

**Predicting Health-related Quality of Life in Early  
Adulthood from Individual Differences in Adolescent  
Neurocognitive Development**

by

**Jennifer M. Fedor**

BS, Neuroscience and Psychology, University of Pittsburgh, 2014

Submitted to the Graduate Faculty of  
the Department of Biostatistics  
Graduate School of Public Health in partial fulfillment  
of the requirements for the degree of  
**Master of Science**

University of Pittsburgh

2020

UNIVERSITY OF PITTSBURGH  
GRADUATE SCHOOL OF PUBLIC HEALTH

This thesis was presented

by

Jennifer M. Fedor

It was defended on

April 20, 2020

and approved by

**Thesis Advisor:**

Hanna Bandos, PhD, Research Assistant Professor, Department of Biostatistics, Graduate School  
of Public Health, University of Pittsburgh

**Committee Members:**

Jeanine Buchanich, MPH, MEd, PhD, Research Associate Professor, Department of Biostatistics,  
Graduate School of Public Health, University of Pittsburgh

Jenna Carlson, PhD, Assistant Professor, Department of Biostatistics, Graduate School of Public  
Health, University of Pittsburgh

Beatriz Luna, PhD, Professor, Departments of Psychiatry and Psychology, School of Medicine  
and Kenneth P. Dietrich School of Arts and Sciences, University of Pittsburgh

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

Copyright © by Jennifer M. Fedor  
2020

**Predicting Health-related Quality of Life in Early Adulthood from Individual Differences in Adolescent Neurocognitive Development**

Jennifer M. Fedor, MS

University of Pittsburgh, 2020

**Abstract**

**Background:** Inhibitory control and the brain systems that support this cognitive process are known to undergo a protracted maturation through adolescence. Relatively little work, however, has examined how individual differences in the normative adolescent development of inhibitory control may be associated with outcomes such as health-related quality of life (QOL) in early adulthood.

**Methods:** We analyzed data from an accelerated longitudinal study of healthy individuals initially aged 8-30 who, at approximately yearly intervals, completed an inhibitory control task while functional magnetic resonance imaging data were acquired. Generalized additive mixed models were utilized to characterize age-related change in inhibitory control behaviorally and in regional brain activation. Random intercepts and slopes from these models, representing person-specific deviations from group-level developmental trajectories, were utilized as covariates in a bootstrap-enhanced elastic net regression procedure to predict QOL, which was assessed with a self-report questionnaire at the study endpoint.

**Results:** There were significant developmental improvements in inhibitory control behaviorally that continued into young adulthood. Among examined motor response and executive control brain regions, there were significant age-related decreases in activation during correctly performed task trials occurring until mid-adolescence in the L frontal eye fields (FEF), bilateral posterior parietal cortex (pPC), and R dorsolateral prefrontal cortex (dlPFC). In the performance monitoring region, dorsal anterior cingulate (dACC), activation during error-corrected trials significantly increased with age and reached mature levels in young adulthood. The bootstrap-enhanced elastic net model indicated that person-specific deviations from these adolescent developmental trajectories did not significantly predict QOL in early adulthood, although variable inclusion probabilities suggested

that performance monitoring behaviorally and activation in L FEF and R dlPFC may be relatively important predictors.

**Conclusions:** Findings show that subtle age-related changes in inhibitory control may continue later into adolescence and young adulthood than previously reported. Individual differences in adolescent development of aspects of inhibitory control may potentially be important predictors of QOL in early adulthood, but require further investigation.

**Public health significance:** Establishing the relationship between individual trajectories of neurocognitive maturation and subsequent QOL may help to inform the development of personalized interventions that can be applied during adolescence to promote optimal adult outcomes.

## Table of Contents

<b>1.0 Introduction . . . . .</b>	1
1.1 Inhibitory control . . . . .	2
1.1.1 Brain systems supporting inhibitory control . . . . .	3
1.2 Health-related quality of life . . . . .	4
1.3 Generalized additive (mixed) models . . . . .	6
1.3.1 Generalized linear models . . . . .	6
1.3.2 Generalized additive models . . . . .	7
1.3.3 Generalized additive mixed models . . . . .	8
1.4 Regularized regression methods . . . . .	9
1.4.1 Ridge regression . . . . .	10
1.4.2 Lasso regression . . . . .	11
1.4.3 Elastic net regression . . . . .	12
1.5 Objectives . . . . .	12
<b>2.0 Methods . . . . .</b>	14
2.1 Participants . . . . .	14
2.2 Experimental procedure and data acquisition . . . . .	16
2.2.1 Antisaccade task . . . . .	16
2.2.2 Eye-tracking data acquisition and preprocessing . . . . .	16
2.2.3 Neuroimaging data acquisition and preprocessing . . . . .	17
2.2.4 WHO-QOL questionnaire . . . . .	18
2.3 Statistical analyses . . . . .	19
2.3.1 Generalized additive mixed models . . . . .	19
2.3.2 Bootstrap-enhanced elastic net regression . . . . .	22
<b>3.0 Results . . . . .</b>	25
3.1 Characterizing adolescent development of inhibitory control . . . . .	25
3.1.1 Behavioral measures . . . . .	25
3.1.2 Brain function measures . . . . .	27
3.1.2.1 Motor response regions . . . . .	27

3.1.2.2 Executive control regions . . . . .	27
3.1.2.3 Performance monitoring region . . . . .	27
3.2 Predicting health-related quality of life . . . . .	33
<b>4.0 Discussion</b> . . . . .	38
<b>Appendix. R code</b> . . . . .	42
<b>Bibliography</b> . . . . .	90

## List of Tables

1	Estimated coefficients and significance of sex for each outcome measure, controlling for age . . . . .	31
2	Global significance of maternal education for each outcome measure, controlling for age . . . . .	32
3	Significance of smooth term for age for each final GAMM . . . . .	33
4	Estimated coefficients and bootstrapped 95% confidence intervals, and variable inclusion probabilities for each candidate predictor in the elastic net model . . . . .	35

## List of Figures

1	Age distribution of the final sample . . . . .	15
2	Effects of age for each behavioral measure . . . . .	26
3	Effects of age for activation during correct antisaccade trials in each motor response region . . . . .	28
4	Effects of age for activation during correct antisaccade trials in each executive control region . . . . .	29
5	Effects of age for activation in the performance monitoring region . . . . .	30
6	Estimated developmental trajectories in high and low health-related quality of life groups . . . . .	36

## 1.0 Introduction

Adolescence is a unique period of development during which many social, biological, and cognitive changes take place. In particular, significant changes in brain structure and function, such as synaptic pruning and myelination [1] and the functional integration of distributed circuitry [2], occur during this period. These changes in brain structure and function during adolescence are thought to support continued refinements in cognitive control, the ability to voluntarily and adaptively coordinate goal-directed behavior [3]. Cognitive control is an essential process, as it is necessary for the performance of more complex cognitive tasks such as planning, reasoning, and decision-making [4]; is involved in emotion regulation [5]; and plays a significant role in health and health behavior [6].

Cognitive control is of particular interest during adolescence because of its relationships with risk-taking behavior and psychopathology, which increase and emerge, respectively, during this stage of development. More specifically, immaturities in cognitive control that persist into adolescence, coupled with concomitant increases in sensation-seeking and sensitivity to rewards, may make adolescents particularly vulnerable to risk-taking and sub-optimal decision-making [7]. This is evidenced by transient increases in mortality due to preventable injury, including motor vehicle crashes and suicide, during this period [8, 9]. Additionally, psychiatric disorders, many of which are characterized by deficits in cognitive control [10], and substance use are significant contributors to morbidity in adolescence [11], and can have lasting effects on health into adulthood.

Improving the health and well-being of adolescents and young adults, and supporting healthy development during the transition from adolescence to young adulthood, have been identified as important public health goals [12]. Prior research has been instrumental in characterizing the normative development of cognitive control and underlying brain regions during adolescence. However, adolescents are a heterogeneous group; relatively little work has explored individual differences in adolescent developmental trajectories, and how such individual differences may contribute to outcomes such as health-related quality of life in early adulthood. As a subjective measure of health and well-being, quality of life can help to identify hidden morbidities that would otherwise go undetected, and thus may be a particularly informative outcome measure. Because adolescence is a time of significant plasticity, it is amenable to and thus a promising target for public health intervention. Therefore, establishing the relationship between individual trajectories of neurocog-

nitive maturation in adolescence and health-related quality of life in adulthood may help to inform the development of personalized interventions that can be applied during adolescence in order to ultimately promote optimal adult outcomes.

### 1.1 Inhibitory control

Cognitive control is comprised of distinct, but overlapping, component processes including working memory, task-switching, and inhibitory control [13–15]. Inhibitory control, sometimes referred to as (voluntary) response inhibition, describes the ability to inhibit a reflexive or prepotent, goal-irrelevant response in favor of a goal-appropriate response [4]. Inhibitory control is present early in development but undergoes a protracted maturation during adolescence.

Inhibitory control is evident as early as infancy, albeit in a basic form. Infants are able to inhibit prepotent reaching responses [16] and suppress attention to distractor stimuli [17]. As inhibitory control begins to improve in early childhood [18] and continues into adolescence [19, 20], individuals are able to perform more complex inhibitory tasks, although the age at which performance reaches adult-like levels may depend on the specific task being examined. Popular paradigms for measuring inhibitory control include the Stroop task, in which individuals must respond with the color in which a word is printed, which on some trials is incongruent with the word itself (e.g., “red” printed in green); the stop-signal task, in which individuals must suppress an already-initiated response such as a button press when an unpredictable cue is presented; and the antisaccade task [3, 4].

The antisaccade task [21] is a well-established oculomotor task that has been widely used to study age-related changes in inhibitory control. In the antisaccade task, participants are presented with a peripheral stimulus, such as a flash of light. In order to perform the task correctly, participants must inhibit the prepotent response to make a saccade to the stimulus location (i.e., a prosaccade), and instead make a saccade to the mirror-image location (i.e., an antisaccade). Young children perform relatively poorly on the antisaccade task. Compared to older age groups, children aged 5 to 8 have more variable and slower overall saccade latencies, and make more errors due to difficulty in suppressing prepotent prosaccades [19]. Extending these findings, in a cross-sectional sample of individuals aged 8 to 30, Luna and colleagues [20] reported that antisaccade latency and the proportion of prosaccade-suppression errors declined rapidly from late childhood through adolescence, following an inverse curve trajectory. Change-point analysis revealed that adult levels of

antisaccade performance were reached at age 14. These results provide evidence that the capacity for inhibitory control is available in childhood, as all children in the study were able to correctly perform an antisaccade on at least one task trial, but the ability to instantiate inhibitory control consistently improves during adolescence. This suggests that improvement in inhibitory control during adolescence reflects refinement of an extant cognitive ability, rather than the acquisition of a fundamentally new process [3, 20]. These findings were replicated in a recent longitudinal study, which also found that antisaccade latencies and error rates decelerated from late childhood into young adulthood [22].

### 1.1.1 Brain systems supporting inhibitory control

Improvements in inhibitory control during adolescence are supported by concurrent changes in regional brain activity. Functional magnetic resonance imaging (fMRI) studies have been instrumental in advancing our understanding of the neural substrates that support the transition to mature levels of inhibitory control during adolescence. Blood oxygenation level dependent (BOLD) fMRI is a non-invasive technique that detects regional changes in blood flow and oxygenation that occur when neuronal activity is upregulated in response to a cognitive task; thus, fMRI provides an indirect measure of brain activity [23].

Traditionally, fMRI studies investigating the development of the neural substrates of inhibitory control have focused on the prefrontal cortex (PFC), a region that has an integral role in the top-down control of behavior [14, 24] and is known to undergo a protracted maturation structurally [25]. Results of these studies have been inconsistent, with both age-related increases [2, 26, 27] and decreases [28, 29] in inhibitory task-related activity in PFC regions being reported. Providing more compelling evidence for the latter finding, a recent longitudinal study investigating the functional development of brain systems supporting antisaccade performance reported that activity in the right dorsolateral PFC (dlPFC) significantly decreased with age through adolescence, although this was not associated with improvements in task performance [22]. Developmental decreases in PFC activation have been interpreted as potentially reflecting reduced effort required for implementing inhibitory control in adulthood [30]. Such decreases have also been posited to reflect developmental increases in the functional integration of a number of brain regions that support inhibitory control, which consequently reduces demands on these regions individually [31, 32].

Indeed, inhibitory control is supported by a distributed circuitry comprised of key executive regions, including the dlPFC and ventrolateral prefrontal cortex (vlPFC); motor regions such as the supplementary motor area, posterior parietal cortex, and putamen; and the dorsal anterior cingulate (dACC) [2, 33]. Motor regions are essential for planning, preparing, and executing motor responses, such as eye movements in the antisaccade task, but typically do not exhibit age-related changes in activation from late childhood into adolescence or adulthood [22, 34]. This suggests that motor response systems necessary for inhibitory control are in place early in development, consistent with evidence that, structurally, sensorimotor areas mature relatively earlier than do higher-order association areas including the PFC [25]. The dACC contributes to the development of inhibitory control due to its role in performance monitoring and error processing [35]. Activation of the dACC has been found to increase with age on error-corrected antisaccade trials, where a saccade to the correct location follows an initial failure to inhibit the prepotent prosaccade, and activity in this region reaches mature levels later than does the dlPFC [22, 34]. Further, dACC activation was found to significantly mediate the relationship between antisaccade performance and age [22], suggesting that dACC activity uniquely supports developmental improvements in inhibitory control observed behaviorally.

In sum, the literature suggests that behavioral refinements in inhibitory control during adolescence are supported by the transition from primary reliance on prefrontal executive systems, to reliance on a circuitry comprised of prefrontal executive, sensorimotor, and attentional systems. Specifically, age-related decreases in PFC activation may reflect reductions in the amount of effort involved in reliably instantiating inhibitory control from childhood through adolescence, as well as the integration and recruitment of additional brain regions to support inhibitory control processes. One such additional area that appears to be of particular importance is the dACC due to its function in performance monitoring. Altogether, this evidence is consistent with changes in brain structure known to occur during adolescence (e.g., myelination), which are thought to support efficient processing locally as well as promote connections between distributed systems.

## 1.2 Health-related quality of life

Health-related quality of life refers to an individual's subjective perception of their health and well-being; it is a multidimensional concept that encompasses physical, psychological, and social

domains [36]. This conceptualization dovetails with the World Health Organization's (WHO) definition of health as a "state of complete physical, mental and social well-being and not merely the absence of disease" [37]. Much of the initial work investigating determinants of health-related quality of life focused on contextual factors, such as sociodemographic characteristics. A number of sociodemographic factors including male gender [38, 39], high socioeconomic status and subjective social class [40], and high level of educational attainment [40, 41] have been linked to better health-related quality of life, overall as well as in specific domains. More recently, however, there has been a focus on the contribution of intrapersonal factors, including affect and cognition, which evidence suggests are stronger or more long-term predictors of subjective well-being than are contextual factors [42].

Previous research explicitly investigating the relationship between inhibitory control and health-related quality of life has been sparse and has yielded conflicting results. One study of older men and women with cardiovascular disease found that inhibitory control, assessed using a modified version of the Stroop task, was significantly related to a measure of independent functioning in daily activities [43]. Another study of older but otherwise healthy women aged 65 to 75 found no significant association between inhibitory control, as measured by performance on the Stroop task, and quality-adjusted life years, although other components of cognitive control such as working memory were found to be significantly related to this measure of health-related quality of life [44]. The inconsistencies between these findings may potentially be explained by differences in the populations examined, or in how health-related quality of life was operationalized. Additionally, because both of these studies focused on older individuals, it is unclear how or if these results may generalize to younger individuals. A recent study among young adults aged 18 to 29 who had engaged in gambling at least once in the preceding year found that impulsivity, a personality trait distinct from but related to inhibitory control deficits [45], was associated with lower self-reported quality of life; however, there was no significant association between performance on an inhibitory control task (stop-signal task) and quality of life in this sample [46]. Thus, the relationship between inhibitory control and health-related quality of life, particularly among adolescents and young adults, remains to be elucidated.

### 1.3 Generalized additive (mixed) models

#### 1.3.1 Generalized linear models

The generalized linear model (GLM) [47] is a widely-utilized statistical tool for regression and classification problems. A GLM consists of three components, namely, a random component, a systematic component, and a link function. The random component specifies the distribution of the response variable, conditional on the values of the explanatory variables in the model; traditionally, this distribution belongs to the exponential family. The systematic component, or linear predictor, is a linear combination of the explanatory variables. Finally, the link function relates the expected value of the response variable to the linear predictor. Different link functions may be used depending on the distribution of the response variable (e.g., the identity link is traditionally used for normally-distributed data, as is the logit link for binomially-distributed data) [47, 48]. The GLM has the following general form

$$g(\mu) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p \quad (1.1)$$

where  $g(\cdot)$  represents the link function;  $\mu = E(Y)$  is the conditional expectation of the response variable  $Y$ , and  $x_j$  represents the  $j^{th}$  of  $p$  covariates. Estimates for the unknown regression parameters,  $\beta_j$ , and standard errors are obtained through maximum likelihood procedures [48].

A central assumption of the GLM is that of a linear relationship between the response variable (via the link function) and the explanatory variable(s). However, this assumption is often violated. When the relationship between the response variable and an explanatory variable is not linear, a commonly used and relatively straightforward approach is to introduce polynomial terms (e.g., quadratic, cubic) into the model in an attempt to better capture this nonlinearity. However, this approach has several disadvantages. Specifically, the inclusion of polynomial terms may introduce problems with collinearity [49] and result in overfitting [50]. Additionally, this method requires the researcher to make a decision about the shape of the observed relationship, which is rarely if ever truly known. This is also a limitation of other methods such as growth curve modeling that *a priori* specify a functional form for the relationship between the response and explanatory variables (e.g., the Gompertz function [51]). Together, this suggests that a more flexible approach may be beneficial when modeling complex, nonlinear relationships.

### 1.3.2 Generalized additive models

The generalized additive model (GAM) [52] is a flexible, semiparametric method for identifying and estimating nonlinear effects of explanatory variables on the response variable.

$$g(\mu) = \beta_0 + s_1(x_1) + \dots + s_p(x_p) \quad (1.2)$$

Here,  $g(\cdot)$  represents the link function;  $\mu = E(Y)$  is the conditional expectation of the response variable  $Y$ ,  $x_j$  represents the value of the  $j^{th}$  of  $p$  covariates, and  $s_j(\cdot)$  represents an unknown smooth function of covariate  $x_j$ . The smooth functions  $s_j(\cdot)$  can be estimated using a number of different methods, including running means, kernel estimates, or splines [52]. However, splines are most commonly used to represent these smooth functions [53, 54]. Using splines, each smooth function  $s_j(x_j)$  can be represented by a sum of basis functions

$$s_j(x_j) = \sum_k^{K_j} \beta_{jk} b_{jk}(x_j) \quad (1.3)$$

where  $K_j$  is the basis dimension (the number of basis functions in the set),  $b_{jk}(x_j)$  are specified basis functions, and  $\beta_{jk}$  are parameters to be estimated [53]. Penalized maximum likelihood is used to estimate these coefficients, and in practice this is typically accomplished using penalized iteratively re-weighted least squares. The penalty terms, or smoothing parameters  $\boldsymbol{\lambda} = \lambda_1, \dots, \lambda_j$  balance smoothness and flexibility [53]. If  $\boldsymbol{\lambda}$  is too large, oversmoothing occurs, while, conversely, if  $\boldsymbol{\lambda}$  is too small, overfitting occurs. Thus, it is crucial to choose optimal values for  $\boldsymbol{\lambda}$ , and this can be accomplished using generalized cross validation [53].

There are several broad classes of splines that can be employed in GAMs, including regression splines, natural splines, and smoothing splines. Regression and natural splines generate a curve composed of  $k$  segments, which are constructed in between  $k+1$  points, or “knots;” the  $k$  individual curves are joined together at these knots such that the entire curve is continuous [53]. However, a disadvantage of these spline methods is that they require the researcher to specify the number and spacing of these knots, which can be problematic because model fit often depends on knot placement [53].

Smoothing splines, particularly thin plate splines, are in many ways optimal smoothers. Because thin plate splines place a knot at every data point, they do not require the specification of the number or location of knots as do regression or natural splines. Thin plate splines are often referred to as “full-rank” because the basis dimension is equal to the number of observations [55].

However, as a result, the number of unknown parameters to be estimated is equal to the number of observations, and thus there is a significant associated computational cost [53, 55, 56]. Thin plate regression splines [56] offer an attractive alternative to thin plate splines. Thin plate regression splines approximate thin plate splines by “truncating” the full-rank basis of thin plate splines in order to obtain a lower-rank smoother, retaining the advantages thin plate splines provide (e.g., no knot placement) while improving computational efficiency [53, 56]. Thin plate regression splines are the default basis for smooths in a popular R package for fitting GAMs [57].

### 1.3.3 Generalized additive mixed models

One important assumption of both GLMs and GAMs is that observations are independent. However, this assumption is violated in the case of longitudinal data, in which repeated measurements are acquired over time from a sample of participants. In this case, additional methods are needed that can account for the correlation between observations from a given participant.

Mixed effects regression models, which incorporate both fixed and random effects, are important tools for longitudinal data analysis as they account for this intra-subject correlation, and have several advantages over other approaches (e.g., generalized estimating equations). First, participants are not required to have the same number of observations, thus allowing participants with incomplete data to be included in analyses. Participants are also not required to have been measured at the same time points, which is particularly useful in longitudinal studies where follow-up times may, and likely do, vary across participants. Finally, mixed effects regression approaches allow for the estimation of individual-level change in addition to the estimation of average or population-level change [58]. Just as generalized linear mixed models (GLMMs) were developed as mixed model extension of GLMs, generalized additive mixed models (GAMM) [59] were developed as a mixed model extension of GAMs. GAMMs have the following general form

$$g(\mu_{it}) = \beta_0 + s_1(x_{1it}) + \dots + s_p(x_{pit}) + z'_{it}b_i \quad (1.4)$$

where  $\mu_{it} = E(Y_{it})$  is the conditional expectation of the response variable  $Y$  for subject  $i$ ,  $i = 1, \dots, n$  at time  $t$ ,  $t = 1, \dots, T_i$ ;  $g(\cdot)$  is the link function;  $s_j(\cdot)$  are the unknown smooth functions;  $x_{jut}$  is the  $j^{th}$  of  $p$  covariates associated with fixed effects;  $z'_{it}$  is a vector of  $q$  covariates associated with random effects; and  $b_i$  is a vector of random effects [60].

Three possible types of random effects can be specified in GAMMs, including random intercepts, slopes, and smooths. Random intercepts characterize subject-specific deviation from the mean or population-average response, while random slopes characterize subject-specific deviation from the population-average covariate effect over time [58]. Although random intercepts and slopes can also be specified in GLMMs, GAMMs are unique in that they additionally allow for random smooths [53, 61]. Random smooths are similar to random slopes, but are more flexible. In GAMMs, random slopes essentially “rotate and stretch” the same curve to best fit a given subject’s trajectory, while random smooths fit a separate curve for each trajectory [61]. However, models with random smooths require  $n \times k$  basis functions to be fit and thus may be prohibitively resource-intensive [61].

Previous work from our laboratory characterizing the functional development of brain regions supporting inhibitory control in a longitudinal sample of adolescents and young adults employed GLMMs, using a polynomial term for age to capture nonlinearity [22]. Although this study provided us with important insight into normative neurocognitive developmental trajectories during adolescence, there are some limitations associated with this analytical approach. First, as discussed previously, the use of a polynomial term assumes that nonlinear age effects follow some known form such as a quadratic, cubic, or inverse function. The application of flexible GAMMs to model neurocognitive development may provide us with a more nuanced understanding of the shapes of these trajectories during adolescence. GAMMs have previously been used to model longitudinal changes in brain structure [62–64], but to our knowledge have yet to be employed in modeling longitudinal changes in brain function. Additionally, because GAMMs permit the estimation of random subject effects including random intercepts and slopes, as do GLMMs, as well as random smooths, after model-fitting these estimates can be extracted and used as covariates in subsequent analyses. In this way, we may be able to determine how individual differences in neurocognitive developmental trajectories during adolescence relate to other factors, such as subjective measures of health and well-being in young adulthood.

#### 1.4 Regularized regression methods

In a linear model of the form  $Y = X\beta + \epsilon$ , where  $Y$  is a vector of observed response variable values,  $X$  is an  $n \times p$  design matrix that contains the values of each of  $p$  covariates for each of  $n$  observations, and  $\epsilon$  is a vector of error terms, we may wish to both predict the response variable

based on the covariates, as well as determine which covariates are important for this prediction [65]. The unknown parameters  $\beta = \beta_1, \dots, \beta_p$  are traditionally estimated through least squares, by minimizing the residual sum of squares (RSS).

$$\text{RSS} = \sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 \quad (1.5)$$

When the number of predictors  $p$  is less than the sample size  $n$ , the method of least squares performs well. Although the least squares estimates have high variance, they are unbiased, and thus mean squared error (MSE), which is equal to the sum of the variance of the estimate, the squared bias of the estimate, and some irreducible error, will be relatively low. A low MSE corresponds to more accurate prediction performance. However, when  $p$  approaches  $n$ , although bias remains low, the variance of the least squares estimates increases, consequently increasing MSE. Finally, in the case of high-dimensional data where  $p > n$ , least squares has no unique solution, and any of these solutions will be overfit to the data [65, 66]. Regularized regression or “shrinkage” methods are biased, but are also less flexible and thus have less variance. In situations where  $p > n$ , unlike least squares, these methods can still perform well by trading a small increase in bias for a reduction in variance, consequently reducing MSE [66].

Problems with relatively high-dimensional data (i.e.,  $p > n$ ) may often be encountered in neuroimaging research. Small sample sizes are common in fMRI studies due to the high cost associated with acquiring these data [67]. A recent systematic review of task-based fMRI papers published in 2017 reported a median sample size of only  $n = 33$ , with almost 75% of studies having 50 or fewer subjects [68]. Thus, regularized regression methods may offer a useful alternative for modeling data from neuroimaging studies, where  $p$  can conceivably approach or exceed  $n$ .

#### 1.4.1 Ridge regression

In ridge regression [69], the coefficients  $\beta_1, \dots, \beta_p$  are estimated by minimizing the following objective, which is similar to that of least squares, but includes a penalty term.

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p \beta_j^2 \quad (1.6)$$

The penalty term  $\lambda \sum_j \beta_j^2$  is referred to as the  $\ell_2$  penalty. The parameter  $\lambda$  is a non-negative tuning parameter that can be determined through cross-validation. When  $\lambda = 0$ , the estimated

coefficients are the same as those produced by least squares. However, as  $\lambda$  is increased away from zero, the magnitudes of the estimated coefficients,  $\hat{\beta}_1, \dots, \hat{\beta}_j$ , decrease or “shrink” in aggregate toward zero [66]. Although the  $\ell_2$  penalty shrinks coefficient estimates toward zero, it does not force any estimates to be exactly zero [66]. Thus one notable disadvantage of ridge regression is that it does not perform variable selection; all  $p$  covariates will be included in the final model in some capacity. This lack of parsimony can complicate model interpretation, particularly when modeling high-dimensional data.

### 1.4.2 Lasso regression

Lasso regression [70] is another regularized regression method, which, unlike ridge regression, can perform variable selection. In lasso regression, the regression coefficients  $\beta_1, \dots, \beta_p$  are estimated by minimizing the following objective. Like ridge regression, this objective is similar to that of least squares, but includes a penalty term.

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p |\beta_j| \quad (1.7)$$

The lasso penalty term,  $\lambda \sum_j^p |\beta_j|$ , is referred to as the  $\ell_1$  penalty. Unlike with the  $\ell_2$  penalty of ridge regression, with the  $\ell_1$  penalty (assuming a sufficiently large value for  $\lambda$ ), some coefficient estimates are forced to be exactly zero [66]. Thus, lasso regression can perform variable selection and yield a “sparse” model. The subset of covariates included in the final model depends on the value of  $\lambda$ ; as the value of  $\lambda$  increases, more coefficients will shrink to zero and thus fewer covariates will be included.

Although lasso regression is advantageous for performing variable selection, it has some limitations. Ridge regression typically performs better than does lasso regression when there are fewer covariates than observations (i.e., when  $p < n$ ) [66]. In the case of high-dimensional data where  $p > n$ , lasso regression can only select at most  $n$  covariates before it saturates [71]. Additionally, if a group of covariates is highly correlated, lasso regression tends to arbitrarily choose one of them and exclude the others [71].

### 1.4.3 Elastic net regression

Elastic net regression [71] was recently developed in an attempt to overcome the individual limitations of both ridge and lasso regression approaches. The elastic net method utilizes a linear combination of the  $\ell_2$  penalty of ridge regression and the  $\ell_1$  penalty of lasso regression. In elastic net regression, coefficients are estimated by minimizing the following objective.

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda_1 \sum_{j=1}^p |\beta_j| + \lambda_2 \sum_{j=1}^p \beta_j^2 \quad (1.8)$$

The solution to the above is referred to as the “naive” elastic net solution. The naive elastic net follows a sequential procedure, in which the ridge regression coefficients are determined for a fixed value of  $\lambda_2$ , and then the lasso shrinkage is subsequently performed. However, this procedure results in excessive shrinkage, which increases bias and has little impact on variance, resulting in poorer prediction performance. To mitigate this problem, the naive elastic net estimates are rescaled by a factor of  $(1 + \lambda_2)$ .

$$\hat{\beta}_{\text{elastic net}} = (1 + \lambda_2) \hat{\beta}_{\text{naive elastic net}} \quad (1.9)$$

This transformation preserves the variable selection property of the lasso step while undoing excessive shrinkage [71]. Additionally, elastic net is strictly convex when  $\lambda_2 > 0$ , which allows for groups of highly correlated predictors to have similar or equal regression coefficients [71]. Thus groups of highly related predictors will tend to be all included in or all excluded from the model, depending on the magnitudes of the coefficients. Elastic net has been shown to perform better in terms of prediction accuracy and sensitivity compared to ridge and lasso methods [72, 73].

## 1.5 Objectives

The present study aimed to establish the relationship between individual differences in the development of inhibitory control and its neural substrates in adolescence and subsequent health-related quality of life in adulthood. We used data from an accelerated longitudinal cohort of healthy individuals, initially aged 8 to 30 years, who completed up to 13 study visits at approximately yearly intervals. At each study visit, participants performed the antisaccade task, a well-established oculomotor task probing inhibitory control, while fMRI data were acquired. At the conclusion of the

study, participants completed the World Health Organization Quality of Life (WHO-QOL) questionnaire, a self-report measure assessing subjective health and well-being in physical, psychological, social, and environmental domains.

There were two major objectives of the present study: (1) To characterize the normative adolescent development of inhibitory control (as measured by performance on the antisaccade task), as well as activation in brain regions supporting inhibitory control, in our longitudinal sample using GAMMs; and (2) To determine how individual differences in these adolescent developmental trajectories (i.e., random intercepts and slopes) may predict subsequent health-related quality of life in early adulthood using elastic net regression.

## 2.0 Methods

### 2.1 Participants

One-hundred and sixty-seven participants (93 female) were studied in an accelerated longitudinal cohort design, in which participants spanning a broad cross-section of age were recruited and followed prospectively. Participants were initially aged 8 to 30 years and were followed for up to 13 time points (mean = 3.65, SD = 2.94) at approximately yearly intervals. At each time point, participants completed an fMRI scan and a separate behavioral session consisting of neuropsychological testing, computerized cognitive tasks, and surveys. Participants were healthy individuals without history of head injury with a loss of consciousness exceeding one hour; vision or eye-movement problems not corrected in childhood; Tourette syndrome or tic disorders; seizures; meningitis; encephalitis; diabetes; genetic disorder; or a personal or family history of psychiatric disorders including autism spectrum disorders, attention deficit hyperactivity disorder, bipolar disorder, major depression, schizophrenia, or any major neurological disorder such as Huntington's chorea. Participants also had no contraindications for scanning such as non-removable metal, weight greater than 250 pounds, or current pregnancy. IQ was measured using the Weschler Abbreviated Scale of Intelligence [74], and all participants had a full-scale IQ greater than 80. The study was approved by the Institutional Review Board at the University of Pittsburgh. All participants, and parents of minor participants, provided written informed consent. Participants were compensated for their time.

Over the course of the study, 41 participants were dropped due to diagnosis of a major neurological or psychiatric disorder in a first-degree relative ( $n = 15$ ), claustrophobia or unwillingness to scan ( $n = 11$ ), poor performance ( $n = 8$ ), brain abnormality ( $n = 3$ ), diagnosis of a psychiatric disorder or use of psychiatric medication ( $n = 3$ ), or seizure episodes ( $n = 1$ ), resulting in an eligible sample of 126 participants. At the conclusion of the study, participants were contacted to complete a brief follow-up study consisting of the World Health Organization Quality of Life (WHO-QOL) questionnaire. Of the participants contacted for the follow-up study, 71 completed the questionnaire. Among these 71 participants, 21 were excluded from the present analyses because they had usable data for fewer than three time points, the minimum number required to allow for the identification of nonlinear age effects. This resulted in a final sample size of 50 participants

(30 female) with a total of 316 scans (Figure 1). On average, each participant completed 6.32 scans ( $SD = 2.71$ , min = 3, max = 13), with an average between-scan interval of 1.65 years ( $SD = 1.58$ , min = 0.41, max = 10.16).

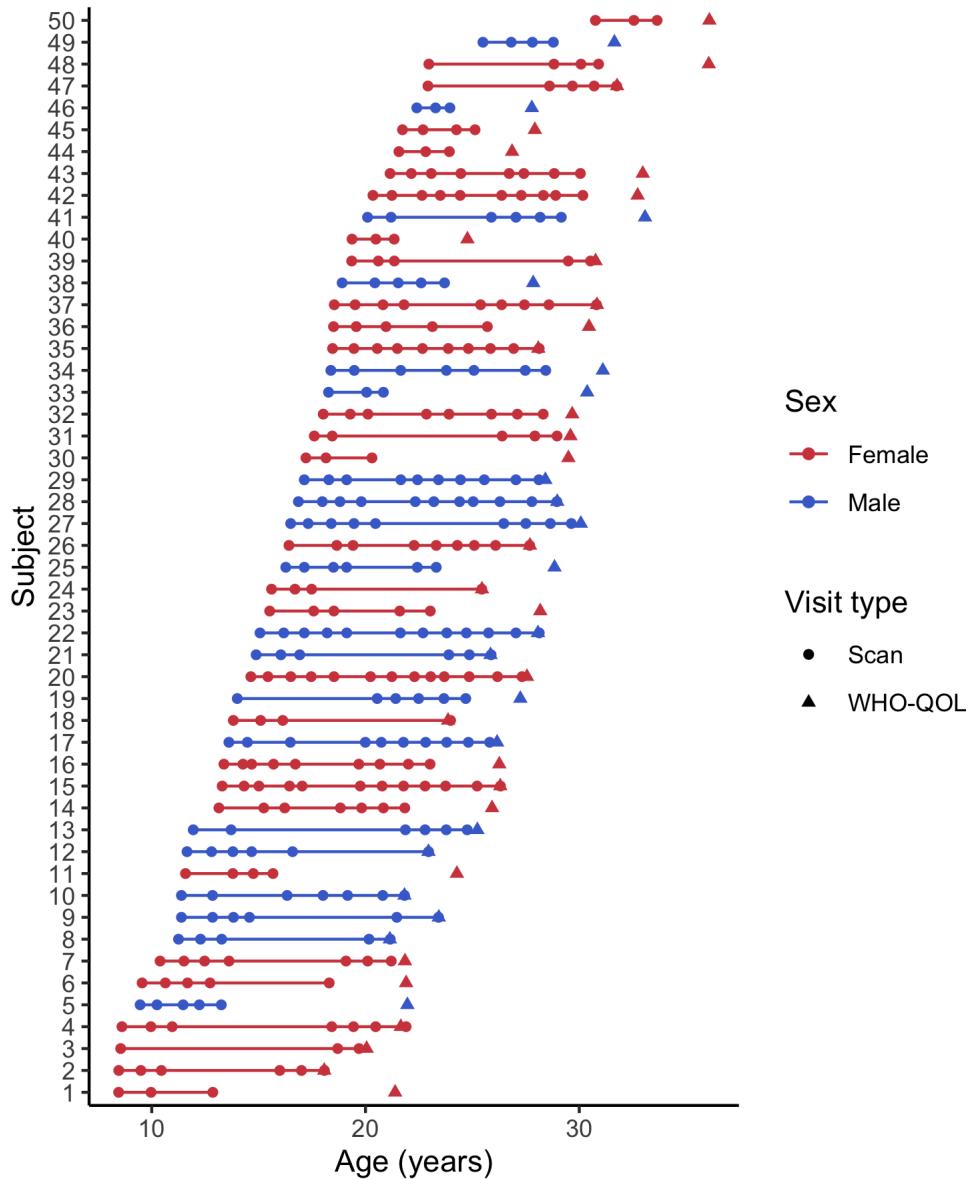


Figure 1: Age distribution of the final sample. Lines interconnect scans for a single participant, where each circle denotes a scan acquisition. For each participant, the left-most circle corresponds to age at baseline scan. Triangles correspond to age at WHO-QOL administration for each participant

## **2.2 Experimental procedure and data acquisition**

### **2.2.1 Antisaccade task**

At each visit, participants completed a total of four task runs in the scanner. Each task run consisted of a block of the antisaccade task and a block of a prosaccade task, each of which was preceded and followed by a block of fixation (i.e., a central cross to which the participant was instructed to look). The order in which the antisaccade and prosaccade task blocks were presented within each run was counterbalanced across runs and participants. In each task run, participants completed 12 trials of the antisaccade task and 12 trials of the prosaccade task, for a total of 48 trials of each type in the experiment. On antisaccade trials, participants first fixated on a red central fixation cross for 3 s. After 3 s, the central fixation disappeared and the saccade target stimulus, a yellow circle, immediately appeared in one of six possible locations in the periphery (presented on the horizontal meridian at  $\pm 3$ , 6, or  $9^\circ$  visual angle) for 1.5 s. Stimulus location order was randomized within each task run. In order to perform the antisaccade task, participants were instructed to inhibit the prepotent prosaccade to the stimulus, and instead make an antisaccade to the opposite or mirror-image location. Inter-trial intervals consisted of a white central fixation cross and varied in duration between 3 and 9 s in order to permit estimation of trial-related activation. The procedure for the prosaccade task was identical, with the exception that each trial began with a green central fixation cross, and participants were instructed to make the prepotent prosaccade to the stimulus location when the stimulus appeared. Only the antisaccade task is considered in the present analyses. A PC running E-Prime software was used to control stimulus display. Stimuli were projected onto a translucent screen affixed behind the scanner bore, visible to the participant via a mirror attached to the head coil.

### **2.2.2 Eye-tracking data acquisition and preprocessing**

In order to assess antisaccade task performance, eye movement data were obtained during scanning using a long-range optics eye tracking system (model R-LRO6, Applied Science Laboratories, Bedford, MA) with a sampling rate of 60 Hz. Lights in the scanner room were dimmed to reduce glare and maximize pupil size. The eye tracker field of view was manually centered on the participant's pupil before beginning the scan. Immediately before the task, the eye tracker was calibrated using an evenly-distributed, nine-point stimulus grid. Calibration verification was per-

formed by instructing the participant to re-scan the nine-point grid, and adjustments were made by the experimenter when necessary.

Eye movement data were scored offline with an in-house, automated scoring program using tools from ILAB [75]. Saccades were identified using a velocity algorithm with a  $20^{\circ}/\text{s}$  criterion. Saccades following the presentation of each stimulus were subsequently scored as correct (the participant successfully inhibited the prosaccade and made the antisaccade), error-corrected (the participant made the antisaccade after initially failing to inhibit the prosaccade), incorrect (the participant failed to inhibit the prosaccade), or dropped (unable to be scored, e.g., due to signal dropout or excessive blinks). Primary measures of interest included the proportions of correct and error-corrected antisaccade trials among those trials that were not dropped, as well as the average saccade latency, measured as the time (in ms) from the stimulus onset to the onset of the saccade, for each of these trial types.

### 2.2.3 Neuroimaging data acquisition and preprocessing

Neuroimaging data were acquired on a 3 Tesla (3T) Siemens Allegra scanner with a standard 8-channel, radio-frequency (RF) head coil at the Neuroscience Imaging Center at the University of Pittsburgh. A sagittal magnetization-prepared rapid gradient echo (MPRAGE) T1-weighted pulse sequence (repetition time (TR) = 1630 ms, echo time (TE) = 2.48 ms, flip angle =  $8^{\circ}$ , voxel size = 0.8 mm isotropic) was used to obtain structural images with which to coregister functional images. Functional images were obtained using an echo-planar imaging (EPI) sequence sensitive to BOLD contrast (TR = 1500 ms, TE = 25 ms, flip angle =  $70^{\circ}$ , voxel size =  $3.1 \times 3.1 \times 4$  mm, 29 contiguous axial slices aligned parallel to the anterior-posterior commissure (AC-PC) plane acquired during each TR). Each functional run was 6 min, 12 s in duration.

Data preprocessing followed our laboratory's standard pipeline, incorporating tools from FM-RIB Software Library (FSL) and Analysis of Functional Neuroimages (AFNI) [76]. Functional images were corrected for slice-timing, and motion using a rigid-body rotation and translation algorithm. Functional and structural images were coregistered and aligned to the Montreal Neurological Institute (MNI) 2.3 mm template using a series of affine and nonlinear transformations. Functional images were spatially smoothed using a weighted 5 mm full-width at half-maximum Gaussian kernel, and a 0.025 Hz high-pass temporal filter was applied to remove low-frequency signal drifts due to noise. Finally, the voxel time series was normalized to permit calculation of

percent signal change. Preprocessed fMRI data were analyzed using voxelwise GLMs performed in AFNI. These analyses were performed to estimate changes in the BOLD signal associated with correct and error-corrected antisaccade trials relative to baseline fixation trials in each voxel. Nuisance regressors included baseline signal drifts, mean white matter and cerebrospinal fluid time courses, and six standard motion parameters (translations along the x, y, and z axes, and roll, pitch, and yaw rotations). Subject motion was also addressed by censoring TR values with significant motion.

We examined activation in the same brain regions of interest (ROIs) reported in Ordaz and colleagues [22]. These ROIs included canonical regions involved in inhibitory control, but not specific to the antisaccade task. In particular, this encompassed executive regions including the bilateral dorsolateral and ventrolateral prefrontal cortex (dlPFC and vlPFC, respectively); motor response regions including the supplementary eye fields (SEF), pre-supplementary motor area (pre-SMA), and bilateral frontal eye fields (FEF), posterior parietal cortex (pPC), putamen; and performance and error monitoring regions including the dorsal anterior cingulate cortex (dACC). ROIs were created based on coordinates obtained from Neurosynth [77], a platform for reverse-inference meta-analyses of fMRI studies. ROIs were defined as voxels within a 7-12 mm radius of the respective mean peak coordinates obtained from Neurosynth. The previously estimated beta coefficients reflecting the response magnitude for each voxel within each ROI were averaged to produce a mean percent signal change measure for each ROI.

#### **2.2.4 WHO-QOL questionnaire**

Participants completed the WHO-QOL questionnaire in-person at their final study visit, or, for those who were no longer regularly participating (e.g., due to relocation), the questionnaire was completed via an online survey platform. The WHO-QOL questionnaire is a self-report measure assessing subjective health and well-being in physical, psychological, social, and environmental domains. The questionnaire contains 26, 5-point Likert scale items. The first item is an overall rating of the participant's quality of life, and the second item is an overall rating of the participant's satisfaction with their health. The subsequent items contribute to the calculation of four domain-specific quality of life scores. Domain-specific scores were calculated from the raw questionnaire data according to scoring guidelines in the WHO-QOL manual. This included replacing missing values for items corresponding to a given domain with the mean of the participant's responses to the other items in the same domain, and summing response values across individual items

corresponding to each domain in order to obtain four domain-specific quality of life scores (namely, physical, psychological, social, and environmental) for each participant. Domain-specific scores were z-scored such that all domains were on comparable scales. This was necessary because the number of items and the range of possible scores varied across each of the four domains, and thus domain scores were not directly comparable. These four standardized domain-specific scores were then averaged for each participant to create an overall or composite measure of health-related quality of life for each individual.

## 2.3 Statistical analyses

### 2.3.1 Generalized additive mixed models

The first objective of this thesis was to characterize the normative adolescent development of inhibitory control, and the brain regions that support this cognitive process, in our longitudinal sample. Outcome variables of interest were modeled separately and included each of the following behavioral and brain measures: proportion correct antisaccade trials, proportion error-corrected antisaccade trials, average saccade latency on correct trials, average saccade latency on error-corrected trials, activation (percent BOLD signal change) during correct trials in dlPFC, vlPFC, SEF, pre-SMA, FEF, pPC, putamen, dACC, and activation during error-corrected trials in dACC. The primary covariate of interest in each model was age, which we expected, based on previous findings, may exhibit a nonlinear relationship with each outcome. Age was centered to permit meaningful interpretation of model intercept terms. Given that there are notable between-sex differences in brain structure and function [78], and that maternal education is a strong predictor of brain and cognitive development [79, 80], we assessed whether the potential confounding variables of sex (female, male) and maternal education level at the time of the subject’s baseline scan (categorized as completed high school, college, or graduate/professional school) should be controlled for in each model.

The generalized additive mixed model (GAMM) [59] is a flexible, semiparametric method for identifying and estimating nonlinear effects of covariates on the outcome variable when observations

are not independent, such as with longitudinal data. The key feature of GAMMs is that the mean of the outcome variable depends on the covariates through a sum of smooth terms [53, 56].

$$g(\mu_{it}) = \beta_0 + s_1(x_{1it}) + \dots + s_p(x_{pit}) + z'_{it} b_i \quad (2.1)$$

where  $\mu_{it} = E(Y_{it})$  is the conditional expectation of the response variable  $Y$  for subject  $i$ ,  $i = 1, \dots, n$  at time  $t$ ,  $t = 1, \dots, T_i$ ;  $g(\cdot)$  is the link function;  $s_j(\cdot)$  are unknown smooth functions to be estimated;  $x_{j�}$  is the  $j^{th}$  of  $p$  covariates associated with fixed effects;  $z'_{it}$  is a vector of  $q$  covariates associated with random effects; and  $b_i$  is a vector of random effects [60]. Splines are most commonly used to represent these smooth functions [53, 54]. Using splines, each smooth function  $s_j(x_j)$  can be represented by a linear combination of a set of simpler functions that have a known, closed form. These functions are referred to as basis functions.

$$s_j(x_j) = \sum_k^{K_j} \beta_{jk} b_{jk}(x_j) \quad (2.2)$$

where  $K_j$  is the basis dimension (the number of basis functions in the set),  $b_{jk}(x_j)$  are specified basis functions, and  $\beta_{jk}$  are parameters to be estimated [53]. Penalized maximum likelihood is used to estimate these coefficients, and in practice this is typically accomplished using penalized iteratively re-weighted least squares. The penalty terms, or smoothing parameters  $\lambda = \lambda_1, \dots, \lambda_j$  balance smoothness and flexibility [53]. If  $\lambda$  is too large, oversmoothing occurs, while, conversely, if  $\lambda$  is too small, overfitting occurs. Thus, it is crucial to choose optimal values for  $\lambda$ , and this can be accomplished using generalized cross validation [53].

There are several broad classes of splines that can be employed in GAMMs, including regression splines, natural splines, and smoothing splines. Regression and natural splines generate a curve composed of  $k$  segments, which are constructed in between  $k+1$  points, or “knots;” the  $k$  individual curves are joined together at these knots such that the entire curve is continuous [53]. However, a disadvantage of these spline methods is that they require the researcher to specify the number and spacing of these knots, which can be problematic because model fit often depends on knot placement [53].

Smoothing splines, particularly thin plate splines, are optimal smoothers. Because thin plate splines place a knot at every data point, they do not require the specification of the number or location of knots as do regression or natural splines. Thin plate splines are often referred to as “full-rank” because the basis dimension is equal to the number of observations [55]. However, as

a result, or a sample of  $n$  data points, thin plate splines require  $n$  parameters plus an additional smoothing parameter to be estimated, and thus there is a significant associated computational cost [53, 56]. Thin plate regression splines [56] offer an attractive alternative to thin plate splines. Thin plate regression splines approximate thin plate splines by “truncating” the full-rank basis of thin plate splines in order to obtain a lower-rank smoother, retaining the advantages thin plate splines provide (e.g., no knot placement) while improving computational efficiency [53, 56].

In GAMMs, three types of random effects are permitted to be specified. These include random intercepts, which characterize subject-specific deviation from the mean or population-average response; random slopes, which characterize subject-specific deviation from the population-average covariate effect over time; and random or factor smooths, which characterize subject-specific smooth functions of the covariate; these smooths share the same smoothing parameter  $\lambda$  but may have different shapes [53, 61]. As the random smooths may inherently capture differences in intercepts or slopes, random smooths for a given covariate are generally not used in conjunction with random intercepts and slopes for that same covariate [61, 81].

In the present analyses, the *mgcv* package for R (version 1.8-31 [57]) was used to fit a series of GAMMs for the outcomes of interest, each with a smooth function of age as a covariate, using a thin plate regression spline basis to estimate this smooth function. Random effects in each GAMM included subject-specific intercepts and slopes for age. Because the antisaccade accuracy measures are proportions, the quasi-binomial family and logit link function was used for these outcomes. For all other outcomes, which are continuous, a Gaussian family and identity link function was used. When evaluating the significance of each smooth term for age, the Benjamini-Hochberg procedure [82] was used to account for multiple comparisons by controlling the false discovery rate. However, because the aim of this set of analyses is to characterize development, age was retained in each model even if it did not have a statistically significant effect on the respective outcome measure. Each model was also refit including additional fixed effects for the potential confounders of sex and maternal education level. The significance of these covariates, determined by t tests (sex) or partial F tests (maternal education level), were used as criteria to determine if these confounders should be retained in each of the final models.

For those models for which there was a statistically significant effect of age, the *gratia* package for R (version 0.3.0 [83]) was used to conduct exploratory post-hoc analyses to identify significant periods of developmental change. Specifically, the derivatives of each estimated smooth function of age were approximated using the method of finite differences, and a simultaneous 95% confidence

band for the derivatives were calculated. The age range(s) for which each 95% confidence band did not contain zero was considered to represent a period of significant developmental change. The random intercepts and slopes from each of these final models were extracted and saved for use in subsequent analyses.

### 2.3.2 Bootstrap-enhanced elastic net regression

The second objective of this thesis was to determine how individual differences in the adolescent development of inhibitory control and brain regions supporting this cognitive process predict subsequent health-related quality of life in early adulthood. The outcome measure for this analysis was the standardized composite quality of life score obtained by averaging the four standardized domain-specific scores from the WHO-QOL questionnaire. Covariates included random slopes and intercepts extracted from those GAMM models for which there was a significant effect of age. Random intercepts are considered to represent person-specific deviations from the mean level of the respective measure of inhibitory control at the mean age of the sample, while random slopes are considered to represent person-specific deviations in the rate of development of that measure. Because potential confounders were controlled for at the GAMM step, no confounding variables were controlled for in this portion of the analysis.

Because the number of predictors ( $p_{\max} = 36$ ) could potentially have approached the size of our sample ( $n = 50$ ), a regularized regression method was appropriate. Elastic net regression is a regularized regression method that utilizes a linear combination of the  $\ell_2$  penalty of ridge regression and the  $\ell_1$  penalty of lasso regression. In naive elastic net regression, coefficients are estimated by minimizing the following objective.

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda_1 \sum_{j=1}^p |\beta_j| + \lambda_2 \sum_{j=1}^p \beta_j^2 \quad (2.3)$$

where  $y_i$  represents the response for the  $i^{th}$  of  $n$  subjects,  $x_{ij}$  represents the value of the  $j^{th}$  of  $p$  covariates for subject  $i$ ,  $\lambda_1$  is the tuning parameter corresponding to the  $\ell_1$  lasso penalty, and  $\lambda_2$  is the tuning parameter corresponding to the  $\ell_2$  ridge penalty. The  $\ell_1$  portion of the elastic net penalty generates a sparse model (i.e., allows for variable selection), while the  $\ell_2$  portion encourages a grouping effect by which highly correlated predictors tend to be included or excluded from the model in aggregate [71].

The above objective can be reparameterized as the following.

$$\sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij} \right)^2 + \lambda \left( \frac{1-\alpha}{2} \sum_{j=1}^p \beta_j^2 + \alpha \sum_{j=1}^p |\beta_j| \right) \quad (2.4)$$

where  $\alpha \in [0, 1]$  is a “mixing” parameter [71]. When  $\alpha = 0$ , the solution to the above is equivalent to that of ridge regression, and when  $\alpha = 1$ , the solution to the above is equivalent to that of lasso regression. Thus the  $\alpha$  parameter controls the relative contributions of the ridge and lasso penalty terms. The tuning parameter  $\lambda$  controls the overall strength of the elastic net penalty [71]. Optimal values for these hyperparameters can be determined using k-fold cross validation. In k-fold cross validation, the dataset is randomly split into  $k$  partitions or “folds,” where  $k$  is typically equal to 5 or 10. On each of the  $k$  iterations,  $k - 1$  folds are used in combination to train the model, and the remaining fold is reserved as a test fold to validate the model. A cross-validated accuracy measure can be obtained by averaging the mean squared errors (MSE) computed from each of the test folds [66].

$$CV_{(k)} = \frac{1}{k} \sum_{i=1}^k \text{MSE}_i \quad (2.5)$$

Using grid search, the k-fold cross validation procedure is repeated for each possible combination of  $\alpha$  and  $\lambda$  values. The pair of  $\alpha$  and  $\lambda$  values that result in the minimum cross-validated error are considered the optimal hyperparameter values.

One disadvantage associated with regularized regression approaches, including elastic net regression, is that confidence intervals and p-values are generally not available for the coefficients estimated by these methods, and thus inference is not possible [84]. However, a bootstrap-enhanced elastic net method was developed by Bunea and colleagues [85], and recently expanded by Abram and colleagues [86], in order to overcome this issue. In the bootstrap-enhanced elastic net method,  $B$  bootstrap samples of size  $n$  are randomly sampled from the data with replacement. The elastic net model is then refit separately to each of the  $B$  bootstrap samples to produce  $B$  different estimates of each of the coefficients. The distributions of these coefficient estimates (i.e., bootstrap distributions) can then be used to assess the importance of the associated predictors, by one of two approaches.

In the quantile approach [86], the quantiles of the bootstrap distribution of  $\beta_j$  are used to determine the significance of the  $j^{th}$  predictor. Assuming a significance level  $\alpha^*$ , the  $100(1 - \alpha^*)\%$  bootstrap confidence interval for  $\beta_j$  is given by  $[Q_{j,\alpha^*/2}, Q_{j,1-\alpha^*/2}]$ , where  $Q_{j,\alpha}$  is the quantile value

such that  $\alpha = \frac{1}{B} \sum_{b=1}^B I(\hat{\beta}_{jb} \leq Q_{j,\alpha})$ , where  $I(\hat{\beta}_{jb} \leq Q_{j,\alpha})$  is an indicator function that equals 1 if  $\hat{\beta}_{jb} \leq Q_{j,\alpha}$  and equals 0 otherwise, and  $\hat{\beta}_{jb}$  is the estimated coefficient for the  $j^{th}$  predictor on the  $b^{th}$  bootstrap iteration [86]. If the  $100(1 - \alpha^*)\%$  bootstrap confidence interval for a given coefficient does not contain zero, the associated variable is considered to be a statistically significant predictor of the outcome.

Alternatively, the variable inclusion probability (VIP) approach [85] can be utilized. The VIP quantifies the importance of a given predictor as the proportion of the  $B$  iterations of the bootstrap procedure on which the predictor received a nonzero coefficient estimate. Assuming the significance level  $\alpha^*$ , a VIP greater than  $(1 - \alpha^*)\%$  indicates that a variable is a statistically significant predictor. The VIP may also be interpreted in a Bayesian context. Specifically, assuming a Laplace prior on  $\beta_j$ , the VIP reflects the posterior probability of the  $j^{th}$  predictor being included in the model [85].

$$\text{VIP}_j = P(\beta_j \neq 0 \mid \text{Data}), \quad j = 1, \dots, p \quad (2.6)$$

The bootstrap-enhanced elastic net procedure was utilized in the present analyses because it allows for automatic variable selection and also provides methods for determining which variables are important or statistically significant predictors of the outcome. The *ensr* (version 0.1.0 [87]) and *glmnet* (version 3.0-2 [88]) packages for R were used to tune hyperparameters and fit the optimal elastic net regression model, respectively. Grid search was used in conjunction with ten-fold cross validation to perform simultaneous hyperparameter tuning. The elastic net model was fit using the pair of hyperparameter values which minimized the cross-validated mean squared error (CV-MSE). The model was then refit on  $B = 5000$  bootstrap samples of size  $n = 50$  to obtain bootstrap distributions for each of the estimated coefficients. The 2.5<sup>th</sup> and 97.5<sup>th</sup> percentiles of each distribution were calculated and used to construct 95% bootstrap confidence intervals for each coefficient. Those predictors for which the 95% bootstrap confidence interval did not contain zero were considered statistically significant. Additionally, variable inclusion probabilities (VIP) were calculated for each predictor as the proportion of the 5000 bootstrap iterations on which the predictor received a nonzero coefficient estimate. A VIP greater than or equal to 95% was also considered as a criterion for statistical significance. To visualize and better understand potential associations between predictors and health-related quality of life, a post-hoc analysis included performing a median split to categorize participants into high and low quality of life groups. Individual GAMMs were then refit including a smooth age by quality of life group interaction term, and estimated developmental trajectories for each quality of life group were plotted.

### 3.0 Results

#### 3.1 Characterizing adolescent development of inhibitory control

The first objective of the present analysis was to characterize the normative adolescent development of inhibitory control in our longitudinal sample. A separate GAMM was fit with a smooth function of age as a covariate, and subject-specific intercepts and slopes for age included as random effects, for each outcome measure of interest including proportion correct antisaccade trials, proportion error-corrected antisaccade trials, average saccade latency on correct trials, average saccade latency on error-corrected trials, activation (percent BOLD signal change relative to baseline) during correct trials in dlPFC, vIPFC, SEF, pre-SMA, FEF, pPC, putamen, and dACC, and activation during error-corrected trials in dACC. Each GAMM was also refit separately including additional fixed effects for the potential confounders of sex (Table 1) and maternal education level (Table 2) to determine if either or both of these covariates should be used as adjustments in the respective final models. For each final model, the effective degrees of freedom (edf), F statistic, p-value, and false discovery rate adjusted p-value (q-value) for the smooth term for age are reported in Table 3. All statistically significant findings described below survived correction for multiple comparisons (i.e., q-value < 0.05) unless otherwise specified.

##### 3.1.1 Behavioral measures

There were statistically significant effects of age for the proportion of correct antisaccade trials ( $p < 0.0001$ ), average latency on correct antisaccade trials ( $p < 0.0001$ ), and the proportion of error-corrected antisaccade trials ( $p < 0.0001$ ; Table 3). There was not a significant effect of age for average latency on error-corrected antisaccade trials ( $p = 0.0550$ ; Table 3), although age was retained in the model because age is the primary covariate of interest. In the models with age as the covariate of interest, there were no statistically significant effects of sex for the proportion of correct antisaccade trials, the proportion of error-corrected antisaccade trials, average latency on correct antisaccade trials, or average latency on error-corrected antisaccade trials ( $p \geq 0.1971$ ; Table 1). Additionally, in the models with age as the covariate of interest, there were no statistically significant global effects of maternal education level for any of the behavioral measures ( $p \geq 0.1080$ );

Table 2). Therefore, the final models for each of the behavioral measures included only the smooth term for age (Figure 2). For those outcome measures for which there were significant effects of age, post-hoc analyses indicated that the proportion of correct antisaccade trials increased from age 8.45 to 21.36 years, while average latency on correct antisaccade trials and the proportion of error-corrected antisaccade trials decreased from 8.45 to 23.01 years, and from 11.48 to 20.98 years, respectively.

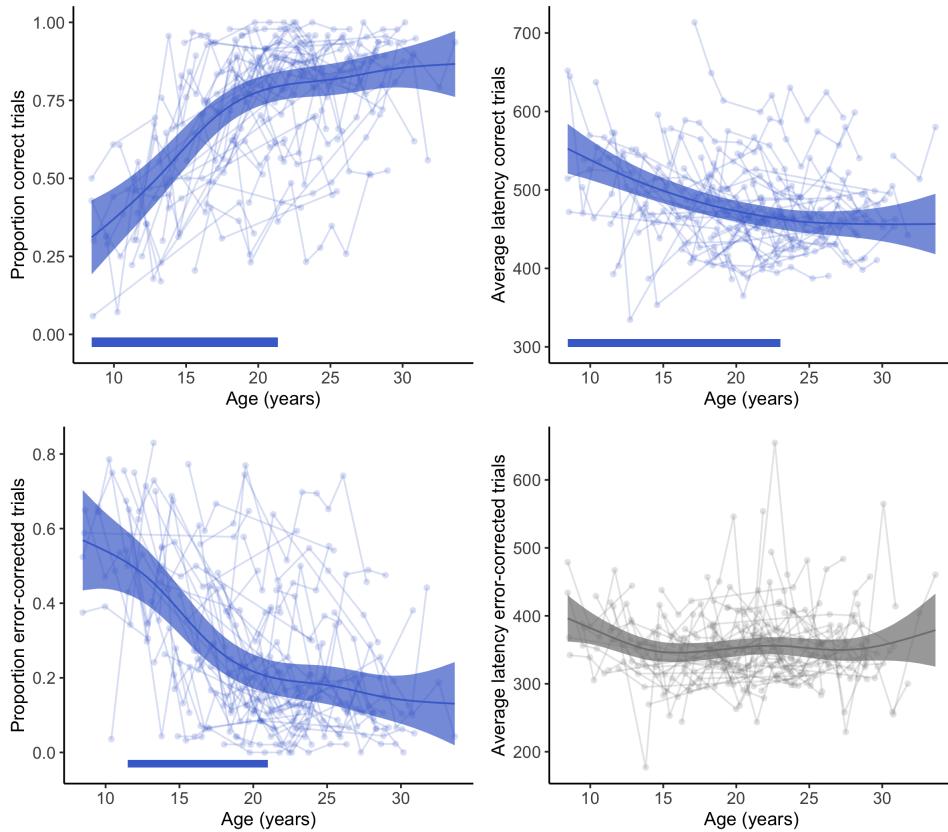


Figure 2: Effects of age for each behavioral measure. A spaghetti plot of individual subject trajectories for each behavioral measure is overlaid with a plot of fitted values and point-wise 95% confidence intervals for the smooth function of age estimated in each respective GAMM. Plots displayed in blue indicate that the age effect was statistically significant, with the bar above the x-axis indicating the period of significant developmental change. Plots displayed in grey indicate that the age effect was not statistically significant. There were significant effects of age for the proportions of correct (top left) and error-corrected (bottom left) antisaccade trials as well as average saccade latency on correct antisaccade trials (top right), continuing into young adulthood

### 3.1.2 Brain function measures

**3.1.2.1 Motor response regions** Among the motor response ROIs, there were statistically significant effects of age for activation during correct antisaccade trials in L FEF ( $p = 0.0142$ ), L pPC ( $p = 0.0060$ ), and R pPC ( $p = 0.0055$ ); there were no statistically significant age effects in SEF ( $p = 0.9644$ ), pre-SMA ( $p = 0.3244$ ), R FEF ( $p = 0.0785$ ), L putamen ( $p = 0.5025$ ), or R putamen ( $p = 0.7368$ ; Table 3). In the models with age as the covariate of interest, there were no statistically significant effects of sex ( $p \geq 0.0705$ ; Table 1) or maternal education level ( $p \geq 0.5705$ ; Table 2) on activation in any of the motor response ROIs. Therefore, the final models for activation in each of the motor response ROIs includes only the smooth term for age (Figure 3). For those ROIs for which there were significant age effects, post-hoc analyses indicated that activation decreased from age 10.98 to 16.93 years in L FEF, from 11.11 to 17.31 years in L pPC, and from 9.71 to 16.67 years in R pPC.

**3.1.2.2 Executive control regions** Among the executive control ROIs, there was a statistically significant effect of age only for activation during correct antisaccade trials in R dlPFC ( $p = 0.0064$ ; Table 3). There were no significant effects of age in L dlPFC ( $p = 0.9828$ ), L vIPFC ( $p = 0.7332$ ), or R vIPFC ( $p = 0.0581$ ; Table 3). Additionally, in the models with age as the covariate of interest, there were no statistically significant effects of either sex ( $p \geq 0.1697$ ; Table 1) or maternal education level ( $p \geq 0.3502$ ; Table 2) in any of the executive control ROIs. Therefore, the final models for activation in each of the executive control ROIs includes only the smooth term for age (Figure 4). For the ROI for which there was a significant age effect, a post-hoc analysis indicated that there were developmental decreases in R dlPFC activation from age 10.60 to 16.67 years.

**3.1.2.3 Performance monitoring region** In the performance monitoring ROI, dACC, there was not a significant effect of age for activation during correct antisaccade trials ( $p = 0.3770$ ; Table 3). Additionally, in the model with age as the covariate of interest, there were no statistically significant effects of sex ( $p = 0.3185$ ; Table 1) or maternal education level ( $p = 0.0847$ ; Table 2). Therefore, the final model for dACC activation during correct antisaccade trials includes only the smooth term for age (Figure 5).

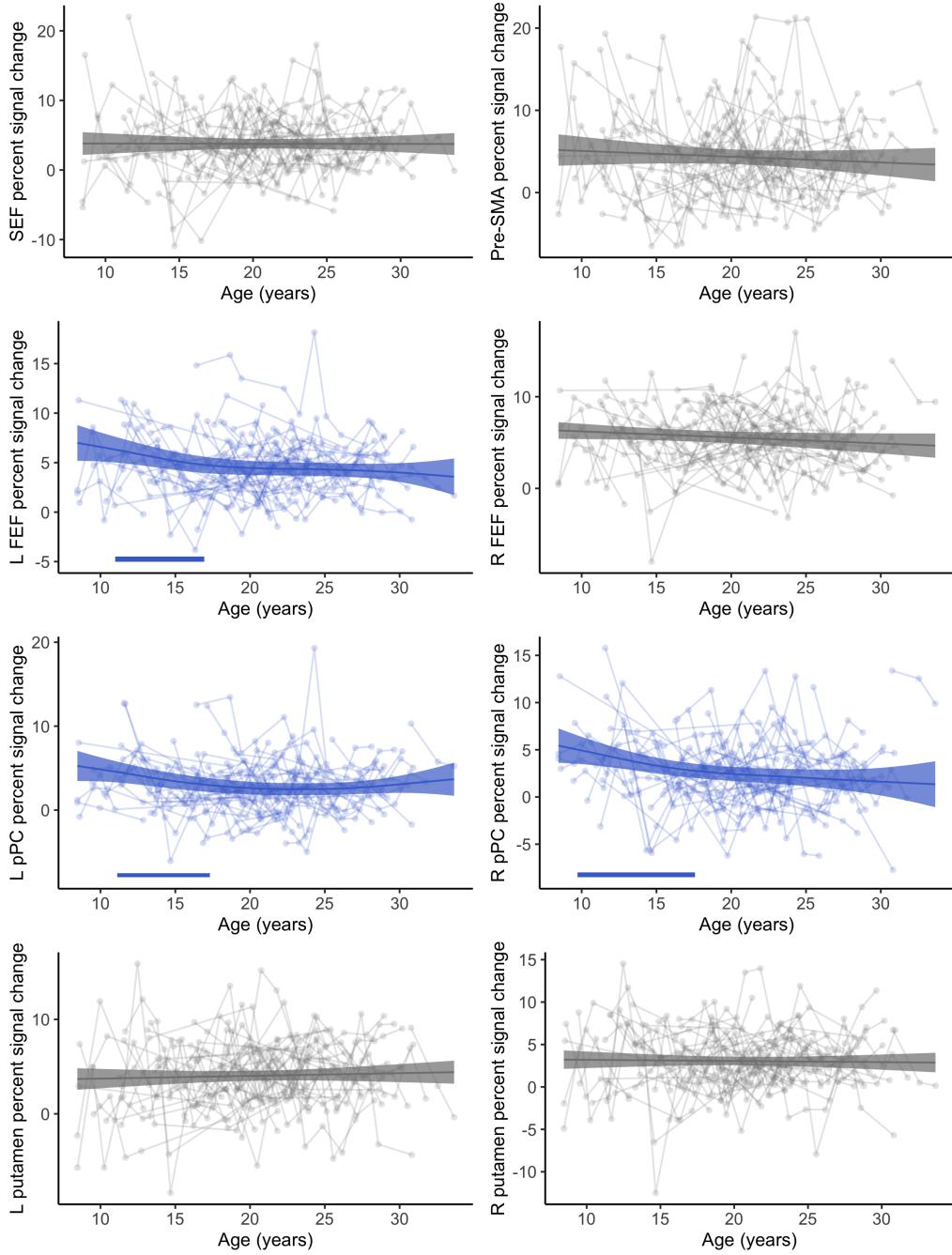


Figure 3: Effects of age for activation during correct antisaccade trials in each motor response region. A spaghetti plot of individual subject trajectories for activation (percent signal change) in each motor response ROI is overlaid with a plot of fitted values and point-wise 95% confidence intervals for the smooth function of age estimated with the respective final GAMM. There were significant effects of age for activation during correct antisaccade trials in L FEF, L pPC, and R pPC (displayed in blue), with age-related change continuing into mid-adolescence

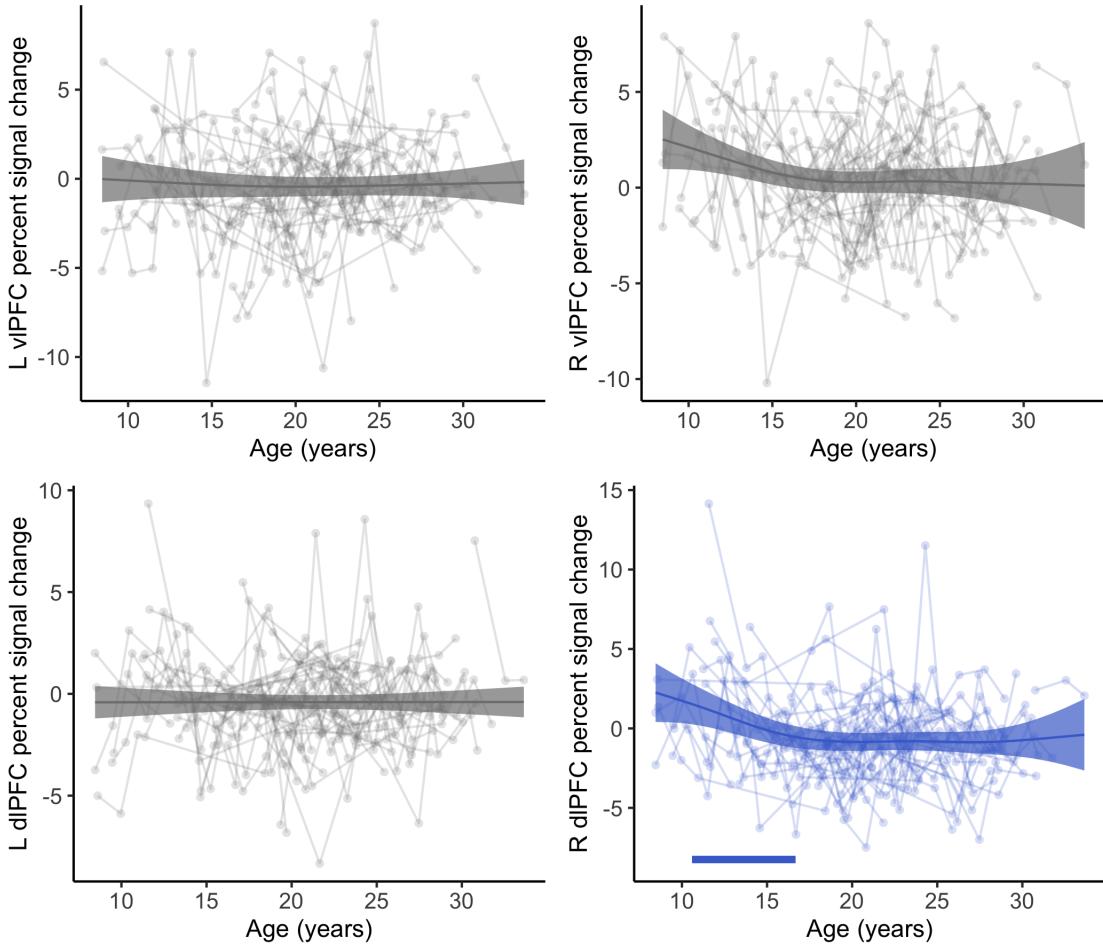


Figure 4: Effects of age for activation during correct antisaccade trials in each executive control region. A spaghetti plot of individual subject trajectories for activation (percent signal change) in each executive control ROI is overlaid with a plot of fitted values and point-wise 95% confidence intervals for the smooth function of age estimated with the respective final GAMM. Plots displayed in blue indicate that the age effect was statistically significant, with the bar above the x-axis indicating the period of significant developmental change. Plots displayed in grey indicate that the age effect was not statistically significant. There was a significant effect of age for activation during correct antisaccade trials only in R dlPFC (bottom right), with age-related change continuing into mid-adolescence

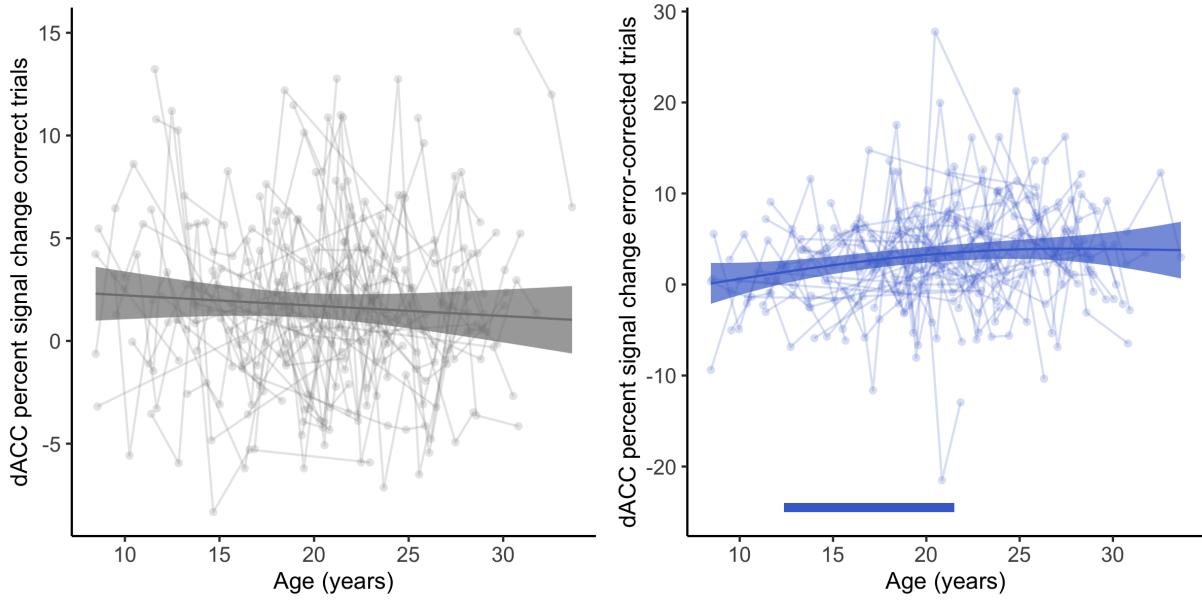


Figure 5: Effects of age for activation in the performance monitoring region, dACC. A spaghetti plot of individual subject trajectories for activation (percent signal change) during correct (left) and error-corrected (right) antisaccade trials in dACC is overlaid with a plot of fitted values and point-wise 95% confidence intervals for the smooth function of age estimated with the respective final GAMM. Adjusting for sex, there was a significant effect of age for dACC activation during error-corrected antisaccade trials, with age-related change continuing into young adulthood

There was, however, a significant effect of age in dACC for activation during error-corrected antisaccade trials ( $p = 0.0184$ ; Table 3). In the model with age as the covariate of interest, there was a significant effect of sex ( $p = 0.0161$ ; Table 1), with average dACC activation during error-corrected antisaccade trials about 1.67% higher in males compared to females. There was not a significant effect of maternal education level ( $p = 0.3654$ ; Table 2). Therefore, the final model for dACC activation during error-corrected antisaccade trials includes both the smooth term for age and sex as covariates (Figure 5); including sex in the model did not substantially change the significance of the smooth term for age ( $p = 0.0120$ ). A post-hoc analysis indicated that, adjusting for sex, there were developmental increases in activation in dACC during error-corrected antisaccade trials from age 12.37 to 21.49 years.

Table 1: Estimated coefficients and significance of sex for each outcome measure, controlling for age

Outcome measure	$\hat{\beta}$	t	p-value
Proportion correct trials	0.0873	0.3424	0.7322
Proportion error-corrected trials	-0.0698	-0.2851	0.7756
Average latency correct trials	-3.9599	-0.3069	0.7590
Average latency error-corrected trials	-12.8374	-1.2925	0.1971
SEF activation correct trials	-1.2471	-1.8144	0.0705
Pre-SMA activation correct trials	0.0283	0.0304	0.9757
L FEF activation correct trials	0.1769	0.2812	0.7787
R FEF activation correct trials	0.2924	0.4442	0.6571
L putamen activation correct trials	0.5367	0.9830	0.3263
R putamen activation correct trials	0.3923	0.7509	0.4532
L pPC activation correct trials	-0.6196	-0.9154	0.3606
R pPC activation correct trials	-0.8914	-1.2740	0.2035
L dlPFC activation correct trials	0.0617	0.1728	0.8628
R dlPFC activation correct trials	0.0925	0.1932	0.8468
L vlPFC activation correct trials	-0.6830	-1.3485	0.1784
R vlPFC activation correct trials	0.6435	1.3760	0.1697
dACC activation correct trials	-0.6099	-0.9989	0.3185
dACC activation error-corrected trials	1.6726	2.4183	<b>0.0161</b>

Table 2: Global significance of maternal education for each outcome measure, controlling for age

Outcome measure	df	F	p-value
Proportion correct trials	2	0.2916	0.7472
Proportion error-corrected trials	2	0.3686	0.6919
Average latency correct trials	2	2.2412	0.1080
Average latency error-corrected trials	2	0.7669	0.4653
SEF activation correct trials	2	0.0813	0.9219
Pre-SMA activation correct trials	2	0.2180	0.8042
L FEF activation correct trials	2	0.5621	0.5705
R FEF activation correct trials	2	0.4468	0.6400
L putamen activation correct trials	2	0.4275	0.6525
R putamen activation correct trials	2	0.2743	0.7602
L pPC activation correct trials	2	0.0316	0.9688
R pPC activation correct trials	2	0.4523	0.6365
L dlPFC activation correct trials	2	0.8012	0.4497
R dlPFC activation correct trials	2	0.7833	0.4577
L vlPFC activation correct trials	2	0.1464	0.8638
R vlPFC activation correct trials	2	1.0525	0.3502
dACC activation correct trials	2	2.4872	0.0847
dACC activation error-corrected trials	2	1.0098	0.3654

Table 3: Significance of smooth term for age for each final GAMM

Outcome measure	edf	F	p-value	q-value
Proportion correct trials	3.64	24.43	< 0.0001	< <b>0.0001</b>
Proportion error-corrected trials	3.87	17.22	< 0.0001	< <b>0.0001</b>
Average latency correct trials	2.91	12.94	< 0.0001	< <b>0.0001</b>
Average latency error-corrected trials	4.02	2.29	0.0550	0.1046
SEF activation correct trials	1.00	0.00	0.9644	0.9828
Pre-SMA activation correct trials	1.00	0.97	0.3244	0.4867
L FEF activation correct trials	2.67	4.44	0.0142	<b>0.0320</b>
R FEF activation correct trials	1.00	3.11	0.0785	0.1285
L putamen activation correct trials	1.00	0.45	0.5025	0.6461
R putamen activation correct trials	1.00	0.11	0.7368	0.8289
L pPC activation correct trials	2.74	4.49	0.0060	<b>0.0193</b>
R pPC activation correct trials	2.30	4.63	0.0055	<b>0.0193</b>
L dlPFC activation correct trials	1.00	0.00	0.9828	0.9828
R dlPFC activation correct trials	3.15	4.20	0.0064	<b>0.0193</b>
L vlPFC activation correct trials	1.54	0.27	0.7332	0.8289
R vlPFC activation correct trials	2.57	2.23	0.0581	0.1046
dACC activation correct trials	1.00	0.84	0.3570	0.4948
dACC activation error-corrected trials*	1.51	4.55	0.0120	<b>0.0309</b>

\*Model adjusted for sex

### 3.2 Predicting health-related quality of life

The second objective of the present analysis was to determine how individual differences in the adolescent development of inhibitory control, as estimated using GAMMs, may predict subsequent health-related quality of life in early adulthood. Random intercepts and slopes from the final GAMM models for which there was a significant effect of age on the respective outcome measure, after controlling for the false discovery rate, were standardized and used as candidate predictors in

a bootstrap-enhanced elastic net regression model to predict the composite health-related quality of life score derived from the WHO-QOL questionnaire. Random intercepts are considered to represent person-specific deviations from the mean level of the respective outcome measure at the mean age of the sample (20.71 years), while random slopes are considered to represent person-specific deviations in the rate of development of that measure. The candidate predictors for the elastic net model included the standardized random intercepts and slopes for the proportion of correct trials on the antisaccade task, proportion of error-corrected trials on the antisaccade task, average latency on correct trials, activation (percent BOLD signal change relative to baseline) during correct antisaccade trials in L FEF, L pPC, R pPC, R dLPFC, and activation during error-corrected antisaccade trials in dACC.

Ten-fold cross validation was first used to simultaneously tune the elastic net hyperparameters,  $\alpha$  and  $\lambda$ , which control the relative contributions of the ridge and lasso penalty terms, and the level of shrinkage, respectively. The hyperparameter values that minimized the cross-validated MSE (CV-MSE) were  $\hat{\alpha} = 0.1$  and  $\hat{\lambda} = 0.7753$ , with CV-MSE = 0.9866. With these optimal hyperparameter values, the elastic net model selected 10 of the 16 candidate predictors, including random intercepts and slopes for the proportion of correct antisaccade trials, proportion of error-corrected antisaccade trials, average latency on correct antisaccade trials, and activation during correct antisaccade trials in L FEF, as well as random intercepts for activation during correct antisaccade trials in R pPC and R dLPFC. Estimated regression coefficients for the fitted model are presented in Table 4. A coefficient estimate exactly equal to zero indicates that the predictor was not selected for inclusion in the model. The model had an  $R^2 = 0.1884$ , indicating that only about 18.84% of the total variability in composite health-related quality of life score was explained by the elastic net model containing this set of selected covariates.

Results from the bootstrap-enhanced procedure, in which the model was refit on  $B = 5000$  bootstrap samples, suggested that none of the selected covariates were, given the other covariates in the model, statistically significant predictors of composite health-related quality of life score (Table 4). Specifically, all of the 95% bootstrapped confidence intervals (CI) for the estimated regression coefficients contained zero, suggesting that none of these coefficients significantly differed from zero. Additionally, the variable inclusion probability (VIP) for each predictor, which quantifies the importance of that predictor as the proportion of times, out of the 5000 bootstrap iterations, the predictor received a non-zero coefficient estimate, were also below 95%. Although below this chosen threshold for statistical significance, several predictors had relatively high VIPs, including random

slopes for activation during correct antisaccade trials in L FEF (VIP = 0.8916) and the proportion of error-corrected antisaccade trials (VIP = 0.8200), as well as random intercepts for activation during correct antisaccade trials in R dlPFC (VIP = 0.8456), indicating that these individual predictors may be relatively more important.

Table 4: Estimated coefficients and bootstrapped 95% confidence intervals, and variable inclusion probabilities for each candidate predictor in the elastic net model

Predictor	$\hat{\beta}$	95% CI	VIP
Antisaccade proportion correct trials intercept	0.0286	(-0.0014, 0.1216)	0.6630
Antisaccade proportion correct trials slope	0.0225	(0.0000, 0.1012)	0.6216
Antisaccade proportion error-corrected trials intercept	-0.0247	(-0.1332, 0.0104)	0.6468
Antisaccade proportion error-corrected trials slope	-0.0712	(-0.1500, 0.0000)	0.8200
Antisaccade average latency correct trials intercept	0.0284	(0.0000, 0.1282)	0.6590
Antisaccade average latency correct trials slope	-0.0118	(-0.1647, 0.0305)	0.6178
L FEF activation correct trials intercept	-0.0334	(-0.1163, 0.0000)	0.7206
L FEF activation correct trials slope	0.0858	(0.0000, 0.1713)	0.8916
L pPC activation correct trials intercept	0	(-0.0606, 0.0281)	0.3616
L pPC activation correct trials slope	0	(-0.0111, 0.0738)	0.4216
R pPC activation correct trials intercept	-0.0735	(-0.1725, 0.0000)	0.7520
R pPC activation correct trials slope	0	(-0.1277, 0.0621)	0.5396
R dlPFC activation correct trials intercept	-0.0888	(-0.2028, 0.0000)	0.8456
R dlPFC activation correct trials slope	0	(-0.1264, 0.0520)	0.4602
dACC activation error-corrected trials intercept	0	(-0.0694, 0.0775)	0.4476
dACC activation error-corrected trials slope	0	(-0.1004, 0.0462)	0.4746

To better understand the potential associations between the three predictors with relatively high VIPs and health-related quality of life, a median split was used to categorize individuals into high and low health-related quality of life groups, the GAMMs for these three measures were refit including a smooth age by health-related quality of life group interaction term, and the estimated developmental trajectories for each group were plotted (Figure 6).

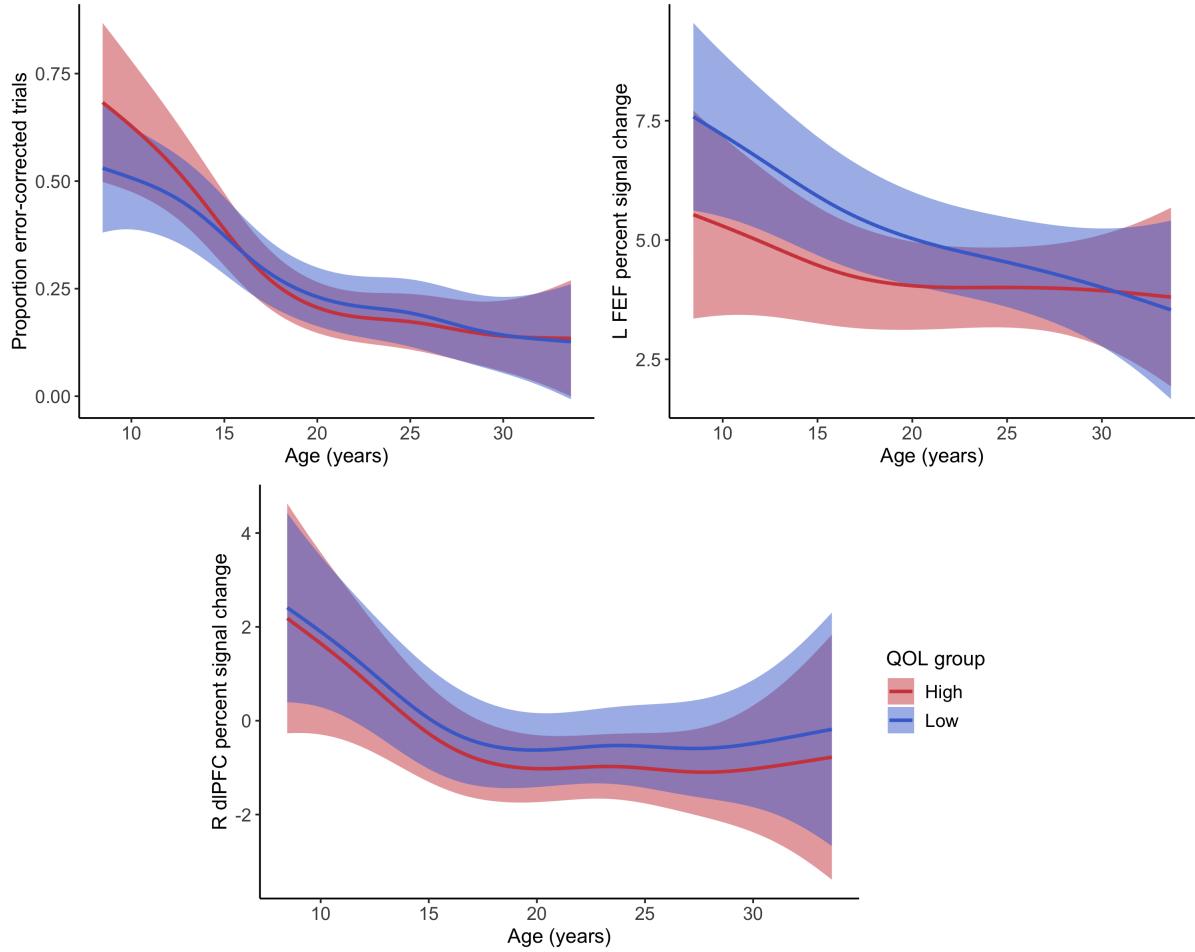


Figure 6: Estimated developmental trajectories in high (red) and low (blue) health-related quality of life (QOL) groups. Individuals with high QOL in adulthood corrected more antisaccade errors early in development than did those with low QOL in adulthood, although trajectories converged by adolescence (top left). Individuals with low QOL in adulthood had greater activation in L FEF (top right) and R dlPFC (bottom) over development compared to those with high QOL in adulthood

For the two measures for which the random slope terms were important in the elastic net model, namely, the proportion of error-corrected antisaccade trials and activation in L FEF, the plots suggest between-group differences early in development with equifinality in trajectories by adulthood. Specifically, the group with high health-related quality of life in adulthood corrected a larger proportion of antisaccade errors early in development than did those with low health-related quality of life, although these differences normalized by adolescence. Individuals with low health-

related quality of life in adulthood had greater activation in L FEF during correct antisaccade trials than did those with high quality of life, and these differences continued into young adulthood. For the measure for which the random intercept term was important in the elastic net model, activation in R dlPFC, the plot suggests parallel trajectories in both the high and low health-related quality of life groups, although activation was slightly greater across age among those with low health-related quality of life in adulthood. However, none of these smooth age by health-related quality of life group interaction effects were significant ( $p \geq 0.2040$ ), and the bootstrap-enhanced elastic net procedure ultimately indicated that individual differences in the development of inhibitory control during adolescence were not predictive of health-related quality of life in early adulthood.

## 4.0 Discussion

The goals of the present study were to characterize the normative adolescent development of inhibitory control, in terms of both behavior and the function of brain regions supporting this cognitive process, and to determine how individual differences in these adolescent developmental trajectories may predict health-related quality of life in early adulthood. We examined the development of behavioral and brain function measures using GAMM, a flexible, semiparametric method that is relatively novel to functional neuroimaging research. The use of GAMMs to model the development of aspects of inhibitory control allowed us to capture nonlinear effects of age without assuming functional forms for these relationships, thus providing us with a more nuanced understanding of the shapes of developmental trajectories during adolescence and into young adulthood.

Consistent with the literature [19, 20, 22], we found significant developmental improvements in inhibitory control behaviorally, with the proportions of correctly performed and error-corrected antisaccade trials increasing and decreasing with age, respectively; average latency on correct antisaccade trials also declined with age. Post-hoc analyses revealed that developmental change in these behavioral measures continued into early adulthood, reaching mature levels as late as age 23, providing novel evidence that, behaviorally, inhibitory control may undergo a more protracted development than previously reported [20].

We also found significant effects of age for a number of brain function measures. Consistent with previous findings [22], among examined executive control regions, we found that R dlPFC activation during correct antisaccade trials significantly decreased with age. Although the majority of the examined motor response regions did not exhibit developmental changes in activation during correct antisaccade trials, significant effects of age were found for activation in motor response regions including L FEF, L pPC, and R pPC. This contrasts with previous research reporting that motor areas do not exhibit age-related changes in activation during adolescence, rather reaching mature levels of activation in childhood [22, 34]. The present finding could possibly be explained by our use of GAMMs, which may have permitted us to identify subtle age-related changes that were not evident in previous work employing linear or polynomial models, or comparisons of broad age groups. Additionally, previous findings have suggested that executive regions including the R dlPFC develop later than do motor response regions, with the former continuing into adolescence [22]. However, we found that the development of R dlPFC activation during correct antisaccade

trials occurred in parallel with that of L FEF, L pPC, and R pPC, with age-related decreases in activation in these regions beginning at approximately age 10 and continuing until approximately age 17. A previous cross-sectional study of functional connectivity associated with the antisaccade task found that the strength of long-range connections between regions in the frontal and parietal cortices increased with age [89]. Therefore, our finding of parallel developmental trajectories in such regions in the present study could reflect the integration of these regions into such a circuitry supporting inhibitory control collectively, thereby reducing processing demands on each region individually. Finally, we found that activation in dACC changed significantly with age and matured relatively later at 21.49 years, but only for error-corrected antisaccade trials, potentially reflecting its unique role in inhibitory control as a performance monitoring and error-processing region [22, 34, 35].

Importantly, the use of a mixed model approach such as GAMM in this series of analyses allowed us to not only estimate population-level effects of age on aspects of inhibitory control, but also to estimate individual differences, which were utilized as covariates in a subsequent analysis. Specifically, we estimated random slopes, representing person-specific deviations in the rate of development through adolescence of each behavioral and brain measure, as well as random intercepts, representing person-specific deviations from the mean level of each measure at the mean age of the sample, 20.71 years. Additionally, because we found that many of the examined measures reached or were approaching maturity by this age, random intercepts also, and perhaps more meaningfully, may be thought to reflect person-specific deviations from the mean level of each measure approximately at or near the age of maturation.

These estimated individual differences were utilized as candidate predictors in a bootstrap-enhanced elastic net regression approach to predict health-related quality of life in young adulthood. The use of a regularized regression method such as elastic net was advantageous due to the relatively small size of our sample, and the potential for the number of predictors to be relatively large in comparison. Further, the bootstrap-enhanced elastic net procedure allowed us to not only perform variable selection, but also to conduct inference on the selected model, which is generally not possible with regularized regression methods. The elastic net procedure selected a model including a set of 10 of the 16 candidate predictors encompassing individual differences in adolescent development of both brain and behavioral measures. However, the model had little explanatory power, explaining only about 18.84% of the variability in the composite health-related quality of life score. This was not entirely unexpected, given that quality of life is also likely to be related

to a number of sociodemographic and situational factors [38–41] in addition to cognitive factors. Further, both the 95% bootstrap confidence intervals for the estimated coefficients and the variable inclusion probabilities indicated that, conditional on the other covariates in the model, none of the selected variables were statistically significant predictors of health-related quality of life in young adulthood. Nevertheless, an examination of the variable inclusion probabilities did suggest that three predictors in particular (random slopes for the proportion of error-corrected antisaccade trials and activation in L FEF, and random intercepts for activation in R dlPFC) may be relatively important. Post-hoc analyses indicated that those with higher health-related quality of life in adulthood corrected more antisaccade errors in late childhood and early adolescence than did those with lower health-related quality of life, although trajectories converged by mid-adolescence. This suggests that greater engagement in performance monitoring early in development may be associated with better quality of life in early adulthood. Additionally, we found that those with lower health-related quality of life in adulthood had greater activation in L FEF and R dlPFC during correctly performed antisaccade trials, potentially reflecting increased effort to perform inhibitory control as compared to individuals with higher health-related quality of life. For activation in L FEF, we observed differing trajectories between groups in late childhood and adolescence, with equifinality in adulthood, while, for activation in R dlPFC, we observed parallel developmental trajectories in the two health-related quality of life groups; this is consistent with the random slope term for L FEF activation, and the random intercept term for R dlPFC activation, being relatively more important in the elastic net model. Although these results were not statistically significant, they do motivate further investigation.

Several limitations of the present study should be considered. A large portion of our initial longitudinal sample was not eligible for inclusion in these analyses. Over time, a substantial number of participants met exclusion criteria and were dropped from the study, or were lost to follow-up and thus had fewer than three analyzable time points, were unable to be contacted to complete the WHO-QOL questionnaire at the study endpoint, or both. Additionally, among our eligible sample, the total number of analyzable observations was large ( $n = 316$ ), but this comprised data from only 50 participants. This small sample size may have reduced our statistical power to identify significant effects, or conversely, reduced the likelihood that those results we found to be statistically significant actually reflected a true effect. Low statistical power is a common limitation in neuroscience research, and particularly for human neuroimaging studies [90]. There are also limitations associated with fMRI as a neuroimaging modality. FMRI data are inherently noisy,

with potential sources of noise including the participant's respiration, heart rate, and movement in the scanner, as well as scanner signal drifts and image artifacts [91]. Noise can result in reduced BOLD contrast sensitivity, thus compromising the ability to detect task-related changes in the BOLD signal [91]. Although we censored volumes with substantial motion and included motion parameters and baseline signal drifts as nuisance regressors in the voxelwise GLMs, we did not control for physiological recordings including respiration and heart rate. Finally, the scope of the present study was limited; in addition to behavioral measures of inhibitory control, we considered only activation in a small subset of brain regions identified from previous literature, and did not examine other regions or measures such as functional connectivity between regions.

In sum, by using a flexible, semiparametric approach that is relatively novel in the context of functional neuroimaging, we have extended important previous work by providing additional insight into the developmental time course of inhibitory control and the brain systems that support this cognitive process during adolescence. Our present findings indicate that the maturation of inhibitory control may occur later than previously reported, with refinements in behavior and brain function continuing through late adolescence and in some cases into early adulthood. These findings may have important implications for public health, as they suggest that there may be a greater window of opportunity during which interventions can be applied to effectively promote optimal adult outcomes. The relationship between the adolescent development of inhibitory control and subsequent health-related quality of life in young adulthood, however, remains to be elucidated. Although our findings did not reach the threshold for statistical significance, results did indicate that individual differences in the adolescent development of a small number of the examined brain and behavioral measures were relatively more important for predicting health-related quality of life than were others, thus motivating further work to investigate these potential associations and their implications. Future research should additionally consider examining the relationship between individual differences in the adolescent development of inhibitory control and specific domains of health-related quality of life (e.g., psychological) in early adulthood, as well as how health-related quality of life may be related to cognitive control more generally among healthy adolescents and young adults.

## Appendix

### R code

```
1 #####  
2 ### Score WHO-QOL questionnaire data ###  
3 #####  
4  
5 setwd( '/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')  
6  
7 library(dplyr)  
8 library(lubridate)  
9  
10 ## import data collected at behavioral visits  
11 whoqol_beh <- read.csv('Data/CogLong_WHOQOL.csv',  
12 header = TRUE,  
13 stringsAsFactors = FALSE)  
14  
15 # filter empty rows, select only needed columns  
16 whoqol_beh <- whoqol_beh %>%  
17 filter(!is.na(ID)) %>%  
18 select(-starts_with('D_')) %>%  
19 rename('lunaid' = 'ID') %>%  
20 mutate(date = as.POSIXct(mdy(date)))  
21  
22 ## import data collected via R03 online surveys  
23 whoqol_ro3_a <- read.csv('Data/R03+Research+Follow-up+Battery_November+14,+2019_-  
11.43.csv',  
24 header = TRUE,  
25 stringsAsFactors = FALSE)  
26  
27 # filter unneeded rows and select only needed columns  
28 whoqol_ro3_a <- whoqol_ro3_a[3:nrow(whoqol_ro3_a), ] %>%  
29 filter(Status != 'Survey Preview') %>%  
30 select(c('ExternalReference', 'EndDate', 'Q60_1':'Q68')) %>%  
31 mutate(EndDate = as.POSIXct(EndDate))  
32  
33 # rename columns with consistent names  
34 names(whoqol_ro3_a) <- names(whoqol_beh)  
35  
36 ## import data collected via R03 online surveys  
37 whoqol_ro3_b <- read.csv('Data/R03+WHOQOL-BREF_November+15,+2019_08.48.csv',  
38 header = TRUE,  
39 stringsAsFactors = FALSE)  
40  
41 # filter unneeded rows and select only needed columns  
42 whoqol_ro3_b <- whoqol_ro3_b[3:nrow(whoqol_ro3_b), ] %>%  
43 filter(Status != 'Survey Preview') %>%  
44 select(c('ExternalReference', 'EndDate', 'Q3_1':'Q15')) %>%  
45 mutate(EndDate = as.POSIXct(EndDate, format = '%m/%d/%Y %H:%M', tz = ''))  
46  
47 # rename columns with consistent names
```

```

48 names(whoqol_ro3_b) <- names(whoqol_beh)
49
50 ## merge two online survey versions
51 whoqol_ro3 <- rbind(whoqol_ro3_a, whoqol_ro3_b, stringsAsFactors = FALSE)
52
53 ## function to recode online survey items from text input to numeric values
54 whoqol_text_to_num <- function(x) {
55   x <- gsub(x, pattern = '^Very Poor$', replacement = '1')
56   x <- gsub(x, pattern = '^Poor$', replacement = '2')
57   x <- gsub(x, pattern = '^Neither [Pp]oor nor [Gg]ood$', replacement = '3')
58   x <- gsub(x, pattern = '^Good$', replacement = '4')
59   x <- gsub(x, pattern = '^Very [Gg]ood$', replacement = '5')
60
61   x <- gsub(x, pattern = '^Very Dissatisfied$', replacement = '1')
62   x <- gsub(x, pattern = '^Dissatisfied$', replacement = '2')
63   x <- gsub(x, pattern = '^Neither [Dd]issatisfied nor [Ss]atisfied$', replacement = '3')
64   x <- gsub(x, pattern = '^Satisfied$', replacement = '4')
65   x <- gsub(x, pattern = '^Very Satisfied$', replacement = '5')
66
67   x <- gsub(x, pattern = '^Not at all$', replacement = '1')
68   x <- gsub(x, pattern = '^A little$', replacement = '2')
69   x <- gsub(x, pattern = '^A moderate amount$', replacement = '3')
70   x <- gsub(x, pattern = '^Very much$', replacement = '4')
71   x <- gsub(x, pattern = '^An extreme amount$', replacement = '5')
72
73   x <- gsub(x, pattern = '^Extremely$', replacement = '5')
74
75   x <- gsub(x, pattern = '^Moderately$', replacement = '3')
76   x <- gsub(x, pattern = '^Mostly$', replacement = '4')
77   x <- gsub(x, pattern = '^Completely$', replacement = '5')
78
79   x <- gsub(x, pattern = '^Never$', replacement = '5')
80   x <- gsub(x, pattern = '^Seldom$', replacement = '4')
81   x <- gsub(x, pattern = '^Quite Often$', replacement = '3')
82   x <- gsub(x, pattern = '^Very Often$', replacement = '2')
83   x <- gsub(x, pattern = '^Always$', replacement = '1')
84
85   x <- as.numeric(x)
86   return(x)
87 }
88
89 # recode online survey items from text input to numeric values
90 whoqol_ro3[, 3:28] <- sapply(whoqol_ro3[, 3:28], whoqol_text_to_num)
91
92 # items 3 and 4 are reverse scored
93 whoqol_ro3$X3 <- 6 - whoqol_ro3$X3
94 whoqol_ro3$X4 <- 6 - whoqol_ro3$X4
95
96 # replace missing value with mean of other domain items per scoring guidelines
97 which(is.na(whoqol_ro3), arr.ind = TRUE)
98 whoqol_ro3[19, 17] <- mean(unlist(whoqol_ro3[19, c('X3', 'X4', 'X10', 'X16', 'X17', 'X18')]))
99
100 ## merge online survey data with behavioral visit data
101 whoqol <- rbind(whoqol_ro3, whoqol_beh, stringsAsFactors = FALSE)
102
103 ## score whoqol per scoring guidelines
104 whoqol <- whoqol %>%

```

```

105 mutate(D_1_Raw = X3 + X4 + X10 + X15 + X16 + X17 + X18) %>% # physical
106 mutate(D_2_Raw = X5 + X6 + X7 + X11 + X19 + X26) %>% # psychological
107 mutate(D_3_Raw = X20 + X21 + X22) %>% # social
108 mutate(D_4_Raw = X8 + X9 + X12 + X13 + X14 + X23 + X24 + X25) # environment
109
110 ## functions to rescale raw scores
111 scale_domain_1 <- function(x) {
112   y <- case_when(x == 7 ~ 4,
113     x == 8 | x == 9 ~ 5,
114     x == 10 | x == 11 ~ 6,
115     x == 12 | x == 13 ~ 7,
116     x == 14 ~ 8,
117     x == 15 | x == 16 ~ 9,
118     x == 17 | x == 18 ~ 10,
119     x == 19 | x == 20 ~ 11,
120     x == 21 ~ 12,
121     x == 22 | x == 23 ~ 13,
122     x == 24 | x == 25 ~ 14,
123     x == 26 | x == 27 ~ 15,
124     x == 28 ~ 16,
125     x == 29 | x == 30 ~ 17,
126     x == 31 | x == 32 ~ 18,
127     x == 33 | x == 34 ~ 19,
128     x == 35 ~ 20)
129   return(y)
130 }
131
132 scale_domain_2 <- function(x) {
133   y <- case_when(x == 6 ~ 4,
134     x == 7 | x == 8 ~ 5,
135     x == 9 ~ 6,
136     x == 10 | x == 11 ~ 7,
137     x == 12 ~ 8,
138     x == 13 | x == 14 ~ 9,
139     x == 15 ~ 10,
140     x == 16 | x == 17 ~ 11,
141     x == 18 ~ 12,
142     x == 19 | x == 20 ~ 13,
143     x == 21 ~ 14,
144     x == 22 | x == 23 ~ 15,
145     x == 24 ~ 16,
146     x == 25 | x == 26 ~ 17,
147     x == 27 ~ 18,
148     x == 28 | x == 29 ~ 19,
149     x == 30 ~ 20)
150   return(y)
151 }
152
153 scale_domain_3 <- function(x) {
154   y <- case_when(x == 3 ~ 4,
155     x == 4 ~ 5,
156     x == 5 ~ 7,
157     x == 6 ~ 8,
158     x == 7 ~ 9,
159     x == 8 ~ 11,
160     x == 9 ~ 12,
161     x == 10 ~ 13,
162     x == 11 ~ 15,
163     x == 12 ~ 16,

```

```

164         x == 13 ~ 17,
165         x == 14 ~ 19,
166         x == 15 ~ 20)
167     return(y)
168 }
169
170 scale_domain_4 <- function(x) {
171   y <- case_when(x == 8 ~ 4,
172                   x == 9 | x == 10 ~ 5,
173                   x == 11 | x == 12 ~ 6,
174                   x == 13 | x == 14 ~ 7,
175                   x == 15 | x == 16 ~ 8,
176                   x == 17 | x == 18 ~ 9,
177                   x == 19 | x == 20 ~ 10,
178                   x == 21 | x == 22 ~ 11,
179                   x == 23 | x == 24 ~ 12,
180                   x == 25 | x == 26 ~ 13,
181                   x == 27 | x == 28 ~ 14,
182                   x == 29 | x == 30 ~ 15,
183                   x == 31 | x == 32 ~ 16,
184                   x == 33 | x == 34 ~ 17,
185                   x == 35 | x == 36 ~ 18,
186                   x == 37 | x == 38 ~ 19,
187                   x == 39 | x == 40 ~ 20)
188   return(y)
189 }
190
191 scale_20_to_100 <- function(x) {
192   y <- case_when(x == 4 ~ 0,
193                   x == 5 ~ 6,
194                   x == 6 ~ 13,
195                   x == 7 ~ 19,
196                   x == 8 ~ 25,
197                   x == 9 ~ 31,
198                   x == 10 ~ 38,
199                   x == 11 ~ 44,
200                   x == 12 ~ 50,
201                   x == 13 ~ 56,
202                   x == 14 ~ 63,
203                   x == 15 ~ 69,
204                   x == 16 ~ 75,
205                   x == 17 ~ 81,
206                   x == 18 ~ 88,
207                   x == 19 ~ 94,
208                   x == 20 ~ 100)
209   return(y)
210 }
211
212 # transform raw domain scores to scaled scores
213 whoqol <- whoqol %>%
214   mutate(D_1_Trans_4_20 = scale_domain_1(D_1_Raw)) %>%
215   mutate(D_2_Trans_4_20 = scale_domain_2(D_2_Raw)) %>%
216   mutate(D_3_Trans_4_20 = scale_domain_3(D_3_Raw)) %>%
217   mutate(D_4_Trans_4_20 = scale_domain_4(D_4_Raw)) %>%
218   mutate(D_1_Trans_0_100 = scale_20_to_100(D_1_Trans_4_20)) %>%
219   mutate(D_2_Trans_0_100 = scale_20_to_100(D_2_Trans_4_20)) %>%
220   mutate(D_3_Trans_0_100 = scale_20_to_100(D_3_Trans_4_20)) %>%
221   mutate(D_4_Trans_0_100 = scale_20_to_100(D_4_Trans_4_20))
222

```

```

223 ## some subjects completed the questionnaire more than once
224 # create counter variable to keep track of repeated administrations
225 whoqol <- whoqol %>%
226   group_by(lunaid) %>%
227   arrange(date, .by_group = TRUE) %>%
228   mutate(counter = row_number())
229
230
231 # select only scores from subject's most recent administration
232 whoqol_most_recent <- whoqol %>%
233   group_by(lunaid) %>%
234   filter(counter == max(counter))
235
236 write.csv(whoqol_most_recent, 'Data/whoqol_ages_scores_20200204.csv')
237
238
239 ## descriptive stats
240 whoqol_variability <- data.frame(matrix(NA, nrow = 1, ncol = 6))
241 whoqol_variability <- whoqol_variability[-1, , drop = FALSE]
242 names(whoqol_variability) <- c('Measure', 'Mean', 'SD', 'Median', 'Min', 'Max')
243 columns <- names(whoqol_most_recent)[30:41]
244
245 for (i in seq(1, length(columns), by = 1)){
246   whoqol_variability[i, 'Measure'] <- columns[i]
247   whoqol_variability[i, 'Mean'] <- mean(as.numeric(unlist(whoqol_most_recent[, columns[i]])))
248   whoqol_variability[i, 'SD'] <- sd(as.numeric(unlist(whoqol_most_recent[, columns[i]])))
249   whoqol_variability[i, 'Median'] <- median(as.numeric(unlist(whoqol_most_recent[, columns[i]])))
250   whoqol_variability[i, 'Min'] <- range(as.numeric(unlist(whoqol_most_recent[, columns[i]])))[1]
251   whoqol_variability[i, 'Max'] <- range(as.numeric(unlist(whoqol_most_recent[, columns[i]])))[2]
252 }

```

```

1 ##########
2 ### Plot of final sample age distribution and descriptive stats #####
3 #########
4
5 setwd('/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)
8 library(tidyr)
9 library(ggplot2)
10 library(lubridate)
11
12 # cog subject DOB and sex pulled from database
13 dob_sex <- read.csv('Data/cog_allsubs_dobs_sex.csv',
14   header = TRUE,
15   stringsAsFactors = FALSE)
16 dob_sex <- dob_sex %>%
17   rename('lunaid' = 'id') %>%
18   mutate(dob = as.Date(as.character(dob), format = '%Y-%m-%d'))
19
20
21 # scored whoqol data compiled with prev script
22 whoqol <- read.csv('Data/whoqol_ages_scores_20200204.csv',

```

```

23     header = TRUE,
24     stringsAsFactors = FALSE)
25 whoqol <- whoqol %>%
26   select(-c('X')) %>%
27   mutate(date = as.Date(as.character(date), format = '%Y-%m-%d'))
28
29 # join dob and whoqol data
30 whoqol_age <- whoqol %>%
31   left_join(dob_sex, by = 'lunaid') %>%
32   mutate(age = interval(start = dob, end = date)/duration(n = 1, unit = 'years'))
33   %>%
34   mutate(age_type = 'WHO-QOL') %>% # indicator if age is for scan or for whoqol
35   mutate(note = '') %>%
36   select(c(lunaid, date, dob, sex, age, note, age_type)) %>%
37   rename('vtimestamp' = 'date')
38
39 # subjects who completed whoqol
40 subids <- unique(whoqol_age$lunaid)
41 length(subids)
42
43 # Cog scan dates and ages pulled from database
44 scan_age <- read.csv('Data/cog_scans_dates_ages.csv',
45   header = TRUE,
46   stringsAsFactors = FALSE)
47 scan_age <- scan_age %>%
48   mutate(dob = as.Date(as.character(dob), format = '%Y-%m-%d')) %>%
49   mutate(vtimestamp = as.Date(as.character(vtimestamp), format = '%Y-%m-%d')) %>%
50   rename('lunaid' = 'id') %>%
51   mutate(age_type = 'Scan') %>% # indicator if age is for scan or for whoqol
52   mutate(visit_identifier = paste(lunaid, vtimestamp, sep = '_'))
53
54 # remove dropped visits
55 scan_age_nodrop <- scan_age %>%
56   filter(lunaid %in% subids) %>% # select only subjects who completed whoqol
57   filter(vtimestamp > '2005-11-11') %>% # dates prior to Nov 2005 = pilot data
58   filter(visit_identifier != '10188_2005-12-14') %>%
59   filter(visit_identifier != '10192_2005-12-22') %>%
60   filter(visit_identifier != '10463_2007-10-24') %>%
61   filter(visit_identifier != '10463_2008-11-08') %>%
62   filter(visit_identifier != '10365_2017-05-23') %>%
63   filter(visit_identifier != '10133_2009-01-14') %>%
64   filter(visit_identifier != '10248_2006-06-26') %>%
65   filter(visit_identifier != '10136_2007-07-21') %>%
66   filter(visit_identifier != '10132_2008-12-23') %>%
67   filter(visit_identifier != '10184_2010-09-25') %>%
68   filter(visit_identifier != '10252_2011-02-09') %>%
69   filter(visit_identifier != '10202_2009-01-06') %>%
70   select(-c('visit_identifier'))
71
72 # sessions to preprocess scan data for
73 write.csv(scan_age_nodrop, 'Data/cog_subs_to_prepoc_20200217.csv')
74
75
76 ## final sample age distribution
77 # subjects who have whoqol data and 3+ usable/preprocessed scan time points
78 final_subs <- read.csv('Data/final_subs_for_analysis_20200228.csv',
79   stringsAsFactors = FALSE)
80 final_subs <- final_subs %>%

```

```

81 select(-c(X, n_scans)) %>%
82 mutate(date = as.Date(date, format = '%Y-%m-%d')) %>%
83 rename('lunaid' = 'id') %>%
84 rename('vtimestamp' = 'date')
85
86 final_subids <- unique(final_subs$lunaid)
87
88 # merge whoqol, scan age, and final sample data
89 dat <- final_subs %>%
90   left_join(scan_age_nodrop, by = c('lunaid', 'vtimestamp')) %>%
91   mutate(visit_identifier = paste(lunaid, vtimestamp, sep = '_')) %>%
92   filter(!duplicated(visit_identifier)) %>%
93   select(-c('visit_identifier')) %>%
94   bind_rows(whoqol_age) %>%
95   filter(lunaid %in% final_subids)
96
97 write.csv(dat, 'Data/whoqol_and_scan_ages_20200228.csv')
98
99 # number of scans per subject
100 numscans <- as.data.frame(table(dat$lunaid) - 1) # one observation is for whoqol
101 colnames(numscans)[1] <- 'lunaid'
102
103 # order subjects by age at baseline scan
104 min_ages <- dat %>%
105   filter(age_type == 'Scan') %>%
106   select(lunaid, age) %>%
107   group_by(lunaid) %>%
108   filter(age == min(age))
109 min_ages <- min_ages[order(min_ages$age), ]
110 min_ages$order_for_plot <- seq(1:nrow(min_ages))
111 range(min_ages$age)
112
113 # data for plotting
114 dat_to_plot <- min_ages %>%
115   select(lunaid, order_for_plot) %>%
116   full_join(dat, by = 'lunaid')
117
118 # data for plotting - scans only
119 dat_to_plot_nowhoqol <- dat_to_plot %>%
120   filter(age_type != 'WHOQOL')
121
122 # create plot of longitudinal subject age distribution
123 subject_plot <- ggplot(dat_to_plot_nowhoqol) +
124   aes(y = as.factor(order_for_plot),
125       x = age,
126       color = factor(sex, labels = c('Female', 'Male')),
127       group = as.factor(order_for_plot)) +
128   geom_line() +
129   geom_point(data = dat_to_plot,
130             aes(shape = as.factor(age_type))) +
131   scale_colour_manual(values = c('#D1474B', '#476FD1')) +
132   theme_bw() +
133   theme(panel.border = element_blank(), panel.grid.major = element_blank(),
134         panel.grid.minor = element_blank(), axis.line = element_line(colour = "black")) +
135   labs(x = 'Age (years)', y = 'Subject', color = 'Sex', shape = 'Visit type')
136 subject_plot
137
138 ggsave('subjects_20200228.png', plot = subject_plot, width = 5, height = 6, units =

```

```

    'in')

139 ## descriptive statistics
140 # create variables for intervals (years) between each scan timepoint
141 bw_scan_intervals <- dat_to_plot_nowhoqol %>%
142   select(c(lunaid, vtimestep)) %>%
143   group_by(lunaid) %>%
144   arrange(vtimestep, .by_group = TRUE) %>%
145   mutate(counter = row_number()) %>%
146   pivot_wider(id_cols = lunaid, names_from = counter, values_from = vtimestep) %>%
147   mutate(interval1 = interval(start = '1', end = '2')/duration(n = 1, unit = 'years'
148         )) %>%
149   mutate(interval2 = interval(start = '2', end = '3')/duration(n = 1, unit = 'years
150         )) %>%
151   mutate(interval3 = interval(start = '3', end = '4')/duration(n = 1, unit = 'years
152         )) %>%
153   mutate(interval4 = interval(start = '4', end = '5')/duration(n = 1, unit = 'years
154         )) %>%
155   mutate(interval5 = interval(start = '5', end = '6')/duration(n = 1, unit = 'years
156         )) %>%
157   mutate(interval6 = interval(start = '6', end = '7')/duration(n = 1, unit = 'years
158         )) %>%
159   mutate(interval7 = interval(start = '7', end = '8')/duration(n = 1, unit = 'years
160         )) %>%
161   mutate(interval8 = interval(start = '8', end = '9')/duration(n = 1, unit = 'years
162         )) %>%
163   mutate(interval9 = interval(start = '9', end = '10')/duration(n = 1, unit =
164     'years')) %>%
165   mutate(interval10 = interval(start = '10', end = '11')/duration(n = 1, unit =
166     'years')) %>%
167   mutate(interval11 = interval(start = '11', end = '12')/duration(n = 1, unit =
168     'years')) %>%
169   mutate(interval12 = interval(start = '12', end = '13')/duration(n = 1, unit =
170     'years')) %>%
171   select(-c('1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13'))
172   %>%
173   pivot_longer(cols = starts_with('interval'),
174                 names_to = 'bw_scan_interval_length')

# average interval between scans
175 mean(as.numeric(bw_scan_intervals$value), na.rm = TRUE)
176 sd(as.numeric(bw_scan_intervals$value), na.rm = TRUE)
177 min(as.numeric(bw_scan_intervals$value), na.rm = TRUE)
178 max(as.numeric(bw_scan_intervals$value), na.rm = TRUE)

# average number of scans among the final sample (those with 3+ useable time points)
179 mean(as.numeric(numscans$Freq))
180 sd(as.numeric(numscans$Freq))
181 min(as.numeric(numscans$Freq))
182 max(as.numeric(numscans$Freq))

# average baseline age (age at first scan)
183 mean(as.numeric(min_ages$age))
184 sd(as.numeric(min_ages$age))
185 min(as.numeric(min_ages$age))
186 max(as.numeric(min_ages$age))

# average endpoint age (age at final scan)
187 max_ages <- dat %>%

```

```

184 filter(age_type == 'Scan') %>%
185 select(lunaid, age) %>%
186 group_by(lunaid) %>%
187 filter(age == max(age))
188 mean(as.numeric(max_ages$age))
189 sd(as.numeric(max_ages$age))
190 min(as.numeric(max_ages$age))
191 max(as.numeric(max_ages$age))
192
193 # n and proportion male and female
194 whoqol_age %>%
195 filter(lunaid %in% final_subids) %>%
196 group_by(sex) %>%
197 summarise(n = n(),
198           prop = n() / length(final_subids))

```

```

1 #####
2 ### Score antisaccade eye-tracking data #####
3 #####
4
5 setwd('/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)
8 library(lubridate)
9
10 # autoeyescored eye data
11 eye_data <- read.table('Data/all_trail_et.tsv',
12                         sep = '\t',
13                         header = TRUE,
14                         stringsAsFactors = FALSE)
15
16 eye_data <- eye_data %>%
17   mutate(date = as.Date(as.character(date), format = '%Y-%m-%d')) %>%
18   mutate(fstCorrect <- as.numeric(fstCorrect)) %>%
19   mutate(ErrCorr <- as.numeric(ErrCorr)) %>%
20   mutate(Incorrect = ifelse(Count == 0, 1, 0)) %>%
21   mutate(Dropped = ifelse(Count == -1, 1, 0))
22
23 ## accuracy
24 # proportion of each trial type in each session (considering all 4 anti runs
# together)
25 anti_perf <- eye_data %>%
26   filter(AS == 'AS') %>%
27   group_by(id, date) %>%
28   summarise(anti_perc_correct = mean(fstCorrect),
29             anti_perc_error_corrected = mean(ErrCorr),
30             anti_perc_incorrect = mean(Incorrect),
31             anti_perc_dropped_trials = mean(Dropped))
32
33 # proportion of each trial type in each session among non-dropped trials
34 anti_perf_nodrop <- eye_data %>%
35   filter(AS == 'AS') %>%
36   filter(Dropped == 0) %>%
37   group_by(id, date) %>%
38   summarise(anti_perc_correct_nodrop = mean(fstCorrect),
39             anti_perc_error_corrected_nodrop = mean(ErrCorr))
40
41 ## saccade latency

```

```

42 # average antisaccade latency on correct trials only
43 anti_lat_corr <- eye_data %>%
44   filter(AS == 'AS') %>%
45   filter(Count == 1) %>%
46   group_by(id, date) %>%
47   summarise(anti_avg_lat_correct_trials = mean(lat))
48
49 # average antisaccade latency on error corrected trials
50 anti_lat_errcorr <- eye_data %>%
51   filter(AS == 'AS') %>%
52   filter(Count == 2) %>%
53   group_by(id, date) %>%
54   summarise(anti_avg_lat_error_corrected_trials = mean(lat))
55
56
57 # join accuracy and latency data
58 anti_perf_lat_20200228 <- anti_perf %>%
59   full_join(anti_perf_nodrop, by = c('id', 'date')) %>%
60   full_join(anti_lat_corr, by = c('id', 'date')) %>%
61   full_join(anti_lat_errcorr, by = c('id', 'date')) %>%
62   rename('lunaid' = 'id')
63
64 write.csv(anti_perf_lat_20200228, 'Data/anti_perf_lat_20200228.csv')

```

```

1 #####
2 ### Merge WHO-QOL and eye data ###
3 #####
4
5 setwd('/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)
8
9 # whoqol data
10 whoqol <- read.csv('Data/whoqol_and_scan_ages_20200228.csv',
11                      header = TRUE,
12                      stringsAsFactors = FALSE)
13
14 whoqol <- whoqol %>%
15   select(-c('X')) %>%
16   mutate(dob = as.Date(as.character(dob), format = '%Y-%m-%d')) %>%
17   mutate(vtimestamp = as.Date(as.character(vtimestamp), format = '%Y-%m-%d')) %>%
18   rename('date' = 'vtimestamp')
19
20 # import scored anti task eye data
21 eye <- read.csv('Data/anti_perf_lat_20200228.csv',
22                   header = TRUE,
23                   stringsAsFactors = FALSE)
24
25 eye <- eye %>%
26   select(-c('X')) %>%
27   mutate(date = as.Date(as.character(date), format = '%Y-%m-%d')) %>%
28   rename('lunaid' = 'id')
29
30 # join whoqol and eye data
31 whoqol_plus_eye <- whoqol %>%
32   left_join(eye, by = c('lunaid', 'date')) %>%
33   write.csv('Data/whoqol_plus_eye_20200228.csv')

```

```

1 ##### Create stim time files for AFNI 3dDeconvolve #####
2 #### Create stim time files for AFNI 3dDeconvolve #####
3 #####
4
5 setwd( '/Users/jenniferefedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(readxl)
8 library(dplyr)
9 library(tidyr)
10 library(LNCDR)
11 library(stringr)
12
13 # empty dataframe
14 antistate_timings <- data.frame(FRAME = integer(),
15                                   'Start Tm' = double(),
16                                   'End Tm' = double(),
17                                   Event = character(),
18                                   Location = integer(),
19                                   Run = character(),
20                                   stringsAsFactors = FALSE)
21
22 # read in relevant columns from each sheet of Katerina's task timing Excel sheet and
23 # append to the dataframe
24 for (i in 1:12){
25   av <- readxl::read_excel('Data/Anti_Mix_Design_Lists_FINAL_minusfix_fixblockrecord
26   _FINMOD.xls',
27                           sheet = sprintf('List%AV', i),
28                           range = cell_cols('BB:BF'))
29   av$Run <- sprintf('%sAV', i)
30   va <- readxl::read_excel('Data/Anti_Mix_Design_Lists_FINAL_minusfix_fixblockrecord
31   _FINMOD.xls',
32                           sheet = sprintf('List%VA', i),
33                           range = cell_cols('BB:BF'))
34   va$Run <- sprintf('%sVA', i)
35   av_va <- rbind(av, va)
36   antistate_timings <- rbind(antistate_timings, av_va)
37 }
38
39 write.csv(antistate_timings, 'Data/antistate_timings_from_kat.csv')
40
41 # subject-date run orders
42 sub_runs <- readxl::read_excel('Data/Anti-State&MGSEncode_data.xls',
43                                 sheet = 'Anti-State_sorted_by-year')
44 sub_runs$Scan_Date <- as.Date(sub_runs$Scan_Date)
45
46 # cleaning subject, date, run data
47 sub_runs <- sub_runs %>%
48   select(id = Oxford_ID,
49         date = Scan_Date,
50         bircid = BIRC_ID,
51         AS1 = 'Anti-State_1',
52         AS2 = 'Anti-State_2',
53         AS3 = 'Anti-State_3',
54         AS4 = 'Anti-State_4') %>%
55   pivot_longer(cols = AS1:AS4,
56               names_to = 'run',
57               values_to = 'run_type')

```

```

55 sub_runs$run <- str_replace_all(sub_runs$run, "^\AS", "")
56 sub_runs$run <- as.integer(sub_runs$run)
57 sub_runs$run_type <- str_replace_all(sub_runs$run_type, "^.\\\"(", ""))
58 sub_runs$run_type <- str_replace_all(sub_runs$run_type, "^.\\\"(", ""))
59 sub_runs$run_type <- str_replace_all(sub_runs$run_type, "^.\\\"(", ""))
60 sub_runs$run_type <- str_replace_all(sub_runs$run_type, "\\\\$", "")
61 sub_runs$run_type <- str_replace_all(sub_runs$run_type, "\\\\$", "")

62
63 # scored eye data (per trial, all subjects)
64 eye_data <- read.csv('Data/all_trail_et.tsv',
65                      sep = '\t',
66                      stringsAsFactors = FALSE)
67
68 eye_data <- eye_data %>%
69   mutate(date = as.Date(as.character(date), format = '%Y-%m-%d')) %>%
70   mutate(fstCorrect = as.numeric(fstCorrect)) %>%
71   mutate(ErrCorr = as.numeric(ErrCorr)) %>%
72   mutate(Incorrect = ifelse(Count == 0, 1, 0)) %>%
73   mutate(Dropped = ifelse(Count == -1, 1, 0))

74
75 eye_data_sub_runs <- eye_data %>%
76   left_join(sub_runs, by = c('id', 'date', 'run'))
77 write.csv(eye_data_sub_runs, 'Data/antistate_scored_eyd_plus_run_numbers.csv')
78
79 eye_data_sub_runs %>%
80   filter(is.na(run_type)) %>%
81   select(id, date, run, run_type) %>%
82   tally()

83 # subjects for analysis (have whoql and 3+ scans)
84 subs_for_analysis <- read.csv('Data/cog_subs_to_prepoc_20200217.csv',
85                               stringsAsFactors = FALSE)
86
87
88 subs_for_analysis <- subs_for_analysis %>%
89   mutate(vtimestamp = as.Date(vtimestamp, format = '%m/%d/%Y'))
90 names(subs_for_analysis) <- c('id', 'date', 'age', 'note')
91
92 eye_data_sub_runs_analysis <- subs_for_analysis %>%
93   left_join(eye_data_sub_runs, by = c('id', 'date'))

94
95 # filter trials that are fixation, merge for saving 1D files later
96 antistate_timings_excl_fix <- antistate_timings %>%
97   filter(!grepl('FIX$', Event)) %>% # including 'FIXCUE'
98   group_by(run_type) %>%
99   filter(Event != lag(Event) | is.na(lag(Event))) %>%
100  mutate(trial = cumsum(grepl('(ANTI|VGS)CUE', Event)))

101 write.csv(antistate_timings_excl_fix, 'Data/antistate_timings_from_kat_excl_fix.csv')
102
103
104 merged_for_1d <- eye_data_sub_runs_analysis %>%
105   full_join(antistate_timings_excl_fix, by = c('trial', 'run_type'))
106
107 write.csv(merged_for_1d, 'Data/cog_antistate_timings_merged_for_1d_20200219.csv')

108
109 # create new column in merged file to specify left/right
110 unique(merged_for_1d$Location)
111 merged_for_1d <- merged_for_1d %>%
112   mutate(LeftRight = case_when(Location < 320 ~ 'Left',

```

```

113             Location > 320 ~ 'Right'))  

114  

115 # write 1D files for 3Ddeconvolve  

116 oned_return <- function(f, ...){  

117   save1D(fname = f, ...)  

118   return(f)  

119 }  

120  

121 merged_for_1d <- merged_for_1d %>%  

122   filter(!is.na(id))  

123  

124 for(subid in unique(merged_for_1d$id)){  

125   temp <- merged_for_1d %>%  

126     filter(id == subid)  

127  

128   for(subdate in unique(temp$date)){  

129     temp %>%  

130       rename(block = run) %>%  

131         filter(LeftRight == 'Left') %%  

132         filter(Event == 'ANTICUE') %%  

133         filter(Count == 1) %%  

134         filter(date == subdate) %%  

135         summarize(d = strftime(first(date), format = '%Y%m%d'),  

136           f = sprintf('1D_files/%s.%s_left_correct.1D', first(id), d),  

137           r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))  

138  

139   temp %>%  

140     rename(block = run) %>%  

141       filter(LeftRight == 'Left') %%  

142       filter(Event == 'ANTICUE') %%  

143       filter(Count == 2) %%  

144       filter(date == subdate) %%  

145       summarize(d = strftime(first(date), format = '%Y%m%d'),  

146         f = sprintf('1D_files/%s.%s_left_errcorr.1D', first(id), d),  

147         r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))  

148  

149  

150   temp %>%  

151     rename(block = run) %>%  

152       filter(LeftRight == 'Left') %%  

153       filter(Event == 'ANTICUE') %%  

154       filter(Count == 0) %%  

155       filter(date == subdate) %%  

156       summarize(d = strftime(first(date), format = '%Y%m%d'),  

157         f = sprintf('1D_files/%s.%s_left_incorr.1D', first(id), d),  

158         r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))  

159  

160  

161   temp %>%  

162     rename(block = run) %>%  

163       filter(LeftRight == 'Left') %%  

164       filter(Event == 'ANTICUE') %%  

165       filter(Count == -1) %%  

166       filter(date == subdate) %%  

167       summarize(d = strftime(first(date), format = '%Y%m%d'),  

168         f = sprintf('1D_files/%s.%s_left_dropped.1D', first(id), d),  

169         r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))  

170  

171   temp %>%

```

```

172 rename(block = run) %>%
173   filter(LeftRight == 'Right') %>%
174   filter(Event == 'ANTICUE') %>%
175   filter(Count == 1) %>%
176   filter(date == subdate) %>%
177   summarize(d = strftime(first(date), format = '%Y%m%d')),
178     f = sprintf('1D_files/%s.%s_anti_right_correct.1D', first(id), d),
179     r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
180
181 temp %>%
182   rename(block = run) %>%
183     filter(LeftRight == 'Right') %>%
184     filter(Event == 'ANTICUE') %>%
185     filter(Count == 2) %>%
186     filter(date == subdate) %>%
187     summarize(d = strftime(first(date), format = '%Y%m%d')),
188       f = sprintf('1D_files/%s.%s_anti_right_errcorr.1D', first(id), d),
189       r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
190
191 temp %>%
192   rename(block = run) %>%
193     filter(LeftRight == 'Right') %>%
194     filter(Event == 'ANTICUE') %>%
195     filter(Count == 0) %>%
196     filter(date == subdate) %>%
197     summarize(d = strftime(first(date), format = '%Y%m%d')),
198       f = sprintf('1D_files/%s.%s_anti_right_incorr.1D', first(id), d),
199       r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
200
201 temp %>%
202   rename(block = run) %>%
203     filter(LeftRight == 'Right') %>%
204     filter(Event == 'ANTICUE') %>%
205     filter(Count == -1) %>%
206     filter(date == subdate) %>%
207     summarize(d = strftime(first(date), format = '%Y%m%d')),
208       f = sprintf('1D_files/%s.%s_anti_right_dropped.1D', first(id), d),
209       r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
210
211 temp %>%
212   rename(block = run) %>%
213     filter(LeftRight == 'Left') %>%
214     filter(Event == 'VGSCUE') %>%
215     filter(Count != -1) %>%
216     filter(date == subdate) %>%
217     summarize(d = strftime(first(date), format = '%Y%m%d')),
218       f = sprintf('1D_files/%s.%s_vgs_left.1D', first(id), d),
219       r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
220
221 temp %>%
222   rename(block = run) %>%
223     filter(LeftRight == 'Left') %>%
224     filter(Event == 'VGSCUE') %>%
225     filter(Count == -1) %>%
226     filter(date == subdate) %>%
227     summarize(d = strftime(first(date), format = '%Y%m%d')),
228       f = sprintf('1D_files/%s.%s_vgs_left_dropped.1D', first(id), d),
229       r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
230

```

```

231 temp %>%
232   rename(block = run) %>%
233   filter(LeftRight == 'Right') %>%
234   filter(Event == 'VGSCUE') %>%
235   filter(Count != -1) %>%
236   filter(date == subdate) %>%
237   summarize(d = strftime(first(date), format = '%Y%m%d'),
238             f = sprintf('1D_files/%s.%s_vgs_right.1D', first(id), d),
239             r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
240
241 temp %>%
242   rename(block = run) %>%
243   filter(LeftRight == 'Right') %>%
244   filter(Event == 'VGSCUE') %>%
245   filter(Count == -1) %>%
246   filter(date == subdate) %>%
247   summarize(d = strftime(first(date), format = '%Y%m%d'),
248             f = sprintf('1D_files/%s.%s_vgs_right_dropped.1D', first(id), d),
249             r = oned_return(f, ., colname = 'Start Tm', nblocks = 4))
250
251 }
252 }
```

```

1 #####
2 ### Average activation in ROIs ###
3 #####
4
5 setwd('/Users/jenniferefedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(tidyr)
8 library(dplyr)
9 library(stringr)
10 library(lubridate)
11
12 # roi names and coordinates from Ordaz et al. 2013 paper
13 roi_names <- read.csv('Data/sarah_roi_20200225.csv',
14                       header = TRUE,
15                       stringsAsFactors = FALSE)
16
17 # average activation per roi per session - output from AFNI 3dROIstats
18 roi_vals <- read.table('Data/jen_roi_stats_20200228.txt',
19                        header = TRUE,
20                        stringsAsFactors = FALSE)
21
22 # rename AFNI sub-brick labels to be more descriptive
23 roi_vals <- roi_vals %>%
24   mutate(condition = case_when(Sub.brick == '0[anti_corr]' ~ 'anti_corr_v_baseline',
25                               Sub.brick == '1[anti_errc]' ~ 'anti_errcorr_v_',
26                               Sub.brick == '2[anti_inco]' ~ 'anti_incorr_v_baseline',
27                               ,
28                               Sub.brick == '3[anti_corr]' ~ 'anti_corr_v_errcorr',
29                               Sub.brick == '4[anti_corr]' ~ 'anti_corr_v_incorr',
30                               Sub.brick == '5[anti_corr]' ~ 'anti_corr_v_vgs')) %>%
31   mutate(File2 = str_replace_all(File, '^..//subjs/...../','')) %>%
32   mutate(File2 = str_replace_all(File2, '_bucket.nii.gz.*','')) %>%
33   separate(col = File2,
```

```

34     sep = ' ') %>%
35   select(lunaid, date, condition, everything()) %>%
36   select(-c('File', 'Sub.brick')) %>%
37   mutate(date = as.Date(date, format = '%Y-%m-%d'))
38
39 # rename columns
40 colnames(roi_vals) <- c('id', 'date', 'condition',
41                       'SEF', 'pre_SMA', 'L_FEF', 'R_FEF',
42                       'L_putamen', 'R_putamen', 'L_pPC', 'R_pPC',
43                       'L_dlPFC', 'R_dlPFC', 'L_vlPFC', 'R_vlPFC',
44                       'dACC')
45
46 # pivot dataframe from long to wide format
47 roi_vals_wide <- roi_vals %>%
48   pivot_wider(id_cols = c('id', 'date'),
49               names_from = c('condition'),
50               values_from = c('SEF', 'pre_SMA', 'L_FEF', 'R_FEF',
51                               'L_putamen', 'R_putamen', 'L_pPC', 'R_pPC',
52                               'L_dlPFC', 'R_dlPFC', 'L_vlPFC', 'R_vlPFC',
53                               'dACC'))
54
55 roi_vals_wide %% write.csv('Data/roi_percent_signal_change_20200228.csv')

```

```

1 #####
2 ### Maternal education level #####
3 #####
4
5 setwd('/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)
8 library(tidyverse)
9
10 # demographic data for Cog subjects pulled from LNCD database
11 demos <- read.csv('Data/Cog_Demos_20190815.csv',
12                     stringsAsFactors = FALSE)
13 colnames(demos)[1] <- 'lunaid'
14 demos$vtimestamp <- as.Date(demos$vtimestamp,
15                             format = '%Y-%m-%d')
16
17 # maternal edu level at baseline visit
18 mat_edu <- demos %>%
19   filter(level_edu_mother != 'null') %>%
20   filter(level_edu_mother != '-9') %>%
21   group_by(lunaid) %>%
22   filter(vtimestamp == min(vtimestamp)) %>%
23   select(lunaid, level_edu_mother)
24
25 # convert numeric labels in db to corresponding labels on demo form
26 mat_edu <- mat_edu %>%
27   mutate(level_edu_mother_cat =
28         case_when(level_edu_mother == 1 ~ 'Less than high school',
29                    level_edu_mother == 4 ~ 'Completed high school',
30                    level_edu_mother == 5 ~ 'Completed high school',
31                    level_edu_mother == 6 ~ 'Completed college',
32                    level_edu_mother == 7 ~ 'Completed post-graduate'))
33
34 write.csv(mat_edu, 'Data/maternal_education_20200219.csv')

```

```

1 ######
2 ### Compile data for analysis ###
3 #####
4
5 setwd( '/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)
8 library(tidyr)
9 library(lubridate)
10
11 # read in scan data (316 deconvolved scans among 50 subjects)
12 scan_data <- read.csv('Data/roi_percent_signal_change_20200228.csv',
13                         stringsAsFactors = FALSE) %>%
14   select(-c('X')) %>%
15   rename('lunaid' = 'id') %>%
16   mutate(date = as.Date(date, format = '%Y-%m-%d'))
17
18 # read in scored anti eye data
19 eye_data <- read.csv('Data/whoqol_plus_eye_20200228.csv',
20                         stringsAsFactors = FALSE) %>%
21   filter(whoqol_age == 'Scan') %>%
22   select(-c('X', 'note', 'age_type')) %>%
23   mutate(date = as.Date(date, format = '%Y-%m-%d')) %>%
24   mutate(dob = as.Date(dob, format = '%Y-%m-%d'))
25
26 # read in maternal education level data
27 mat_edu <- read.csv('Data/maternal_education_20200219.csv',
28                         stringsAsFactors = FALSE) %>%
29   select(-c('X'))
30
31 # change reference category for maternal education level
32 mat_edu <- within(mat_edu,
33   level_edu_mother_cat <- relevel(as.factor(level_edu_mother_cat),
34                                     ref = 'Completed high school'))
35
36 # merge eye and scan data
37 data_for_gamms <- scan_data %>%
38   left_join(eye_data, by = c('lunaid', 'date')) %>%
39   mutate(sex = as.factor(sex)) %>%
40   mutate(duplicate_row = duplicated(.)) %>%
41   filter(duplicate_row == FALSE) %>%
42   select(-c('duplicate_row')) %>%
43   mutate(age_c = age - mean(age))
44
45 data_for_gamms <- data_for_gamms %>%
46   left_join(mat_edu, by = c('lunaid')) %>%
47   mutate(lunaid = as.factor(lunaid))
48
49 data_for_gamms %>% write.csv('Data/final_eye_scan_data_for_analysis_20200228.csv')

```

```

1 #####
2 ## GAMMs ##
3 #####
4
5 setwd( '/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)

```

```

8 library(ggplot2)
9 library(gratia)
10 library(gridExtra)
11 library(lubridate)
12 library(mgcv)
13 library(purrr)
14 library(stringr)
15 library(tidyr)
16 library(xtable)
17
18
19 # function to plot raw data, fitted values and pointwise 95% confidence intervals
20 # from a GAMM
21 plot_gamm <- function(model, y_axis_label = "", sig = TRUE) {
22   outcome <- str_split(model$gam$formula, ",~")[[2]]
23
24   newdat <- data.frame(dat$age_c) # observed centered ages
25   names(newdat) <- 'age_c'
26
27   pred <- predict.gam(model$gam,
28                         newdata = newdat,
29                         type = 'response',
30                         se.fit = TRUE)
31
32   plotdat <- data.frame(cbind(dat$age, pred$fit, pred$se))
33   plotdat <- plotdat %>%
34     mutate(outcome = 1)
35   names(plotdat) <- c('age', 'fit', 'se', outcome)
36
37   if (sig == TRUE){
38     plotcolor <- '#476FD1'
39   } else {
40     plotcolor <- 'grey50'
41   }
42
43   gammplot <- dat %>%
44     ggplot(aes(x = age,
45                y = eval(parse(text = outcome)))) +
46     geom_line(alpha = .2,
47                color = plotcolor,
48                aes(group = lunaid)) +
49     geom_point(alpha = .2,
50                shape = 16,
51                color = plotcolor) +
52     geom_ribbon(data = plotdat,
53                 alpha = .7,
54                 aes(ymin = fit -(1.96*se), ymax = fit +(1.96*se)),
55                 show.legend = FALSE,
56                 fill = plotcolor) +
57     geom_line(data = plotdat,
58               aes(y = fit),
59               show.legend = FALSE,
60               color = plotcolor) +
61     labs(x = 'Age (years)',
62          y = y_axis_label) +
63     theme_bw() +
64     theme(panel.border = element_blank(),
65           panel.grid.major = element_blank(),
66           panel.grid.minor = element_blank())

```

```

66     axis.line = element_line(colour = 'black')) +
67     theme(axis.title = element_text(size = 11),
68           axis.text = element_text(size = 10))
69   return(gammplot)
70 }
71
72 # function for post-hoc analysis to identify periods of developmental change
73 calc_dev_change <- function(model) {
74   deriv <- gratia::derivatives(model)
75   deriv <- deriv %>%
76     mutate(sig = !(0 > lower & 0 < upper))
77   ages <- list(c(min(deriv$data[deriv$sig == TRUE]) + mean(dat$age),
78                  max(deriv$data[deriv$sig == TRUE]) + mean(dat$age)))
79   return(ages[[1]])
80 }
81
82
83 # read in final data file and format variables
84 dat <- read.csv('Data/final_eye_scan_data_for_analysis_20200228.csv',
85                  stringsAsFactors = FALSE)
86 dat <- dat %>%
87   select(-c('X')) %>%
88   mutate(lunaid = as.factor(lunaid)) %>%
89   mutate(date = as.Date(date, format = '%Y-%m-%d')) %>%
90   mutate(dob = as.Date(dob, format = '%Y-%m-%d')) %>%
91   mutate(sex = as.factor(sex))
92 dat <- within(dat, level_edu_mother_cat <- relevel(as.factor(level_edu_mother_cat),
93                                                 ref = 'Completed high school'))
94
95 #### I. Behavioral outcomes
96
97 ## 1. Antisaccade proportion correct trials
98 # fit the model with smooth term for age
99 model_formula <- as.formula("anti_perc_correct_nodrop ~ s(age_c, k = 10, fx = FALSE,
100                                bs = 'tp')")
100 gamm_anti_perc_correct_nodrop <- gamm(as.formula(model_formula),
101                                         family = 'quasibinomial',
102                                         random = list(lunaid=~age_c),
103                                         data = dat,
104                                         method = 'REML')
105 summary(gamm_anti_perc_correct_nodrop$gam)
106 summary(gamm_anti_perc_correct_nodrop$lme)
107
108 # model diagnostics
109 par(mfrow = c(2,2))
110 gam.check(gamm_anti_perc_correct_nodrop$gam)
111
112 # identify significant periods of developmental change
113 anti_perc_correct_devchange <- calc_dev_change(gamm_anti_perc_correct_nodrop)
114 anti_perc_correct_devchange
115
116 # plot
117 anti_perc_correct_plot <- plot_gamm(
118   model = gamm_anti_perc_correct_nodrop,
119   y_axis_lab = 'Proportion correct trials',
120   sig = TRUE)
121 anti_perc_correct_plot
122
123 # add bar to plot reflecting period of sig dev change

```

```

124 anti_perc_correct_plot <- anti_perc_correct_plot +
125   annotate(geom = "rect",
126             xmin = anti_perc_correct_devchange[1],
127             xmax = anti_perc_correct_devchange[2],
128             ymin = -0.04,
129             ymax = -0.01,
130             fill = '#476FD1')
131 anti_perc_correct_plot
132
133
134 # fit the model w/ fixed effects of sex
135 model_formula <- as.formula("anti_perc_correct_nodrop ~ s(age_c, k = 10, fx = FALSE,
136   bs = 'tp') + sex")
137 gamm_anti_perc_correct_nodrop_sex <- gamm(as.formula(model_formula),
138                                              family = 'quasibinomial',
139                                              random = list(lunaid=~age_c),
140                                              data = dat,
141                                              method = 'REML')
142 summary(gamm_anti_perc_correct_nodrop_sex$gam)
143 summary(gamm_anti_perc_correct_nodrop_sex$lme)
144
145 # fit the model w/ fixed effects of maternal education level
146 model_formula <- as.formula("anti_perc_correct_nodrop ~ s(age_c, k = 10, fx = FALSE,
147   bs = 'tp') + level_edu_mother_cat")
148 gamm_anti_perc_correct_nodrop_medu <- gamm(as.formula(model_formula),
149                                              family = 'quasibinomial',
150                                              random = list(lunaid=~age_c),
151                                              data = dat,
152                                              method = 'REML')
153 summary(gamm_anti_perc_correct_nodrop_medu$gam)
154 summary(gamm_anti_perc_correct_nodrop_medu$lme)
155
156 # save random slopes and intercepts from final model
157 ranef_anti_perc_correct_nodrop <- ranef(gamm_anti_perc_correct_nodrop$lme)$lunaid
158   %>%
159   rename(anti_perc_correct_nodrop_int = '(Intercept)',
160         anti_perc_correct_nodrop_rslope = age_c) %>%
161   tibble::rownames_to_column(., 'lunaid') %>%
162   mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
163
164 ## 2. Antisaccade proportion error-corrected trials
165 # fit the model with smooth term for age
166 model_formula <- as.formula("anti_perc_error_corrected_nodrop ~ s(age_c, k = 10, fx
167   = FALSE, bs = 'tp')")
168 gamm_anti_perc_error_corrected_nodrop <- gamm(as.formula(model_formula),
169                                              family = 'quasibinomial',
170                                              random = list(lunaid=~age_c),
171                                              data = dat,
172                                              method = 'REML')
173 summary(gamm_anti_perc_error_corrected_nodrop$gam)
174 summary(gamm_anti_perc_error_corrected_nodrop$lme)
175
176 # model diagnostics
177 par(mfrow = c(2,2))
178 gam.check(gamm_anti_perc_error_corrected_nodrop$gam)
179
180 # identify significant periods of developmental change

```

```

179 anti_perc_error_corrected_nodrop_devchange <- calc_dev_change(gamm_anti_perc_error-
    corrected_nodrop)
180 anti_perc_error_corrected_nodrop_devchange
181
182 # plot
183 anti_perc_error_corrected_nodrop_plot <- plot_gamm(
184   model = gamm_anti_perc_error_corrected_nodrop,
185   y_axis_label = 'Proportion error-corrected trials',
186   sig = TRUE)
187 anti_perc_error_corrected_nodrop_plot
188
189 anti_perc_error_corrected_nodrop_plot <- anti_perc_error_corrected_nodrop_plot +
190   annotate(geom = 'rect',
191     xmin = anti_perc_error_corrected_nodrop_devchange[1],
192     xmax = anti_perc_error_corrected_nodrop_devchange[2],
193     ymin = -0.04,
194     ymax = -0.02,
195     fill = '#476FD1')
196 anti_perc_error_corrected_nodrop_plot
197
198 # fit the model w/ fixed effects of sex
199 model_formula <- as.formula("anti_perc_error_corrected_nodrop ~ s(age_c, k = 10, fx
  = FALSE, bs = 'tp') + sex")
200 gamm_anti_perc_error_corrected_nodrop_sex <- gamm(as.formula(model_formula),
201   family = 'quasibinomial',
202   random = list(lunaid=~age_c),
203   data = dat,
204   method = 'REML')
205 summary(gamm_anti_perc_error_corrected_nodrop_sex$gam)
206 summary(gamm_anti_perc_error_corrected_nodrop_sex$lme)
207
208 # fit the model w/ fixed effects of maternal education level
209 model_formula <- as.formula("anti_perc_error_corrected_nodrop ~ s(age_c, k = 10, fx
  = FALSE, bs = 'tp') + level_edu_mother_cat")
210 gamm_anti_perc_error_corrected_nodrop_medu <- gamm(as.formula(model_formula),
211   family = 'quasibinomial',
212   random = list(lunaid=~age_c),
213   data = dat,
214   method = 'REML')
215 summary(gamm_anti_perc_error_corrected_nodrop_medu$gam)
216 summary(gamm_anti_perc_error_corrected_nodrop_medu$lme)
217
218 # save random slopes and intercepts from final model
219 ranef_anti_perc_error_corrected_nodrop <- ranef(gamm_anti_perc_error_corrected-
    nodrop$lme)$lunaid %%
220   rename(anti_perc_error_corrected_nodrop_int = '(Intercept)',
221         anti_perc_error_corrected_rslope = age_c) %%
222   tibble::rownames_to_column(., 'lunaid') %%
223   mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
224
225
226 ## 3. Average latency on correct trials
227 # fit the model with smooth term for age
228 model_formula <- as.formula("anti_avg_lat_correct_trials ~ s(age_c, k = 10, fx =
  FALSE, bs = 'tp')")
229 gamm_anti_avg_lat_correct_trials <- gamm(as.formula(model_formula),
230   random = list(lunaid=~age_c),
231   data = dat,
232   method = 'REML')

```

```

233 summary(gamm_anti_avg_lat_correct_trials$gam)
234 summary(gamm_anti_avg_lat_correct_trials$lme)
235
236 # model diagnostics
237 par(mfrow = c(2,2))
238 gam.check(gamm_anti_avg_lat_correct_trials$gam)
239
240 # identify significant periods of developmental change
241 anti_avg_lat_correct_trials_devchange <- calc_dev_change(gamm_anti_avg_lat_correct_
242 trials)
242 anti_avg_lat_correct_trials_devchange
243
244 # plot
245 anti_avg_lat_correct_trials_plot <- plot_gamm(
246   model = gamm_anti_avg_lat_correct_trials,
247   y_axis_label = 'Average latency correct trials',
248   sig = TRUE)
249 anti_avg_lat_correct_trials_plot
250
251 anti_avg_lat_correct_trials_plot <- anti_avg_lat_correct_trials_plot +
252 annotate(geom = 'rect',
253   xmin = anti_avg_lat_correct_trials_devchange[1],
254   xmax = anti_avg_lat_correct_trials_devchange[2],
255   ymin = 300,
256   ymax = 310,
257   fill = '#476FD1')
258 anti_avg_lat_correct_trials_plot
259
260 # fit the model w/ fixed effects of sex
261 model_formula <- as.formula("anti_avg_lat_correct_trials ~ s(age_c, k = 10, fx =
262 FALSE, bs = 'tp') + sex")
262 gamm_anti_avg_lat_correct_trials_sex <- gamm(as.formula(model_formula),
263   random = list(lunaid=~age_c),
264   data = dat,
265   method = 'REML')
266 summary(gamm_anti_avg_lat_correct_trials_sex$gam)
267 summary(gamm_anti_avg_lat_correct_trials_sex$lme)
268
269 # fit the model w/ fixed effects of maternal education level
270 model_formula <- as.formula("anti_avg_lat_correct_trials ~ s(age_c, k = 10, fx =
271 FALSE, bs = 'tp') + level_edu_mother_cat")
271 gamm_anti_avg_lat_correct_trials_medu <- gamm(as.formula(model_formula),
272   random = list(lunaid=~age_c),
273   data = dat,
274   method = 'REML')
275 summary(gamm_anti_avg_lat_correct_trials_medu$gam)
276 summary(gamm_anti_avg_lat_correct_trials_medu$lme)
277
278
279 # save random slopes and intercepts from final model
280 ranef_anti_avg_lat_correct_trials <- ranef(gamm_anti_avg_lat_correct_trials$lme)$
281   lunaid %>%
282   rename(anti_avg_lat_correct_trials_int = '(Intercept)',
283     anti_avg_lat_correct_trials_rslope = age_c) %>%
284   tibble::rownames_to_column(., 'lunaid') %>%
285   mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
286
287 ## 4. Average latency on error-corrected trials

```

```

288 # fit the model with smooth term for age
289 model_formula <- as.formula("anti_avg_lat_error_corrected_trials ~ s(age_c, k = 10,
290   fx = FALSE, bs = 'tp')")
290 gamm_anti_avg_lat_error_corrected_trials <- gamm(model_formula,
291   random = list(lunaid=~age_c),
292   data = dat,
293   method = 'REML')
294 summary(gamm_anti_avg_lat_error_corrected_trials$gam)
295 summary(gamm_anti_avg_lat_error_corrected_trials$lme)
296
297 # model diagnostics
298 par(mfrow = c(2,2))
299 gam.check(gamm_anti_avg_lat_error_corrected_trials$gam)
300
301 # plot
302 anti_avg_lat_error_corrected_trials_plot <- plot_gamm(
303   model = gamm_anti_avg_lat_error_corrected_trials,
304   y_axis_label = 'Average latency error-corrected trials',
305   sig = FALSE)
306 anti_avg_lat_error_corrected_trials_plot
307
308
309 # fit the model w/ fixed effects of sex
310 model_formula <- as.formula("anti_avg_lat_error_corrected_trials ~ s(age_c, k = 10,
311   fx = FALSE, bs = 'tp') + sex")
311 gamm_anti_avg_lat_error_corrected_trials_sex <- gamm(as.formula(model_formula),
312   random = list(lunaid=~age_c),
313   data = dat,
314   method = 'REML')
315 summary(gamm_anti_avg_lat_error_corrected_trials_sex$gam)
316 summary(gamm_anti_avg_lat_error_corrected_trials_sex$lme)
317
318 # fit the model w/ fixed effects of maternal education level
319 model_formula <- as.formula("anti_avg_lat_error_corrected_trials ~ s(age_c, k = 10,
320   fx = FALSE, bs = 'tp') + level_edu_mother_cat")
320 gamm_anti_avg_lat_error_corrected_trials_medu <- gamm(as.formula(model_formula),
321   random = list(lunaid=~age_c),
322   data = dat,
323   method = 'REML')
324 summary(gamm_anti_avg_lat_error_corrected_trials_medu$gam)
325 summary(gamm_anti_avg_lat_error_corrected_trials_medu$lme)
326
327
328 # save random slopes and intercepts from final model
329 ranef_anti_avg_lat_error_corrected_trials <- ranef(gamm_anti_avg_lat_error_corrected_
330   _trials$lme)$lunaid %%
331   rename(anti_avg_lat_error_corrected_trials_int = '(Intercept)',
332   anti_avg_lat_error_corrected_trials_rslope = age_c) %%
333   tibble::rownames_to_column(., 'lunaid') %%
334   mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
335
336 #### II. Brain function outcomes - antisaccade correct trials vs. baseline
337 ## 1. SEF
338 # fit the model with smooth term for age
339 model_formula <- as.formula("SEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx = FALSE,
340   bs = 'tp')")
340 gamm_SEF_anti_corr_v_baseline <- gamm(model_formula,
341   random = list(lunaid=~age_c),

```

```

342                     data = dat,
343                     method = 'REML')
344 summary(gamm_SEF_anti_corr_v_baseline$gam)
345 summary(gamm_SEF_anti_corr_v_baseline$lme)
346
347 # model diagnostics
348 par(mfrow = c(2,2))
349 gam.check(gamm_SEF_anti_corr_v_baseline$gam)
350
351 # plot
352 SEF_anti_corr_v_baseline_plot <- plot.gamm(
353   model = gamm_SEF_anti_corr_v_baseline,
354   y_axis_label = 'SEF percent signal change',
355   sig = FALSE)
356 SEF_anti_corr_v_baseline_plot
357
358
359 # fit the model w/ fixed effects of sex
360 model_formula <- as.formula("SEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx = FALSE,
361   bs = 'tp') + sex")
361 gamm_SEF_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
362   random = list(lunaid=~age_c),
363   data = dat,
364   method = 'REML')
365 summary(gamm_SEF_anti_corr_v_baseline_sex$gam)
366 summary(gamm_SEF_anti_corr_v_baseline_sex$lme)
367
368 # fit the model w/ fixed effects of maternal education level
369 model_formula <- as.formula("SEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx = FALSE,
370   bs = 'tp') + level_edu_mother_cat")
370 gamm_SEF_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
371   random = list(lunaid=~age_c),
372   data = dat,
373   method = 'REML')
374 summary(gamm_SEF_anti_corr_v_baseline_medu$gam)
375 summary(gamm_SEF_anti_corr_v_baseline_medu$lme)
376
377 # save random slopes and intercepts
378 ranef_SEF_anti_corr_v_baseline <- ranef(gamm_SEF_anti_corr_v_baseline$lme)$lunaid
379   %>%
380   rename(SEF_anti_corr_v_baseline_int = '(Intercept)',
381         SEF_anti_corr_v_baseline_rslope = age_c) %>%
382   tibble::rownames_to_column(., 'lunaid') %>%
383   mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
384
385 ## 2. pre-SMA
386 # fit the model with smooth term for age
387 model_formula <- as.formula("pre_SMA_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
388   FALSE, bs = 'tp')")
389 gamm_pre_SMA_anti_corr_v_baseline <- gamm(model_formula,
390   random = list(lunaid=~age_c),
391   data = dat,
392   method = 'REML')
393 summary(gamm_pre_SMA_anti_corr_v_baseline$gam)
394 summary(gamm_pre_SMA_anti_corr_v_baseline$lme)
395
396 # model diagnostics
397 par(mfrow = c(2,2))

```

```

397 gam.check(gamm_pre_SMA_anti_corr_v_baseline$gam)
398
399 # plot
400 pre_SMA_anti_corr_v_baseline_plot <- plot_gamm(
401   model = gamm_pre_SMA_anti_corr_v_baseline ,
402   y_axis_label = 'Pre-SMA percent signal change',
403   sig = FALSE)
404 pre_SMA_anti_corr_v_baseline_plot
405
406
407 # fit the model w/ fixed effects of sex
408 model_formula <- as.formula("pre_SMA_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
409   FALSE, bs = 'tp') + sex")
410 gamm_pre_SMA_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
411                                               random = list(lunaid=~age_c) ,
412                                               data = dat,
413                                               method = 'REML')
414 summary(gamm_pre_SMA_anti_corr_v_baseline_sex$gam)
415 summary(gamm_pre_SMA_anti_corr_v_baseline_sex$lme)
416
417 # fit the model w/ fixed effects of maternal education level
418 model_formula <- as.formula("pre_SMA_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
419   FALSE, bs = 'tp') + level_edu_mother_cat")
420 gamm_pre_SMA_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
421                                               random = list(lunaid=~age_c) ,
422                                               data = dat,
423                                               method = 'REML')
424 summary(gamm_pre_SMA_anti_corr_v_baseline_medu$gam)
425 summary(gamm_pre_SMA_anti_corr_v_baseline_medu$lme)
426
427 # save random slopes and intercepts from final model
428 ranef_pre_SMA_anti_corr_v_baseline <- ranef(gamm_pre_SMA_anti_corr_v_baseline$lme)$
429   lunaid %>%
430   rename(pre_SMA_anti_corr_v_baseline_int = '(Intercept)' ,
431         pre_SMA_anti_corr_v_baseline_rslope = age_c) %>%
432   tibble::rownames_to_column(., 'lunaid') %>%
433   mutate(lunaid = as.factor(gsub('1', ' ', lunaid)))
434
435 ## 3. L FEF
436 # fit the model with smooth term for age
437 model_formula <- as.formula("L_FEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
438   FALSE, bs = 'tp')")
439 gamm_L_FEF_anti_corr_v_baseline <- gamm(model_formula ,
440                                               random = list(lunaid=~age_c) ,
441                                               data = dat,
442                                               method = 'REML')
443 summary(gamm_L_FEF_anti_corr_v_baseline$gam)
444 summary(gamm_L_FEF_anti_corr_v_baseline$lme)
445
446 # model diagnostics
447 par(mfrow = c(2,2))
448 gam.check(gamm_L_FEF_anti_corr_v_baseline$gam)
449
450 # identify significant periods of developmental change
451 L_FEF_devchange <- calc_dev_change(gamm_L_FEF_anti_corr_v_baseline)
452 L_FEF_devchange
453
454 # plot

```

```

452 L_FEF_anti_corr_v_baseline_plot <- plot_gamm(
453   model = gamm_L_FEF_anti_corr_v_baseline ,
454   y_axis_label = 'L FEF percent signal change')
455 L_FEF_anti_corr_v_baseline_plot
456
457 L_FEF_anti_corr_v_baseline_plot <- L_FEF_anti_corr_v_baseline_plot +
458   annotate(geom = 'rect',
459             xmin = L_FEF_devchange[1] ,
460             xmax = L_FEF_devchange[2] ,
461             ymin = -5, ymax = -4.5,
462             fill = '#476FD1')
463 L_FEF_anti_corr_v_baseline_plot
464
465
466 # fit the model w/ fixed effects of sex
467 model_formula <- as.formula("L_FEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
468   FALSE, bs = 'tp') + sex")
469 gamm_L_FEF_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula) ,
470                                               random = list(lunaid=~age_c) ,
471                                               data = dat,
472                                               method = 'REML')
473 summary(gamm_L_FEF_anti_corr_v_baseline_sex$gam)
474 summary(gamm_L_FEF_anti_corr_v_baseline_sex$lme)
475
476 # fit the model w/ fixed effects of maternal education level
477 model_formula <- as.formula("L_FEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
478   FALSE, bs = 'tp') + level_edu_mother_cat")
479 gamm_L_FEF_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula) ,
480                                               random = list(lunaid=~age_c) ,
481                                               data = dat,
482                                               method = 'REML')
483 summary(gamm_L_FEF_anti_corr_v_baseline_medu$gam)
484 summary(gamm_L_FEF_anti_corr_v_baseline_medu$lme)
485
486 # save random slopes and intercepts from final model
487 ranef_L_FEF_anti_corr_v_baseline <- ranef(gamm_L_FEF_anti_corr_v_baseline$lme)$
488   lunaid %>%
489   rename(L_FEF_anti_corr_v_baseline_int = '(Intercept)' ,
490         L_FEF_anti_corr_v_baseline_rslope = age_c) %>%
491   tibble::rownames_to_column(., 'lunaid') %>%
492   mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
493
494 ## 4. R FEF
495 # fit the model with smooth term for age
496 model_formula <- as.formula("R_FEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
497   FALSE, bs = 'tp')")
498 gamm_R_FEF_anti_corr_v_baseline <- gamm(model_formula ,
499                                               random = list(lunaid=~age_c) ,
500                                               data = dat,
501                                               method = 'REML',
502                                               control = lmeControl(maxIter = 50))
503 summary(gamm_R_FEF_anti_corr_v_baseline$gam)
504 summary(gamm_R_FEF_anti_corr_v_baseline$lme)
505
506 # model diagnostics
507 par(mfrow = c(2,2))
508 gam.check(gamm_R_FEF_anti_corr_v_baseline$gam)

```

```

507 # plot
508 R_FEF_anti_corr_v_baseline_plot <- plot_gamm(
509   model = gamm_R_FEF_anti_corr_v_baseline ,
510   y_axis_label = 'R FEF percent signal change' ,
511   sig = FALSE)
512 R_FEF_anti_corr_v_baseline_plot
513
514 # fit the model w/ fixed effects of sex
515 model_formula <- as.formula("R_FEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
516   FALSE, bs = 'tp') + sex")
516 gamm_R_FEF_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula) ,
517   random = list(lunaid=~age_c) ,
518   data = dat ,
519   method = 'REML' ,
520   control = lmeControl(opt = 'optim'))
521 summary(gamm_R_FEF_anti_corr_v_baseline_sex$gam)
522 summary(gamm_R_FEF_anti_corr_v_baseline_sex$lme)
523
524 # fit the model w/ fixed effects of maternal education level
525 model_formula <- as.formula("R_FEF_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
526   FALSE, bs = 'tp') + level_edu_mother_cat")
526 gamm_R_FEF_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula) ,
527   random = list(lunaid=~age_c) ,
528   data = dat ,
529   method = 'REML' ,
530   control = lmeControl(opt = 'optim'))
531 summary(gamm_R_FEF_anti_corr_v_baseline_medu$gam)
532 summary(gamm_R_FEF_anti_corr_v_baseline_medu$lme)
533
534 # save random slopes and intercepts
535 ranef_R_FEF_anti_corr_v_baseline <- ranef(gamm_R_FEF_anti_corr_v_baseline$lme)$
536   lunaid %>%
537     rename(R_FEF_anti_corr_v_baseline_rint = '(Intercept)' ,
538           R_FEF_anti_corr_v_baseline_rslope = age_c) %>%
539     tibble::rownames_to_column(., 'lunaid') %>%
540     mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
541
542 ## 5. L putamen
543 # fit the model with smooth term for age
544 model_formula <- as.formula("L_putamen_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
545   FALSE, bs = 'tp')")
546 gamm_L_putamen_anti_corr_v_baseline <- gamm(model_formula ,
547   random = list(lunaid=~age_c) ,
548   data = dat ,
549   method = 'REML' ,
550   control = lmeControl(maxIter = 50))
551 summary(gamm_L_putamen_anti_corr_v_baseline$gam)
552 summary(gamm_L_putamen_anti_corr_v_baseline$lme)
553
554 # model diagnostics
555 par(mfrow = c(2,2))
556 gam.check(gamm_L_putamen_anti_corr_v_baseline$gam)
557
558 # plot
559 L_putamen_anti_corr_v_baseline_plot <- plot_gamm(
560   model = gamm_L_putamen_anti_corr_v_baseline ,
561   y_axis_label = 'L putamen percent signal change' ,
562   sig = FALSE)

```

```

562 L_putamen_anti_corr_v_baseline_plot
563
564
565 # fit the model w/ fixed effects of sex
566 model_formula <- as.formula("L_putamen_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
567 FALSE, bs = 'tp') + sex")
568 gamm_L_putamen_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
569                                         random = list(lunaid=~age_c),
570                                         data = dat,
571                                         method = 'REML')
572 summary(gamm_L_putamen_anti_corr_v_baseline_sex$gam)
573 summary(gamm_L_putamen_anti_corr_v_baseline_sex$lme)
574
575 # fit the model w/ fixed effects of maternal education level
576 model_formula <- as.formula("L_putamen_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
577 FALSE, bs = 'tp') + level_edu_mother_cat")
578 gamm_L_putamen_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
579                                         random = list(lunaid=~age_c),
580                                         data = dat,
581                                         method = 'REML')
582 summary(gamm_L_putamen_anti_corr_v_baseline_medu$gam)
583 summary(gamm_L_putamen_anti_corr_v_baseline_medu$lme)
584
585 # save random slopes and intercepts
586 ranef_L_putamen_anti_corr_v_baseline <- ranef(gamm_L_putamen_anti_corr_v_baseline$  

587 lme)$lunaid %>%
588 rename(L_putamen_anti_corr_v_baseline_int = '(Intercept)',
589        L_putamen_anti_corr_v_baseline_rslope = age_c) %>%
590 tibble::rownames_to_column(., 'lunaid') %>%
591 mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
592
593 ## 6. R putamen
594 # fit the model with smooth term for age
595 model_formula <- as.formula("R_putamen_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
596 FALSE, bs = 'tp')")
597 gamm_R_putamen_anti_corr_v_baseline <- gamm(model_formula,
598                                         random = list(lunaid=~age_c),
599                                         data = dat,
600                                         method = 'REML',
601                                         control = lmeControl(maxIter = 50))
602 summary(gamm_R_putamen_anti_corr_v_baseline$gam)
603 summary(gamm_R_putamen_anti_corr_v_baseline$lme)
604
605 # model diagnostics
606 par(mfrow = c(2,2))
607 gam.check(gamm_R_putamen_anti_corr_v_baseline$gam)
608
609 # plot
610 R_putamen_anti_corr_v_baseline_plot <- plot_gamm(
611   model = gamm_R_putamen_anti_corr_v_baseline,
612   y_axis_label = 'R putamen percent signal change',
613   sig = FALSE
614 )
615 R_putamen_anti_corr_v_baseline_plot
616
617 # fit the model w/ fixed effects of sex
618 model_formula <- as.formula("R_putamen_anti_corr_v_baseline ~ s(age_c, k = 10, fx =

```

```

      FALSE, bs = 'tp') + sex")
617 gamm_R_putamen_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
618                                         random = list(lunaid=~age_c),
619                                         data = dat,
620                                         method = 'REML')
621 summary(gamm_R_putamen_anti_corr_v_baseline_sex$gam)
622 summary(gamm_R_putamen_anti_corr_v_baseline_sex$lme)
623
624 # fit the model w/ fixed effects of maternal education level
625 model_formula <- as.formula("R_putamen_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
626   FALSE, bs = 'tp') + level_edu_mother_cat")
627 gamm_R_putamen_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
628                                         random = list(lunaid=~age_c),
629                                         data = dat,
630                                         method = 'REML')
631 summary(gamm_R_putamen_anti_corr_v_baseline_medu$gam)
632 summary(gamm_R_putamen_anti_corr_v_baseline_medu$lme)
633
634 # save random slopes and intercepts from final model
635 ranef_R_putamen_anti_corr_v_baseline <- ranef(gamm_R_putamen_anti_corr_v_baseline$lme)$lunaid %%
636   rename(R_putamen_anti_corr_v_baseline_int = '(Intercept)',
637         R_putamen_anti_corr_v_baseline_rslope = age_c) %%
638   tibble::rownames_to_column(., 'lunaid') %%
639   mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
640
641 ## 7. L pPC
642 # fit the model with smooth term for age
643 model_formula <- as.formula("L_pPC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
644   FALSE, bs = 'tp')")
645 gamm_L_pPC_anti_corr_v_baseline <- gamm(model_formula,
646                                         random = list(lunaid=~age_c),
647                                         data = dat,
648                                         method = 'REML',
649                                         control = lmeControl(maxIter = 50))
650 summary(gamm_L_pPC_anti_corr_v_baseline$gam)
651 summary(gamm_L_pPC_anti_corr_v_baseline$lme)
652
653 # model diagnostics
654 par(mfrow = c(2,2))
655 gam.check(gamm_L_pPC_anti_corr_v_baseline$gam)
656
657 # identify significant periods of developmental change
658 L_pPC_devchange <- calc_dev_change(gamm_L_pPC_anti_corr_v_baseline)
659 L_pPC_devchange
660
661 # plot
662 L_pPC_anti_corr_v_baseline_plot <- plot_gamm(
663   model = gamm_L_pPC_anti_corr_v_baseline,
664   y_axis_label = 'L pPC percent signal change',
665   sig = TRUE)
666 L_pPC_anti_corr_v_baseline_plot
667
668 L_pPC_anti_corr_v_baseline_plot <- L_pPC_anti_corr_v_baseline_plot +
669   annotate(geom = 'rect',
670             xmin = L_pPC_devchange[1],
671             xmax = L_pPC_devchange[2],
672             ymin = -8,

```

```

672     ymax = -7.5,
673     fill = '#476FD1')
674 L_pPC_anti_corr_v_baseline_plot
675
676
677 # fit the model w/ fixed effects of sex
678 model_formula <- as.formula("L_pPC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
679   FALSE, bs = 'tp') + sex")
680 gamm_L_pPC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
681                                             random = list(lunaid=~age_c),
682                                             data = dat,
683                                             method = 'REML')
684 summary(gamm_L_pPC_anti_corr_v_baseline_sex$gam)
685 summary(gamm_L_pPC_anti_corr_v_baseline_sex$lme)
686
687 # fit the model w/ fixed effects of maternal education level
688 model_formula <- as.formula("L_pPC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
689   FALSE, bs = 'tp') + level_edu_mother_cat")
690 gamm_L_pPC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
691                                             random = list(lunaid=~age_c),
692                                             data = dat,
693                                             method = 'REML')
694 summary(gamm_L_pPC_anti_corr_v_baseline_medu$gam)
695 summary(gamm_L_pPC_anti_corr_v_baseline_medu$lme)
696
697 # save random slopes and intercepts from final model
698 ranef_L_pPC_anti_corr_v_baseline <- ranef(gamm_L_pPC_anti_corr_v_baseline$lme)$
699   lunaid %>%
700   rename(L_pPC_anti_corr_v_baseline_int = '(Intercept)',
701         L_pPC_anti_corr_v_baseline_rslope = age_c) %>%
702   tibble::rownames_to_column(., 'lunaid') %>%
703   mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
704
705 ## 8. R pPC
706 # fit the model with smooth term for age
707 model_formula <- as.formula("R_pPC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
708   FALSE, bs = 'tp')")
709 gamm_R_pPC_anti_corr_v_baseline <- gamm(model_formula,
710                                             random = list(lunaid=~age_c),
711                                             data = dat,
712                                             method = 'REML',
713                                             control = lmeControl(maxIter = 50))
714 summary(gamm_R_pPC_anti_corr_v_baseline$gam)
715 summary(gamm_R_pPC_anti_corr_v_baseline$lme)
716
717 # model diagnostics
718 par(mfrow = c(2,2))
719 gam.check(gamm_R_pPC_anti_corr_v_baseline$gam)
720
721 # identify significant periods of developmental change
722 R_pPC_devchange <- calc_dev_change(gamm_R_pPC_anti_corr_v_baseline)
723 R_pPC_devchange
724
725 # plot
726 R_pPC_anti_corr_v_baseline_plot <- plot_gamm(
727   model = gamm_R_pPC_anti_corr_v_baseline,
728   y_axis_label = 'R pPC percent signal change',
729   sig = TRUE)

```

```

727 R_pPC_anti_corr_v_baseline_plot
728
729 R_pPC_anti_corr_v_baseline_plot <- R_pPC_anti_corr_v_baseline_plot +
730   annotate(geom = 'rect',
731             xmin = R_pPC_devchange[1],
732             xmax = R_pPC_devchange[2],
733             ymin = -8.5,
734             ymax = -8,
735             fill = '#476FD1')
736 R_pPC_anti_corr_v_baseline_plot
737
738 # fit the model w/ fixed effects of sex
739 model_formula <- as.formula("R_pPC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
740   FALSE, bs = 'tp') + sex")
740 gamm_R_pPC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
741                                               random = list(lunaid=~age_c),
742                                               data = dat,
743                                               method = 'REML')
744 summary(gamm_R_pPC_anti_corr_v_baseline_sex$gam)
745 summary(gamm_R_pPC_anti_corr_v_baseline_sex$lme)
746
747 # fit the model w/ fixed effects of maternal education level
748 model_formula <- as.formula("R_pPC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
749   FALSE, bs = 'tp') + level_edu_mother_cat")
750 gamm_R_pPC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
751                                               random = list(lunaid=~age_c),
752                                               data = dat,
753                                               method = 'REML')
754 summary(gamm_R_pPC_anti_corr_v_baseline_medu$gam)
755 summary(gamm_R_pPC_anti_corr_v_baseline_medu$lme)
756
757 # save random slopes and intercepts
758 ranef_R_pPC_anti_corr_v_baseline <- ranef(gamm_R_pPC_anti_corr_v_baseline$lme)$
759   lunaid %>%
760     rename(R_pPC_anti_corr_v_baseline_int = '(Intercept)',
761           R_pPC_anti_corr_v_baseline_rslope = age_c) %>%
762     tibble::rownames_to_column(., 'lunaid') %>%
763     mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
764
765 ## 9. L dlPFC
766 # fit the model with smooth term for age
767 model_formula <- as.formula("L_dlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
768   FALSE, bs = 'tp')")
769 gamm_L_dlPFC_anti_corr_v_baseline <- gamm(model_formula,
770                                               random = list(lunaid=~age_c),
771                                               data = dat,
772                                               method = 'REML',
773                                               control = lmeControl(maxIter = 50))
774 summary(gamm_L_dlPFC_anti_corr_v_baseline$gam)
775 summary(gamm_L_dlPFC_anti_corr_v_baseline$lme)
776
777 # model diagnostics
778 par(mfrow = c(2,2))
779 gam.check(gamm_L_dlPFC_anti_corr_v_baseline$gam)
780
781 # plot
782 L_dlPFC_anti_corr_v_baseline_plot <- plot_gamm(
783   model = gamm_L_dlPFC_anti_corr_v_baseline,

```

```

782   y_axis_label = 'L dlPFC percent signal change',
783   sig = FALSE)
784 L_dlPFC_anti_corr_v_baseline_plot
785
786 # fit the model w/ fixed effects of sex
787 model_formula <- as.formula("L_dlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
788   FALSE, bs = 'tp') + sex")
789 gamm_L_dlPFC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
790   random = list(lunaid=~age_c),
791   data = dat,
792   method = 'REML')
793 summary(gamm_L_dlPFC_anti_corr_v_baseline_sex$gam)
794 summary(gamm_L_dlPFC_anti_corr_v_baseline_sex$lme)
795
796 # fit the model w/ fixed effects of maternal education level
797 model_formula <- as.formula("L_dlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
798   FALSE, bs = 'tp') + level_edu_mother_cat")
799 gamm_L_dlPFC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
800   random = list(lunaid=~age_c),
801   data = dat,
802   method = 'REML')
803 summary(gamm_L_dlPFC_anti_corr_v_baseline_medu$gam)
804 summary(gamm_L_dlPFC_anti_corr_v_baseline_medu$lme)
805
806 # save random slopes and intercepts
807 ranef_L_dlPFC_anti_corr_v_baseline <- ranef(gamm_L_dlPFC_anti_corr_v_baseline$lme)$
808   lunaid %>%
809   rename(L_dlPFC_anti_corr_v_baseline_int = '(Intercept)',
810     L_dlPFC_anti_corr_v_baseline_rslope = age_c) %>%
811   tibble::rownames_to_column(., 'lunaid') %>%
812   mutate(lunaid = as.factor(gsub('/', '', lunaid)))
813
814 ## 10. R dlPFC
815 # fit the model with smooth term for age
816 model_formula <- as.formula("R_dlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
817   FALSE, bs = 'tp')")
818 gamm_R_dlPFC_anti_corr_v_baseline <- gamm(model_formula,
819   random = list(lunaid=~age_c),
820   data = dat,
821   method = 'REML',
822   control = lmeControl(maxIter = 50))
823 summary(gamm_R_dlPFC_anti_corr_v_baseline$gam)
824 summary(gamm_R_dlPFC_anti_corr_v_baseline$lme)
825
826 # model diagnostics
827 par(mfrow = c(2,2))
828 gam.check(gamm_R_dlPFC_anti_corr_v_baseline$gam)
829
830 # identify significant periods of developmental change
831 R_dlPFC_devchange <- calc_dev_change(gamm_R_dlPFC_anti_corr_v_baseline)
832 R_dlPFC_devchange
833
834 # plot
835 R_dlPFC_anti_corr_v_baseline_plot <- plot_gamm(
836   model = gamm_R_dlPFC_anti_corr_v_baseline,
837   y_axis_label = 'R dlPFC percent signal change',
838   sig = TRUE)
839 R_dlPFC_anti_corr_v_baseline_plot

```

```

837 R_dlPFC_anti_corr_v_baseline_plot <- R_dlPFC_anti_corr_v_baseline_plot +
838   annotate(geom = 'rect',
839             xmin = R_dlPFC_devchange[1],
840             xmax = R_dlPFC_devchange[2],
841             ymin = -8.5, ymax = -8,
842             fill = '#476FD1')
843 R_dlPFC_anti_corr_v_baseline_plot
845
846 # fit the model w/ fixed effects of sex
847 model_formula <- as.formula("R_dlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
848   FALSE, bs = 'tp') + sex")
849 gamm_R_dlPFC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
850                                                 random = list(lunaid=~age_c),
851                                                 data = dat,
852                                                 method = 'REML')
853 summary(gamm_R_dlPFC_anti_corr_v_baseline_sex$gam)
854 summary(gamm_R_dlPFC_anti_corr_v_baseline_sex$lme)
855
856 # fit the model w/ fixed effects of maternal education level
857 model_formula <- as.formula("R_dlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
858   FALSE, bs = 'tp') + level_edu_mother_cat")
859 gamm_R_dlPFC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
860                                                 random = list(lunaid=~age_c),
861                                                 data = dat,
862                                                 method = 'REML')
863 summary(gamm_R_dlPFC_anti_corr_v_baseline_medu$gam)
864 summary(gamm_R_dlPFC_anti_corr_v_baseline_medu$lme)
865
866 # save random slopes and intercepts from final model
867 ranef_R_dlPFC_anti_corr_v_baseline <- ranef(gamm_R_dlPFC_anti_corr_v_baseline$lme)$
868   lunaid %>%
869   rename(R_dlPFC_anti_corr_v_baseline_int = '(Intercept)',
870         R_dlPFC_anti_corr_v_baseline_rslope = age_c) %>%
871   tibble::rownames_to_column(., 'lunaid') %>%
872   mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
873
874 ## 11. L_vIPFC
875 # fit the model with smooth term for age
876 model_formula <- as.formula("L_vIPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
877   FALSE, bs = 'tp')")
878 gamm_L_vIPFC_anti_corr_v_baseline <- gamm(model_formula,
879                                                 random = list(lunaid=~age_c),
880                                                 data = dat,
881                                                 method = 'REML',
882                                                 control = lmeControl(maxIter = 100,
883                                                       msMaxIter = 100, niterEM = 50, opt = ,
884                                                       optim'))
885 summary(gamm_L_vIPFC_anti_corr_v_baseline$gam)
886 summary(gamm_L_vIPFC_anti_corr_v_baseline$lme)
887
888 # model diagnostics
889 par(mfrow = c(2,2))
890 gam.check(gamm_L_vIPFC_anti_corr_v_baseline$gam)
891
892 # plot
893 L_vIPFC_anti_corr_v_baseline_plot <- plot_gamm(
894   model = gamm_L_vIPFC_anti_corr_v_baseline,

```

```

890   y_axis_label = 'L_vlPFC percent signal change',
891   sig = FALSE)
892 L_vlPFC_anti_corr_v_baseline_plot
893
894 # fit the model w/ fixed effects of sex
895 model_formula <- as.formula("L_vlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
896   FALSE, bs = 'tp') + sex")
897 gamm_L_vlPFC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
898   random = list(lunaid=~age_c),
899   data = dat,
900   method = 'REML',
901   control = lmeControl(opt = 'optim'))
902 summary(gamm_L_vlPFC_anti_corr_v_baseline_sex$gam)
903 summary(gamm_L_vlPFC_anti_corr_v_baseline_sex$lme)
904
905 # fit the model w/ fixed effects of maternal education level
906 model_formula <- as.formula("L_vlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
907   FALSE, bs = 'tp') + level_edu_mother_cat")
908 gamm_L_vlPFC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
909   random = list(lunaid=~age_c),
910   data = dat,
911   method = 'REML',
912   control = lmeControl(opt = 'optim'))
913 summary(gamm_L_vlPFC_anti_corr_v_baseline_medu$gam)
914 summary(gamm_L_vlPFC_anti_corr_v_baseline_medu$lme)
915
916 # save random slopes and intercepts from final model
917 ranef_L_vlPFC_anti_corr_v_baseline <- ranef(gamm_L_vlPFC_anti_corr_v_baseline$lme)$
918   lunaid %>%
919   rename(L_vlPFC_anti_corr_v_baseline_int = '(Intercept)',
920         L_vlPFC_anti_corr_v_baseline_rslope = age_c) %>%
921   tibble::rownames_to_column(., 'lunaid') %>%
922   mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
923
924 ## 12. R_vlPFC
925 # fit the model with smooth term for age
926 model_formula <- as.formula("R_vlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
927   FALSE, bs = 'tp')")
928 gamm_R_vlPFC_anti_corr_v_baseline <- gamm(model_formula,
929   random = list(lunaid=~age_c),
930   data = dat,
931   method = 'REML',
932   control = lmeControl(maxIter = 50))
933 summary(gamm_R_vlPFC_anti_corr_v_baseline$gam)
934 summary(gamm_R_vlPFC_anti_corr_v_baseline$lme)
935
936 # model diagnostics
937 par(mfrow = c(2,2))
938 gam.check(gamm_R_vlPFC_anti_corr_v_baseline$gam)
939
940 # plot
941 R_vlPFC_anti_corr_v_baseline_plot <- plot_gamm(
942   model = gamm_R_vlPFC_anti_corr_v_baseline,
943   y_axis_label = 'R_vlPFC percent signal change',
944   sig = FALSE)
945 R_vlPFC_anti_corr_v_baseline_plot
946
947 # fit the model w/ fixed effects of sex

```

```

945 model_formula <- as.formula("R_vlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
946   FALSE, bs = 'tp') + sex")
947 gamm_R_vlPFC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
948   random = list(lunaid=~age_c),
949   data = dat,
950   method = 'REML')
951 summary(gamm_R_vlPFC_anti_corr_v_baseline_sex$gam)
952 summary(gamm_R_vlPFC_anti_corr_v_baseline_sex$lme)
953
954 # fit the model w/ fixed effects of maternal education level
955 model_formula <- as.formula("R_vlPFC_anti_corr_v_baseline ~ s(age_c, k = 10, fx =
956   FALSE, bs = 'tp') + level_edu_mother_cat")
957 gamm_R_vlPFC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
958   random = list(lunaid=~age_c),
959   data = dat,
960   method = 'REML')
961 summary(gamm_R_vlPFC_anti_corr_v_baseline_medu$gam)
962 summary(gamm_R_vlPFC_anti_corr_v_baseline_medu$lme)
963
964 # save random slopes and intercepts from final model
965 ranef_R_vlPFC_anti_corr_v_baseline <- ranef(gamm_R_vlPFC_anti_corr_v_baseline$lme)$
966   lunaid %>%
967   rename(R_vlPFC_anti_corr_v_baseline_int = '(Intercept)',
968     R_vlPFC_anti_corr_v_baseline_rslope = age_c) %>%
969   tibble::rownames_to_column(., 'lunaid') %>%
970   mutate(lunaid = as.factor(gsub('1/', '', lunaid)))
971
972 ## 13. dACC
973 # fit the model with smooth term for age
974 model_formula <- as.formula("dACC_anti_corr_v_baseline ~ s(age_c, k = 10, fx = FALSE
975   , bs = 'tp')")
976 gamm_dACC_anti_corr_v_baseline <- gamm(model_formula,
977   random = list(lunaid=~age_c),
978   data = dat,
979   method = 'REML',
980   control = lmeControl(maxIter = 50))
981 summary(gamm_dACC_anti_corr_v_baseline$gam)
982 summary(gamm_dACC_anti_corr_v_baseline$lme)
983
984 # model diagnostics
985 par(mfrow = c(2,2))
986 gam.check(gamm_dACC_anti_corr_v_baseline$gam)
987
988 # plot
989 dACC_anti_corr_v_baseline_plot <- plot_gamm(
990   model = gamm_dACC_anti_corr_v_baseline,
991   y_axis_label = 'dACC percent signal change correct trials',
992   sig = FALSE)
993 dACC_anti_corr_v_baseline_plot
994
995 # fit the model w/ fixed effects of gender
996 model_formula <- as.formula("dACC_anti_corr_v_baseline ~ s(age_c, k = 10, fx = FALSE
997   , bs = 'tp') + sex")
998 gamm_dACC_anti_corr_v_baseline_sex <- gamm(as.formula(model_formula),
999   random = list(lunaid=~age_c),
1000   data = dat,
1001   method = 'REML')
1002 summary(gamm_dACC_anti_corr_v_baseline_sex$gam)

```

```

999 summary(gamm_dACC_anti_corr_v_baseline_sex$lme)
1000
1001 # fit the model w/ fixed effects of maternal education level
1002 model_formula <- as.formula("dACC_anti_corr_v_baseline ~ s(age_c, k = 10, fx = FALSE
1003 , bs = 'tp') + level_edu_mother_cat")
1004 gamm_dACC_anti_corr_v_baseline_medu <- gamm(as.formula(model_formula),
1005 random = list(lunaid=~age_c),
1006 data = dat,
1007 method = 'REML')
1008 summary(gamm_dACC_anti_corr_v_baseline_medu$gam)
1009 summary(gamm_dACC_anti_corr_v_baseline_medu$lme)
1010
1011 # save random slopes and intercepts, adjusting for maternal education
1012 ranef_dACC_anti_corr_v_baseline <- ranef(gamm_dACC_anti_corr_v_baseline_medu$lme)$
1013 lunaid %>%
1014 rename(dACC_anti_corr_v_baseline_int = '(Intercept)',
1015 dACC_anti_corr_v_baseline_rslope = age_c) %>%
1016 tibble::rownames_to_column(., 'lunaid') %>%
1017 mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
1018
1019 ### III. Brain function outcomes – antisaccade error–corrected trials vs. baseline
1020 ## 1. dACC
1021 # fit the model with smooth term for age
1022 model_formula <- as.formula("dACC_anti_errcorr_v_baseline ~ s(age_c, k = 10, fx =
1023 FALSE, bs = 'tp')")
1024 gamm_dACC_anti_errcorr_v_baseline <- gamm(model_formula,
1025 random = list(lunaid=~age_c),
1026 data = dat,
1027 method = 'REML',
1028 control = lmeControl(opt = 'optim'))
1029 summary(gamm_dACC_anti_errcorr_v_baseline$gam)
1030 summary(gamm_dACC_anti_errcorr_v_baseline$lme)
1031
1032 # model diagnostics
1033 par(mfrow = c(2,2))
1034 gam.check(gamm_dACC_anti_errcorr_v_baseline$gam)
1035
1036 # identify significant periods of developmental change
1037 dACC_errcorr_devchange <- calc_dev_change(gamm_dACC_anti_errcorr_v_baseline)
1038 dACC_errcorr_devchange
1039
1040 # plot
1041 dACC_anti_errcorr_v_baseline_plot <- plot_gamm(
1042 model = gamm_dACC_anti_errcorr_v_baseline,
1043 y_axis_label = 'dACC percent signal change error–corrected trials',
1044 sig = TRUE)
1045 dACC_anti_errcorr_v_baseline_plot
1046
1047 # fit the model w/ fixed effects of sex
1048 model_formula <- as.formula("dACC_anti_errcorr_v_baseline ~ s(age_c, k = 10, fx =
1049 FALSE, bs = 'tp') + sex")
1050 gamm_dACC_anti_errcorr_v_baseline_sex <- gamm(as.formula(model_formula),
1051 random = list(lunaid=~age_c),
1052 data = dat,
1053 method = 'REML')
1054 summary(gamm_dACC_anti_errcorr_v_baseline_sex$gam)
1055 summary(gamm_dACC_anti_errcorr_v_baseline_sex$lme)

```

```

1054 # identify significant periods of developmental change, adjusting for sex
1055 dACC_errcorr_sex_devchange <- calc_dev_change(gamm_dACC_anti_errcorr_v_baseline_sex)
1056 dACC_errcorr_sex_devchange
1057
1058 dACC_anti_errcorr_v_baseline_plot <- dACC_anti_errcorr_v_baseline_plot +
1059   annotate(geom = 'rect',
1060             xmin = dACC_errcorr_sex_devchange[1],
1061             xmax = dACC_errcorr_sex_devchange[2],
1062             ymin = -25,
1063             ymax = -24,
1064             fill = '#476FD1')
1065 dACC_anti_errcorr_v_baseline_plot
1066
1067 # fit the model w/ fixed effects of maternal education level
1068 model_formula <- as.formula("dACC_anti_errcorr_v_baseline ~ s(age_c, k = 10, fx =
1069   FALSE, bs = 'tp') + level_edu_mother_cat")
1070 gamm_dACC_anti_errcorr_v_baseline_medu <- gamm(as.formula(model_formula),
1071                                                 random = list(lunaid=~age_c),
1072                                                 data = dat,
1073                                                 method = 'REML')
1074 summary(gamm_dACC_anti_errcorr_v_baseline_medu$gam)
1075 summary(gamm_dACC_anti_errcorr_v_baseline_medu$lme)
1076
1077 # save random slopes and intercepts, adjusting for sex
1078 ranef_dACC_anti_errcorr_v_baseline <- ranef(gamm_dACC_anti_errcorr_v_baseline_sex$`lme`)$lunaid %>%
1079   rename(dACC_anti_errcorr_v_baseline_int = '(Intercept)',
1080         dACC_anti_errcorr_v_baseline_rslope = age_c) %>%
1081   tibble::rownames_to_column(., 'lunaid') %>%
1082   mutate(lunaid = as.factor(gsub('1/', ' ', lunaid)))
1083
1084 ## test of global effect of maternal education level for each model
1085 mods <- c('gamm_anti_perc_correct_nodrop',
1086           'gamm_anti_perc_error_corrected_nodrop',
1087           'gamm_anti_avg_lat_correct_trials',
1088           'gamm_anti_avg_lat_error_corrected_trials',
1089           'gamm_SEF_anti_corr_v_baseline',
1090           'gamm_pre_SMA_anti_corr_v_baseline',
1091           'gamm_L_FEF_anti_corr_v_baseline',
1092           'gamm_R_FEF_anti_corr_v_baseline',
1093           'gamm_L_putamen_anti_corr_v_baseline',
1094           'gamm_R_putamen_anti_corr_v_baseline',
1095           'gamm_L_pPC_anti_corr_v_baseline',
1096           'gamm_R_pPC_anti_corr_v_baseline',
1097           'gamm_L_dlPFC_anti_corr_v_baseline',
1098           'gamm_R_dlPFC_anti_corr_v_baseline',
1099           'gamm_L_vIPFC_anti_corr_v_baseline',
1100           'gamm_R_vIPFC_anti_corr_v_baseline',
1101           'gamm_dACC_anti_corr_v_baseline',
1102           'gamm_dACC_anti_errcorr_v_baseline')
1103
1104 global_test_medu <- matrix(nrow = length(mods), ncol = 4)
1105 for(i in 1:length(mods)){
1106   mod_medu <- paste0(mods[i], '_medu')
1107   global_test_medu[i, 1] <- mod_medu
1108   global_test_medu[i, 2:4] <- anova(get(mod_medu)$gam)$pTerms.table[1:3]
1109 }
1110 colnames(global_test_medu) <- c('model', 'df', 'F', 'p-value for global medu')

```

```

1111 global_test_medu
1112 print(xtable(global_test_medu, type = "latex"),
1113       file = "gamm-test_global_sig_medu_terms.tex")
1114
1115 ## tests of effect of sex for each model
1116 test_sex <- matrix(nrow = length(mods), ncol = 4)
1117 for (i in 1:length(mods)){
1118   mod_sex <- paste0(mods[i], '_sex')
1119   test_sex[i, 1] <- mod_sex
1120   test_sex[i, 2:4] <- summary(get(mod_sex)$gam)$p.table[2, c(1, 3:4)]
1121 }
1122 colnames(test_sex) <- c('model', 'estimate', 't', 'p-value for sex')
1123 test_sex
1124 print(xtable(test_sex, type = "latex"),
1125       file = "gamm-test_sig_sex_terms.tex")
1126
1127
1128 #### IV. Compile results and plots
1129 # summary of significance of smooth terms for age for final models
1130 mods <- c('gamm_anti_perc_correct_nodrop',
1131           'gamm_anti_perc_error_corrected_nodrop',
1132           'gamm_anti_avg_lat_correct_trials',
1133           "gamm_anti_avg_lat_error_corrected_trials",
1134           "gamm_SEF_anti_corr_v_baseline",
1135           "gamm_pre_SMA_anti_corr_v_baseline",
1136           "gamm_L_FEF_anti_corr_v_baseline",
1137           "gamm_R_FEF_anti_corr_v_baseline",
1138           "gamm_L_putamen_anti_corr_v_baseline",
1139           "gamm_R_putamen_anti_corr_v_baseline",
1140           "gamm_L_pPC_anti_corr_v_baseline",
1141           "gamm_R_pPC_anti_corr_v_baseline",
1142           "gamm_L_dlPFC_anti_corr_v_baseline",
1143           "gamm_R_dlPFC_anti_corr_v_baseline",
1144           "gamm_L_vLPFC_anti_corr_v_baseline",
1145           "gamm_R_vLPFC_anti_corr_v_baseline",
1146           "gamm_dACC_anti_corr_v_baseline",
1147           "gamm_dACC_anti_errcorr_v_baseline_sex")
1148 sumtab <- sapply(mods, function (x) summary(get(x)$gam)$s.table)
1149 rownames(sumtab) <- c('edf', 'Ref.df', 'F', 'p-value')
1150 sumtab <- as.data.frame(t(sumtab)) %>%
1151   tibble::rownames_to_column(., 'model') %>%
1152   select(-c('Ref.df'))
1153 sumtab <- cbind(sumtab, p.adjust(sumtab$p-value, method = 'fdr'))
1154 names(sumtab)[ncol(sumtab)] <- 'q-value'
1155 sumtab <- sumtab %>%
1156   mutate(survive_multcomp = case_when('q-value' < 0.05 ~ 1,
1157                                         'q-value' >= 0.05 ~ 0))
1158 print(xtable(sumtab %>% select(-c(survive_multcomp)),
1159               digits = c(0,0,2,2,7,7),
1160               type = "latex"),
1161       file = "sumtab_age_terms.tex")
1162
1163 # merge random effects and save for further analysis
1164 random_effects_list <- do.call('list', mget(grep('ranef', names(.GlobalEnv), value =
1165 TRUE)))
1165 random_effects <- random_effects_list %>%
1166   purrr::reduce(left_join, by = 'lunaid')
1167 write.csv(random_effects, 'Data/random_effects_from_gamms_20200312.csv')
1168
```

```

1169 ## plots
1170 # behavioral (accuracy and latency) plot
1171 behav_plot <- grid.arrange(
1172   anti_perc_correct_plot,
1173   anti_avg_lat_correct_trials_plot,
1174   anti_perc_error_corrected_nodrop_plot,
1175   anti_avg_lat_error_corrected_trials_plot,
1176   ncol = 2)
1177 behav_plot
1178 ggsave(filename = 'Plots/behav_plot.png',
1179         plot = behav_plot,
1180         width = 7, height = 7, units = 'in')
1181
1182 # motor response ROIs
1183 motor_resp_rois_plot <- grid.arrange(
1184   SEF_anti_corr_v_baseline_plot,
1185   pre_SMA_anti_corr_v_baseline_plot,
1186   L_FEF_anti_corr_v_baseline_plot,
1187   R_FEF_anti_corr_v_baseline_plot,
1188   L_pPC_anti_corr_v_baseline_plot,
1189   R_pPC_anti_corr_v_baseline_plot,
1190   L_putamen_anti_corr_v_baseline_plot,
1191   R_putamen_anti_corr_v_baseline_plot,
1192   ncol = 2)
1193 motor_resp_rois_plot
1194 ggsave(filename = 'Plots/motor_rois.png',
1195         plot = motor_resp_rois_plot,
1196         width = 8, height = 10.5, units = 'in')
1197
1198 # executive control rois
1199 exec_rois_plot <- grid.arrange(
1200   L_vlPFC_anti_corr_v_baseline_plot,
1201   R_vlPFC_anti_corr_v_baseline_plot,
1202   L_dlPFC_anti_corr_v_baseline_plot,
1203   R_dlPFC_anti_corr_v_baseline_plot,
1204   ncol = 2)
1205 exec_rois_plot
1206 ggsave(filename = 'Plots/exec_rois.png',
1207         plot = exec_rois_plot,
1208         width = 7, height = 6, units = 'in')
1209
1210 # dACC roi
1211 dACC_plot <- grid.arrange(
1212   dACC_anti_corr_v_baseline_plot,
1213   dACC_anti_errcorr_v_baseline_plot,
1214   ncol = 2)
1215 dACC_plot
1216 ggsave(filename = 'Plots/dACC.png',
1217         plot = dACC_plot,
1218         width = 8, height = 4, units = 'in')

```

```

1 #####
2 ### Bootstrap-enhanced elastic net ####
3 #####
4
5 setwd('/Users/jenniferefedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(caret)

```

```

8 library(dplyr)
9 library(ensr)
10 library(ggplot2)
11 library(glmnet)
12 library(gridExtra)
13 library(lubridate)
14 library(purrr)
15 library(tidyr)
16 library(xtable)
17
18 # response measure (whoqol score)
19 whoqol <- read.csv('Data/whoqol_ages_scores_20200204.csv',
20                     stringsAsFactors = FALSE) %>%
21   mutate(date = as.Date(date)) %>%
22   select(-c('X'))
23
24 # predictors (random effects from GAMM models)
25 ranefs <- read.csv('Data/random_effects_from_gamms_20200312.csv',
26                      stringsAsFactors = FALSE) %>%
27   select(-c(X))
28
29 # scale predictors and response variable
30 ranefs <- ranefs %>%
31   mutate_at(vars(matches('anti')), scale)
32 head(ranefs)
33
34 # join response and predictors
35 dat <- whoqol %>%
36   right_join(ranefs, by = 'lunaid') %>%
37   # z-score WHO-QOL domain scores
38   mutate(D_1_Raw_z = scale(D_1_Raw)) %>%
39   mutate(D_2_Raw_z = scale(D_2_Raw)) %>%
40   mutate(D_3_Raw_z = scale(D_3_Raw)) %>%
41   mutate(D_4_Raw_z = scale(D_4_Raw)) %>%
42   # for each subject, average their four z-scored domain scores
43   rowwise() %>%
44   mutate(whoqol_avg_score = mean(c(D_1_Raw_z, D_2_Raw_z, D_3_Raw_z, D_4_Raw_z)))
45
46 whoqol_avg_score_z <- scale(dat$whoqol_avg_score)[1:50]
47
48 dat <- cbind(dat, whoqol_avg_score_z) %>%
49   select(-(date:counter)) %>%
50   select(-(D_1_Raw_z:whoqol_avg_score))
51
52 write.csv(dat, 'Data/data_for_elastic_net_20200317.csv')
53
54 # function to plot histogram of whoqol scores
55 plot_whoqol <- function(x_var, x_axis_label, binw, centering) {
56   p <- ggplot(dat2, aes_string(x = x_var)) +
57     geom_histogram(color = 'black',
58                    fill = 'white',
59                    binwidth = binw,
60                    center = centering) +
61     labs(x = x_axis_label,
62          y = 'Frequency') +
63     theme_bw() +
64     theme(panel.border = element_blank(), panel.grid.major = element_blank(),
65           panel.grid.minor = element_blank(), axis.line = element_line(colour = 'black')) +

```



```

124 R_pPC_anti_corr_v_baseline_rint ,
125 R_pPC_anti_corr_v_baseline_rslope ,
126 R_dLPFC_anti_corr_v_baseline_rint ,
127 R_dLPFC_anti_corr_v_baseline_rslope ,
128 dACC_anti_errcorr_v_baseline_rint ,
129 dACC_anti_errcorr_v_baseline_rslope)))
130
131 ## fit the initial elastic net model
132 # sequence of alpha parameter values to try
133 # does not include 0 or 1 because those values correspond to ridge and lasso
134 alphas <- seq(from = 0.1, to = 0.9, length.out = 17)
135
136 # create k = 10 folds for cross-validation
137 # explicitly setting folds to allow for reproducibility
138 set.seed(100)
139 folds <- createFolds(y = y, k = 10, list = FALSE, returnTrain = FALSE)
140
141 # simultaneous tuning of alpha and lambda
142 fit <- ensr(x = X_sig,
143               y = y,
144               alphas = alphas,
145               foldid = folds,
146               standardize = FALSE)
147
148 # alpha and lambda values that minimized CV-MSE
149 summary(fit)[cvm == min(cvm)]
150 summary(fit)[cvm == min(cvm)]$alpha
151 summary(fit)[cvm == min(cvm)]$lambda
152
153 # fit the model using the alpha and lambda values that minimized CV-MSE
154 glmnetfit <- glmnet(x = X_sig,
155                       y = y,
156                       alpha = summary(fit)[cvm == min(cvm)]$alpha,
157                       lambda = summary(fit)[cvm == min(cvm)]$lambda,
158                       family = 'gaussian',
159                       standardize = FALSE)
160
161 coef(glmnetfit) # estimated beta coefficients
162 glmnetfit$dev.ratio # model R^2
163
164
165 # organize coefficients into dataframe
166 coefs_glmnet_fit <- as.data.frame(as.matrix(coef(glmnetfit)))
167 colnames(coefs_glmnet_fit) <- 'beta_hat'
168 coefs_glmnet_fit <- tibble::rownames_to_column(coefs_glmnet_fit, 'predictor')
169 coefs_glmnet_fit
170
171
172 ## use bootstrap-enhanced procedure to:
173 ## (1) derive confidence intervals and
174 ## (2) calculate variable inclusion probabilities
175 # number of bootstrap samples and subjects
176 B <- 5000
177 subjs <- 1:nrow(data_for_boot)
178
179 coefs_boot <- data.frame(Predictor = predictors, stringsAsFactors = FALSE)
180
181 # perform bootstrap procedure
182 for (i in 1:B){

```

```

183 # resample the data with replacement
184 set.seed(i*10)
185 boot_sample <- sample(subjs, size = 50, replace = TRUE)
186 y_boot <- y[boot_sample]
187 X_sig_boot <- X_sig[boot_sample, 1:ncol(X_sig)]
188
189 # fit the elastic net model on the bootstrap sample
190 glmnet_boot <- glmnet(x = X_sig_boot,
191                         y = y_boot,
192                         alpha = summary(fit)[cvm == min(cvm)]$alpha,
193                         lambda = summary(fit)[cvm == min(cvm)]$lambda,
194                         family = 'gaussian',
195                         standardize = FALSE)
196
197 # store the estimated coefficients
198 coefs <- as.data.frame(as.matrix(coef(glmnet_boot))) %>%
199   tibble::rownames_to_column(var = 'Predictor')
200 colnames(coefs_nocv)[2] <- paste0('Beta_boot_', i)
201
202 coefs_boot <- full_join(coefs_boot, coefs, by = 'Predictor')
203 }
204
205 coefs_boot
206
207 # transpose bootstrap coefficient dataframe from wide to long format
208 coefs_boot_long_colnames <- coefs_boot_nocv[, 1]
209 coefs_boot_long <- as.data.frame(t(coefs_boot[, -1]), stringsAsFactors = FALSE)
210 colnames(coefs_boot_long) <- coefs_boot_long_colnames
211 coefs_boot_long
212
213 # 95% bootstrapped CIs
214 coefs_quantiles <- as.data.frame(t(sapply(coefs_boot_long,
215                                         quantile,
216                                         probs = c(0.025, 0.975),
217                                         names = TRUE)))
218 coefs_quantiles <- tibble::rownames_to_column(coefs_quantiles, 'predictor')
219 coefs_quantiles
220
221 # variable inclusion probabilities
222 VIP <- as.data.frame(cbind(coefs_boot[, 1],
223                             rowSums(coefs_boot[, -1] != 0)),
224                           stringsAsFactors = FALSE)
225 colnames(VIP) <- c('predictor', 'times_selected')
226 VIP$times_selected <- as.numeric(VIP$times_selected)
227 VIP$prop_selected <- VIP$times_selected/B
228 VIP
229
230 # join the results
231 coefs_and_CI <- full_join(coefs_glmnet_fit, coefs_quantiles)
232 coefs_and_CI_and_VIP <- full_join(coefs_and_CI, VIP) %>%
233   select(-c('times_selected')) %>%
234   rename(VIP = prop_selected)
235 coefs_and_CI_and_VIP
236
237 # save the results
238 write.csv(coefs_and_CI_and_VIP, 'Data/bootstrap_enet_results_20200329.csv')
239
240
241 ## create LaTeX table to include in document

```

```

242 # clean up predictor names
243 coefs_and_CI_and_VIP$predictor <- gsub(pattern = '_',
244                                         replacement = ' ',
245                                         x = coefs_and_CI_and_VIP$predictor)
246 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'anti corr v baseline',
247                                         replacement = 'percent signal change correct
248                                         trials',
249                                         x = coefs_and_CI_and_VIP$predictor)
250 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'anti errcorr v baseline',
251                                         replacement = 'percent signal change error-
252                                         corrected trials',
253                                         x = coefs_and_CI_and_VIP$predictor)
254 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'rint',
255                                         replacement = 'intercept',
256                                         x = coefs_and_CI_and_VIP$predictor)
257 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'rslope',
258                                         replacement = 'slope',
259                                         x = coefs_and_CI_and_VIP$predictor)
260 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'anti perc',
261                                         replacement = 'Antisaccade proportion',
262                                         x = coefs_and_CI_and_VIP$predictor)
263 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'error corrected',
264                                         replacement = 'error-corrected trials',
265                                         x = coefs_and_CI_and_VIP$predictor)
266 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'nodrop',
267                                         replacement = '',
268                                         x = coefs_and_CI_and_VIP$predictor)
269 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'anti avg lat',
270                                         replacement = 'Antisaccade average latency',
271                                         x = coefs_and_CI_and_VIP$predictor)
272 coefs_and_CI_and_VIP$predictor <- gsub(pattern = ',',
273                                         replacement = ',',
274                                         x = coefs_and_CI_and_VIP$predictor)
275 coefs_and_CI_and_VIP$predictor <- gsub(pattern = 'Antisaccade proportion correct',
276                                         replacement = 'Antisaccade proportion correct
277                                         trials',
278                                         x = coefs_and_CI_and_VIP$predictor)
279
280 # save latex table code
281 print(xtable(coefs_and_CI_and_VIP,
282               digits = c(0,0,4,4,4,4),
283               type = 'latex'),
284               file = 'bootstrap_enet_20200329.tex')

```

```

1 #####
2 ### Post-hoc analysis for elastic net ####
3 #####
4
5 setwd('/Users/jenniferfedor/Documents/Biostats MS/Spring 2020/Thesis')
6
7 library(dplyr)
8 library(tidyr)
9 library(lubridate)
10 library(ggplot2)
11 library(gridExtra)

```

```

12 library(mgcv)
13
14 ## data for gamms
15 dat_gamm <- read.csv('Data/final_eye_scan_data_for_analysis_20200228.csv',
16   stringsAsFactors = FALSE)
17
18 dat_gamm <- dat_gamm %>%
19   select(-c('X')) %>%
20   mutate(lunaid = as.factor(lunaid)) %>%
21   mutate(date = as.Date(date, format = '%Y-%m-%d')) %>%
22   mutate(dob = as.Date(dob, format = '%Y-%m-%d')) %>%
23   mutate(sex = as.factor(sex))
24
25 dat_gamm <- within(dat_gamm,
26   level_edu_mother_cat <- relevel(as.factor(level_edu_mother_cat),
27     ref = 'Completed high school'))
28 dat_gamm$age_c <- dat$age - mean(dat$age)
29
30
31 ## whoqol data
32 dat_whoqol <- read.csv('Data/data_for_elastic_net_20200317.csv',
33   stringsAsFactors = FALSE)
34
35 # median whoqol composite score
36 med_score <- median(dat_whoqol$whoqol_avg_score_z)
37 # median split to create high and low QOL groups
38 dat_whoqol <- dat_whoqol %>%
39   select(c('lunaid', 'whoqol_avg_score_z')) %>%
40   mutate(whoqol_group = case_when(whoqol_avg_score_z < med_score ~ 'low',
41     whoqol_avg_score_z >= med_score ~ 'high'))
42
43 write.csv(dat_whoqol, 'Data/whoqol_group_labels_20200414.csv')
44
45
46 ## merge data for gamms and whoqol groups
47 dat_plus_whoqol_grp <- merge(dat_gamm, dat_whoqol)
48 levels(dat_plus_whoqol_grp$whoqol_group) <- c('High', 'Low')
49 # ordered factor
50 dat_plus_whoqol_grp$ordered_whoqol_group <- ordered(dat_plus_whoqol_grp$whoqol_group
51   ,
52   levels = c('High', 'Low'))
53
54 ## refit GAMMS for three predictors that had high VIPs with smooth age x QOL group
55   interaction and plot two group trajectories
56 # 1. Proportion error corrected antisaccade trials
57 model_formula <- as.formula("anti_perc_error_corrected_nodrop ~ ordered_whoqol_group
58   +
59   s(age_c, k = 10, fx = FALSE, bs = 'tp') +
60   s(age_c, by = ordered_whoqol_group, k = 10, fx = FALSE,
61     bs = 'tp')")
62
63 gamm_anti_perc_error_corrected_nodrop_bywhoqol <- gamm(as.formula(model_formula),
64   family = 'quasibinomial',
65   random = list(lunaid=~age_c),
66   data = dat_plus_whoqol_grp,
67   method = 'REML')
68
69 summary(gamm_anti_perc_error_corrected_nodrop_bywhoqol$gam)

```

```

67
68
69 # 2. L FEF activation
70 model_formula <- as.formula("L_FEF_anti_corr_v_baseline ~ ordered_whoqol_group +
71                         s(age_c, k = 10, fx = FALSE, bs = 'tp') +
72                         s(age_c, by = ordered_whoqol_group, k = 10, fx = FALSE,
73                           bs = 'tp')")
74
75 gamm_L_FEF_anti_corr_v_baseline_bywhoqol <- gamm(as.formula(model_formula),
76                                         random = list(lunaid=~age_c),
77                                         data = dat_plus_whoqol_grp,
78                                         method = 'REML')
79
80 summary(gamm_L_FEF_anti_corr_v_baseline_bywhoqol$gam)
81
82 # 3. R dlPFC activation
83 model_formula <- as.formula("R_dlPFC_anti_corr_v_baseline ~ ordered_whoqol_group +
84                         s(age_c, k = 10, fx = FALSE, bs = 'tp') +
85                         s(age_c, by = ordered_whoqol_group, k = 10, fx = FALSE,
86                           bs = 'tp')")
87
88 gamm_R_dlPFC_anti_corr_v_baseline_bywhoqol <- gamm(as.formula(model_formula),
89                                         random = list(lunaid=~age_c),
90                                         data = dat_plus_whoqol_grp,
91                                         method = 'REML')
92
93 summary(gamm_R_dlPFC_anti_corr_v_baseline_bywhoqol$gam)
94
95 ## function to plot groups' developmental trajectories
96 plot_gamm_int <- function(model, int_var, smooth_var, y_axis_label = ''){
97   mod <- get(model)$gam
98   s <- summary(mod)
99   df <- mod$model
100  df <- df %>% mutate(age = (age = age_c + mean(dat_gamm$age)))
101
102  response <- as.character(mod$terms[[2]])
103  labs <- levels(df[, int_var]) # groups for interaction term
104
105  # new data to predict for
106  np <- 10000 # number of values
107  newdat <- data.frame(matrix(data = NA, nrow = np, ncol = 0))
108  newdat[, smooth_var] <- seq(min(df[, smooth_var], na.rm = TRUE),
109                               max(df[, smooth_var], na.rm = TRUE),
110                               length.out = np)
111  newdat <- do.call('rbind',
112                      replicate(length(labs),
113                               newdat,
114                               simplify = FALSE))
115
116  newdat[, int_var] <- ordered(rep(labs, each = np), levels = labs)
117
118  # fitted values and standard errors for pointwise CIs
119  pred <- data.frame(predict.gam(mod,
120                       newdata = newdat,
121                       type = 'response',
122                       se.fit = TRUE))
123  pred <- cbind(newdat, pred)

```

```

124 pred[, response] <- 1
125 pred$age <- pred$age_c + mean(dat$age)
126 pred$ci_low <- pred$fit - 1.96*pred$se.fit
127 pred$ci_high <- pred$fit + 1.96*pred$se.fit
128
129 # plot trajectories for each group
130 p <- ggplot() +
131   geom_ribbon(data = pred,
132     aes_string(x = 'age',
133                 ymin = 'ci_low',
134                 ymax = 'ci_high',
135                 fill = int_var),
136     alpha = .5, linetype = 0) +
137   scale_fill_manual(values = c('#D1474B', '#476FD1')) +
138   geom_line(data = pred,
139     aes_string(x = 'age',
140                 y = 'fit',
141                 color = int_var), size = 1) +
142   scale_color_manual(values = c('#D1474B', '#476FD1')) +
143   labs(x = 'Age (years)',
144         y = y_axis_label,
145         fill = 'QOL group',
146         color = 'QOL group') +
147   theme_bw() +
148   theme(panel.border = element_blank(),
149         panel.grid.major = element_blank(),
150         #legend.position = 'none',
151         panel.grid.minor = element_blank(),
152         axis.line = element_line(colour = "black")) +
153   theme(axis.title = element_text(size = 11),
154         axis.text = element_text(size = 10),
155         legend.title=element_text(size = 11),
156         legend.text = element_text(size = 10))
157 return(p)
158 }
159
160
161 ## plots of group trajectories for the three models
162 p1 <- plot_gamm_int(model = 'gamm_anti_perc_error_corrected_nodrop_bywhoqol',
163                       int_var = 'ordered_whoqol_group',
164                       smooth_var = 'age_c',
165                       y_axis_label = 'Proportion error-corrected trials')
166 p1 <- p1 + theme(legend.position = 'none')
167 p1
168
169 p2 <- plot_gamm_int(model = 'gamm_L_FEF_anti_corr_v_baseline_bywhoqol',
170                       int_var = 'ordered_whoqol_group',
171                       smooth_var = 'age_c',
172                       y_axis_label = 'L FEF percent signal change')
173 p2 <- p2 + theme(legend.position = 'none')
174 p2
175
176 p3 <- plot_gamm_int(model = 'gamm_R_dlPFC_anti_corr_v_baseline_bywhoqol',
177                       int_var = 'ordered_whoqol_group',
178                       smooth_var = 'age_c',
179                       y_axis_label = 'R dlPFC percent signal change')
180 p3
181
182 # arrange plots

```

```
183 plots_by_whoqol_group <- gridExtra::grid.arrange(  
184   p1, p2, grid::nullGrob(), p3, grid::nullGrob(),  
185   layout_matrix = matrix(c(1,1,1,2,2,2,  
186     3,4,4,4,4,5),  
187     byrow = TRUE,  
188     ncol = 6))  
189 plots_by_whoqol_group  
190  
191 ggsave('Plots/plots_by_whoqol_group.png',  
192   plots_by_whoqol_group,  
193   width = 10, height = 8, units = 'in')
```

## Bibliography

1. Huttenlocher, P. R. & Dabholkar, A. S. Regional differences in synaptogenesis in human cerebral cortex. *The Journal of Comparative Neurology* **387**, 167–178. ISSN: 0021-9967 (Oct. 20, 1997).
2. Luna, B. *et al.* Maturation of widely distributed brain function subserves cognitive development. *NeuroImage* **13**, 786–793. ISSN: 1053-8119 (May 2001).
3. Luna, B. Developmental changes in cognitive control through adolescence. *Advances in Child Development and Behavior* **37**, 233–278. ISSN: 0065-2407 (2009).
4. Diamond, A. Executive Functions. *Annual review of psychology* **64**, 135–168. ISSN: 0066-4308 (2013).
5. Ochsner, K. & Gross, J. The cognitive control of emotion. *Trends in Cognitive Sciences* **9**, 242–249. ISSN: 13646613 (May 2005).
6. Williams, P. G., Tinajero, R. & Suchy, Y. *Executive Functioning and Health* (Oxford University Press, Nov. 6, 2017).
7. Luna, B., Paulsen, D. J., Padmanabhan, A. & Geier, C. The Teenage Brain: Cognitive Control and Motivation. *Current Directions in Psychological Science* **22**, 94–100. ISSN: 0963-7214 (Apr. 1, 2013).
8. Cunningham, R. M., Walton, M. A. & Carter, P. M. The Major Causes of Death in Children and Adolescents in the United States. *The New England journal of medicine* **379**, 2468–2475. ISSN: 0028-4793 (Dec. 20, 2018).
9. Centers for Disease Control and Prevention (CDC) National Center for Health Statistics. *Adolescent Health* <https://www.cdc.gov/nchs/fastats/adolescent-health.htm> (2020).
10. Arnsten, A. F. T. & Rubia, K. Neurobiological Circuits Regulating Attention, Cognitive Control, Motivation, and Emotion: Disruptions in Neurodevelopmental Psychiatric Disorders. *Journal of the American Academy of Child & Adolescent Psychiatry* **51**, 356–367. ISSN: 0890-8567 (Apr. 1, 2012).
11. World Health Organization (WHO). *Adolescent health epidemiology* WHO. [https://www.who.int/maternal\\_child\\_adolescent/epidemiology/adolescence/en/](https://www.who.int/maternal_child_adolescent/epidemiology/adolescence/en/) (2020).

12. Healthy People 2020. *Adolescent Health* U.S. Department of Health and Human Services, Office of Disease Prevention and Health Promotion. <https://www.healthypeople.gov/2020/topics-objectives/topic/Adolescent-Health> (2020).
13. Lenartowicz, A., Kalar, D. J., Congdon, E. & Poldrack, R. A. Towards an Ontology of Cognitive Control. *Topics in Cognitive Science* **2**, 678–692. ISSN: 1756-8765 (2010).
14. Miller, E. K. & Cohen, J. D. An integrative theory of prefrontal cortex function. *Annual Review of Neuroscience* **24**, 167–202. ISSN: 0147-006X (2001).
15. Miyake, A. *et al.* The unity and diversity of executive functions and their contributions to complex "Frontal Lobe" tasks: a latent variable analysis. *Cognitive Psychology* **41**, 49–100. ISSN: 0010-0285 (Aug. 2000).
16. Diamond, A. Developmental time course in human infants and infant monkeys, and the neural bases of, inhibitory control in reaching. *Annals of the New York Academy of Sciences* **608**, 637–669. ISSN: 0077-8923 (1990).
17. Diamond, A. & Goldman-Rakic, P. S. Comparison of human infants and rhesus monkeys on Piaget's AB task: evidence for dependence on dorsolateral prefrontal cortex. *Experimental Brain Research* **74**, 24–40. ISSN: 0014-4819 (1989).
18. Moilanen, K. L., Shaw, D. S., Dishion, T. J., Gardner, F. & Wilson, M. Predictors of Longitudinal Growth in Inhibitory Control in Early Childhood. *Social Development* **19**, 326–347. ISSN: 1467-9507 (2010).
19. Munoz, D. P., Broughton, J. R., Goldring, J. E. & Armstrong, I. T. Age-related performance of human subjects on saccadic eye movement tasks. *Experimental Brain Research* **121**, 391–400. ISSN: 0014-4819 (Aug. 1998).
20. Luna, B., Garver, K. E., Urban, T. A., Lazar, N. A. & Sweeney, J. A. Maturation of cognitive processes from late childhood to adulthood. *Child Development* **75**, 1357–1372. ISSN: 0009-3920 (Oct. 2004).
21. Hallett, P. E. Primary and secondary saccades to goals defined by instructions. *Vision Research* **18**, 1279–1296. ISSN: 0042-6989 (Jan. 1, 1978).
22. Ordaz, S. J., Foran, W., Velanova, K. & Luna, B. Longitudinal Growth Curves of Brain Function Underlying Inhibitory Control through Adolescence. *The Journal of Neuroscience* **33**, 18109–18124. ISSN: 0270-6474 (Nov. 13, 2013).
23. Glover, G. H. Overview of Functional Magnetic Resonance Imaging. *Neurosurgery clinics of North America* **22**, 133–139. ISSN: 1042-3680 (Apr. 2011).

24. Fuster, J. *The Prefrontal Cortex* 3rd ed. 461 pp. ISBN: 978-0-12-408060-7 (Raven Press, New York, 1997).
25. Gogtay, N. *et al.* Dynamic mapping of human cortical development during childhood through early adulthood. *Proceedings of the National Academy of Sciences of the United States of America* **101**, 8174–8179. ISSN: 0027-8424 (May 25, 2004).
26. Tamm, L., Menon, V. & Reiss, A. L. Maturation of brain function associated with response inhibition. *Journal of the American Academy of Child and Adolescent Psychiatry* **41**, 1231–1238. ISSN: 0890-8567 (Oct. 2002).
27. Rubia, K. *et al.* Functional frontalis with age: mapping neurodevelopmental trajectories with fMRI. *Neuroscience and Biobehavioral Reviews* **24**, 13–19. ISSN: 0149-7634 (Jan. 2000).
28. Casey, B. J. *et al.* A Developmental Functional MRI Study of Prefrontal Activation during Performance of a Go-No-Go Task. *Journal of Cognitive Neuroscience* **9**, 835–847. ISSN: 0898-929X (Nov. 1997).
29. Durston, S. *et al.* A neural basis for the development of inhibitory control. *Developmental Science* **5**, F9–F16. ISSN: 1467-7687. <https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-7687.00235> (2020) (2002).
30. Luna, B., Padmanabhan, A. & O’Hearn, K. What has fMRI told us about the Development of Cognitive Control through Adolescence? *Brain and cognition* **72**, 101. ISSN: 0278-2626 (Feb. 2010).
31. Luna, B., Marek, S., Larsen, B., Tervo-Clemmens, B. & Chahal, R. An Integrative Model of the Maturation of Cognitive Control. *Annual review of neuroscience* **38**, 151–170. ISSN: 0147-006X (July 8, 2015).
32. Ghuman, A. S., Bar, M., Dobbins, I. G. & Schnyer, D. M. The effects of priming on frontal-temporal communication. *Proceedings of the National Academy of Sciences of the United States of America* **105**, 8405–8409. ISSN: 1091-6490 (June 17, 2008).
33. Sweeney, J. A. *et al.* Positron emission tomography study of voluntary saccadic eye movements and spatial working memory. *Journal of Neurophysiology* **75**, 454–468. ISSN: 0022-3077 (Jan. 1996).
34. Velanova, K., Wheeler, M. E. & Luna, B. Maturational Changes in Anterior Cingulate and Frontoparietal Recruitment Support the Development of Error Processing and Inhibitory Control. *Cerebral Cortex* **18**, 2505–2522. ISSN: 1047-3211 (Nov. 2008).

35. Carter, C. S. *et al.* Anterior cingulate cortex, error detection, and the online monitoring of performance. *Science* **280**, 747–749. ISSN: 0036-8075 (May 1, 1998).
36. Hayes, R. D. & Reeve, B. B. in *Epidemiology and demography in public health* (eds Killewo, J., Heggenhougen, H. K. & Quah, S. R.) 195–205 (Academic Press, San Diego, 2010).
37. The Whoqol Group. The World Health Organization Quality of Life assessment (WHO-QOL): position paper from the World Health Organization. *Social Science & Medicine* **41**, 1403–1409. ISSN: 0277-9536 (Nov. 1995).
38. Cherepanov, D. *et al.* Gender Differences in Multiple Underlying Dimensions of Health-Related Quality of Life are Associated with Sociodemographic and Socioeconomic Status. *Medical care* **49**, 1021–1030. ISSN: 0025-7079 (Nov. 2011).
39. Meade, T. & Dowswell, E. Adolescents' health-related quality of life (HRQoL) changes over time: a three year longitudinal study. *Health and Quality of Life Outcomes* **14**. ISSN: 1477-7525 (Jan. 25, 2016).
40. Kim, J.-H. & Park, E.-C. Impact of socioeconomic status and subjective social class on overall and health-related quality of life. *BMC Public Health* **15**, 1–15. ISSN: 1471-2458 (Dec. 2015).
41. Wang, W.-C., Yao, G., Tsai, Y.-J., Wang, J.-D. & Hsieh, C.-L. Validating, improving reliability, and estimating correlation of the four subscales in the WHOQOL-BREF using multi-dimensional Rasch analysis. *Quality of Life Research: An International Journal of Quality of Life Aspects of Treatment, Care and Rehabilitation* **15**, 607–620. ISSN: 0962-9343 (May 2006).
42. Diener, E. & Ryan, K. Subjective Well-Being: A General Overview. *South African Journal of Psychology* **39**, 391–406. ISSN: 0081-2463 (Dec. 1, 2009).
43. Jefferson, A. L., Paul, R. H., Ozonoff, A. & Cohen, R. A. Evaluating elements of executive functioning as predictors of instrumental activities of daily living (IADLs). *Archives of clinical neuropsychology : the official journal of the National Academy of Neuropsychologists* **21**, 311–320. ISSN: 0887-6177 (May 2006).
44. Davis, J. C., Marra, C. A., Najafzadeh, M. & Liu-Ambrose, T. The independent contribution of executive functions to health related quality of life in older women. *BMC Geriatrics* **10**, 16. ISSN: 1471-2318 (Apr. 1, 2010).

45. Enticott, P. G., Ogloff, J. R. P. & Bradshaw, J. L. Associations between laboratory measures of executive inhibitory control and self-reported impulsivity. *Personality and Individual Differences* **41**, 285–294. ISSN: 0191-8869 (July 1, 2006).
46. Chamberlain, S. R. & Grant, J. E. Relationship between quality of life in young adults and impulsivity/compulsivity. *Psychiatry Research* **271**, 253–258. ISSN: 0165-1781 (Jan. 2019).
47. Nelder, J. A. & Wedderburn, R. W. M. Generalized Linear Models. *Journal of the Royal Statistical Society. Series A (General)* **135**, 370. ISSN: 00359238 (1972).
48. Fox, J. *Applied Regression Analysis and Generalized Linear Models* 817 pp. ISBN: 978-1-4833-1088-6 (SAGE Publications, Mar. 18, 2015).
49. *Regression methods in biostatistics: linear, logistic, survival, and repeated measures models* (ed Vittinghoff, E.) (Springer, New York, 2005). 340 pp. ISBN: 978-0-387-20275-4.
50. Hawkins, D. M. The Problem of Overfitting. *Journal of chemical information and computer sciences* **44**, 1–12. ISSN: 0095-2338 (Jan. 1, 2004).
51. Tjørve, K. M. C. & Tjørve, E. The use of Gompertz models in growth analyses, and new Gompertz-model approach: An addition to the Unified-Richards family. *PLOS ONE* **12**, e0178691. ISSN: 1932-6203 (June 5, 2017).
52. Hastie, T. & Tibshirani, R. Generalized Additive Models. *Statistical Science* **1**, 297–310. ISSN: 0883-4237, 2168-8745 (Aug. 1986).
53. Wood, S. N. *Generalized Additive Models: an introduction with R* (Chapman & Hall/CRC, New York, 2006).
54. Lin, X. & Zhang, D. Inference in Generalized Additive Mixed Models by Using Smoothing Splines. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **61**, 381–400. ISSN: 1369-7412 (1999).
55. Marra, G. & Radice, R. Penalised regression splines: theory and application to medical research. *Statistical Methods in Medical Research* **19**, 107–125. ISSN: 0962-2802 (Apr. 1, 2010).
56. Wood, S. N. Thin plate regression splines. *Journal of the Royal Statistical Society, Series B* **65**, 95–114 (2003).
57. Wood, S. *mgcv: Mixed GAM Computation Vehicle with Automatic Smoothness Estimation* R package version 1.8-28. 2019. <https://CRAN.R-project.org/package=mgcv>.
58. Hedeker, D. & Gibbons, R. D. *Longitudinal Data Analysis* 369 pp. ISBN: 978-0-470-03647-1 (John Wiley & Sons, Hoboken, NJ, May 12, 2006).

59. Chen, C. Generalized additive mixed models. *Communications in Statistics - Theory and Methods* **29**, 1257–1271. ISSN: 0361-0926 (Jan. 1, 2000).
60. Mullah, M. A. S., Hanley, J. A. & Benedetti, A. Modeling perinatal mortality in twins via generalized additive mixed models: a comparison of estimation approaches. *BMC Medical Research Methodology* **19**. ISSN: 1471-2288 (Nov. 15, 2019).
61. Sóskuthy, M. Generalised additive mixed models for dynamic analysis in linguistics: a practical introduction. *arXiv:1703.05339 [stat]*. <http://arxiv.org/abs/1703.05339> (Mar. 15, 2017).
62. Alexander-Bloch, A. F. *et al.* Abnormal Cortical Growth in Schizophrenia Targets Normative Modules of Synchronized Development. *Biological Psychiatry. Brain Development and Connectivity in Schizophrenia* **76**, 438–446. ISSN: 0006-3223 (Sept. 15, 2014).
63. Forde, N. J. *et al.* Healthy cortical development through adolescence and early adulthood. *Brain Structure & Function* **222**, 3653–3663. ISSN: 1863-2661 (Nov. 2017).
64. Tamnes, C. K. *et al.* Development of the Cerebral Cortex across Adolescence: A Multisample Study of Inter-Related Longitudinal Changes in Cortical Volume, Surface Area, and Thickness. *The Journal of Neuroscience* **37**, 3402–3412. ISSN: 0270-6474 (Mar. 22, 2017).
65. Hastie, T., Tibshirani, R. & Wainwright, M. *Statistical Learning with Sparsity: The Lasso and Generalizations* 354 pp. ISBN: 978-1-4987-1217-0 (CRC Press, May 7, 2015).
66. *An introduction to statistical learning: with applications in R* (eds James, G., Witten, D., Hastie, T. & Tibshirani, R.) *Springer texts in statistics* **103** (Springer, New York, 2013). 426 pp. ISBN: 978-1-4614-7137-0.
67. Mumford, J. A. & Nichols, T. Power Calculation for Group fMRI Studies Accounting for Arbitrary Design and Temporal Autocorrelation. *NeuroImage* **39**, 261–268. ISSN: 1053-8119 (Jan. 1, 2008).
68. Yeung, A. W. K. An Updated Survey on Statistical Thresholding and Sample Size of fMRI Studies. *Frontiers in Human Neuroscience* **12**. ISSN: 1662-5161 (2018).
69. Hoerl, A. E. & Kennard, R. W. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* **12**, 55–67. ISSN: 0040-1706. <https://amstat.tandfonline.com/doi/abs/10.1080/00401706.1970.10488634> (2020) (Feb. 1, 1970).
70. Tibshirani, R. Regression Shrinkage and Selection Via the Lasso. *Journal of the Royal Statistical Society: Series B (Methodological)* **58**, 267–288. ISSN: 2517-6161 (1996).

71. Zou, H. & Hastie, T. Regularization and variable selection via the Elastic Net. *Journal of the Royal Statistical Society, Series B* **67**, 301–320 (2005).
72. Waldmann, P., Mészáros, G., Gredler, B., Fürst, C. & Sölkner, J. Evaluation of the lasso and the elastic net in genome-wide association studies. *Frontiers in Genetics* **4**. ISSN: 1664-8021 (2013).
73. Ryali, S., Chen, T., Supekar, K. & Menon, V. Estimation of functional connectivity in fMRI data using stability selection-based sparse partial correlation with elastic net penalty. *NeuroImage* **59**, 3852–3861. ISSN: 1095-9572 (Feb. 15, 2012).
74. Weschler, D. *Weschler Adult Intelligence Scale - Revised* (The Psychological Corporation, San Antonio, TX, 1999).
75. Gitelman, D. R. ILAB: a program for postexperimental eye movement analysis. *Behavior Research Methods, Instruments, & Computers: A Journal of the Psychonomic Society, Inc* **34**, 605–612. ISSN: 0743-3808 (Nov. 2002).
76. Cox, R. W. AFNI: software for analysis and visualization of functional magnetic resonance neuroimages. *Computers and Biomedical Research* **29**, 162–173 (1996).
77. Yarkoni, T., Poldrack, R. A., Nichols, T. E., Van Essen, D. C. & Wager, T. D. Large-scale automated synthesis of human functional neuroimaging data. *Nature Methods* **8**, 665–670. ISSN: 1548-7091, 1548-7105 (Aug. 2011).
78. Cosgrove, K. P., Mazure, C. M. & Staley, J. K. Evolving Knowledge of Sex Differences in Brain Structure, Function, and Chemistry. *Biological Psychiatry* **62**, 847–855. ISSN: 0006-3223, 1873-2402 (Oct. 15, 2007).
79. Jackson, M., Kiernan, K. & McLanahan, S. Maternal Education, Changing Family Circumstances, and Children’s Skill Development in the United States and UK. *The Annals of the American Academy of Political and Social Science* **674**, 59–84. ISSN: 0002-7162 (Nov. 2017).
80. Harding, J. F., Morris, P. A. & Hughes, D. The Relationship Between Maternal Education and Children’s Academic Outcomes: A Theoretical Framework. *Journal of Marriage and Family* **77**, 60–76. ISSN: 1741-3737 (2015).
81. Sullivan, K. J., Shadish, W. R. & Steiner, P. M. An introduction to modeling longitudinal data with generalized additive models: Applications to single-case designs. *Psychological Methods* **20**, 26–42. ISSN: 1082-989X (2015).

82. Benjamini, Y. & Hochberg, Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)* **57**, 289–300. ISSN: 0035-9246 (1995).
83. Simpson, G. *gratia: Graceful 'ggplot'-Based Graphics and Other Functions for GAMs Fitted Using 'mgcv'* R package version 0.3.0. 2020. <https://CRAN.R-project.org/package=gratia>.
84. Lockhart, R., Taylor, J., Tibshirani, R. J. & Tibshirani, R. A significance test for the lasso. *The Annals of Statistics* **42**, 413–468. ISSN: 0090-5364 (Apr. 2014).
85. Bunea, F. *et al.* Penalized Least Squares Regression Methods and Applications to Neuroimaging. *NeuroImage* **55**, 1519–1527. ISSN: 1053-8119 (Apr. 15, 2011).
86. Abram, S. V. *et al.* Bootstrap Enhanced Penalized Regression for Variable Selection with Neuroimaging Data. *Frontiers in Neuroscience* **10**. ISSN: 1662-4548 (July 28, 2016).
87. DeWitt, P. *ensr: Elastic Net Searcher* R package version 0.1.0. 2019. <https://CRAN.R-project.org/package=ensr>.
88. Friedman, J. *et al.* *glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models* R package version 3.0-2. 2019. <https://CRAN.R-project.org/package=glmnet>.
89. Hwang, K., Velanova, K. & Luna, B. Strengthening of Top-Down Frontal Cognitive Control Networks Underlying the Development of Inhibitory Control: A Functional Magnetic Resonance Imaging Effective Connectivity Study. *The Journal of Neuroscience* **30**, 15535–15545. ISSN: 0270-6474 (Nov. 17, 2010).
90. Button, K. S. *et al.* Power failure: why small sample size undermines the reliability of neuroscience. *Nature Reviews Neuroscience* **14**, 365–376. ISSN: 1471-003X, 1471-0048 (May 2013).
91. Thomas, C. G., Harshman, R. A. & Menon, R. S. Noise reduction in BOLD-based fMRI using component analysis. *NeuroImage* **17**, 1521–1537. ISSN: 1053-8119 (Nov. 2002).