pittsburgh

Robert W.S. Coulter, PhD, Assistant Professor, Behavioral and Community Health Sciences
School of Public Health, University of Pittsburgh

Jaime Sidani, PhD, Assistant Professor, Behavioral and Community Health Sciences
School of Public Health, University of Pittsburgh

Exact Regression for Small and Wide Data: Analyzing Transgender Participation in a Game-Based Intervention

Violet Hecmanczuk, M.S.

University of Pittsburgh, 2023

Background: The public health significance of this work is to push forward new strategies for measuring transgender populations, especially study populations that face small sample size challenges. Transgender individuals face unique health disparities. Intervention methods to mitigate these disparities are still evolving and it is of interest to see what factors within transgender populations are associated with intervention fidelity.

Methods: Data was pulled from the intervention arm of a randomized controlled trial (n = 120). Participants had been instructed to download and play a computer game aimed at LGBT adolescents. Transgender and cisgender participation were compared across three outcomes: download (binary), hours played, and completion (binary). Among transgender participants (n = 62), participation in those three outcomes was modeled on social covariates. A purposeful selection process that combined field knowledge and statistical testing was used to select social variables. Linear and logistic regression models were estimated. Additionally, exact logistic regression was implemented in final models measuring completion and download within the transgender subgroup.

Results: Compared to their cisgender LGBT peers, transgender adolescents were equally likely to play the game, averaged about the same amount of hours, and were more likely to complete the game (OR: 2.95). Within the transgender population, family social support, the presence of a gender-sexuality alliance, and how a participant felt their gender mannerisms were perceived affected participation. Higher self-reported support from family was associated with playing more hours (CI: 0.10, 0.81). Students who knew their school had a gay straight alliance also tended to play more hours (CI: 0.2, 2.5) and were more likely to download (Exact CI: 0.78, 173.99).

Conclusion: Exact inference aided in reasonably estimating social covariates. Future studies facing sample size challenges should consider using this method.

Table of Contents

List of Tables

List of Figures

Preface

Terms Used

Nomenclature surrounding transgender individuals remains a new and evolving concept for many individuals at the time of writing this. I explain my choice of terms here. Italicized terms are used explicitly in the paper. Non-italicized defined terms are defined to contextualize the italicized terms.

Transgender individuals are people whose gender identity is different from their assigned sex at birth. Gender identity is whether someone is a boy, girl, combination of the two, or something else. “Assigned sex at birth” is what a doctor marks on a newborn’s birth certificate. Notably, the term “biological sex” should not be used as a replacement for “assigned sex at birth” due to both inaccuracy and insensitivity. Transgender women who have medically transitioned share far more “biological” characteristics in common with cisgender women than with cisgender men, and the inverse is true in the case of transgender men. As biologically alive organisms who are men, it’s most accurate to call transgender men “biological men”; likewise transgender women are most accurately called “biological women”.

Cisgender individuals are individuals who are not transgender, whose sex assigned at birth match their current gender identity.

Trans may be used as shorthand for “transgender”, and *cis* as shorthand for “cisgender”.

Non-binary individuals have a gender identity that can neither be accurately described as “male” or “female”. It should be noted that non-binary individuals have varying identities: they may be a third gender other than male and female, may have no gender at all, or may be a combination of two or more genders. In this thesis, there was no distinction made between these specific identities; individuals outside the binary of “boy” and “girl” were all simply recorded as “non-binary”. Being non-binary automatically means one’s gender identity conflicts with their assigned gender at birth; as such it meets the above definition of “transgender” and all non-binary individuals are coded as “transgender” in this thesis.

Outness refers to a measure of how “out” a transgender individual is. Being “out” refers generally in LGBT context to other people knowing about one’s LGBT status, and in the case of this paper refers to other people knowing that the individual is transgender. If a transgender individual has “high outness” it means more people in their lives know that they are transgender.

Basic gendered terms like *boy* and *girl* refer always to gender identity in this paper, and never to sex assigned at birth.

The choices made in nomenclature noted above are made with the hope of guiding future papers. Understanding transgender populations requires understanding transgender individuals to be ontologically, consistently, the gender they identify as.

Dedication

I don't have the support of a mother, sibling, or father.

I am glad to have professors who support me.

To Dr. Carlson, thank you for taking me on as a student. And for your dedication and passion to making us good statisticians.

To Dr. Coulter, thank you for your insight, and for letting me use this dataset. The hardest days of thesis writing were easier knowing I had a topic I cared deeply about.

To Dr. Youk, thank you for suggesting the lost art of exact inference. The concept excites me as someone who has been seeking ways to deal with small data.

To Dr. Sidani, thank you for taking me on in a BCHS study. And for being a great boss. I feel welcome and affirmed in your team setting.

To everyone else in Public Health -- professor, student, administrator, staff-- who has helped me, thank you. I wouldn't have gotten this far without getting the occasional coffee from Carmen in the Public Health building.

Whom I owe the biggest thanks to are my community of friends. In no particular order:

To Alex: Thanks for giving me the coming out support I needed. It helped me live as boldly as I have the past two years. Thanks for being my longest friend.

To Jade, you taught me people like us can weather any storm, and you've been my umbrella more times than I can count. There’s a lot that can’t fit here. But thanks for being my ride or die.

To Taylor, thank you for being someone I can simply have girl time with. And for half-accidentally building half of my wardrobe.

To Sabella, our impromptu hangouts are often the breaks I didn't know I needed. Thank you for staying around.

To Anya, whose timing for encouragement has been impeccable.

To Rika, who has taught me I'm allowed to be loud, and made me a better dancer.

To Maddie, for being around to do budget-friendly fun. My bank account thanks you.

To Matt, Luke, Robby, and William, thanks for the weekly break from the stresses of adulthood. Thanks for creating a welcoming environment for both Nissa and Violet.

To Helena, for being the first person I shared my joy with. For proving it's possible to have a girls' night in with a friend in a different continent.

To Laura J, you’ve helped me unlock parts of myself I often doubted that I ever could.

To Jaide R, thanks for being so easy to talk to.

To many more friends who are part of my blossoming new life. The past two years --despite massive challenges and hardships-- have been faraway better than any of the previous twenty-five, and whether I named you or not, everyone who has treated me as their female friend has been part of that.

This is also dedicated to every transgender university student in the U.S. I'm sorry that so many institutions have permitted discussion of killing us. Lean on your community-- friends, trusted professors, mentors, kind everyday strangers. They have your back.

To every academic and scientist who has done research demonstrating that we are who we say we are. To the cultural movements that pushed for that question in the first place. To the individuals who engaged with either and are leading the fight for our liberation.

And to whoever decides it's worth their time to hire a dorky, peppy, annoying blonde M.S. biostatistician who just so happens to be a trans woman.

1.0 Introduction

Transgender individuals face unique challenges, and evaluating how studies and interventions appeal to them specifically is vital to the future of public health. Connolly (2016) found that transgender youth are at greater risk of adverse mental health outcomes including depression and eating disorders. Connolly also found that mental health and psychological functioning improve significantly in transgender youth who either receive gender affirming medical therapy, are supported in social transition, or both. Tordoff (2022) similarly observed that transgender individuals aged between 13-20 years old who received gender affirming hormones, puberty blockers, or both, had 60% lower odds of depression and 73% lower odds of suicidality. De Vries, et al (2014) conducted a seven-year longitudinal study on transgender individuals who received puberty blockers at mean age 13.6, affirming hormones at mean age 16.7, and gender reassignment surgery at mean age 19.7, and found that dysphoria was alleviated, psychological functioning improved, and "well-being was similar to or better than same-age young adults from the general population". However, affirming care and safe social transition remain inaccessible to transgender youth who are trapped with unsupportive families, communities, or state legislatures. It is of interest to explore what other interventions and existing variables may affect mental and behavioral health outcomes in transgender youth.

Studies often combine transgender populations with lesbian and gay populations. A case-control study at the University of Pittsburgh sought to measure the effectiveness of a game-based intervention aimed at "Sexual and Gender Minority Youth" in reducing risk-seeking behaviors (Egan et al, 2021). The population for Egan's study was high school students, aged 14-18, living in the United States, who self-reported being any of the following: gay/lesbian, bisexual, transgender, asexual, queer, or any combination of multiple of these. The students in the intervention group were instructed to download and play a game called “Singularities” that featured notable parallels to LGBT experiences: the protagonist of the game encountered prejudice due to their own uniqueness. Students in the intervention group self-reported whether they downloaded the game, how many hours they played the game for, and whether they completed the game.

The purpose of this thesis is to measure social factors driving transgender participation in *Singularities*, as well as to see how their overall participation compares to their cisgender peers. The sample of interest, as is often the case when studying transgender people, is small: less than one hundred. Because of this smallness, attempting to model a wide set of variables is susceptible to breaking down (Heinze 2018). Rather than include the tens of recorded input variables, then, the approach in this thesis begins by defining a subset of variables of interest through area expertise and intellectual curiosity. Participation in a game-based intervention is a *behavioral* outcome, so previous studies on behavioral and mental health in transgender individuals are logical choices to guide selection. Previous studies have found that family support is vastly influential on the mental health of transgender adolescents (Simons 2013). Wilson’s study (2016) of transgender women aged 16-24 found that their mental health was impacted adversely by racial discrimination, and impacted positively by family support. Ryan (2010) found that family acceptance improved health and reduced risk of depression, substance abuse, and suicidal ideation and behaviors. Specifically, this thesis chooses to investigate the potential influence of: how participants report others perceiving their gender by mannerisms and appearance, their race (white or nonwhite), their outness (to family, friends, and school), whether a participant’s school has a gay-straight alliance group, and how they rate their support systems (family and friends). It was initially of interest to consider models that included specific gender (girl, boy, or nonbinary), but the low count of transgender girls in this sample (n = 5) made such evaluation infeasible.

"Purposeful selection" of variables seeks to balance area expertise with mathematical insight in choosing which variables to include in a model (Hosmer 2013). The aforementioned subset of variables was subject to rigorous testing to determine which covariates should be included in final models; separately for three outcomes of interest: download (yes or no), completion (yes or no), and hours played (continuous).

Exact inference is a common method used to handle small sample sizes (Agresti, 1992). The *elrm* package for R software implements a Markov Chain Monte Carlo approximation of exact logistic regression (Zamar 2007). This package was used to conduct exact logistic regression for the two binary outcomes: download and completion. These exact logistic models were compared to "traditional" maximum likelihood logistic regression models.

By highlighting variables of interest, implementing a purposeful selection process suitable to small data, and exploring the potential utility of exact inference, this thesis hopes to provide future intuition for ways to analyze transgender datasets, often a challenging task due to small sample size coinciding with many recorded variables.

\

2.0 Methods

2.1 Data

The original dataset is from a study by Egan et al (2019) that examined the effects of a game based intervention on sexual and gender minority youth (SGMY). All participants were either a minority in sexual orientation (gay, lesbian, bisexual, asexual, or queer) or were transgender, and additionally had been victims of either cyberbullying or in-person bullying in the past year. Additionally, all participants were English literate, lived in the United States, were between 14-18 years old, had the means to download the game, and had an email address. All of this criteria was evaluated through a web-based screener that potential participants were required to complete.

This thesis seeks to examine intervention fidelity—that is, participation in the game-- in the trial. Consequently, it is concerned with the intervention arm of the trial only, which had a sample size of 120. This paper examines participation in *Singularities* through three outcome variables: *download*, *hours played*, and *completion*. *Download* records whether a participant self-reported downloading the *Singularities* game. Students were considered to have downloaded if they reported "yes", were considered to *not* have downloaded if they reported "no" or "unsure", and were treated as missing if they had no response. *Hours played* is the self-reported number of hours played by the participant; missing values were treated as missing. *Completion* is whether the student self-reported completing the game or not. Students were considered to have completed the game if they reported "yes", and were considered to *not* have completed if either (i) they reported "no" or "unsure" OR (ii) they had a missing value, but had reported "no" for having downloaded the game. If students had missing responses for both download and completion they were treated as having missing values for completion.

Hours played, download, and completion were assessed at three different time intervals; the newly transformed dataset takes the values from each student's most recent, nonmissing entry for each variable (for instance, a student who reported 3 hours of gameplay at interval 1, 12 hours gameplay at interval 2, and had missing values at interval 3, is assigned a "total hours played" of 12). These are the only variables that were repeated over time; all other variables were recorded once per participant, at t = 1 (baseline).

2.1.1 Variables

2.1.2 Race/Ethnicity

Race/Ethnicity was treated as a two-level categorical variable. Students who answered “white” were coded as “white”, while students who answered any of the following were coded “nonwhite”: Black, Latinx, Asian or Pacific Islander, or multiracial.

2.1.3 Discrete Scales

Many of the variables used were discrete scales with several levels, and were estimated as continuous. This includes: gender appearance, gender mannerisms, outness to family, outness to friends, outness to school, family social support, and friend social support.

Table 1 summarizes the treatment of variables. Categorical variables of three or more levels were either dichotomized or treated as continuous when estimated in models.

Table 1: Variable Handling

Variable	**Initial Data Type**	**Coefficient estimation in models**
Perceived Gender Appearance	Discrete 7-point scale	Continuous
Perceived Gender Mannerisms	Discrete 7-point scale	Continuous
Family Social Support	Average of four 7-point scales.	Continuous
Outness to family	Discrete 5-point scale	Continuous
Outness to friends	Discrete 5-point scale	Continuous
Outness to school	Discrete 5-point scale	Continuous
Gay-Straight Alliance	3-level categorical variable: Yes, No, Unknown	Binary: Yes or No/Unknown.
Race	5-level categorical variable	Binary: White or Nonwhite
Cisgender or Transgender	Binary	Binary

2.2 Models

2.2.1 High Level Overview

Participation is measured through three outcomes: download rate, hours played, and completion rate. For each outcome, two questions were asked. Firstly, does participation differ by transgender versus cisgender participants? Secondly, among transgender participants, what effect do social covariates, such as outness and support, have on participation?

For measuring transgender versus cisgender participation, two univariate logistic models were fit (for download and completion) and one univariate linear regression model was fit (for hours played). For social covariates, for hours played, one linear regression model was fit. For social covariates, for download and completion, a total of six models were fit. In each case, one logistic model was fit, and its estimates were compared to those obtained from two exact logistic models.

Table 2: Summary of the Ten Models

Outcome Variable	**Population**	**Regression Method**	**Model**	**Input Variables**
Download	All (N = 120)	Logistic	Model I	Gender (cis or trans)
	Transgender only (N = 63)	Logistic	Model II	Gay-Straight Alliance, Family Social Support
		Exact Logistic	Model III	Gay-Straight Alliance, Family Social Support*
			Model IV	Gay-Straight Alliance*, Family Social Support
Hours	All	Linear	Model V	Gender (cis or trans)
	Transgender only	Linear	Model VI	Gay-Straight Alliance, Family Social Support, Outness to Family
Completion	All	Logistic	Model VII	Gender (cis or trans)

	Transgender Only	Logistic	Model VIII	Gay-Straight Alliance, Gender Mannerisms
		Exact Logistic	Model IX	Gender Mannerisms, Gay-Straight Alliance*
			Model X	Gender Mannerisms*, Gay-Straight Alliance
	*in *formula* parameter only; not placed in *interest* parameter and therefore not calculated, but still used to influence calculation of other variables			

2.2.2 Cisgender and Transgender Participants

To measure whether participation differs for transgender individuals compared to cisgender individuals, simple univariate regression models were fit. To test the difference between groups in hours played, linear regression was used. For download (binary) and completion (binary), logistic regression was used.

2.2.3 Social Covariate Selection

Each outcome (download, hours played, and completion) separately underwent the same initial variable selection process for social covariates. One outcome's results did not influence the process for other outcomes.

The selection process is heavily influenced by Hosmer et al (2013). Firstly, a subset of nine variables of interest were selected: race (white or nonwhite), gender appearance, gender mannerisms, outness to family, outness to friends, outness to school, whether they had a gay-straight alliance (often now called *gender-sexuality alliances*) at school, family support, and friend support. Gender (boy, girl, or nonbinary) was also considered, but eliminated due to the low count of transgender girls (n = 5).

For each variable within the subset of nine, the univariate model was measured by partial likelihood-ratio test compared to the null model. An initial multivariable model was then fit, using only the variables with a partial likelihood-ratio test p-value of 0.25 or lower. With that model fit, Wald test p values were then used to identify more variables that could potentially be deleted (p > 0.25) . Each variable was checked for confounding upon deletion: if the coefficient values for other variables changed by more than 20%, the variable would have been added back. No variables' deletion had such an effect. The model at the end of this will be referred to as the *preliminary effects model*.

After this, the initially deleted variables (from the univariate LRT step) were also tested, one at a time, for confounding to see if they might be added back. Coefficient values in the preliminary effects model were compared to coefficient values in that model with the addition of one of the deleted variables, and this process repeated for each initially deleted variables. The same threshold of 20% was used. No variables had such an effect on coefficient values. As a result, the preliminary effects model became the *final model*. It was the case that no confounding was found in all three outcomes (download, hours played, and completion).

Notably, this selection process did not include assessing for interactions. The reason for this decision is that the transgender dataset is quite small (n = 63). With such a small sample size, tests of interactions would be underpowered. In such a case, the great reduction to power that an interaction term brings about is undesirable, so no interactions were included.

The above concluded the variable selection process for Models I, II, V, VI, VII, and VIII. Models III, IV, IX, and X, which are the exact logistic regression models, inherited values from the selection process used in their corresponding (same outcome) non-exact logistic regression models, but in some cases applied necessary transformations.

In evaluating the estimates obtained in the final models, $p < 0.05$ is considered the cutoff for statistical significance.

2.2.4 Exact Logistic Regression

It is common to use logistic regression to model data with binary outcome variables. Logistic regression is a generalized linear model that uses the logit link function to model based on the natural log of the odds of an event occurring. Let *i* denote an individual observation in a random sample of size *n*. Given vector $X = \{x_{1i}, \ldots x_{ji}\}$ for *j* covariates corresponding to random outcome variable Y_i, let parameter *p* be the probability that each random outcome variable $Y_i = 1$. Then logistic regression models the relationship:

$$\log\left(\frac{p_i}{1-p_i}\right) = x_i\beta + \gamma \tag{1}$$

Normally, to obtain estimates, the resulting likelihood function for $Y_1, Y_2, \ldots Y_n$ is maximized with respect to parameters of interest.

In this case, the sample of transgender adolescents who were played the *Singularities* game was small (n = 63). The goal of this thesis is to make inferences about how multiple social covariates may or may not effect odds of download and odds of completion for the game. Such inference becomes difficult using the conventional approach to logistic regression of maximum likelihood, as Forster (2002) points out. Harrell et al (2015) recommended at least 15 events per variable (EPV) and Heinze's work (2018) suggests that some cases may necessitate many as 50 EPV. Studies of transgender adolescents often are simultaneously interested in many variables and plagued by small sample sizes, so remedies to such a challenge are of interest. In this thesis's case, the original pool of variables of interest was ten. This sets up an EPV of 6.3, or 4.1 if missing data is accounted for. Using Hosmer's (2008) process, does allow univariate tests to shrink the pool of variables and thus obtain a higher EPV. However, it is not guaranteed that it will lead to such a reduction in the number of variables. If all variables were to have p values below 0.25 against the null model in univariate likelihood ratio tests, then the EPV could remain as low as 4.1. In our case, we did end up benefiting from many variables having high p-values in their univariate likelihood-ratio tests against the null model. This allowed us to proceed with less than five variables in the multivariate stages, alleviating concerns somewhat about EPV. However, it it is easy to imagine a scenario in which univariate tests do not delete enough variables, and the process becomes at risk of having to rely on trusting the testing of multivariate models that have EPV < 10. When it is in question whether EPV is at an acceptable ratio, exact logistic regression becomes a preferable way of making inferences about the variables. The results obtained using each method are compared in this paper to show how they can differ.

While maximum likelihood methods for logistic regression do not distinguish between covariates of "interest" and "nuisance" covariates, exact logistic regression is different. Exact logistic regression is based on the conditional distribution of the parameter of interest's sufficient statistics, given the nuisance parameters' sufficient statistics. In the case of the elrm package used in this analysis, the *interest* argument in the *elrm*() function tells the program which variables are of interest. Meanwhile, all variables that are not present in interest but are present in the *formula* argument are treated as nuisance parameters. To maximize the accuracy of a single variable's estimate, Zamar (2007) recommends running *elrm()* with only that variable in the *interest* argument. In this analysis, for each outcome (download and completion), we do exactly that. For instance, in using exact logistic regression to model completion, we had two variables (inherited from the selection process used to select variables for the maximum likelihood model), and we ran *elrm()* twice: each time the *formula* argument contained both variables, and the variable specified in the *interest* argument changed each time. The same process was followed when conducting exact logistic regression for the completion outcome.

The elrm, or Exact Logistic Regression Model, package implements an approximation of exact logistic regression using a slightly modified version of the Metropolis-Hastings algorithm developed by Forster et al (2002). Forster et al's algorithm is one of multiple Markov Chain Monte Carlo (MCMC) methods used to approximate exact logistic regression. MCMC methods are used because enumeration of exact logistic regression is often infeasible due to complexity. Forster showed that, in cases where enumeration was feasible, their algorithm obtained very similar p-values to those obtained through enumeration. For a review of other approaches to MCMC, see Oster (2002) and Oster (2003).

Implementing *elrm* requires creating a special type of data frame. In such a frame, a row is created for each possible combination of the input variables. A row contains those input variable variables, the number of observations which had that combination, and the number of observations which had a "success" (i.e. which had a value of "yes" for either download or completion, depending on which was being modeled.

For each outcome of interest, the input variables used in *formula* were the same variables selected in the final non-exact logistic regression model for that outcome.

The formula argument in the *elrm* function specifies the input variables that should be iterated over. Given one or more of such variables, exact logistic regression iterates over each possible combination of values for the input variables and counts both the number of observations with that combination ("trials") and, among those combinations, how many have a given outcome value ("successes"). After iterating, the *elrm* package gives coefficient estimates for the variable(s) specified in the *interest* argument. As mentioned above, in this analysis, all variables inherited from the selection process were always passed to the *formula* argument, while among those variables, one variable at a time was passed to the *interest* argument, allowing the distribution to avoid becoming degenerate and thus producing an NA result (Zamar 2007).

In addition to the *interest* and *formula* argument, elrm() allows the user to specify the number of Markov chain iterations to make, and a burn-in period (during which values of the chain are discarded) (Zamar 2007). For all models fit, 22000 iterations with a burn-in period of 2000 was used. Random seed was set to 11037 before fitting all models. These numbers were based on those used by the UCLA statistical consulting group (Exact Logistic Regression | R Data Analysis Examples, 2023).

After obtaining the estimated coefficients, confidence intervals, and p-values for each variable through exact logistic regression, those results are compared to that variable's estimated coefficients, confidence intervals, and p-values obtained using "typical" logistic regression.

3.0 Results

3.1 Descriptive Statistics

3.1.1 Comparing Participation for Transgender and Cisgender Participants

Table 3 summarizes descriptive statistics for the three outcomes, and for gender minority status (i.e. whether the student is transgender or cisgender). While the overwhelming majority of students downloaded the game (80%), only 31% of participants completed the game. Participants tended to play about three hours. Of the 120 participants, 63 were transgender, and 57 were cisgender. We can see that between transgender and cisgender students, rate of download was similar (80% and 79%, respectively). The median and quartiles for hours played were also quite similar. Completion rate had a more noticeable difference. The rate of completion for transgender students was 34%, while only 16% of cisgender students completed the game. Partial likelihood-ratio tests against the null model were used to compute p-values.

Table 3: Participation Outcomes, All Participants

Characteristic	***All Students, N = 120[1]***	***Transgender Students, N = 63[1]***	***Cisgender Sexual Minority Students, N = 57[1]***	***P-value[2]***
Game Download				0.642
No	17 (20%)	8 (20%)	9 (21%)	
Yes	67 (80%)	33 (80%)	34 (79%)	
Unknown	36	22	14	
Completed Game?				**0.05**
No	64 (69%)	27 (66%)	37 (84%)	
Yes	21 (24%)	14 (34%)	7 (16%)	
Unknown	35	22	13	
Hours Played	3.00 (1.00, 4.00)	3.00 (1.00, 4.00)	2.00 (1.00, 3.25)	0.642

Unknown	35	22	13

[1] n (%); Median (IQR)

[2]chi-square test

We see a low, significant p-value for completion only.

3.1.2 Transgender Participation by Covariates of Interest

3.1.3 Download

Table 4 summarizes download rate by covariates of interest among the transgender participants. Notably, for the 63 transgender participants, 33 downloaded the game, 8 did not, and 22 were missing a response. We can see that p-values using common statistical tests are quite high for most variables. Lowest are family support (0.2) and gay-straight alliance (0.013).

Table 4: Download Rate by Covariates of Interest for Trans Participants

Characteristic	***No, N = 8[1]***	***Yes, N = 33[1]***	***p-value***
Gender Mannerisms			0.5[2]
Very Feminine	0 (0%)	3 (9.1%)	
Mostly Feminine	1 (12%)	2 (6.1%)	
Somewhat Feminine	4 (50%)	10 (30%)	
Equally Feminine and Masculine	2 (25%)	9 (27%)	
Somewhat Masculine	0 (0%)	7 (21%)	
Mostly Masculine	1 (12%)	2 (6.1%)	
Very Masculine	0 (0%)	0 (0%)	
Race			>0.9[2]
Nonwhite	2 (25%)	11 (33%)	
White	6 (75%)	22 (67%)	
Gender			0.5[2]
Boys	4 (50%)	12 (36%)	
Girls	1 (12%)	3 (9.1%)	
Nonbinary	3 (38%)	18 (55%)	
Gay-Straight Alliance (Yes)	0 (0%)	17 (52%)	0.013[2]
Gendered Apperance			0.7[2]
Very Feminine	0 (0%)	2 (6.1%)	
Mostly Feminine	1 (12%)	1 (3.0%)	
Somewhat Feminine	1 (12%)	3 (9.1%)	

Equally Feminine and Masculine	3 (38%)	9 (27%)	
Somewhat Masculine	2 (25%)	9 (27%)	
Mostly Masculine	1 (12%)	9 (27%)	
Very Masculine	0 (0%)	0 (0%)	
Family Support	1.88 (1.00, 2.19)	3.00 (1.00, 3.75)	0.2[3]
Friend Support	3.88 (2.88, 5.38)	5.00 (3.75, 5.75)	0.4[3]
Outness to Family			0.7[2]
Everyone Knows	0 (0%)	4 (12%)	
Most People Know	0 (0%)	4 (12%)	
Some People Know	2 (25%)	4 (12%)	
A Few People Know	2 (25%)	9 (27%)	
No one Knows	4 (50%)	12 (36%)	

[1] *n (%); Median (IQR)*

[2] *Fisher's exact test*

[3] *Wilcoxon rank sum test*

Notably, many categorical variables have low cell counts. This is what informed the decision to modify those with three or more levels when modeling. As we will see later, categorical variables with three or more levels, when modeling, were either treated as continuous variables, or collapsed to two levels. The low cell counts were also informed the decision to implement exact logistic regression models.

3.1.4 Hours Played

Figures 1-7 summarize hours played by each covariate of interest for transgender participants. At a high level, playing *Singularities* for more hours appears to be associated with: white race, a nonbinary gender identity, middling gender appearance and mannerisms, and higher family social support.

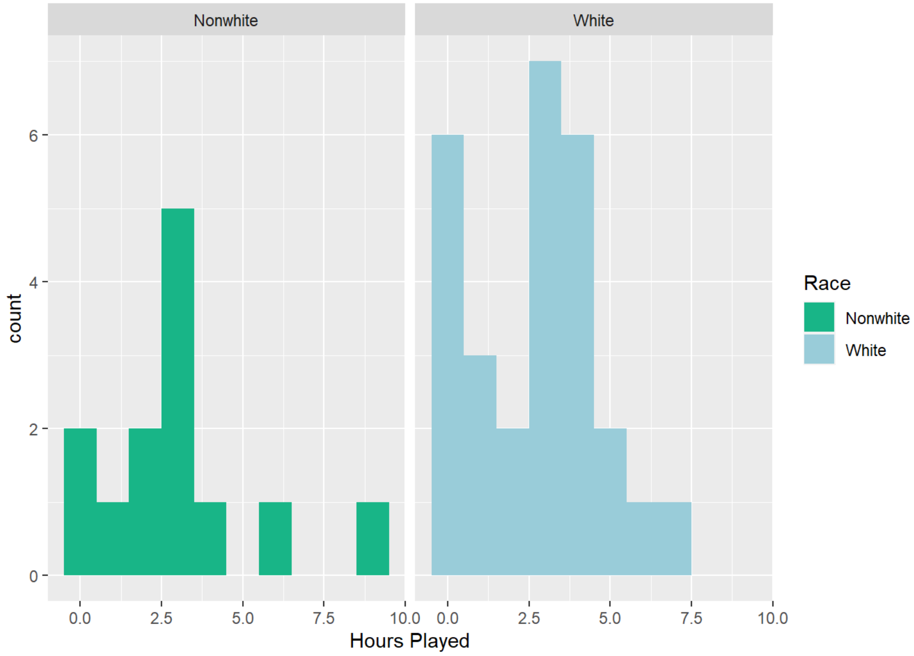

Figure 1: Hours Played by Race for Transgender Participants

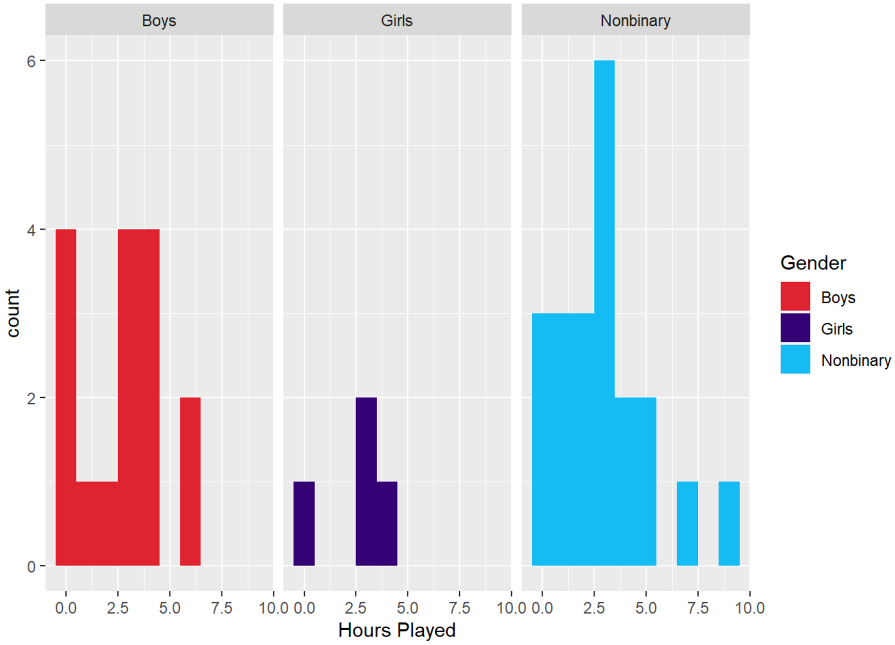

Figure 2: Hours Played by Gender for Transgender Participants

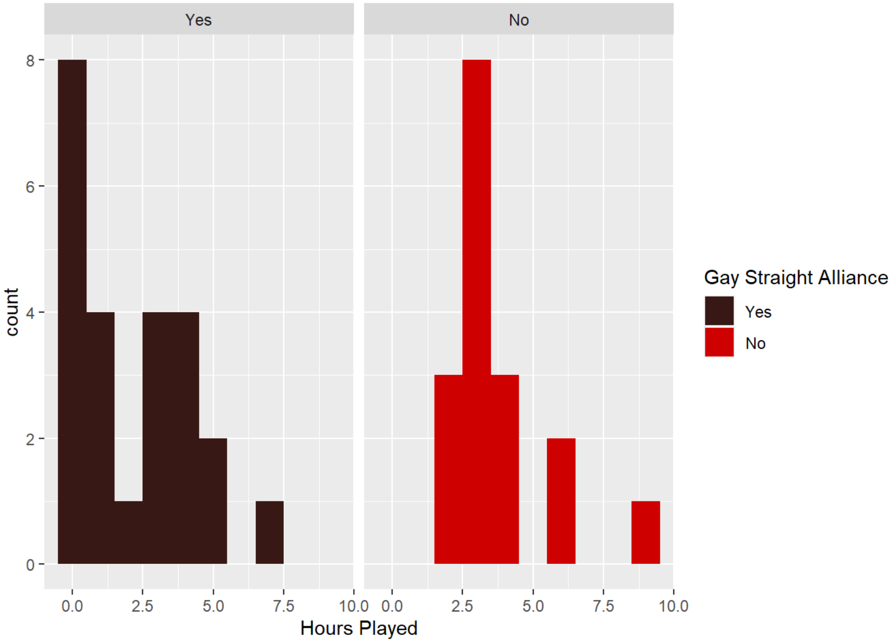

Figure 3: Hours Played by Gay-Straight Alliance for Transgender Participants

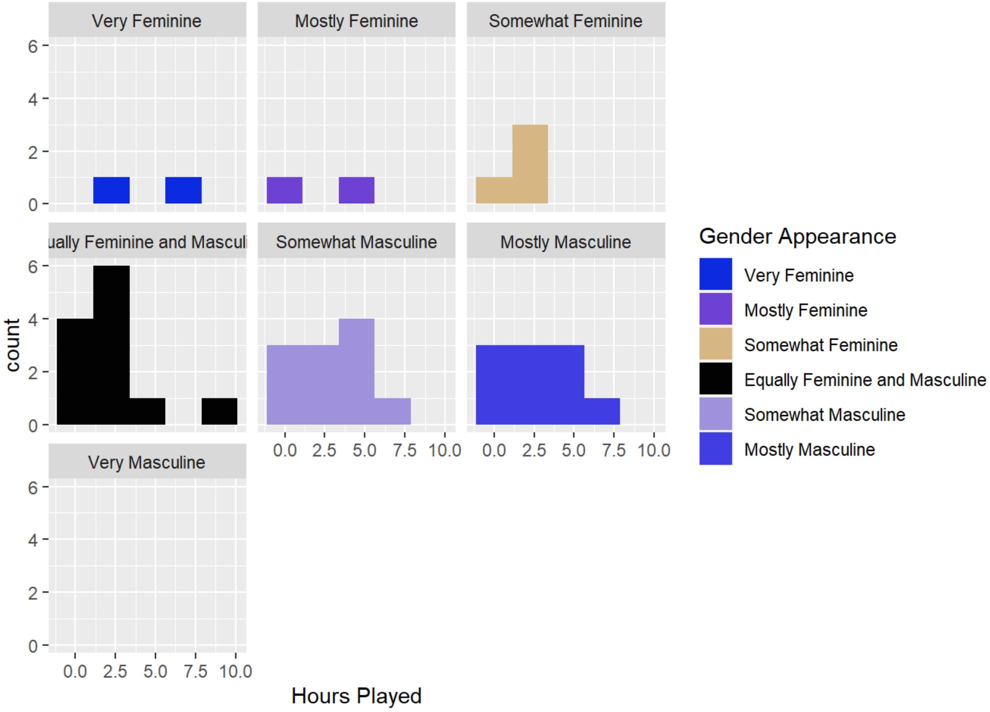

Figure 4: Hours Played by Gender Appearance for Transgender Participants

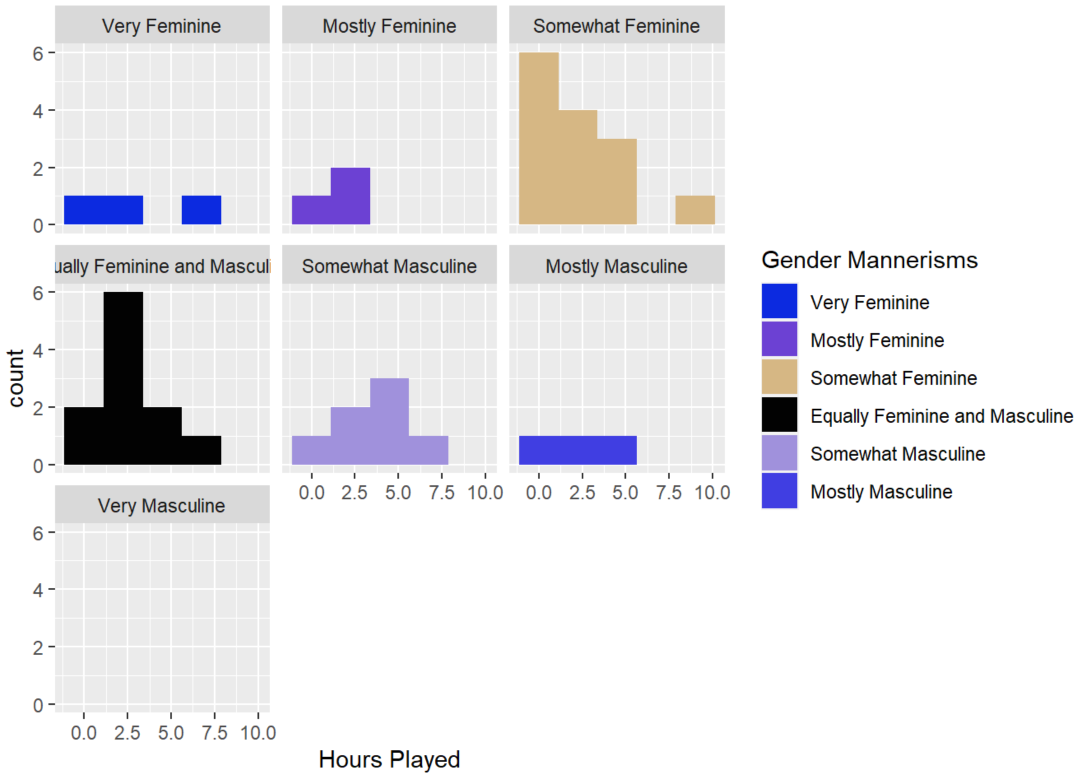

Figure 5: Hours Played by Gender Mannerisms for Transgender Participants

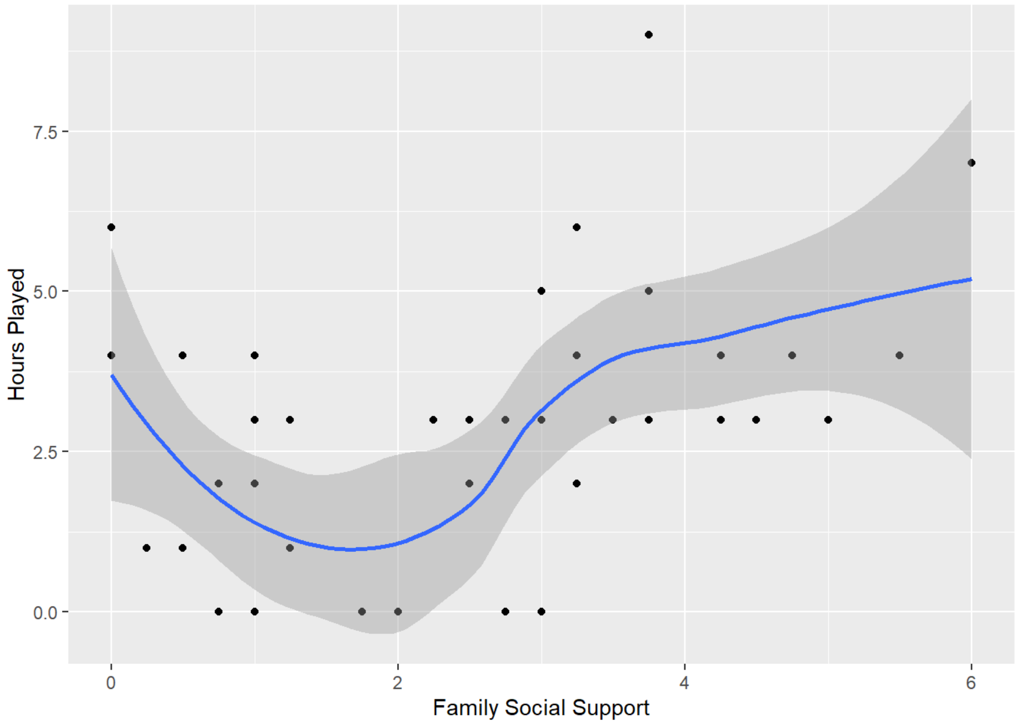

Figure 6: Hours Played by Family Social Support for Transgender Participants

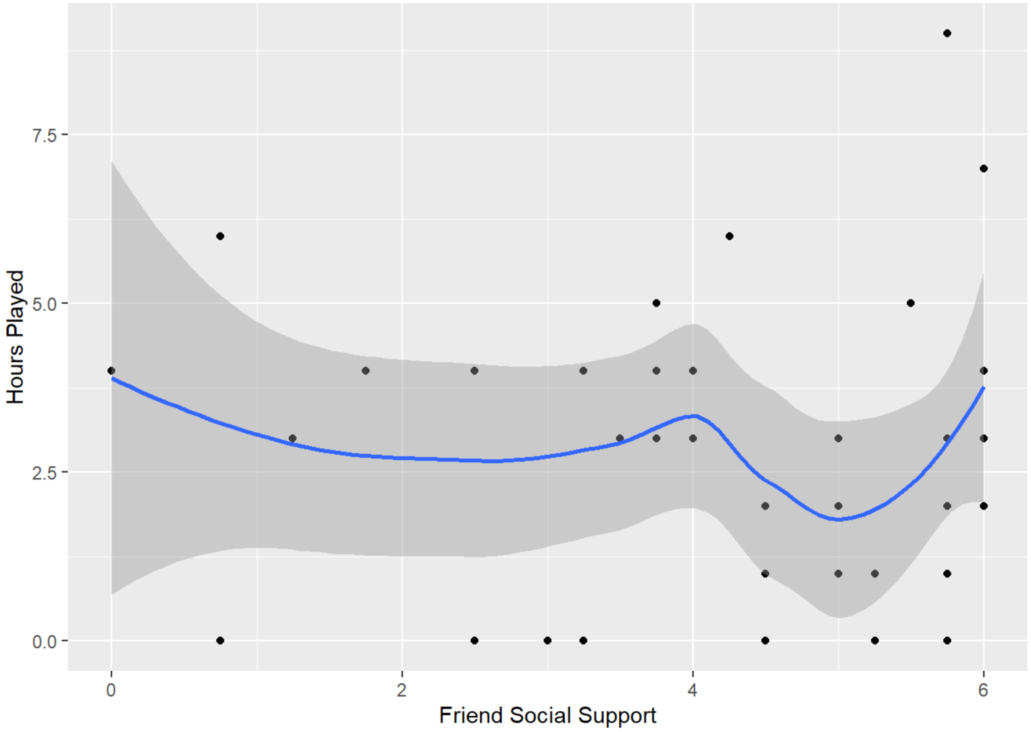

Figure 7: Hours Played by Friend Social Support for Transgender Participants

3.1.5 Completion

Table 5 summarizes completion rate among transgender participants by the covariates of interest. P-values using basic statistical tests are lowest for whether a participant had a *gay-straight alliance* and for how they believe others perceive their *gender mannerisms*.

Table 5: Completion Rate by Covariates of Interest, Trans Participants

Characteristic	***No***, *N = 27*[1]	***Yes***, *N = 14*[1]	***p-value***
Gender Mannerisms			0.13[2]
Very Feminine	3 (11%)	0 (0%)	
Mostly Feminine	2 (7.4%)	1 (7.1%)	
Somewhat Feminine	11 (41%)	3 (21%)	
Equally Feminine and Masculine	6 (22%)	5 (36%)	
Somewhat Masculine	2 (7.4%)	5 (36%)	
Mostly Masculine	3 (11%)	0 (0%)	
Very Masculine	0 (0%)	0 (0%)	
Race			0.7[2]
Nonwhite	8 (30%)	5 (36%)	
White	19 (70%)	9 (64%)	
Gender			0.6[2]
Boys	9 (33%)	7 (50%)	
Girls	3 (11%)	1 (7.1%)	
Nonbinary	15 (56%)	6 (43%)	
Gay-Straight Alliance	8 (30%)	9 (64%)	0.033[3]
Gendered Appearance			0.9[2]
Very Feminine	2 (7.4%)	0 (0%)	
Mostly Feminine	1 (3.7%)	1 (7.1%)	
Somewhat Feminine	3 (11%)	1 (7.1%)	
Equally Feminine and Masculine	7 (26%)	5 (36%)	
Somewhat Masculine	7 (26%)	4 (29%)	
Mostly Masculine	7 (26%)	3 (21%)	
Very Masculine	0 (0%)	0 (0%)	
Family Support	2.25 (1.00, 3.25)	2.88 (1.06, 3.69)	0.8[4]
Friend Support	5.00 (3.75, 5.75)	3.88 (2.75, 5.38)	0.2[4]
outness_gi_family_t1_f			0.5[2]
Everyone Knows	2 (7.4%)	2 (14%)	
Most People Know	2 (7.4%)	2 (14%)	
Some People Know	4 (15%)	2 (14%)	
A Few People Know	6 (22%)	5 (36%)	
No one Knows	13 (48%)	3 (21%)	

[1] *n (%); Median (IQR)*
[2] *Fisher's exact test*
[3] *Pearson's Chi-squared test*
[4] *Wilcoxon rank sum test*

Similar to download rate, there are many low cell counts, which informed the decision to again treat categorical variables of three or more levels as continuous or binary, as well as the implementation of exact logistic regression.

18 transgender participants were nonwhite. Figure 8 summarizes different racial groups among trans students. Most nonwhite students were LatinX or multiracial.

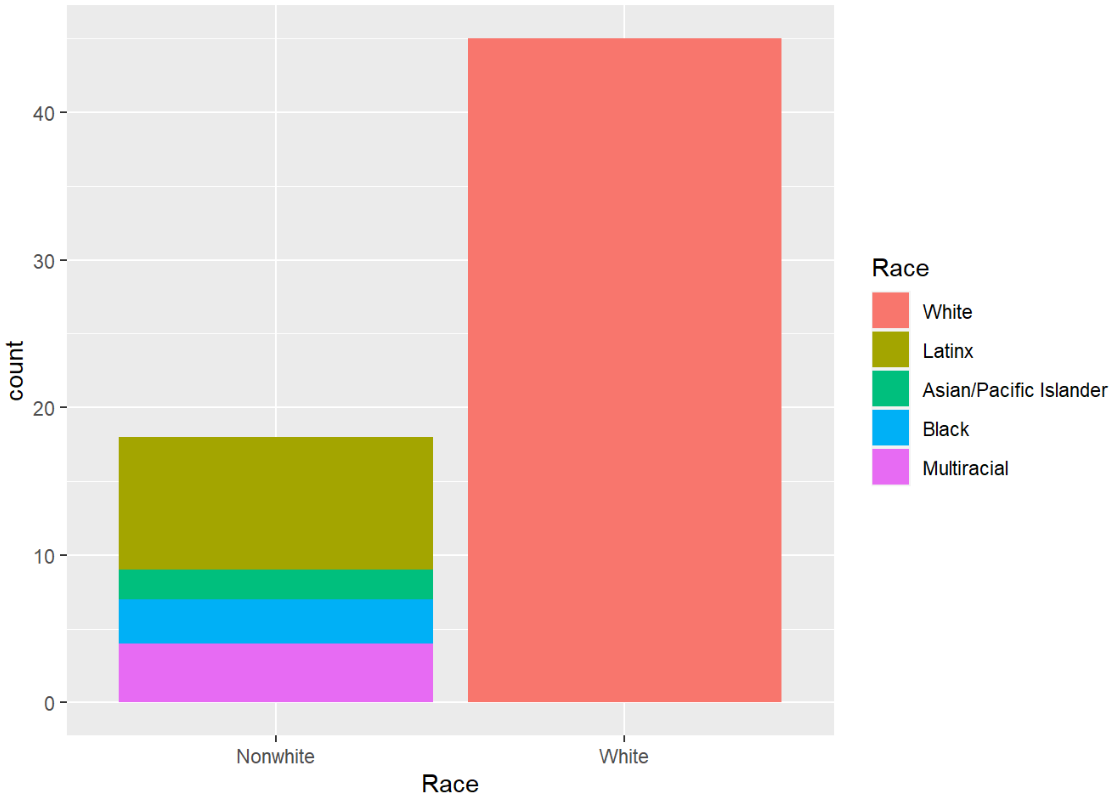

Figure 8: Race of Transgender Participants (N = 63)

3.2 Download Models

3.2.1 Logistic Regression

3.2.1.1 Transgender and Cisgender Univariate Model

In Model I, a simple logistic regression model was used to compare download rates by cisgender and transgender participants. We can see participation was nearly equal, and that modeling failed to reject the null hypothesis of participation being equal among transgender and cisgender participants. Table 6 summarizes the odds ratio (point estimate and 95% confidence interval) and p-values for Model I.

Table 6: Logistic Regression: Download by Cisgender versus Transgender Participants

Characteristic	*OR*[1]	*95% CI*[1]	*p-value*
gender_2groups			
Cisgender	—	—	
Transgender	1.09	0.37, 3.24	0.9

[1] *OR = Odds Ratio, CI = Confidence Interval*

The model has an Akaike Information Criterion (AIC) of 88.52998. There are 120 events per variable.

3.2.1.2 Transgender Covariates of Interest Model

Table 7 summarizes odds ratio estimate and confidence interval, and p-value for the *Gay-Straight-Alliance* and *Family Social Support* covariates from Model II. Gay-Straight Alliance was kept in due to significance found via likelihood ratio testing, but it suffered from high standard error and an unrealistically high point estimate, due largely to there being zero students which fell into the category of “had no alliance and did not download” (Table 8). There were 30 students with missing values for *Gay-Straight Alliance*.

Table 7: Logistic Regression: Download by Covariates of Interest

Characteristic	*OR*[1]	*95% CI*[1]	*p-value*
factor(gsa_t1_mod_labeled)			
Yes	—	—	
No	152,894,090	0.00, NA	0.994
familySS	1.42	0.83, 2.71	0.233

[1] *OR = Odds Ratio, CI = Confidence Interval*

Table 8: Gay-Straight Alliance and Download Outcome

Characteristic	*No, N = 8*[1]	*Yes, N = 33*[1]	*p-value*[2]
Gay-Straight Alliance?			0.013
Yes	8 (100%)	16 (48%)	
No	0 (0%)	17 (52%)	

[1] *n (%)*

[2] *Fisher's exact test*

The model had an Akaike Information Criterion (AIC) of 34.97

3.2.2 Exact Logistic Regression Models

Exact logistic regression fit two models: both models included both *family social support* and *gay-straight alliance* as terms; the difference was that one model specified “family social supports” as the variable of “interest” while the other specified “gay-straight alliance” as the variable of “interest”. Each variable’s calculation comes from the model which used it as the variable of interest. Table 10 summarizes the exact logistic regression models. Compared to logistic regression from maximum likelihood, the estimate for *gay-straight alliance* was relatively more stable when exact inference is used. A bounded confidence interval for the odds ratio was able to be obtained: the 95% confidence interval for gay-straight alliance was (0.775, 173.994). Exact inference found no evidence that Family Social Support scores affect download rate (p = 0.902). Tables 9 summarizes the exact logistic regression models

Table 9: Exact Logistic Regression for Download

Variable	*OR*	*CI*	*P-value*
Gay-Straight Alliance	5.616	(0.775 , 173.994)	0.07275
Family Social Support	0.906	(0.568 , 1.426)	0.90185

The p-values also differed greatly from those in non-exact logistic regression. Table 10 compares, for each variable, the odds ratios and p-values for exact and non-exact regression.

Table 10: Comparison of Exact and Non-Exact Logistic Regression (Download)

	Model	**OR**	**CI**	**p-value**
Gay-Straight Alliance	Non-Exact	152,894,090	(0, NA)	0.9
	Exact	5.616	(0.775, 173.994)	0.073
Family Social Support	Non-Exact	1.42	(0.82, 2.71)	0.2
	Exact	0.906	(0.568, 1.426)	0.901

3.3 Hours Played Models

3.3.1 Transgender and Cisgender Univariate Model

Model V used linear regression to compare hours played for cisgender and transgender participants. Table 11 summarizes the estimated coefficient, 95% confidence interval, and p-values for Model V. We can see there is not sufficient statistical evidence that hours differ among the two groups (p = 0.6).

Table 11: Linear Regression: Hours Played, Cis and Trans

Characteristic	*Beta*	*95% CI*[1]	*p-value*
gender_2groups			
Cisgender	—	—	
Transgender	0.21	-0.71, 1.1	0.6

[1] *CI = Confidence Interval*

3.3.2 Transgender Covariates of Interest Model

The final linear regression model for hours played among trans students (Model VI) used the variables Gay-Straight Alliance, Family Social Support, and outness to family. Figure 2 plots hours played by social support.

Table 12 summarizes Model VI which uses linear regression to measure covariates of interest after model selection. The variable of outness to family was treated as continuous for this model, with higher scores meaning being out to fewer family members. Having a gay straight alliance was found to be associated with playing about 1.3 more hours. A one-unit increase in Family Social Support was associated with playing about 27 minutes (0.45 hours) longer. Being out to more family members was also associated with playing longer.

Table 12: Linear Regression for Hours Played, Trans Covariates

Characteristic	*Beta*	*95% CI*[1]	*p-value*
gsa_t1_mod	1.3	0.20, 2.5	**0.026**
familySS	0.45	0.10, 0.81	**0.016**
outness_gi_family_t1	-0.32	-0.76, 0.12	0.2

[1] *CI = Confidence Interval*

3.4 Completion Models

3.4.1 Logistic Regression

3.4.1.1 Transgender and Cisgender Univariate Model

Model VII was a univariable logistic regression model that tested whether completion differed in cisgender participants compared to transgender participants. Unlike hours and participation, completion differed noticeably between the two groups. Table 13 summarizes the odds ratio (estimate and 95% confidence interval) and p-value for Model VII. We see that transgender students are estimated to be about three times as likely to complete a game with a CI between about 1 and 9. There is statistically significant difference between the two groups (p = 0.049).

Table 13: Logistic Regression, Completion, Cis and Trans

Characteristic	***OR[1]***	***95% CI[1]***	***p-value***
gender_2groups			
Cisgender	—	—	
Transgender	2.95	1.03, 9.08	**0.049**

[1] OR = Odds Ratio, CI = Confidence Interval

3.4.1.2 Transgender Covariates of Interest Model

Model VIII evaluated game completion among transgender participants by *gender mannerisms* (with higher score indicating more “perceived masculine” mannerisms) and whether or not a participant’s school had a *gay-straight alliance*. Table 14 summarizes the point estimates and confidence intervals for odds ratios, as well as p-values. We see that, controlling for gender mannerisms, gay-straight alliance was significant in determining odds of completion; participants with a gay-straight alliance in their school had 1.38 times better odds of completing the game than those who do not have a gay-straight alliance in their school.

Table 14: Logistic Regression, Completion, Trans Covariates

Characteristic	*exp(Beta)*	*95% CI*[1]	*p-value*
gender_mannerisms_t1	1.09	0.98, 1.21	0.14
gsa_t1_mod	1.38	1.05, 1.83	**0.029**

[1] CI = Confidence Interval

3.4.2 Exact Logistic Regression Model

Exact logistic regression fit two models: both models included both mannerisms and gay-straight alliance as terms; the difference was that one model specified "mannerisms" as the variable of "interest" while the other specified "gay-straight alliance" as the variable of "interest". Each variable's calculation comes from the model which used it as the variable of interest. Table 15 summarizes the exact logistic regression models. The results of fitting these models show strongly significant evidence that more masculine gender mannerisms are associated with a higher likelihood of completion. It is estimated a one unit increase towards more "perceived masculine" mannerisms results in odds of completion 1.571 times higher than the unit below.

Table 15: Exact Logistic Regression, Completion

Variable	*OR*	*CI*	*P-value*
Mannerisms	1.571	(1.111 , 2.380)	**0.0005**
Gay-Straight Alliance	0.511	(0.083, 2.533)	0.49..

Exact logistic regression calculated confidence interval for gender mannerisms that was more than five times as wide as that from non-exact logistic regression. The estimated odds ratio for mannerisms is higher than with non-exact regression. Results for *gay-straight alliance* using this MCMC exact regression method were inconclusive. Table 16 compares results obtained in the exact and non-exact methods.

Table 16: Exact and Non-Exact Logistic Regression Compared (Completion)

	Model	OR	CI	p-value
Mannerisms	Non-Exact	1.09	0.98, 1.21	0.14
	Exact	1.571	(1.111 , 2.380)	**0.0005**
Gay-Straight Alliance	Non-Exact	1.38	1.05, 1.83	**0.029**
	Exact	0.511	(0.083, 2.533)	0.49

4.0 Discussion

Given a game-based intervention aimed at LGBT youth, this thesis sought to measure how participation of transgender youth differed from participation of their cisgender peers, and to give insight into which social covariates among the following predict participation in transgender youth: *race*, *friend support, family support, gender mannerisms, gender appearance*, whether or not their school had a *gay-straight alliance,* and their *outness* to family, friends, and school. This thesis also sought to explore the potential of exact inference, specifically the Markov Chain Monte Carlo approximation of exact logistic regression in R's *elrm* package, as a tool in modeling small and wide transgender data.

Being transgender did not seem to effect whether a given student downloaded a game, or how long they played it. Download rates for cisgender LGBT and transgender LGBT youth were nearly identical, with modeling determining no significant difference, and the same was true for mean hours played. However, it was found at significance level 0.05 that transgender students were more likely to complete the game than their cisgender peers. The estimated odds ratio of 2.95 suggests the odds of completion for transgender students is almost three times higher than their cisgender LGBT peers. It is particularly interesting to consider the fact that trans students, having played about the same length of time as cis students, were more likely to complete the game. This could indicate above average skill in the area, or that their engagement with the game was more focused.

Exact logistic regression found that a student's odds of completion may be significantly impacted by whether their school has a *gay-straight alliance* that they know about ($p = 0.07$). In addition, traditional linear regression found that a student's hours played is predicted by such an alliance (p = 0.02). And finally, logistic regression using maximum likelihood found the presence of a gay-straight alliance increases the odds of a student completing the game. In all cases, the presence of such an alliance was associated with higher intervention fidelity: higher odds of download, higher average hours played, and higher odds of completion. A possible hypothesis is that the alliances help students feel that practices such as playing a queer-targeted game are less socially stigmatized, or that the alliances directly improve transgender students' psychological functioning. It is worth noting that a "gay-straight alliance" is not by name explicitly supportive of transgender individuals, but that culturally support of gay populations and support of transgender populations are often intertwined. The results suggest groups with an explicit focus on gay individuals can still be effective in improving mental health outcomes for transgender individuals. It should also be noted that, although the 2021 (Egan) study asked about *gay-straight alliances*, such groups are now often being referred to as *gender and sexuality alliances*. In any case, it appears the alliance leads to a positive behavioral outcome, and future studies of transgender adolescents' mental health and intervention fidelity should consider the presence of a gay-straight alliance in schools as a potential variable of interest.

Family Social Support was associated positively with number of hours played (p = 0.016). Results were less conclusive for odds of download. It was expected based on past studies that family support would have a role to play; such a finding is consistent with Wilson (2016), Ryan (2010), and Simons (2013). It is worth considering the role of selection bias in this study: LGBT youth and especially transgender youth who felt safe enough at home to complete the online screener, we can reasonably infer, are more likely to feel safe enough at home to download, play and complete the game. It could be hypothesized that some students with less family support have *limited windows* of internet autonomy: that is, they feel they can safely browse the web but only, for instance, when their parents are away or distracted. Such students may have downloaded the game and rushed to complete it during such limited windows. It is worth investigating further the extent to which unsupportive family restricts the time which transgender youth can spend engaging with supportive resources. Finally, the aforementioned selection bias is a common challenge faced in studies of stigmatized youth populations, and future studies should seek to explore ways in which youth can be liberated from unsupportive family units which cut off their access to healthy interventions.

Gender mannerisms was found to be a significant predictor of game completion when implementing the Markov Chain Monte Carlo approximation of exact logistic regression. For logistic regression using maximum likelihood, the result was less conclusive ($p = 0.14$). The exact inference result suggests that how masculine or feminine a transgender adolescent believes their peers perceive their gender mannerisms, may be impactful on behavioral health outcomes. It is of interest that the results suggest believing they are perceived more masculine is correlated to higher participation. It is worth noting that this population was made up of 30 non-binary individuals, 28 trans boys, and only 5 trans girls. A possible explanation therefore is that students who felt perceived in a way more closely aligned to their gender performed better. Odds of completion were specifically highest for students who reported their mannerisms were perceived as "somewhat masculine" or "equally masculine and feminine" Modeling interaction between perceived masculinity or femininity of mannerisms and a participant's gender identity was infeasible given the small sample size, but may be of interest if future studies obtain larger samples. It may also be worthwhile to directly ask participants to rate how much they feel their perceived gender mannerisms align with their identity.

Significant impact of any sort on intervention fidelity was *not* found for race, gender appearance, outness (to family, friends, or school, each tested separately), or friend social support.

5.0 Conclusion

The *Singularities* trial saw that, among LGBT youth, intervention fidelity for transgender students was as high or higher than their cisgender peers. Transgender youth were equally likely to download the game, played about the same amount of time, and were more likely to finish the game. Concerns about selection bias, however, may undermine this result. Future studies should explore paths to reaching transgender adolescents who would normally be unlikely to participate due to social risk, especially those at risk of parental abuse.

Analysis showed that, for transgender adolescents (aged 14-18), the known presence of a gay-straight alliance at their school significantly impacted participation in the *Singularities* game: odds of download and completion were higher and hours played tended to be higher as well; most models showed significance level below 0.05. Future studies may want to examine other impacts such an alliance can have on other behavioral health outcomes in transgender teens.

For small samples of transgender youth, like this one, simpler models with fewer variables are often ideal, but . A purposeful selection process that narrows the set of variables using field knowledge and conservative univariate tests, like those used by Hosmer et al (2013) can alleviate small sample problems. However, concerns about events-per-variable are likely to persist in studies that wish to examine many covariates at once. In such cases, exact inference methods can be valuable. Exact logistic regression can be implemented in R using the *elrm()* package, or in other softwares. It should be noted that most softwares, including R, use algorithms to *approximate* the exact model via a Markov Chain Monte Carlo method. However, the Metropolis-Hastings algorithm on which the *elrm* package is based has been shown to be in close agreement with exact results obtained via enumeration, and enumeration is often not possible. A potential recommendation for future implementations of exact inference is to explore techniques for using exact inference as part of the variable *selection* process, and not just for obtaining final estimates.

Appendix A R Code

File 1:

```
---
title: "Model Selection"
output:
  html_document:
    toc: yes
    toc_depth: '4'
    df_print: paged
  html_notebook:
    toc: yes
    toc_depth: 4
---
```

Hosmer's approach

The approach for model selection I am using is that outlined in David Hosmer's *Applied Logistic Regression*.

Some highlights:

-In the initial univariate step, the threshold for variable deletion is a conservative 0.25 as recommended.

This comes from work by Bendel and Afifi (1977) and by Mickey and Greenland (1989), which argue

for a conservative approach to initial deletion by demonstrating that lower thresholds

at the univariate analysis step often lead to important variables being deleted.

-In the initial multivariate step on the other hand, threshold of 0.05-0.10 (depending on assumed importance)

is used to delete. Hosmer recommends "traditional levels of statistical significance" at this step.

-Subjectivity and flexibility exists in handling variables based on existing belief of importance.

A variable that is calculated as statistically significant may be kept in in some cases, as noted when done. This is also argued for in Hosmer.

-Normally, we would as part of the process consider interaction terms. In this case, we will not explore interaction terms because of the small sample size.

~~~
## Setup

```{r}
library(lmtest)
library(tidyverse)
library(elrm)
singu_wide <- read_csv("singularities-wide.csv")

## Create trans-only dataframe for simplicity
singu_wide_t <- singu_wide %>% filter(gender_2groups == "Transgender")
nrow(singu_wide_t)

singu_wide_t <- singu_wide_t %>% mutate(gsa_t1_mod = case_when(gsa_t1 == 99 ~ 1,
                                        TRUE ~ gsa_t1))
```


```{r}
## Data transformations


singu_wide %>% group_by(game_play_t2, game_playtime_t2) %>% tally()
~~~

```
singu_wide <- singu_wide %>% mutate(game_hours_all = game_playtime_t2 + game_playtime_t3)
singu_wide <- singu_wide %>% mutate(game_hours_all =
                              case_when(is.na(game_playtime_t2) ~ game_playtime_t3,
                                        is.na(game_playtime_t3) ~ game_playtime_t2,
                                        TRUE ~ game_playtime_t2 + game_playtime_t3,)
)
singu_wide <- singu_wide %>% mutate(game_download_all = factor(case_when(is.na(game_download_t3) ~ game_download_t2,
                                              TRUE ~ game_download_t3-1), levels = c(0,1))


)


## Labels for categorical variables:
singu_wide <- singu_wide %>% mutate(free_reduced_lunch_t1 = factor(free_reduced_lunch_t1,
                          labels = c("Yes","No","Unsure")))
singu_wide <- singu_wide %>% mutate(sexId6Groups = factor(sexId6Groups,
                          labels = c("Gay or Lesbian", "Bisexual",
                                     "Queer", "Unsure",
                                     "MUltiple", "Other")))
singu_wide <- singu_wide %>% mutate(g2 = factor(gender_appearance_t1,
                          labels = c("Very feminine", "Mostly feminine",
```

```
                              "Somewhat feminine", "Equally masculine and feminine",
                              "Somewhat masculine", "Mostly masculine", "Very
masculine")))
    singu_wide <- singu_wide %>% mutate(gender_3new =
                           case_when((gender_6groups == 0) |  (gender_6groups == 3 ) ~
"Girls",
                               (gender_6groups == 1) |  (gender_6groups ==2 ) ~ "Boys",
                               (gender_6groups == 4) |  (gender_6groups == 5 ) ~
"Nonbinary"))
    singu_wide <- singu_wide %>%  mutate(gender_5 = case_when((gender_6groups == 4) |
(gender_6groups == 5 ) ~ 4,
                             TRUE ~ gender_6groups)
       )
    singu_wide <- singu_wide %>% mutate(gender_5 = factor(gender_5, labels = c("Cis girl",
"Cis boy",
                                     "Trans boy", "Trans girl",
                                     "Nonbinary")))
    singu_wide <- singu_wide %>% mutate(gender_2groups =
factor(case_when((gender_3groups == 0) | (gender_6groups == 1 ) ~ 0,
                             TRUE ~ 1), labels = c("Cisgender","Transgender"))
       )
    singu_wide <- singu_wide %>% mutate(race2 = factor(race2,
```

```
                labels = c("White","Latinx","Asian/Pacific Islander",
"Black","Multiracial")))


    ## Race, 2 levels:
    singu_wide <- singu_wide %>% mutate(race_2groups = case_when(race2 == "White" ~
"White",
                        TRUE ~ "Nonwhite"))




    ## Reverse code the download outcome:
    singu_wide$game_download_all <- factor(1-
(as.numeric(singu_wide$game_download_all)-1))

    ##completion stuff

    singu_wide <- singu_wide %>% mutate(game_complete_all =
factor(case_when(is.na(game_complete_t2) ~ game_complete_t3,
                           TRUE ~ game_complete_t2
    )))
```

```
    singu_wide <- singu_wide %>% mutate(game_complete_all_2lev=
case_when(game_complete_all == "2"~ 0,
                                game_complete_all == "99" ~ 0,
                                game_complete_all =="3" ~ 0,
                                game_download_all == 0 ~ 0,
                                game_complete_all == "1" ~ 1)
    )



    ## Labelling


    singu_wide <- singu_wide %>%
     mutate(game_download_all_labeled = factor(game_download_all, labels =
c("No","Yes")),
        game_complete_all_2lev_labeled = factor(game_complete_all_2lev,
                          labels = c("No","Yes")),
        gender_mannerisms_labeled = factor(gender_mannerisms_t1,
                           labels = c("Very Feminine",
                               "Mostly Feminine",
                               "Somewhat Feminine",
                               "Equally Feminine and Masculine",
                               "Somewhat Masculine",
                               "Mostly Masculine",
```

```
                                        "Very Masculine")),
    gender_appearance_labeled = factor(gender_appearance_t1,
                                   labels = c("Very Feminine",
                                        "Mostly Feminine",
                                        "Somewhat Feminine",
                                        "Equally Feminine and Masculine",
                                        "Somewhat Masculine",
                                        "Mostly Masculine",
                                        "Very Masculine")),
    outness_gi_family_t1_f = factor(outness_gi_family_t1,
                                  labels = c("Everyone Knows",
                                       "Most People Know",
                                       "Some People Know",
                                       "A Few People Know",
                                       "No one Knows"))
    )



## Create trans-only dataframe
singu_wide_t <- singu_wide %>% filter(gender_2groups == "Transgender")
nrow(singu_wide_t)


singu_wide_t <- singu_wide_t %>% mutate(gsa_t1_mod = case_when(gsa_t1 == 99 ~ 1,
```

```
                                   TRUE ~ gsa_t1))

singu_wide_t <- singu_wide_t %>% mutate(gsa_t1_mod_labeled = factor(gsa_t1_mod,
                                   labels = c("Yes", "No")))


## Dichotomize Friend SS for the Exact Model:

singu_wide_t <- singu_wide_t %>%
  mutate(friendSS_2levels = case_when(friendSS < 4 ~ 0,
                         TRUE ~ 1))

```

```{r}

interest <- singu_wide_t %>% select(race_2groups, race2, gender_mannerisms_t1,
            genderMannerisms_baseline, gender_appearance_labeled,
            gender_appearance_t1,                 genderAppearance_baseline,
outness_gi_family_t1,
            outness_gi_friends_t1, outness_gi_school_t1, outness_gi_family_t1_f,
            schoolclimate_1_t1, schoolclimate_2_t1, schoolclimate_3_t1,
            schoolclimate_4_t1, schoolclimate_5_t1, game_complete_all,
```

```
                    game_complete_all_2lev, game_complete_all_2lev_labeled,
                    game_download_all, game_download_all_labeled, game_hours_all,
                    friendSS, friendSS_2levels, familySS, gender_2groups, gender_3new,
                    GIstructuralStigmaBaseline, gsa_t1, gsa_t1_mod, gsa_t1_mod_labeled)
corrplot(interest %>% select(where(is.numeric)))
cor(interest %>% select(where(is.numeric)))
#interestnum <- as.data.frame(apply(interest, 2, as.numeric))
library(sjPlot)
sjp.corr(interest)
```
https://datascience.stackexchange.com/questions/893/how-to-get-correlation-between-two-categorical-variable-and-a-categorical-variab

Cis vs Trans Participation:

```{r}



## Hours
fitGender2 <- lm(game_hours_all ~ gender_2groups, data = singu_wide)
```

~~~~
lrtest(fitGender2)

## Download

fitGender2d <- glm(game_download_all~ gender_2groups, data = singu_wide,
                   family = "binomial")
lrtest(fitGender2d)

## Completion

fitGender2c <- glm(game_complete_all_2lev ~ gender_2groups, data = singu_wide,
                   family = "binomial")
lrtest(fitGender2c)

```
~~~~

Repeat the process for completion and download using exact binomial tests:

~~~~
```{r}
## Download

#gend_down_exact_tab <- xtabs(~game_download_all +
  #                            interaction(race_2groups, gender_2groups),
~~~~

```
  #                     data = singu_wide)
#gddat <- data.frame(gender = rep(0:1, 2), race = rep(0:1, each = 2),
 #                                      not_down = gend_down_exact_tab[2,], n_trials =
colSums(gend_down_exact_tab))
#model.gd_exact <-elrm(formula = not_down/n_trials ~ gender + race, interest = ~gender,
dataset = gddat,
    iter = 22000, burnIn = 2000)
#summary(model.gd_exact)


#gend_down_exact_tab <- xtabs(~game_download_all + gender_2groups,
 ##                     data = singu_wide)
##gddat <- data.frame(gender = c(0,1),
  #                                      not_down = gend_down_exact_tab[2,], n_trials =
colSums(gend_down_exact_tab))
#model.gd_exact <-elrm(formula = not_down/n_trials ~ gender, interest = ~gender, dataset
= gddat,
 #    iter = 2000, burnIn = 500)
#summary(model.gd_exact)



## Completion


gend_comp_exact_tab <- xtabs(~game_complete_all_2lev +
```

~~~~
                         interaction(race_2groups, gender_2groups),
                        data = singu_wide)
     gencompdat <- data.frame(gender = rep(0:1, 2), race = rep(0:1, each = 2),
                  not_down = gend_comp_exact_tab[2,], n_trials =
colSums(gend_comp_exact_tab))
     model.gencomp_exact <-elrm(formula = not_down/n_trials ~ gender + race, interest =
~gender, dataset = gencompdat,
       iter = 22000, burnIn = 2000)
     summary(model.gencomp_exact)



     ## may need to do some more variable selection on these... my thinking was transness
doesn't need covariates since
     ## it's unlikely to get confounded... i actually only included race for exact because it doesn't
seem to work univariate

     ```



     ## Variables affecting Trans Participation


     Up next, among trans students seeing how other variables do or don't affect gameplay
~~~~

Hours Outcome (Vanilla):

Univariate through LRT:

~~~
```{r}

hfitrace <- lm(formula = game_hours_all ~ race_2groups ,
        data = singu_wide_t)
testrace <- lrtest(hfitrace)
paste("Race:", testrace$`Pr(>Chisq)`[2])

hfitapp <- lm(formula = game_hours_all ~ genderAppearance_baseline ,
        data = singu_wide_t)
testapp <- lrtest(hfitapp)
paste("Gender Appearance:", testapp$`Pr(>Chisq)`[2])


hfitgm <- lm(formula = game_hours_all ~ genderMannerisms_baseline ,
        data = singu_wide_t)
testgm <- lrtest(hfitgm)
~~~

```
paste("Gender Mannerisms:", testgm$`Pr(>Chisq)`[2])

hfitoutfam <- lm(formula = game_hours_all ~ outness_gi_family_t1 ,
        data = singu_wide_t)
testoutfam <- lrtest(hfitoutfam)
paste("Outness to Family:", testoutfam$`Pr(>Chisq)`[2])

hfitoutfriends <- lm(formula = game_hours_all ~ outness_gi_friends_t1 ,
        data = singu_wide_t)
testoutfriends <- lrtest(hfitoutfriends)
paste("Outness to Friends:", testoutfriends$`Pr(>Chisq)`[2])

hfitoutschool <- lm(formula = game_hours_all ~ outness_gi_school_t1  ,
        data = singu_wide_t)
testoutschool <- lrtest(hfitoutschool)
paste("Outness to School:", testoutschool$`Pr(>Chisq)`[2])

hfitgsa<- lm(formula = game_hours_all ~ gsa_t1_mod ,
        data = singu_wide_t)
testgsa <- lrtest(hfitgsa)
paste("Gay-Straight Alliance:", testgsa$`Pr(>Chisq)`[2])

hfitfss<- lm(formula = game_hours_all ~ familySS ,
```

```
    data = singu_wide_t)
testfss <- lrtest(hfitfss)
paste("Family Support:", testfss$`Pr(>Chisq)`[2])



hfitfrss<- lm(formula = game_hours_all ~ friendSS ,
    data = singu_wide_t)
testfrss <- lrtest(hfitfrss)
paste("Friend Support:", testfrss$`Pr(>Chisq)`[2])

## generate school climate maybe?

```

Significant: family outness, gsa, family support

Initial Multivariable Model

```{r}
multihoursfit1 <- lm(formula = game_hours_all~ gsa_t1_mod + familySS + gsa_t1_mod
+
        outness_gi_family_t1, data = singu_wide_t)
```

~~~
summary(multihoursfit1)
```
~~~

Outness is a candidate for deletion. We first check the LRT, then for confounding.

~~~
```{r}
multihoursfit2 <- lm(game_hours_all ~ gsa_t1_mod + familySS, data = singu_wide_t)
lrtest(multihoursfit1, multihoursfit2)

(coefficients(multihoursfit1)[1:3] - coefficients(multihoursfit2)) / coefficients(multihoursfit1)[1:3]

```
~~~

The LRT weakly supports deletion and the value of familySS changes by 15 percent. Because there is strong reason to believe outness and social support could confound each other, I will leave outness in the model.

Checking on initially delete variables

~~~
```{r}
~~~

~~~
```

#### Transformations

#### Interaction Terms

### Download Outcome (Vanilla Model):

Univariate through LRT:

```{r}
dfitrace <- glm(formula = game_download_all~ race_2groups ,
        data = singu_wide_t, family = "binomial")
testrace <- lrtest(dfitrace)
paste("Race:", testrace$`Pr(>Chisq)`[2])

dfitapp <- glm(formula = game_download_all~ genderAppearance_baseline ,
        data = singu_wide_t, family = "binomial")
~~~

```
testapp <- lrtest(dfitapp)

paste("Gender Appearance:", testapp$`Pr(>Chisq)`[2])




dfitgm <- glm(formula = game_download_all~ genderMannerisms_baseline ,

        data = singu_wide_t, family = "binomial")

testgm <- lrtest(dfitgm)

paste("Gender Mannerisms:", testgm$`Pr(>Chisq)`[2])


dfitoutfam <- glm(formula = game_download_all~ outness_gi_family_t1 ,

        data = singu_wide_t, family = "binomial")

testoutfam <- lrtest(dfitoutfam)

paste("Outness to Family:", testoutfam$`Pr(>Chisq)`[2])


dfitoutfriends <- glm(formula = game_download_all~ outness_gi_friends_t1 ,

        data = singu_wide_t, family = "binomial")

testoutfriends <- lrtest(dfitoutfriends)

paste("Outness to Friends:", testoutfriends$`Pr(>Chisq)`[2])


dfitoutschool <- glm(formula = game_download_all~ outness_gi_school_t1  ,

        data = singu_wide_t, family = "binomial")

testoutschool <- lrtest(dfitoutschool)

paste("Outness to School:", testoutschool$`Pr(>Chisq)`[2])
```

```
dfitgsa<- glm(formula = game_download_all~ gsa_t1_mod ,
          data = singu_wide_t, family = "binomial")
testgsa <- lrtest(dfitgsa)
paste("Gay-Straight Alliance:", testgsa$`Pr(>Chisq)`[2])


dfitfss<- glm(formula = game_download_all~ familySS ,
          data = singu_wide_t, family = "binomial")
testfss <- lrtest(dfitfss)
paste("Family Support:", testfss$`Pr(>Chisq)`[2])



dfitfrss<- glm(formula = game_download_all~ friendSS ,
          data = singu_wide_t, family = "binomial")
testfrss <- lrtest(dfitfrss)
paste("Friend Support:", testfrss$`Pr(>Chisq)`[2])



dfitsc1<- glm(formula = game_download_all~ schoolclimate_1_t1 ,
          data = singu_wide_t, family = "binomial")
testsc1 <- lrtest(dfitsc1)
```

~~~
#dfitgissb <- glm(formula = game_download_all~ GIstructuralStigmaBaseline ,
         # data = singu_wide_t, family = "binomial")
#testgissb <- lrtest(dfitgissb)
#paste("Gender Identity Structural Stigma (by State):", testgissb$`Pr(>Chisq)`[2])
```
~~~

Significant: gsa, sc1. possibly: family support, outness to family, sc2, sc4

Dig deeper on family support:

~~~
```{r}
dfitss3<- glm(formula = game_download_all~ socialsupport_3_t1 ,
          data = singu_wide_t, family = "binomial")
testss3 <- lrtest(dfitss3)
paste("Family Support, question 3:", testss3$`Pr(>Chisq)`[2])



dfitss4<- glm(formula = game_download_all~ socialsupport_4_t1 ,
          data = singu_wide_t, family = "binomial")
testss4 <- lrtest(dfitss4)
paste("Family Support, question 4:", testss4$`Pr(>Chisq)`[2])

dfitss8<- glm(formula = game_download_all~ socialsupport_8_t1 ,
~~~

~~~
        data = singu_wide_t, family = "binomial")
testss8 <- lrtest(dfitss8)
paste("Family Support, question 8:", testss8$`Pr(>Chisq)`[2])

dfitss11<- glm(formula = game_download_all~ socialsupport_11_t1 ,
        data = singu_wide_t, family = "binomial")
testss11 <- lrtest(dfitss11)
paste("Family Support, question 11:", testss11$`Pr(>Chisq)`[2])



```

```{r}

singu_wide_t <- singu_wide_t %>%
  mutate(familySS_3 = rowMeans(select(., socialsupport_3_t1, socialsupport_4_t1,
                                      socialsupport_8_t1)))
dfitsv<- glm(formula = game_download_all~ familySS_3 ,
        data = singu_wide_t, family = "binomial")
testsv <- lrtest(dfitsv)
paste("Family Support (decision making omitted):", testsv$`Pr(>Chisq)`[2])

singu_wide_t <- singu_wide_t %>%
~~~

~~~~
  mutate(familySS_2 = rowMeans(select(.,socialsupport_4_t1,
                                          socialsupport_8_t1)))
dfitss2<- glm(formula = game_download_all~ familySS_2 ,
         data = singu_wide_t, family = "binomial")
testss2 <- lrtest(dfitss2)
paste("Family Support (4 and 8 only):", testss2$`Pr(>Chisq)`[2])

```



#### Initial Multivariable Model:


```{r}
multidownfit1 <- glm(formula = game_download_all~ gsa_t1_mod  + familySS_3 +
                    outness_gi_family_t1 , data = singu_wide_t, family = "binomial")
summary(multidownfit1)

```




```{r}
~~~~

```
## delete outness?
multidownfit3 <-glm(formula = game_download_all~ factor(gsa_t1_mod)
                    + familySS_3,
                 data = singu_wide_t, family = "binomial")
lrtest(multidownfit1, multidownfit3) # LRT supports deletion
summary(multidownfit3)

multidownfit4 <- glm(formula = game_download_all~ outness_gi_family_t1
                    + familySS_3,
                 data = singu_wide_t, family = "binomial")
lrtest(multidownfit4, multidownfit1) ## LRT opposes deletion

multidownfit6 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
                    familySS_3,
                 data = singu_wide_t, family = "binomial")
summary(multidownfit6)



## delete family support? I don't think so, too much clinical reason.
```

```

#### Check confounding for initially deleted variables:

Test to see if coefficients in multidownfit6 differ by more than 20% if initally deleted variables are added back in.

Initially deleted variables:

-race_2groups

-genderAppearance_baseline

-genderMannerisms_baseline

-outness_gi_friends_t1

-outness_gi_school_t1

-schoolclimate_5_t1

-schoolclimate_3_t1
```

-friendSS

~~~
```{r}
multidownfit7 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
                     familySS_3 + genderAppearance_baseline,
                 data = singu_wide_t, family = "binomial")



multidownfit8 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
                     familySS_3 + genderMannerisms_baseline,
                 data = singu_wide_t, family = "binomial")

multidownfit9 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
                     familySS_3 + outness_gi_friends_t1,
                 data = singu_wide_t, family = "binomial")

multidownfit10  <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
                     familySS_3 + outness_gi_school_t1,
                 data = singu_wide_t, family = "binomial")

multidownfit11 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
~~~

~~~
            familySS_3 + schoolclimate_3_t1,
          data = singu_wide_t, family = "binomial")
multidownfit12 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
            familySS_3 + schoolclimate_5_t1,
          data = singu_wide_t, family = "binomial")
multidownfit13 <- glm(formula = game_download_all~ factor(gsa_t1_mod) +
            familySS_3 + friendSS,
          data = singu_wide_t, family = "binomial")

multidownfit6$coefficients
18.92*.2
-0.39708*0.2
multidownfit6$coefficients
multidownfit7$coefficients

multidownfit8$coefficients
multidownfit9$coefficients
multidownfit10$coefficients
multidownfit11$coefficients
multidownfit12$coefficients
multidownfit13$coefficients
```
~~~

No significant changes. ``multidownfit6`` is the *preliminary effects model*.

Transformations

Interaction Terms

Download Outcome (Exact Model)

We try using Exact Logistic Regression

Univariate:

```{r}
library(elrm)

## This dataframe is too big for R: xtabs(~game_download_all+
##                    interaction(race_2groups,   genderAppearance_baseline,
genderMannerisms_baseline,
 ##          schoolclimate_1_t1, schoolclimate_2_t1, schoolclimate_3_t1,
```

```
    ##          schoolclimate_4_t1, schoolclimate_5_t1, outness_gi_family_t1,
     ##         outness_gi_friends_t1, outness_gi_school_t1, friendSS, familySS,
      ##        gsa_t1), data = singu_wide_t)
## cannot allocate vector of size 4.0 Gb
## Need a smaller model

## sc5 doesn't make as much sense. we've also seen thru univariate LRT evidence
supporting that friend social support and outness to friends aren't important to game download. sc3
as well. the weak evidence for appearance/mannerisms is my justification for swapping to
``manapp`` which combines the two. sc3 looks weak as well


## we can also make school climate a mean scale

singu_wide_t <- singu_wide_t %>% mutate(schoolclimate_avg_4 = (schoolclimate_1_t1
+
                          schoolclimate_2_t1 + schoolclimate_3_t1 +
                           schoolclimate_4_t1)/4)

#exact1 <- xtabs(~game_download_all+
 #interaction(race_2groups, gendermanapp,
     #   schoolclimate_avg_4, outness_gi_family_t1,
      #    outness_gi_school_t1, factor(familySS), factor(friendSS),
```

```
# gsa_t1_mod), data = singu_wide_t)
## length(levels(factor(singu_wide_t$familySS))) CHECK ALL LEVELS WITH
FUNCTIONS LIKE THIS EACH TIME YOU CHANGE

## Number of levels for the variables in exact1: 2, 11, 13, 5, 5, 24, 19, 2

#totalcom <- 2 * 11 * 13 * 5 * 5 * 24 * 19* 2  ## number of possible unique combinations

## for some odd reason a different length is being come up with when i reference exact1
## that length is the product of 2 * 11 * 13 * 5 * 5 * 25 * 18* 2
## try factoring to fix it, didn't work

#ddat <- data.frame(race_nonwhite = rep(0:1, each = totalcom/2),
        #gendermanapp =
rep(as.numeric(levels(factor(singu_wide_t$gendermanapp))),
       #          each = totalcom/11),
        #schoolclimate_avg_4 =
rep(as.numeric(levels(factor(singu_wide_t$schoolclimate_avg_4))),
       #              each = totalcom/13),
        #          outness_family =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_family_t1))),
             #  each = totalcom/5),
```

```
                    #                    outness_school                    =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_school_t1))),
                    #          each = totalcom/5),
                    # familySS = rep(as.numeric(levels(factor(singu_wide_t$familySS))),
                    #          each = totalcom/24),
                    # friendSS = rep(as.numeric(levels(factor(singu_wide_t$friendSS))),
                             # each = totalcom/19),
                    # gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),
                             # each = totalcom/2),
                    #  no_down = exact1[2,], n_trials = colSums(exact1)
                    # )


        ## Try it without familySS and friendSS
        ## what if a value of fr
        exact2 <- xtabs(~game_download_all+
         interaction(race_2groups, gendermanapp,
                   schoolclimate_avg_4, outness_gi_family_t1,
                    outness_gi_school_t1,
                     gsa_t1_mod), data = singu_wide_t)
        totalcom2 <-  2 * 11 * 13 * 5 * 5 * 2
        ```
```
```

~~~
```{r}

ddat2 <- data.frame(race_nonwhite = rep(0:1, each = totalcom2/2),

                    gendermanapp                                                    =
rep(as.numeric(levels(factor(singu_wide_t$gendermanapp))),

                              each = totalcom2/11),

                    schoolclimate_avg_4                                             =
rep(as.numeric(levels(factor(singu_wide_t$schoolclimate_avg_4))),

                                 each = totalcom2/13),

                    outness_family                                                  =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_family_t1))),

                               each = totalcom2/5),

                    outness_school                                                  =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_school_t1))),

                               each = totalcom2/5),

                    gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),

                               each = totalcom2/2),

                    no_down = exact2[2,], n_trials = colSums(exact2)

                    )


## This one doesn't break, but is quite small in trials per group


## Possible ways to deal with this:

## -change the discretization (lower the levels)
~~~

```
## -reduce number of variables


## changing levels (move this higher up in code if it works)
## Generated variables: gendermanapp_3levels


singu_wide_t <- singu_wide_t %>%
  mutate(gendermanapp_3levels = case_when(gendermanapp <= 2 ~ 0, #fem
                                          gendermanapp >= 4 ~ 2, #masc
                                          TRUE ~ 1 ), # neutral
         schoolclimate_avg_4_2levels = case_when(schoolclimate_avg_4 <= 2 ~ 0, # hostile
                                                 TRUE ~ 1), # friendly
         familySS_7levels = round(familySS),
         friendSS_7levels = round(friendSS)
  )


exact3 <- xtabs(~game_download_all+
 interaction(race_2groups, gendermanapp_3levels,
              schoolclimate_avg_4_2levels, outness_gi_family_t1,
                outness_gi_school_t1, factor(familySS_7levels), factor(friendSS_7levels),
                 gsa_t1_mod), data = singu_wide_t)  ## this one is almost as small as exact2
despite having more vars
## new level numbers:
## 2, 3, 2, 5, 5, 6, 6, 2
```

```
totalcom3 <- 2*3*2*5*5*7*7*2

ddat3 <- data.frame(race_nonwhite = rep(0:1, each = totalcom3/2),
          gendermanapp                                                      =
rep(as.numeric(levels(factor(singu_wide_t$gendermanapp_3levels))),
                    each = totalcom3/3),
         schoolclimate_avg_4                                                =
rep(as.numeric(levels(factor(singu_wide_t$schoolclimate_avg_4_2levels))),
                       each = totalcom3/2),
          outness_family                                                    =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_family_t1))),
                     each = totalcom3/5),
          outness_school                                                    =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_school_t1))),
                    each = totalcom3/5),
          familySS = rep(as.numeric(levels(factor(singu_wide_t$familySS_7levels))),
                    each = totalcom3/7),
          friendSS = rep(as.numeric(levels(factor(singu_wide_t$friendSS_7levels))),
                     each = totalcom3/7),
          gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),
                     each = totalcom3/2),
          no_down = exact3[2,], n_trials = colSums(exact3)
          )
```

```
## Reducing levels further

singu_wide_t <- singu_wide_t %>%
  mutate(familySS_2levels = case_when(familySS < 4 ~ 0, TRUE ~ 1),
         friendSS_2levels = case_when(friendSS < 4 ~ 0,TRUE ~ 1),
         outness_gi_family_3levels = case_when(outness_gi_family_t1 == 0 ~ 0, #universal
         outness_gi_family_t1 == 4 ~ 2, # closeted
         TRUE ~ 1),
         outness_gi_school_3levels = case_when(outness_gi_school_t1 == 0 ~ 0, #universal
                                               outness_gi_school_t1 == 4 ~ 2, # closeted
                                               TRUE ~ 1)) # partial

exact4 <- xtabs(~game_download_all+
 interaction(race_2groups, gendermanapp_3levels,
              schoolclimate_avg_4_2levels, outness_gi_family_3levels,
                outness_gi_school_3levels,                         factor(familySS_2levels),
factor(friendSS_2levels),
                 gsa_t1_mod), data = singu_wide_t)

totalcom4 <- 2*3*2*3*3*2*2*2  # this number is acutally cited in the str() description of
exact4. seems retrievable
```

```
        ddat4 <- data.frame(race_nonwhite = rep(0:1, each = totalcom4/2),
                    gendermanapp                                         =
rep(as.numeric(levels(factor(singu_wide_t$gendermanapp_3levels))),
                                each = totalcom4/3),
                   schoolclimate_avg_4                                   =
rep(as.numeric(levels(factor(singu_wide_t$schoolclimate_avg_4_2levels))),
                                    each = totalcom4/2),
                    outness_family                                       =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_family_3levels))),
                                 each = totalcom4/3),
                    outness_school                                       =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_school_3levels))),
                                each = totalcom4/3),
                    familySS = rep(as.numeric(levels(factor(singu_wide_t$familySS_2levels))),
                                each = totalcom4/2),
                    friendSS = rep(as.numeric(levels(factor(singu_wide_t$friendSS_2levels))),
                                 each = totalcom4/2),
                    gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),
                                 each = totalcom4/2),
                    no_down = exact4[2,], n_trials = colSums(exact4)
                    )
```

```
ddat4 %>% filter(n_trials > 0)



## finally, reduce both levels and variables. here we take the last one, and remove
## friendSS and race and school outness and manapp
## all of them had bad LRT values

exact5 <- xtabs(~game_download_all+
 interaction(schoolclimate_avg_4_2levels, outness_gi_family_3levels,
                factor(familySS_2levels), gsa_t1_mod), data = singu_wide_t)

## this model is nice and small

ddat5 <- data.frame(
               schoolclimate_avg_4                                    =
rep(as.numeric(levels(factor(singu_wide_t$schoolclimate_avg_4_2levels))),
                         24/2),
                outness_family                                        =
rep(as.numeric(levels(factor(singu_wide_t$outness_gi_family_3levels))),
                       each = 24/3),
                familySS = rep(as.numeric(levels(factor(singu_wide_t$familySS_2levels))),
                      24/2),
                gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),
```

```
                    each = 24/2),
          no_down = exact4[2,], n_trials = colSums(exact4)
          )

## but even so there are mistakes, it doesn't actually capture all unique combinations

## will we need to conclude that elrm can only be down for very simple combinations? (1 or 2 input variables)

ddat5 %>% filter(n_trials > 0)

## Even so, n_trials is 0 or 1 for most combinations

## the model may get smaller as we move further in selection
library(elrm)
fit_down_exact_dat5 <- elrm(formula = no_down/n_trials ~ schoolclimate_avg_4 +
                outness_family + familySS + gsa, interest = ~schoolclimate_avg_4 +
                outness_family + familySS + gsa, iter = 22000,
  dataset = ddat5 %>% filter(n_trials != 0), burnIn = 2000)

summary(fit_down_exact_dat5)
## some debate to be had over what level of reduction is necessary. given how stringent i was just now, it
```

```
## would be consistent to change schoolclimate to only include questions 1 2 and 4 (not 3)


fit_down_exact_dat5_climate <-elrm(formula = no_down/n_trials ~ schoolclimate_avg_4
,
                interest = ~schoolclimate_avg_4, iter = 22000,
  dataset = ddat5 %>% filter(n_trials != 0), burnIn = 2000)  ## It goes degenerate unless
we remove all others
summary(fit_down_exact_dat5_climate)


fit_down_exact_dat5_support <-elrm(formula = no_down/n_trials ~ familySS,
                interest = ~familySS, iter = 22000,
  dataset = ddat5 %>% filter(n_trials != 0), burnIn = 2000)  ## It goes degenerate unless
we remove all others
summary(fit_down_exact_dat5_support)

fit_down_exact_dat5_gsa <-elrm(formula = no_down/n_trials ~ gsa + outness_family,
                interest = ~gsa, iter = 22000,
  dataset = ddat5 %>% filter(n_trials != 0), burnIn = 2000)  ## It goes degenerate unless
we remove all others
summary(fit_down_exact_dat5_gsa)
```

```
## is the degenerate error because combinations are never > 1? in that case it would most likely be avoided by reducing levels
```

```
## Error in data.frame(race_nonwhite = rep(0:1, each = totalcom/2), gendermanapp = rep(as.numeric(levels(factor(singu_wide_t$gendermanapp))), :
##arguments imply differing number of rows: 6520800, 6435000
## likely an issue from zeros...
```

```
## Exact Modeling with just the 2 variables the selection process gave us:

exact6 <- xtabs(~game_download_all+
 interaction(gsa_t1_mod, familySS), data = singu_wide_t)

ddat6 <- data.frame(
        familySS = rep(as.numeric(levels(factor(singu_wide_t$familySS))),
                2),
        gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),
                 each = 24),
        no_down = exact6[2,], n_trials = colSums(exact6)
        )
```

~~~~
fit_down_exact_dat6_gsa<- elrm(formula =  no_down/n_trials ~ gsa + familySS,
                        interest = ~gsa, iter = 22000,
   dataset = ddat6 %>% filter(n_trials != 0), burnIn = 2000)
summary(fit_down_exact_dat6_gsa)

fit_down_exact_dat6_family<- elrm(formula =  no_down/n_trials ~ gsa + familySS,
                        interest = ~familySS, iter = 22000,
   dataset = ddat6 %>% filter(n_trials != 0), burnIn = 2000)
summary(fit_down_exact_dat6_family)


```



### Completion Outcome (Vanilla Model):


```{r}
cfitrace <- glm(formula = game_complete_all_2lev ~ race_2groups ,
       data = singu_wide_t, family = "binomial")
testrace <- lrtest(cfitrace)
paste("Race:", testrace$`Pr(>Chisq)`[2])
~~~~

```
cfitapp <- glm(formula = game_complete_all_2lev ~ genderAppearance_baseline ,
        data = singu_wide_t, family = "binomial")
testapp <- lrtest(cfitapp)
paste("Gender Appearance:", testapp$`Pr(>Chisq)`[2])



cfitgm <- glm(formula = game_complete_all_2lev ~ genderMannerisms_baseline ,
        data = singu_wide_t, family = "binomial")
testgm <- lrtest(cfitgm)
paste("Gender Mannerisms:", testgm$`Pr(>Chisq)`[2])


cfitoutfam <- glm(formula = game_complete_all_2lev ~ outness_gi_family_t1 ,
        data = singu_wide_t, family = "binomial")
testoutfam <- lrtest(cfitoutfam)
paste("Outness to Family:", testoutfam$`Pr(>Chisq)`[2])


cfitoutfriends <- glm(formula = game_complete_all_2lev ~ outness_gi_friends_t1 ,
        data = singu_wide_t, family = "binomial")
testoutfriends <- lrtest(cfitoutfriends)
paste("Outness to Friends:", testoutfriends$`Pr(>Chisq)`[2])


cfitoutschool <- glm(formula = game_complete_all_2lev ~ outness_gi_school_t1  ,
```

~~~
          data = singu_wide_t, family = "binomial")
testoutschool <- lrtest(cfitoutschool)
paste("Outness to School:", testoutschool$`Pr(>Chisq)`[2])


cfitgsa<- glm(formula = game_complete_all_2lev ~ gsa_t1_mod ,
          data = singu_wide_t, family = "binomial")
testgsa <- lrtest(cfitgsa)
paste("Gay-Straight Alliance:", testgsa$`Pr(>Chisq)`[2])


cfitfss<- glm(formula = game_complete_all_2lev ~ familySS ,
          data = singu_wide_t, family = "binomial")
testfss <- lrtest(cfitfss)
paste("Family Support:", testfss$`Pr(>Chisq)`[2])



cfitfrss<- glm(formula = game_complete_all_2lev ~ friendSS ,
          data = singu_wide_t, family = "binomial")
testfrss <- lrtest(cfitfrss)
paste("Friend Support:", testfrss$`Pr(>Chisq)`[2])


```
~~~

significant: possibly: race, gender mannerisms, gsa, friend support, outness to family.

Initial Multivariable model

~~~
```{r}
multicompfit1 <- glm(game_complete_all_2lev ~ friendSS +
                     gender_mannerisms_t1 + gsa_t1_mod + race_2groups +
                     outness_gi_family_t1, data = singu_wide_t)
summary(multicompfit1)

```
~~~

Wald tests suggest family outness can be deleted. Check the LRT and confounding

~~~
```{r}
multicompfit2 <- glm(game_complete_all_2lev ~ friendSS +
                     gender_mannerisms_t1 + gsa_t1_mod + race_2groups,
                     data = singu_wide_t)
lrtest(multicompfit2, multicompfit1)
~~~

~~~
coefficients(multicompfit2)
coefficients(multicompfit1)


summary(multicompfit2)
```
~~~

Good to delete. Next candidate for deletion is race.

~~~
```{r}
multicompfit3 <-glm(game_complete_all_2lev ~ friendSS +
                    gender_mannerisms_t1 + gsa_t1_mod, data = singu_wide_t)
lrtest(multicompfit3, multicompfit2)


coefficients(multicompfit2)
coefficients(multicompfit3)
```
~~~

Also good to delete.

~~~
```{r}
summary(multicompfit3)
```
~~~

~~~
```{r}
multicompfit4 <- glm(game_complete_all_2lev ~
                  gender_mannerisms_t1 + gsa_t1_mod, data = singu_wide_t)
lrtest(multicompfit3, multicompfit4)
summary(multicompfit4)
multicompfit5 <- glm(game_complete_all_2lev ~ gsa_t1_mod, data = singu_wide_t)
lrtest(multicompfit4, multicompfit5)

```
~~~

checking confounding in initially deleted variables

```
[1] "Race: 0.803568866687939"
[1] "Gender Appearance: 0.956594641595544"
[1] "Gender Mannerisms: 0.16003864957218"
[1] "Outness to Family: 0.321733603733969"
[1] "Outness to Friends: 0.312663223758987"
[1] "Outness to School: 0.684321018313546"
[1] "Gay-Straight Alliance: 0.20529461933293"
[1] "Family Support: 0.790686918875945"
[1] "Friend Support: 0.0754456418420364"
```

~~~
```{r}
multicompfit2$coefficients
glm(game_complete_all_2lev ~ friendSS + gender_mannerisms_t1 + race_2groups,
        data = singu_wide_t)$coefficients
glm(game_complete_all_2lev ~ friendSS + gender_mannerisms_t1 + outness_gi_family_t1,
        data = singu_wide_t)$coefficients
glm(game_complete_all_2lev ~ friendSS + gender_mannerisms_t1 + outness_gi_friends_t1,
        data = singu_wide_t)$coefficients
glm(game_complete_all_2lev ~ friendSS + gender_mannerisms_t1 + outness_gi_school_t1,
        data = singu_wide_t)$coefficients
glm(game_complete_all_2lev ~ friendSS + gender_mannerisms_t1 + gsa_t1_mod,
        data = singu_wide_t)$coefficients
glm(game_complete_all_2lev ~ friendSS + gender_mannerisms_t1 + familySS,
        data = singu_wide_t)$coefficients

```
~~~

No large changes in coefficient values.

~~~
#### transformations



### Exact Model for Completion



```{r}


### To make this model work, we have to collapse friendSS to a categorical variable


exactComp1 <- xtabs(~game_complete_all_2lev + interaction(gender_mannerisms_t1,
                                          gsa_t1_mod),
          data = singu_wide_t)
ddatComp1 <- data.frame(gay_straight = rep(c(0,1), 7),
            mannerisms = rep(c(0:6), each = 2),
            not_down = exactComp1[2,], n_trials = colSums(exactComp1)
)
ddatComp1
~~~

~~~
fit_comp_exact_d1 <- elrm(formula = not_down/n_trials ~ gay_straight + mannerisms,
                    interest = ~gay_straight + mannerisms, iter = 22000,
                    dataset = ddatComp1 %>% filter(n_trials > 0), burnIn = 2000)
summary(fit_comp_exact_d1)
fit_comp_exact_d2 <- elrm(formula = not_down/n_trials ~ gay_straight + mannerisms,
                    interest = ~gay_straight , iter = 22000,
                    dataset = ddatComp1 %>% filter(n_trials > 0), burnIn = 2000)
summary(fit_comp_exact_d2)

```
~~~

(new file)

```
---
title: "Results"
author: "Violet Nova Hecmanczuk"
date: "2023-03-13"
output:
  html_document:
    toc: yes
    df_print: paged
  pdf_document:
    toc: yes
indent: yes
header-includes: \usepackage{indentfirst}
```

~~~~
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(message = FALSE, echo = FALSE, warning = FALSE)
```

```{r}

library(tidyverse)
library(caret)
library(ggplot2)
library(paletteer)
library(peRReo)
library(mice)
library(gtable)
library(elrm)
library(lmtest)
library(gtsummary)
library(gt)
library(webshot)


singu <- read_csv("data16_DataOutput_ForViolet.csv")
~~~~

~~~
```

```{r}
## Data transformations

singu_t1 <- singu %>% filter(time == 1) %>% select(-contains("t2")) %>% select(-contains("t3"))
singu_t2 <- singu %>% filter(time == 2) %>% select(-contains("t1")) %>% select(-contains("t3"))
singu_t3 <- singu %>% filter(time == 3) %>% select(-contains("t1")) %>% select(-contains("t2"))
singu_12 <- inner_join(singu_t1, singu_t2, by = 'record_id') %>% select(-contains('.y'))
singu_wide <- inner_join(singu_12, singu_t3, by = 'record_id') %>% select(-contains('.y')) %>% select(-contains('.x'))


singu_wide <- singu_wide %>% mutate(game_hours_all = game_playtime_t2 + game_playtime_t3)
singu_wide <- singu_wide %>% mutate(game_hours_all =
                          case_when(is.na(game_playtime_t2) ~ game_playtime_t3,
~~~

```
                                        is.na(game_playtime_t3) ~ game_playtime_t2,
                                        TRUE ~ game_playtime_t2 + game_playtime_t3,)
      )
      singu_wide <- singu_wide %>% mutate(game_download_all =
factor(case_when(is.na(game_download_t3) ~ game_download_t2,
                                            TRUE ~ game_download_t3-1), levels = c(0,1))


      )


      ## Labels for categorical variables:
      singu_wide <- singu_wide %>% mutate(free_reduced_lunch_t1 =
factor(free_reduced_lunch_t1,
                                 labels = c("Yes","No","Unsure")))
      singu_wide <- singu_wide %>% mutate(sexId6Groups = factor(sexId6Groups,
                                 labels = c("Gay or Lesbian", "Bisexual",
                                       "Queer", "Unsure",
                                       "MUltiple", "Other")))
      singu_wide <- singu_wide %>% mutate(g2 = factor(gender_appearance_t1,
                                 labels = c("Very feminine", "Mostly feminine",
                                       "Somewhat feminine", "Equally masculine and feminine",
                                       "Somewhat masculine", "Mostly masculine", "Very
masculine")))
      singu_wide <- singu_wide %>% mutate(gender_3new =
```

```
                                        case_when((gender_6groups == 0) |  (gender_6groups == 3 ) ~
"Girls",
                                              (gender_6groups == 1) |  (gender_6groups ==2 ) ~ "Boys",
                                              (gender_6groups == 4) |  (gender_6groups == 5 ) ~
"Nonbinary"))
       singu_wide <- singu_wide %>%  mutate(gender_5 = case_when((gender_6groups == 4) |
(gender_6groups == 5 ) ~ 4,
                                           TRUE ~ gender_6groups)
            )
       singu_wide <- singu_wide %>% mutate(gender_5 = factor(gender_5, labels = c("Cis girl",
"Cis boy",
                                                        "Trans boy", "Trans girl",
                                                        "Nonbinary")))
       singu_wide <- singu_wide %>% mutate(gender_2groups =
factor(case_when((gender_3groups == 0) | (gender_6groups == 1 ) ~ 0,
                                           TRUE ~ 1), labels = c("Cisgender","Transgender"))
            )
       singu_wide <- singu_wide %>% mutate(race2 = factor(race2,
                    labels = c("White","Latinx","Asian/Pacific Islander",
"Black","Multiracial")))



       ## Race, 2 levels:
```

```
    singu_wide <- singu_wide %>% mutate(race_2groups = case_when(race2 == "White" ~
"White",
                                  TRUE ~ "Nonwhite"))




    ## Reverse code the download outcome:
    singu_wide$game_download_all <- factor(1-
(as.numeric(singu_wide$game_download_all)-1))

    ##completion stuff

    singu_wide <- singu_wide %>% mutate(game_complete_all =
factor(case_when(is.na(game_complete_t2) ~ game_complete_t3,
                                     TRUE ~ game_complete_t2
    )))

    singu_wide <- singu_wide %>% mutate(game_complete_all_2lev=
case_when(game_complete_all == "2"~ 0,
                                     game_complete_all == "99" ~ 0,
                                     game_complete_all =="3" ~ 0,
                                     game_download_all == 0 ~ 0,
```

```
                       game_complete_all == "1" ~ 1)
    )


    ## Labelling


    singu_wide <- singu_wide %>%
     mutate(game_download_all_labeled = factor(game_download_all, labels =
c("No","Yes")),
       game_complete_all_2lev_labeled = factor(game_complete_all_2lev,
                  labels = c("No","Yes")),
       gender_mannerisms_labeled = factor(gender_mannerisms_t1,
                   labels = c("Very Feminine",
                      "Mostly Feminine",
                      "Somewhat Feminine",
                      "Equally Feminine and Masculine",
                      "Somewhat Masculine",
                      "Mostly Masculine",
                      "Very Masculine")),
       gender_appearance_labeled = factor(gender_appearance_t1,
                   labels = c("Very Feminine",
                      "Mostly Feminine",
                      "Somewhat Feminine",
```

```
                                      "Equally Feminine and Masculine",
                                      "Somewhat Masculine",
                                      "Mostly Masculine",
                                      "Very Masculine")),
      outness_gi_family_t1_f = factor(outness_gi_family_t1,
                            labels = c("Everyone Knows",
                                    "Most People Know",
                                    "Some People Know",
                                    "A Few People Know",
                                    "No one Knows"))
      )



## Create trans-only dataframe
singu_wide_t <- singu_wide %>% filter(gender_2groups == "Transgender")


singu_wide_t <- singu_wide_t %>% mutate(gsa_t1_mod = case_when(gsa_t1 == 99 ~ 1,
                                           TRUE ~ gsa_t1))


singu_wide_t <- singu_wide_t %>% mutate(gsa_t1_mod_labeled = factor(gsa_t1_mod,
                                         labels = c("Yes", "No")))
```

~~~
## Table 1: Descriptive Statistics for Outcome and Gender Minority Status (All Students)

```{r}


singu_wide %>% select(game_download_all_labeled, game_complete_all_2lev_labeled,
                 game_hours_all)  %>%
  tbl_summary(label = c(game_download_all_labeled ~ "Game Download",
game_complete_all_2lev_labeled ~ "Completed Game?"),
               type = c("game_hours_all" ~ "continuous",
                 "game_complete_all_2lev_labeled" ~ "categorical",
                 'game_download_all_labeled' ~ "categorical")) %>% bold_labels()


singu_wide %>% filter(gender_2groups == "Cisgender") %>%
select(game_download_all_labeled, game_complete_all_2lev_labeled,
                 game_hours_all)  %>%
  tbl_summary(label = c(game_download_all_labeled ~ "Game Download",
game_complete_all_2lev_labeled ~ "Completed Game?"),
               type = c("game_hours_all" ~ "continuous",
~~~

~~~
                              "game_complete_all_2lev_labeled" ~ "categorical",
                              'game_download_all_labeled' ~ "categorical")) %>% bold_labels()



            singu_wide %>% select(game_download_all_labeled, game_complete_all_2lev_labeled,
                              game_hours_all, gender_2groups)  %>%
              tbl_summary(label = c(game_download_all_labeled ~ "Game Download",
                              game_complete_all_2lev_labeled ~ "Completed Game?"),
                       value = c(game_download_all_labeled ~ "{No} {Yes}",
                              game_complete_all_2lev_labeled ~ "{No} {Yes}"),
                       type = c(all_of(c("game_download_all_labeled",
"game_complete_all_2lev_labeled",
                                 "gender_2groups")) ~ "categorical",
                              "game_hours_all" ~ "continuous")) %>% bold_labels()



            ```
~~~

To predict participation among trans students, a second dataframe was made, filtering down to only trans students (n = 63).

Table 2: Univariate Statistics for Covariates of Interest Among Trans Participants

Download:

~~~
```{r}
singu_wide_t %>% select(game_download_all_labeled, gender_mannerisms_labeled,
                        race_2groups, gender_3new, gsa_t1_mod,
                        gender_appearance_labeled, familySS, friendSS,
                        outness_gi_family_t1_f) %>%
  tbl_summary(label = c(game_download_all_labeled ~ "Game Download",
                          race_2groups ~ "Race",
                          gender_mannerisms_labeled ~ "Gender Mannerisms",
                          gender_3new = "Gender",
                          gsa_t1_mod = "Gay-Straight Alliance",
                          gender_appearance_labeled = "Gendered Apperance",
                          familySS = "Family Support",
                          friendSS = "Friend Support",
                          outness_gi_family_t1 = "Outness to Family"),
~~~

~~~
          type = "game_download_all_labeled" ~ "categorical",
          by = game_download_all_labeled
          ) %>% add_p() %>% bold_labels() %>% separate_p_footnotes()
```
~~~

Hours Played

~~~
```{r}
singu_wide_t %>% select(game_hours_all, gender_mannerisms_labeled,
                race_2groups, gender_3new, gsa_t1_mod,
                gender_appearance_labeled, familySS, friendSS,
                outness_gi_family_t1_f) %>%
  tbl_summary(label = c(game_hours_all ~ "Game Download",
                 race_2groups ~ "Race",
                 gender_mannerisms_labeled ~ "Gender Mannerisms",
                 gender_3new = "Gender",
                 gsa_t1_mod = "Gay-Straight Alliance",
                 gender_appearance_labeled = "Gendered Apperance",
                 familySS = "Family Support",
                 friendSS = "Friend Support",
                 outness_gi_family_t1 = "Outness to Family"),
          type ="game_hours_all" ~ "continuous",
~~~

~~~~
        by = game_hours_all
        ) %>% add_p() %>% bold_labels() %>% separate_p_footnotes()
```
~~~~

~~~~
```{r}

g0 <- ggplot(singu_wide_t) + geom_histogram(aes(x = game_hours_all), bins = 10) + xlab("Hours Played")
g0

natti2 <- latin_palette("karolg", n = 2)
rauw3 <- latin_palette("rauw", n = 3)
becky2 <- latin_palette("beckyg", n = 2)
palIvy5 <- latin_palette("ivyqueen", n = 5)
gyal7 <- latin_palette('badgyal', n = 7)
karol7 <- latin_palette("karolg", n = 7)


g0 + facet_grid(~race_2groups) + aes(fill = factor(race_2groups)) +
  scale_fill_manual(values = natti2, name = "Race")
~~~~

```
g0 + facet_grid(~gender_3new) + aes(fill = factor(gender_3new)) +
  scale_fill_manual(values = rauw3, name = "Gender")

g0 + facet_grid(~gsa_t1_mod_labeled) + aes(fill = factor(gsa_t1_mod_labeled)) +
  scale_fill_manual(values = becky2, name = "Gay Straight Alliance")

g0 + facet_wrap(~outness_gi_family_t1_f) +
  aes(fill = factor(outness_gi_family_t1_f)) +
  scale_fill_manual(values = palIvy5, name = "Outness to Family")



ggplot(singu_wide_t) +  geom_histogram(aes(x = game_hours_all), bins = 5) +
  facet_wrap(~gender_appearance_labeled) + xlab("Hours Played") +
  aes(fill = gender_appearance_labeled) +
  scale_fill_manual(values = gyal7, name = "Gender Appearance")

ggplot(singu_wide_t) +  geom_histogram(aes(x = game_hours_all), bins = 5) +
    facet_wrap(~gender_mannerisms_labeled) +
  xlab("Hours Played") + aes(fill = gender_mannerisms_labeled) +
  scale_fill_manual(values = gyal7, name = "Gender Mannerisms")
```

~~~
ggplot(singu_wide_t) + aes(x = familySS, y = game_hours_all) + geom_point() +
  xlab("Family Social Support") + ylab("Hours Played") + geom_smooth()

ggplot(singu_wide_t) + aes(x = friendSS, y = game_hours_all) + geom_point() +
  xlab("Friend Social Support") + ylab("Hours Played") + geom_smooth()

```


Completion

```{r}
singu_wide_t %>% select(game_complete_all_2lev_labeled,
gender_mannerisms_labeled,
                      race_2groups, gender_3new, gsa_t1_mod,
                      gender_appearance_labeled, familySS, friendSS,
                      outness_gi_family_t1_f) %>%
  tbl_summary(label = c(game_complete_all_2lev_labeled ~ "Completed Game?",
                        race_2groups ~ "Race",
                        gender_mannerisms_labeled ~ "Gender Mannerisms",
                        gender_3new = "Gender",
                        gsa_t1_mod = "Gay-Straight Alliance",
                        gender_appearance_labeled = "Gendered Apperance",
~~~

~~~
              familySS = "Family Support",
              friendSS = "Friend Support",
              outness_gi_family_t1 = "Outness to Family"),
        type = "game_complete_all_2lev_labeled" ~ "categorical",
        by = game_complete_all_2lev_labeled
        ) %>% add_p() %>% bold_labels() %>% separate_p_footnotes()

ggplot(singu_wide) +
  aes(x = game_complete_all_2lev_labeled, fill = game_complete_all_2lev_labeled) +
  geom_bar() + xlab("Completed Game") + scale_fill_discrete(name = "Completed Game?") +
  facet_grid(~gender_2groups)

```
~~~

separate tables

~~~
```{r}
singu_wide_t %>% select(gender_mannerisms_labeled,
            race_2groups, gender_3new, gsa_t1_mod,
            gender_appearance_labeled, familySS, friendSS,
            outness_gi_family_t1_f)  %>%
~~~

```
  tbl_summary(label = c(
                    race_2groups ~ "Race",
                    gender_mannerisms_labeled ~ "Gender Mannerisms",
                    gender_3new = "Gender", gsa_t1_mod = "Gay-Straight Alliance",
                    gender_appearance_labeled = "Gendered Apperance",
                    familySS = "Family Support",
                    friendSS = "Friend Support",
                    outness_gi_family_t1 = "Outness to Family"
                    ),
          type = c(all_of(c(
                          "gender_mannerisms_labeled", "race_2groups",
                          "gender_3new",
                          "gender_appearance_labeled")) ~ "categorical",
                   all_of(c("familySS",
                         "friendSS")) ~ "continuous")) %>% bold_labels()



## Support and Outness


singu_wide_t %>% select(familySS, friendSS,
                  outness_gi_family_t1_f, gsa_t1_mod_labeled)  %>%
  tbl_summary(label = c(
                    familySS = "Family Support",
```

```
                    friendSS = "Friend Support",

                    outness_gi_family_t1_f = "Outness to Family",

                    gsa_t1_mod_labeled = "Gay Straight Alliance? (Yes)"

                    ),

            type = c(all_of(c('familySS','friendSS')) ~ "continuous",

                   'outness_gi_family_t1_f' ~ "categorical")

            ) %>% bold_labels()


## Mannerisms, Appearance

singu_wide_t %>% select(gender_mannerisms_labeled,

                  gender_appearance_labeled)  %>%

  tbl_summary(label = c(

                    gender_mannerisms_labeled ~ "Gender Mannerisms",

                    gender_appearance_labeled = "Gendered Apperance"

                    ),

            type = everything() ~ "categorical"

            ) %>% bold_labels()


## Race and Gender

singu_wide_t %>% select(race_2groups, gender_3new)  %>%

  tbl_summary(label = c(race_2groups ~ "Race",

                    gender_3new = "Gender"

                    ),
```

~~~
        type = everything() ~ "categorical"
        ) %>% bold_labels()

```

## Figure 1: Trans Participants by Race

```{r}
ggplot(singu_wide_t, aes(race_2groups, fill = factor(race2))) + geom_bar() +
  xlab("Race")  + scale_fill_discrete(name = "Race")
```

## Logistic Regression Models for Download

### Table 3: Univariable Logistic Regression Model: Comparing Download (1 = Yes) for Transgender and Cisgender Participants
```{r}
fitGender2 <- glm(game_download_all~ gender_2groups, data = singu_wide,
~~~

~~~
                   family = "binomial")
fitGender2 %>% tbl_regression(exponentiate = TRUE) %>% bold_labels()
paste("AIC:", fitGender2$aic)
```





### Table 4: Univariable Models for Download Covariates (Trans participants only)


```{r}
#library(gridExtra)
#jpeg("table-4-uni-down.jpg", height = 500, width = 500)
 singu_wide_t %>% select(game_download_all, genderMannerisms_baseline,
                        gsa_t1_mod, familySS, outness_gi_family_t1,
                        outness_gi_friends_t1, outness_gi_school_t1,
                        friendSS) %>%
  tbl_uvregression(
    method = glm,
    y = game_download_all,
    method.args = list(family = binomial),
    exponentiate = TRUE,
~~~

~~~~
    pvalue_fun = ~ style_pvalue(.x, digits = 2)
  ) %>%
  add_global_p() %>% # add global p-value
  add_nevent() %>% # add number of events of the outcome
  bold_p() %>% # bold p-values under a given threshold (default 0.05)
  bold_p(t = 0.25) %>%
  bold_labels()# %>% as_gt() %>% gtsave("tableUni.html")
#grid.arrange(tableUni)
#dev.off()
```
~~~~

Table 5: Multivariable Logistic Regression Model: Download (Trans participants only)

~~~~
```{r}
multidownfit6 <- glm(formula = game_download_all~ factor(gsa_t1_mod_labeled) +
                       familySS,
                   data = singu_wide_t, family = "binomial"(link = "logit"))
multidownfit6 %>% tbl_regression(exponentiate = TRUE,
                                pvalue_fun = function(x) style_pvalue(x,digits = 3))
paste("AIC:", multidownfit6$aic)
```
~~~~

Table 6: Gay-Straight Alliance (0 = Yes) and Download (1 = Yes) Outcome

~~~~
```{r}
singu_wide_t %>% select(game_download_all_labeled, gsa_t1_mod_labeled) %>%
  tbl_summary(label = c(game_download_all_labeled ~ "Game Download",
                       gsa_t1_mod_labeled ~ "Gay-Straight Alliance?"
                       ),
                       type = everything() ~ "categorical",
              by = game_download_all_labeled
                             ) %>% add_p() %>% bold_labels() #%>% as_gt() %>% gtsave()
```
~~~~

Multivariable Exact Logistic Regression Model: Download

The *elrm* package performs exact logistic regression by using the Markov Chain Monte Carlo (MCMC) algorithm. It is computationally highly intensive, and conditioning on too many variables at once can lead to highly discrete or degenerate distributions.

The Odds Ratio for family social support (discrete) is inconclusive.

```{r}
set.seed(11037)

exact6 <- xtabs(~game_download_all+
 interaction(gsa_t1_mod, familySS), data = singu_wide_t)

ddat6 <- data.frame(
        familySS = rep(as.numeric(levels(factor(singu_wide_t$familySS))),
                2),
        gsa = rep(as.numeric(levels(factor(singu_wide_t$gsa_t1_mod))),
                 each = 24),
        no_down = exact6[2,], n_trials = colSums(exact6)
        )
```

~~~
fit_down_exact_dat6_gsa<- elrm(formula =  no_down/n_trials ~ gsa + familySS,
                         interest = ~gsa, iter = 22000,
   dataset = ddat6 %>% filter(n_trials != 0), burnIn = 2000)
```
~~~

Table 7: Exact Logistic Regression for Download by Gay-Straight Alliance, Controlling for Discretized Family Social Support

~~~
```{r, message = TRUE}
#fit_down_exact_dat6_gsa$coeffs
#fit_down_exact_dat6_gsa$coeffs.ci
tibble("Variable" = "Gay-Straight Alliance",
     "OR" = round(exp(fit_down_exact_dat6_gsa$coeffs), 3),
        "CI" = paste("(", round(exp(fit_down_exact_dat6_gsa$coeffs.ci[[1]]), 3),
                  ",", round(exp(fit_down_exact_dat6_gsa$coeffs.ci[[2]]), 3), ")"),
     "P-value" = fit_down_exact_dat6_gsa$p.values) %>% knitr::kable()
```
~~~

Table 8: Exact Logistic Regression for Download by Family Social Support (Discrete), Controlling for Gay-Straight Alliance

~~~
```{r}
~~~

~~~
fit_down_exact_dat6_family<- elrm(formula =  no_down/n_trials ~ gsa + familySS,
                              interest = ~familySS, iter = 22000,
   dataset = ddat6 %>% filter(n_trials != 0), burnIn = 2000)
```


```{r, message = TRUE}
#fit_down_exact_dat6_family$coeffs
#fit_down_exact_dat6_family$coeffs.ci


tibble("Variable" = "Family Social Support",
      "OR" = round(exp(fit_down_exact_dat6_family$coeffs),3),
         "CI" = paste("(", round(exp(fit_down_exact_dat6_family$coeffs.ci[[1]]), 3),
                   ",", round(exp(fit_down_exact_dat6_family$coeffs.ci[[2]]), 3), ")"),
      "P-value" = fit_down_exact_dat6_family$p.values) %>% knitr::kable()
```



# Hours Models



## Transgender and Cisgender
~~~

Model V uses linear regression to compare hours played for cisgender and transgender participants. Table 9 summarized the estimated coefficient, 95% confidence interval, and p-values for Model V. We can see there is not sufficient statistical evidence that hours differ among the two groups (p = 0.6).

Table 9: Hours Played: Univariable Model (Cis and Trans)

~~~
```{r}
fitGender2_hours <- lm(game_hours_all ~ gender_2groups, data = singu_wide)
fitGender2_hours %>% tbl_regression() %>% bold_labels()
paste("AIC:", fitGender2_hours$aic)
```
~~~

As with modeling for download, univariate analysis was part of the selection process for modeling hours among trans participants by covariates of interest. The same subset of covariates were chosen from, and their univariable results were summarized in Table 10.

Table 10: Univariable Models for Hours Covariates (Trans participants only)

~~~
```{r}
singu_wide_t %>% select(game_hours_all, genderMannerisms_baseline,
~~~

~~~
                   gsa_t1_mod, familySS, outness_gi_family_t1,
                   outness_gi_friends_t1, outness_gi_school_t1,
                   friendSS) %>%
 tbl_uvregression(
   method = glm,
   y = game_hours_all,
   pvalue_fun = ~ style_pvalue(.x, digits = 2)
 )

```
~~~

Once again, variables with p < 0.25 were deleted. Re-adding these variables did not significantly change coefficient estimates, so they remained deleted in the final model.

The final linear regression model for hours played among trans students (Model VI) used the variables *Gay-Straight Alliance*, *Family Social Support*, and *outness to family*. Figure 2 plots hours played by social support. Table 11 summarized Model VI

Figure 2: Game Hours Played by Family Social Support

~~~
```{r}
~~~

~~~~
ggplot(singu_wide_t) + aes(x = familySS, y = game_hours_all) + geom_point() +
  xlab("Family Social Support") + ylab("Game Hours Played") + geom_smooth() +
  ggtitle("Hours Played by Family Social Support, Trans Participants")
```




### Table 11: Multivariable Linear Regression Model: Hours Played


```{r}
multihoursfit1 <- glm(formula = game_hours_all~ gsa_t1_mod + familySS +
                      outness_gi_family_t1, data = singu_wide_t)
multihoursfit1 %>% tbl_regression()
paste("AIC:", multihoursfit1$aic)
```


```{r}
library(car)
crPlots(multihoursfit1)
```
~~~~

Completion

Model VII is a univariable logistic regression model that compares completion rates by cisgender and transgender participants. Unlike hours and participation, completion differed noticeably among the two groups. Table 12 summarized Odds Ratio (estimate and 95% confidence interval) and p-value. We see that transgender students are estimated to be about three times as likely to complete a game.

Table 12 Univariable Logistic Model: Completion (Cis and Trans)

~~~
```{r}
fitGender2c <- glm(game_complete_all_2lev ~ gender_2groups, data = singu_wide,
            family = "binomial")
fitGender2c %>% tbl_regression(exponentiate = TRUE) %>% bold_labels()
paste("AIC:", fitGender2c$aic)
```
~~~

As part of selecting Model VIII, individual models were fit for the covariates of interest. Table 13 summarized the confidence interval, p-value, and odds ratio for each variable.

Table 13: Univariable Models for Completion Covariates (Trans participants only)

~~~
```{r}
singu_wide_t %>% select(game_complete_all_2lev, genderMannerisms_baseline,
                        gsa_t1_mod, familySS, outness_gi_family_t1,
                        outness_gi_friends_t1, outness_gi_school_t1,
                        friendSS) %>%
  tbl_uvregression(
    method = glm,
    y = game_complete_all_2lev,
    method.args = list(family = binomial),
    exponentiate = TRUE,
    pvalue_fun = ~ style_pvalue(.x, digits = 2)
  )
```
~~~

Once again, p-values above the 0.25 threshold led to a variable's deletion. None of the deleted variables were found to be significant confounders, so they were not added back in. Additionally, *gay-straight alliance* was removed due to obtaining a high Wald test p-value, which was confirmed by using the likelihood ratio test to compare the new nested model to the previous model.

Model VIII evaluated game completion by friend social support and gender mannerisms (with higher score indicating more "perceived masculine" mannerisms). Table 14 summarized the point estimates and confidence intervals for Odds Ratios, as well as p-values. We see a weak negative relationship between friend social support and game completion.

Table 14 Multivariable Logistic Regression Model: Completion (Trans only)

~~~~
```{r}
multicompfit4 <- glm(game_complete_all_2lev ~
                gender_mannerisms_t1 + gsa_t1_mod, data = singu_wide_t)
multicompfit4 %>% tbl_regression(exponentiate = TRUE)
paste("AIC:", multicompfit4$aic)
```
~~~~

Exact Logistic Regression: Completion

~~~~
```{r}
~~~~

~~~
set.seed(11037)

### To make this model work, we have to collapse friendSS to a categorical variable

exactComp1 <- xtabs(~game_complete_all_2lev + interaction(gender_mannerisms_t1,
                                              gsa_t1_mod),
               data = singu_wide_t)
ddatComp1 <- data.frame(gay_straight = rep(c(0,1), 7),
                  mannerisms = rep(c(0:6), each = 2),
                  not_down = exactComp1[2,], n_trials = colSums(exactComp1)
)
ddatComp1

fit_comp_exact_d1 <- elrm(formula = not_down/n_trials ~ gay_straight + mannerisms,
                   interest = ~mannerisms, iter = 22000,
                   dataset = ddatComp1 %>% filter(n_trials > 0), burnIn = 2000)
summary(fit_comp_exact_d1)
fit_comp_exact_d2 <- elrm(formula = not_down/n_trials ~ gay_straight + mannerisms,
                   interest = ~gay_straight , iter = 22000,
                   dataset = ddatComp1 %>% filter(n_trials > 0), burnIn = 2000)
summary(fit_comp_exact_d2)
```
~~~

~~~
```{r, message = TRUE}
#fit_comp_exact_d1$coeffs
#fit_comp_exact_d1$coeffs.ci


#fit_comp_exact_d2$coeffs
#fit_comp_exact_d2$coeffs.ci
```



## Table 15: Exact Logisitic Regression Model for Completion by Gender Mannerisms,
Controlling for Gay-Straight Alliance


```{r}
tibble("Variable" = "Mannerisms",
       "OR" = round(exp(fit_comp_exact_d1$coeffs[[1]]), 3),
          "CI" = paste("(", round(exp(fit_comp_exact_d1$coeffs.ci[[1]]), 3),
                    ",", round(exp(fit_comp_exact_d1$coeffs.ci[[2]]), 3), ")"),
       "P-value" = fit_comp_exact_d1$p.values[1]) %>% knitr::kable()


```
~~~

Table 16: Exact Logisitic Regression Model for Completion by Gay-Straight Alliance, Controlling for Gender Mannerisms

~~~
```{r}
tibble("Variable" = "Gay-Straight Allaiance", "OR" = exp(fit_comp_exact_d2$coeffs),
        "CI" = paste("(", round(exp(fit_comp_exact_d2$coeffs.ci[[1]]),3),
                     ",", round(exp(fit_comp_exact_d2$coeffs.ci[[2]]), 3), ")"),
    "P-value" = fit_comp_exact_d2$p.values) %>% knitr::kable()
```
~~~

(New file)

```
---
title: "Results"
output:
  html_document:
    toc: yes
    df_print: paged
  pdf_document:
    toc: yes
---
---
title: "R Notebook"
```

~~~
output: html_notebook
---

```{r}

knitr::opts_chunk$set(message = FALSE, echo = FALSE, warning = FALSE)
```

```{r}
## Libraries

library(tidyverse)
library(caret)
library(ggplot2)
library(paletteer)
library(peRReo)
library(mice)
library(gtable)
library(elrm)
library(lmtest)
~~~

~~~
library(gtsummary)
```




```{r}
## Original df:
singu <- read_csv("data16_DataOutput_ForViolet.csv")
```




```{r}
##Transform to create game_complete merged:

#singu_wide <- singu %>% reshape(idvar = 'record_id', timevar = 'time', direction = 'wide')

#singu_wide <- singu %>% pivot_wider(id_cols = record_id)

## Properly Wide Format:
~~~

~~~
singu_t1 <- singu %>% filter(time == 1) %>% select(-contains("t2")) %>% select(-contains("t3"))
singu_t2 <- singu %>% filter(time == 2) %>% select(-contains("t1")) %>% select(-contains("t3"))
singu_t3 <- singu %>% filter(time == 3) %>% select(-contains("t1")) %>% select(-contains("t2"))
singu_12 <- inner_join(singu_t1, singu_t2, by = 'record_id') %>% select(-contains('.y'))
singu_wide <- inner_join(singu_12, singu_t3, by = 'record_id') %>% select(-contains('.y')) %>% select(-contains('.x'))

```

```{r}
## Mutating singu_wide further:

singu_wide <- singu_wide %>% mutate(game_complete_all = factor(case_when(is.na(game_complete_t2) ~ game_complete_t3,
                                         TRUE ~ game_complete_t2
)))
~~~

```
    singu_wide <- singu_wide %>% mutate(game_complete_all_2lev=
case_when(game_complete_all == "1" ~ 0,
                                        game_complete_all != "1" ~ 1)
    )
```

```
    singu_wide <- singu_wide %>% mutate(game_hours_all = game_playtime_t2 +
game_playtime_t3)
```

```
    singu_wide <- singu_wide %>% mutate(game_hours_all =
                                case_when(is.na(game_playtime_t2) ~ game_playtime_t3,
                                    is.na(game_playtime_t3) ~ game_playtime_t2,
                                    TRUE ~ game_playtime_t2 + game_playtime_t3,)
    )
```

```
    singu_wide <- singu_wide %>% mutate(game_download_all =
factor(case_when(is.na(game_download_t3) ~ game_download_t2,
                                        TRUE ~ game_download_t3-1), levels = c(0,1))


    )
```

```
    ## Labels for categorical variables:
    singu_wide <- singu_wide %>% mutate(free_reduced_lunch_t1 =
factor(free_reduced_lunch_t1,
                        labels = c("Yes","No","Unsure")))

    singu_wide <- singu_wide %>% mutate(sexId6Groups = factor(sexId6Groups,
                        labels = c("Gay or Lesbian", "Bisexual",
                            "Queer", "Unsure",
                            "MUltiple", "Other")))


    singu_wide <- singu_wide %>% mutate(g2 = factor(gender_appearance_t1,
                        labels = c("Very feminine", "Mostly feminine",
                            "Somewhat feminine", "Equally masculine and feminine",
                            "Somewhat masculine", "Mostly masculine", "Very
masculine")))



    singu_wide <- singu_wide %>% mutate(gender_3new =
                          case_when((gender_6groups == 0) |  (gender_6groups == 3 ) ~
"Girls",
                              (gender_6groups == 1) |  (gender_6groups ==2 ) ~ "Boys",
```

```
                                        (gender_6groups == 4) | (gender_6groups == 5 ) ~
"Nonbinary"))

      singu_wide <- singu_wide %>%  mutate(gender_5 = case_when((gender_6groups == 4) |
(gender_6groups == 5 ) ~ 4,
                                      TRUE ~ gender_6groups)
          )

      singu_wide <- singu_wide %>% mutate(gender_5 = factor(gender_5, labels = c("Cis girl",
"Cis boy",
                                                 "Trans boy", "Trans girl",
                                                 "Nonbinary")))



      singu_wide <- singu_wide %>% mutate(gender_2groups =
factor(case_when((gender_3groups == 0) | (gender_6groups == 1 ) ~ 0,
                                      TRUE ~ 1), labels = c("Cisgender","Transgender"))
          )


      singu_wide <- singu_wide %>% mutate(race2 = factor(race2,
                 labels = c("White","Latinx","Asian/Pacific Islander",
"Black","Multiracial")))



      ## Race, 2 levels:
```

~~~
singu_wide <- singu_wide %>% mutate(race_2groups = case_when(race2 == "White" ~ "White",
                                    TRUE ~ "Nonwhite"))

```

```{r}
##Create trans-only dataframe, this will be handy later:

singu_wide_t <- singu_wide %>% filter(gender_2groups == "Transgender")

singu_wide_t <- singu_wide_t %>% mutate(gsa_t1_mod = case_when(gsa_t1 == 99 ~ 1,
                                        TRUE ~ gsa_t1))

```

```{r}
~~~

```
## More created variables:


singu_wide_t <- singu_wide_t %>%
  mutate(familySS_3 = rowMeans(select(., socialsupport_3_t1, socialsupport_4_t1,
                                      socialsupport_8_t1)))
singu_wide_t <- singu_wide_t %>%
  mutate(familySS_2 = rowMeans(select(.,socialsupport_4_t1,
                                      socialsupport_8_t1)))
singu_wide_t <- singu_wide_t %>% mutate(schoolclimate_avg_4 = (schoolclimate_1_t1
+
                                    schoolclimate_2_t1 + schoolclimate_3_t1 +
                                     schoolclimate_4_t1)/4)
singu_wide_t <- singu_wide_t %>%
  mutate(gendermanapp_3levels = case_when(gendermanapp <= 2 ~ 0, #fem
                                    gendermanapp >= 4 ~ 2, #masc
                                    TRUE ~ 1 ), # neutral
         schoolclimate_avg_4_2levels = case_when(schoolclimate_avg_4 <= 2 ~ 0, # hostile
                                       TRUE ~ 1), # friendly
         familySS_7levels = round(familySS),
         friendSS_7levels = round(friendSS)
  )
singu_wide_t <- singu_wide_t %>%
  mutate(familySS_2levels = case_when(familySS < 4 ~ 0,
```

~~~~
                 TRUE ~ 1),
    friendSS_2levels = case_when(friendSS < 4 ~ 0,
                 TRUE ~ 1),
    outness_gi_family_3levels = case_when(outness_gi_family_t1 == 0 ~ 0, #universal
                      outness_gi_family_t1 == 4 ~ 2, # closeted
                      TRUE ~ 1),
    outness_gi_school_3levels = case_when(outness_gi_school_t1 == 0 ~ 0, #universal
                      outness_gi_school_t1 == 4 ~ 2, # closeted
                      TRUE ~ 1))

```


# Univariable Descriptive Statistics

## All Students


```{r}

ggplot(singu_wide %>% filter(!is.na(game_download_all), game_download_all !=3),
   aes(factor(game_download_all), fill = game_download_all)) + geom_bar() +
 scale_fill_discrete(name = "Downloaded Game?", labels = c("Yes","No","Unsure")) +
~~~~

~~~
  xlab("Downloaded Game?") + ggtitle("Download Frequency")

singu_wide %>% group_by(game_download_all) %>% tally() %>% mutate(percent = n/sum(n))
```



```{r}

ggplot(singu_wide %>% filter(!is.na(game_hours_all)),
      aes(x = game_hours_all)) + geom_bar(fill = "purple") + ggtitle("Distribution of Hours Played") +
  xlab("Hours Played")
summary(singu_wide$game_hours_all)
```



```{r}
#factor(singu_wide$game_complete_all)

ggplot(singu_wide %>% filter(!is.na(game_complete_all), game_complete_all !=3),
      aes(game_complete_all, fill = game_complete_all)) + geom_bar() +
~~~

```
  scale_fill_discrete(name = "Completed Game?", labels = c("Yes","No","Unsure")) +
  xlab("Completed Game?") + ggtitle("Completion Rate of Game")

singu_wide %>% group_by(game_complete_all) %>% tally() %>% mutate(percent = n/sum(n))
```

Trans and Cis Breakdown:

```{r}
palCalle2 <- latin_palette("calle13", n = 2)

ggplot(singu_wide, aes(gender_2groups, fill = factor(gender_2groups))) + geom_bar() +
  scale_fill_manual(values = palCalle2, name = "") +
  xlab("") + ggtitle("Proportion of Transgender Students")

singu_wide %>% group_by(gender_2groups) %>% tally() %>% mutate(percent = n/sum(n))
```

```{r}
```

~~~
#table(singu_wide$race1, singu_wide$race2)
```

## Univariate Statistics for Trans Participants

### Outcomes

```{r}

ggplot(singu_wide_t %>% filter(!is.na(game_download_all), game_download_all !=3),
      aes(factor(game_download_all), fill = game_download_all)) + geom_bar() +
  scale_fill_discrete(name = "Downloaded Game?", labels = c("Yes","No","Unsure")) +
  xlab("Downloaded Game?") + ggtitle("Download Frequency")

singu_wide_t %>% group_by(game_download_all) %>% tally() %>% mutate(percent = n/sum(n))
```
~~~

~~~
```{r}

ggplot(singu_wide_t %>% filter(!is.na(game_hours_all)),
    aes(x = game_hours_all)) + geom_bar(fill = "purple") + ggtitle("Distribution of Hours Played") +
  xlab("Hours Played")
summary(singu_wide_t$game_hours_all)
~~~

```{r}
#factor(singu_wide_t$game_complete_all)

ggplot(singu_wide_t %>% filter(!is.na(game_complete_all), game_complete_all !=3),
    aes(game_complete_all, fill = game_complete_all)) + geom_bar() +
  scale_fill_discrete(name = "Completed Game?", labels = c("Yes","No","Unsure")) +
  xlab("Completed Game?") + ggtitle("Completion Rate of Game")

singu_wide_t %>% group_by(game_complete_all) %>% tally() %>% mutate(percent = n/sum(n))
```
```
```

~~~
### Covariates of Interest:

```{r}
pal = latin_palette('badgyal', n = 7)

ggplot(singu_wide_t, aes(gender_mannerisms_t1, fill = factor(gender_mannerisms_t1))) + geom_bar() +
  scale_fill_manual(values = pal, name = "Gender Mannerisms",
                    labels = c("Very Feminine", "Mostly Feminine", "Somewhat Feminine",
                               "Equally Masc/Fem", "Somewhat Masculine",
                               "Mostly Masculine", "Very Masculine")) +
  xlab("Gender Mannerisms") + ggtitle("Trans Participants by Gender Mannerisms")

singu_wide_t %>% group_by(gender_mannerisms_t1) %>% tally() %>% mutate(percent = n/sum(n))
```

```{r}

ggplot(singu_wide_t, aes(race_2groups, fill = factor(race2))) + geom_bar() +
~~~

~~~
  xlab("Race") + ggtitle("Trans Participants by Race")

singu_wide_t %>% group_by(race_2groups) %>% tally() %>% mutate(percent = n/sum(n))
```
~~~

~~~
```{r}
palIvy5 <- latin_palette("ivyqueen", n = 5)

ggplot(singu_wide_t, aes(gender_5, fill = factor(gender_5))) + geom_bar() +
  scale_fill_manual(values = palIvy5, name = "",
                    labels = c("Boy", "Girl", "Non-Binary")) +
  xlab("") + ggtitle("Trans Participants by Gender") + theme(axis.text.x = element_blank())
singu_wide_t %>% group_by(gender_5) %>% tally() %>% mutate(percent = n/sum(n))
```
~~~

Gay-Straight Alliance:

~~~
```{r}
ggplot(singu_wide_t, aes(gsa_t1_mod, fill = factor(gsa_t1_mod))) + geom_bar() +
~~~

~~~
  xlab("") + ggtitle('"Does Your School have a Gay-Straight Alliance?"') +
  scale_fill_manual(values = palCalle2, labels = c("Yes","No or Unsure"), name = "GSA?")

singu_wide_t %>% group_by(gsa_t1_mod) %>% tally() %>% mutate(percent = n/sum(n))
```



```{r}
karol7 <- latin_palette("karolg", n = 7)

ggplot(singu_wide_t, aes(g2, fill = factor(g2))) +
  geom_bar() + xlab("Gender Appearance") + ggtitle("Participants by Gendered Appearance") +
  scale_fill_manual(values = karol7, name = "Gendered Appearance",) +
  theme(axis.text = element_blank())

singu_wide_t %>% group_by(g2) %>% tally() %>% mutate(percent = n/sum(n))
```



```{r}
~~~

~~~
ggplot(singu_wide_t, aes(familySS, fill = cut(familySS, 100))) +
  geom_histogram(show.legend = FALSE) + xlab("Family Social Support") +
  ggtitle("Particpants by Family Support") + scale_fill_discrete(h = c(10, 240))

summary(singu_wide_t$familySS)
```
~~~

~~~
```{r}
ggplot(singu_wide_t, aes(friendSS, fill = cut(friendSS, 100))) +
  geom_histogram(show.legend = FALSE) + xlab("Friend Social Support") +
  ggtitle("Particpants by Friends' Support") + scale_fill_discrete(h = c(120, 350))

summary(singu_wide_t$friendSS)

```
~~~

Gay-Straight Alliance:

~~~
```{r}
singu_wide_t %>% group_by(gsa_t1_mod) %>% tally() %>% mutate(percent = n/sum(n))
~~~

```
```

Bibliography

Arrieta, M.C. Stiemsma, L.T., Dimitriu, P.A., Thorson, L., Russsell, S., Yurist-Doutsch, S., Kuzeljevic, B., Gold, M.J., Britton, H.M., […], Finlay B.B. (2015). Early infancy microbial and metabolic alterations affect risk of childhood asthma. *Science Translational Medicine*, 7 (307). https://doi.org/10.1126/scitranslmed.aab2271

Agresti, A. (1992) A survey of exact inference for contingency tables (with discussion). *Statistical Sciences, 7(1).* 131-177. http://www.jstor.org/stable/2246001.

Connolly MD, Zervos MJ, Barone CJ 2nd, Johnson CC, Joseph CLM. (2016) The Mental Health of Transgender Youth: Advances in Understanding. *J Adolesc Health,* 59 (5). 489-495. https://doi.org/10.1016/j.jadohealth.2016.06.012

de Vries ALC, McGuire JK, Steensma TD, Wagenaar ECF, Doreleijers TAH, Cohen-Kettenis PT. (2014) Young adult psychological outcome after puberty suppression and gender reassignment. *Pediatrics,* 134 (4). 696-704. https://doi.org/10.1542/peds.2013-2958

Egan, J.E.E., Corey, S.L., Henderson, E.R., Abebe, K.Z., Louth-Marquez, W., Espelage, D., Hunter, S.C., DeLucas, M., Miller, E., Morrill, B.A., Hieftje, K., Sang, J.M., Friedman, M.S., Coulter, R.W.S. (2021) Feasibility of a Web-accessible Game-based Intervention Aimed at Improving Help Seeking and Coping Among Sexual and Gender Minority Youth: Results from a Randomized Controlled Trial. *J Adolesc Health, 69 (4),* 604-614. https://doi.org/10.1016/j.jadohealth.2021.03.027

Exact Logistic Regression | R Data Analysis Examples. UCLA: Statistical Consulting Group. From https://stats.oarc.ucla.edu/r/dae/exact-logistic-regression/ Accessed March 2023.

Forster, J.J., McDonald, J.W., Smith, P.W.F. (2003). Markov chain Monte Carlo exact inference for binomial and multinomial logistic regression models. *Statistics and Computing 13*. 169-177. https://doi.org/10.1023/A:1023212726863

Harrell FE (2015) *Regression modeling strategies. With applications to linear models, logistic and ordinal regression, and survival analysis. New York, Berlin, Heidelberg: Springer.* ISBN: 978-3-319-19424-0.

Heinze, G., Wallisch, C., Dunkler, D. (2018) Variable selection – A review and recommendations for the practicing statistician. *Biom J,* 60 (3). 431-449. https://doi.org/10.1002/bimj.201700067

Hirji, KF. (2006) *Exact Analysis of Discrete Data.* Chapman and Hall/CRC, Boca Raton, Florida.

Hosmer, D.W., Lemeshow S., Sturdiant, R.X. (2013) *Applied Logistic Regression*. https://doi.org/10.1002/9781118548387

Mehta, C.R., Patel, N.R. (1995). Exact logistic regression: Theory and Examples. *Statistics in Medicine,* 14 (19). 2143-2160. https://doi.org/10.1002/sim.4780141908

Simons, L., Schrager, S.M., Clark, L.F., Belzer, M., Olson, J. (2013) Parental support and mental health among transgender adolescents. *J Adolesc Health,* 53 (6). 791-793. https://doi.org/10.1016/j.jadohealth.2013.07.019

Tordoff DM, Wanta JW, Collin A, Stepney C, Inwards-Breland DJ, Ahrens K. (2022) Mental Health Outcomes in Transgender and Nonbinary Youths Receiving Gender-Affirming Care. *JAMA Netw Open, 5(2)*. https://doi.org/10.1001/jamanetworkopen.2022.0978

Wilson, E.C., Chen, Y.H., Arayasirikul, S., Raymond, H.F., McFarland, W. (2016) The Impact of Discrimination on the Mental Health of Trans*Female Youth and the Protective Effect of Parental Support. *Aids Behav,* 20 (10). 2203-2211. https://doi.org/10.1007/s10461-016-1409-7

Zamar, D., McNeney, B., Graham, J. (2007). elrm: Software Implementing Exact-like Inference for Logistic Regression Models. *Journal of Statistical Software.* 21 (3). https://doi.org/10.18637/jss.v021.i03