Predicting Body Composition Measurements in Samoan Adults with Multivariate Adaptive Regression Splines and Assessing the Effect of a Missense Variant in *CREBRF* on Body Composition

by

Gregory Procario

B.S. Biological Sciences, University of Pittsburgh, 2019

Submitted to the Graduate Faculty of the

Department of Biostatistics in the

Graduate School of Public Health in partial fulfillment

of the requirements for the degree of

Master of Science

University of Pittsburgh

2021

UNIVERSITY OF PITTSBURGH

GRADUATE SCHOOL OF PUBLIC HEALTH

This thesis was presented

by

Gregory Procario

It was defended on

April 26, 2021

and approved by

Jenna C. Carlson, PhD, Assistant Professor, Biostatistics Graduate School of Public Health, University of Pittsburgh
Ada Youk, PhD, Research Associate Professor, Biostatistics Graduate School of Public Health, University of Pittsburgh
Jeanine M. Buchanich, PhD, Research Associate Professor, Biostatistics Graduate School of Public Health, University of Pittsburgh
Ryan L. Minster, PhD, MSIS, Assistant Professor, Human Genetics Graduate School of Public Health, University of Pittsburgh
Thesis Advisor: Jenna C. Carlson, PhD, Assistant Professor, Biostatistics Graduate School of Public Health, University of Pittsburgh

Predicting Body Composition Measurements in Samoan Adults with Multivariate Adaptive Regression Splines and Assessing the Effect of a Missense Variant in *CREBRF* on Body Composition

Gregory Procario, MS

University of Pittsburgh, 2021

Abstract

Background: The minor A allele of rs373863828, a missense variant in *CREBRF* that is rare in most populations but common in Samoans, was found to have an association with higher BMI yet lower odds of type 2 diabetes. Identifying the physiological mechanisms behind *CREBRF* are of paramount importance in the pursuit of understanding obesity.

Methods: Multivariate Adaptive Regression Splines (MARS) models were developed to predict total fat, total lean, and visceral adipose tissue (VAT) mass in a sample of Samoan adults. These models were developed with a subset of the sample (n = 416) who had precise total fat, total lean, and VAT mass measurements; covariates included demographics (sex and age) along with anthropometric measurements (e.g., weight, hip circumference). Mass measurements were imputed from the MARS models for the larger sample (n = 1,970) who lacked total fat, total lean, and VAT mass measurements. These imputed values were applied as outcomes in genetic mixed models that estimated the relationship between rs373863828 and the mass measurements.

Results: MARS models were less optimal in terms of RMSE, R^2, and MAE in a majority of the cases, compared to alternative linear regression models. After imputing mass values in the larger sample of Samoans with the MARS models, the sex stratified genetic mixed models were fit. Males and females had higher estimated total fat, total lean, and VAT mass per copy of the A

allele (for males: +1,552.2g, +1,634.9g, and +115.7g, respectively, for females: +1,673.5g, +1,050.0g, and +70.7g, respectively).

Conclusion: The *CREBRF* variant rs373863828 was associated with higher average mass for all three measurements in Samoan adults, suggesting a broader role of *CREBRF* in body composition. These effects do not readily explain the paradoxical relationship of this variant with BMI and diabetes. However, our results seem to indicate potential differences by sex in the effects of *CREBRF* on total fat and total lean mass, that future researchers should investigate.

Public Health Significance: This thesis investigated genetic factors related to obesity in Samoan adults, the results of which may help researchers explicate mechanisms behind the disease.

Table of Contents

List of Tables

List of Figures

List of Equations

Preface

I would like to thank my committee members, Dr. Youk, Dr. Buchanich, and Dr. Carlson, for helping me through this process and for everything they have done to get me to this point. I would also like to thank my external committee member, Dr. Minster, who was gracious enough to lend his time and to allow me to work on this project. I would like to give a special thanks to Dr. Carlson, who has been a fantastic advisor; it was wonderful to work on this project with you. This was a long journey, and I am grateful for everything you all have done to help me get to the end.

I would also like to thank the Samoan participants and local village authorities and research assistants over the years. I acknowledge the support of our research collaboration with the Samoa Ministry of Health; the Samoa Bureau of Statistics; the Samoa Ministry of Women, Community and Social Development; and the American Samoa Department of Health. The Samoan Obesity, Lifestyle, and Genetics Adaptations (OLaGA) Study Group investigators are Ranjan Deka, Jenna C. Carlson, Kima Fa‘asalele-Savusa, Nicola L. Hawley, Vaimoana Lupematisila, Stephen T. McGarvey, Ryan L. Minster, Leausa Toleafoa Take Naseri, Muagututi‘a Sefuiva Reupena, Melania Selu, John Tuitele, Asiata Satupa‘itea Viali and Daniel E. Weeks.

1.0 Introduction

1.1 Samoa and Obesity

The Samoan archipelago, part of the Polynesian region of the Pacific Ocean, comprises both the Independent State of Samoa ("Samoa") and the U.S. territory of American Samoa. Samoa, like many of the Pacific Islands communities, had a drastic increase in the prevalence of obesity as the population shifted away from their traditional diets (WHO, 2010). Between 1978 and 2013, obesity (BMI > 30kg/m^2) prevalence in Samoan adults aged 25-64 increased from 27.7% to 53.1% in men, and 44.4% to 76.7% in women (Lin et al., 2017). When using the Polynesian specific[1] BMI cutoff for obesity of 32 kg/m^2, the prevalence is 41.2% and 65.1% respectively. For comparison, in 2013-14 the prevalence of obesity in U.S. adults (20 and over) was 35.0% for men and 40.4% for women (Cheryl D. Fryar, 2016).

The obesity epidemic in Samoa places a strain on their public health system. People with obesity are at a higher risk of developing noncommunicable diseases (NCD: e.g., cancer, diabetes, cardiovascular disease), and NCDs are the main cause of premature mortality and morbidity in Samoa; the financial costs of treating end-stage NCDs within Samoa will continue to burden the health care system and economy as a whole (*Samoa-WHO: Country Cooperation Strategy 2018-2022*, 2017).

---

[1] The World Health Organization's standard guideline is any BMI > 30 kg/m^2 is classified as obese. Swinburn, Ley, Carmichael, and Plank (1999) found that Polynesians' higher muscle to fat ratio necessitates an obesity cutoff of BMI > 32 kg/m^2.

1.2 Body Mass Composition

Classifying obesity with BMI is nearly universal, in part due to the convenience of measuring height and weight. However, classification using BMI thresholds fails to identify those who have excess body fat who are below the 30 kg/m^2 threshold (Frankenfield, Rowe, Cooney, Smith, & Becker, 2001). Because obesity is defined as having excess body fat, measuring the components of body mass (total fat, lean, bone mass, etc.) allows for a more precise classification of obesity. An appealing approach to measuring body mass composition is with dual energy x-ray absorptiometry (DXA or DEXA). DXA scans can measure a whole body and estimate its components with high precision (Albanese, Diessel, & Genant, 2003). Results from a DXA scan that are clinically relevant to obesity include total fat mass, total lean mass, and visceral adipose tissue (VAT) mass.

1.2.1 Health Implications

Total lean mass is the component of overall body mass that encompasses the non-bone and non-fat mass (water, skin, muscle, etc.). Greater lean mass plays a role in maintaining bone density and improving metabolic health (Fielding et al., 2011), and greater lean mass has been associated with better cardiovascular health (O'Donovan et al., 2005). Because BMI directly increases as total mass increases regardless of whether that additional mass is lean or fat, using the metric as a cutoff for obesity can fail to account for the benefits of non-fat mass components (Romero-Corral, Lopez-Jimenez, Sierra-Johnson, & Somers, 2008). In contrast to the benefits of higher lean mass are the consequences of higher fat mass; increased total fat mass defines obesity and is associated with NCDs like diabetes and coronary heart disease (Kopelman, 2000). While total fat mass can

adequately predict negative health outcomes, research has shown that certain fat depots may be more influential on health than others (Ibrahim, 2010).

VAT is a distinct form of fat that is located in the abdominal region, surrounding vital organs. VAT is more hormonally active than subcutaneous fat (SCAT; non-visceral fat located below the skin). The expression of adiponectin is higher in VAT than SCAT (Ibrahim, 2010), and plasma adiponectin levels are negatively associated with BMI and insulin resistance (Motoshima et al., 2002; Prakash, Mittal, Awasthi, Agarwal, & Srivastava, 2013). The metabolic activity of VAT mass is demonstrated through positive associations with insulin resistance, impaired glucose, and impaired lipid metabolism (Ritchie & Connell, 2007; Shuster, Patlas, Pinthus, & Mourtzakis, 2012). These metabolic abnormalities, insulin resistance and impaired glucose metabolism, are a part of "metabolic syndrome" which may lead to adverse health outcomes like type 2 diabetes and cardiovascular disease (Lebovitz, 2001).

1.2.2 Predictive Models

Given the associative links between fat, lean, and VAT mass and disease risk, researchers have sought to develop accurate prediction models for body composition when compartmentalized mass values are unknown. Recently, Cichosz, Rasmussen, Vestergaard, and Hejlesen (2020) developed a neural network model based on anthropometric and demographic data, that could accurately predict total lean and fat mass. However, this neural network model, and other comparable predictive models, were developed from samples with predominantly European ancestry (Lee et al., 2017). The applicability of these models to a population of Polynesian ancestry is questionable, considering Polynesians tend to have higher lean mass and less fat mass

than Europeans at any given BMI (Swinburn, Craig, Daniel, Dent, & Strauss, 1996). Swinburn et al. (1999) established Polynesian specific total fat mass prediction models, including sex-stratified models that solely use BMI to predict total fat mass and a model with height, weight, sex, and age as covariates to predict total fat mass.

1.2.3 Associations with *CREBRF*

Minster et al. (2016) identified a missense variant (rs373863828) in *CREBRF* having a strong association with BMI in Samoans. This variant is prevalent in Samoans (approximately 50% are carriers)(Krishnan et al., 2018) but not in Europeans, Africans, or East Asians (fewer than 0.008% are carriers). Notably, the minor A allele of the *CREBRF* variant rs373863828 is paradoxically associated with lower odds of type 2 diabetes, and higher BMI and obesity risk (Minster et al., 2016). While this contradictory relationship has been replicated in studies (Hanson et al., 2019), none have been successful in identifying the physiological mechanisms that explain this phenomenon. Arslanian et al. (2021) found that rs373863828 is associated with fat-free mass (lean and bone mass) in Samoan infants. Furthermore, Carlson et al. (2020) discovered an association of the minor A allele of the *CREBRF* rs373863828 and greater height in Samoans, utilizing a genetic mixed model.

The associations between the minor A allele of the *CREBRF* rs373863828 and greater fat-free mass, height, and BMI, suggest that this variant may play a role in body composition or development more broadly. This thesis will examine the relationship between the minor A allele of the *CREBRF* rs373863828 and three components of body mass (total fat, lean, and VAT) through a genetic mixed effects model. The mixed effects models were adjusted for distant genetic

relation with three principal components of ancestry, and recent genetic relations with a random subject effect, using an empirical kinship matrix (Minster et al., 2016).

1.3 Objective

The first objective of this thesis is to develop MARS models that predict total fat, lean, and VAT mass in a small cohort of Samoans, and assess their performance against comparable models. These models will use anthropometric measurements and demographic data and will be validated against DXA scan results. The second objective of this thesis is to impute the mass values in a larger cohort of Samoans, and to assess the relationship between these values and the minor A allele of the *CREBRF* rs373863828 with a genetic mixed model.

2.0 Methods

2.1 Data

The data sets of interest originated from a 2010 genome-wide association study (GWAS) of the population of the Independent State of Samoa, performed by Hawley et al. (2014), and from a 2017-19 follow up study on a subset of the original sample (Hawley et al., 2020). These studies were approved by the Health Research Committee of the Samoan Ministry of Health, the American Samoan Department of Health Institutional Review Board (IRB; for Samoa and American Samoa studies) and the Brown University IRB. Participants in these studies gave written informed consent in Samoan. The 2010 sample of the Samoan population was surveyed, genotyped, and anthropometrically measured for the variables outlined in section 2.1.1. The participants in the 2017-19 follow up study were recruited for approximately equal sex distribution, and for the number of copies of the minor allele (A) of a missense variant in *CREBRF*, rs373863828; the sampled genotype ratio was approximately 1:2:2 for AA:AG:GG[2]. These individuals were remeasured for the variables outlined in the prior study, and scanned using DXA to determine total fat, lean, and VAT mass measures (Hawley et al., 2020).

The 2017-19 sample served as the basis sample for predictive models that estimated total fat, lean, and VAT mass, using anthropometric and demographic data as covariates. The resulting models were applied to the 2010 sample (excluding overlapping participants from the 2017-19

[2] Sampling with this ratio gave additional power to detect differences across genotype, while accounting for the rarity of the AA genotype in the population (Hawley et al., 2020).

study), to obtain imputed mass measurements. The effect of rs373863828 on these imputed mass measures for these data was then estimated using a genetic mixed effects model, which utilized three principal components of ancestry and a genetic kinship matrix as previously described by Minster et al. (2016);(Carlson et al., 2020).

2.1.1 Variables of Interest

The primary outcomes of interest for the predictive models were total fat, total lean, and VAT mass, measured in grams via DXA scans. Participants that met criteria (e.g., non-pregnant, no recent X-ray exposure) were scanned by one of four DXA-trained professionals. Individuals whose body exceeded the size of the machine were measured one half at a time. These individuals had their total body composition estimated from the scans of one half of their body (Hawley et al., 2020).

The covariates evaluated for the predictive models were sex, age, height, weight, abdomen circumference, hip circumference, calf circumference, triceps skinfold, forearm skinfold, subscapular skinfold, suprailiac skinfold, and abdomen skinfold. Age and sex were obtained from a questionnaire administered in Samoan; assumed sex was consistent with subsequent genotyping (evidence of two X chromosomes were female, and one X and one Y chromosome were male). Height was measured in centimeters, to the nearest 0.1 cm, using a portable anthropometer. Weight was measured to the nearest 0.1 kg, using a digital weight scale. All anthropometric circumferences were measured to the nearest 0.1 cm, with a tape measure. All anthropometric skinfolds were obtained with skin calipers and measured to the nearest 0.1mm. Participants with skinfold measurements that exceeded the size of the calipers (>67.0 mm) were marked as such in the data. The circumferences and skinfolds were measured twice and averaged for use in analyses

(Hawley et al., 2014). The 2017-19 cohort had their weight and height measurements taken twice and averaged; once during the at home visit, and once at the laboratory visit (Hawley et al., 2020).

The genetic mixed effects model used the genotype of the *CREBRF* variant rs373863828 as a covariate. Genotype was measured via DNA extracted from blood samples, and processed on a genotyping array (Minster et al., 2016) and was coded using an additive genetic model (0 = GG, 1 = AG, 2 = AA). The mixed models were adjusted for genetic relationship with principal components of ancestry and a kinship matrix. The principal components and the kinship matrix were calculated from the SNP array data using PC-Relate (Gogarten et al., 2019) and KING software (Manichaikul et al., 2010), respectively.

2.1.2 Data Cleaning

After the 2010 and 2017-19 samples were loaded into RStudio, cleaning was performed to format the data for model building and analysis. Cleaning took place over two major steps: merging and cleaning 2017-19 cohort data, and cleaning 2010 cohort data. These cleaning steps are outlined below.

Merging and Cleaning 2017-19 Cohort Data

The 2017-19 cohort was loaded in with two separate data sets: DXA scan data, and anthropometric and demographic data. The DXA scan data set had multiple observations per individual, and some missingness in mass measurements for individuals. The multiple observations were a result of individuals whose size necessitated more than one scan to capture their whole body. These participants were scanned one half of their body at a time, with the second scan being a complete scan by mirroring the measures from the first. Subsetting the DXA data on

"Scantouse=1" removed all the non-unique observations with only half of their DXA scan. The resulting dataset (n = 432) was further subsetted for a "Corescan=1", which removed observations with missingness in the outcome variables. These individuals were unable to have their total fat, lean, and VAT mass measured, despite being scanned. This step reduced the dataset to size n = 421.

All unnecessary variables in the reduced DXA data set were dropped, leaving only "IDNumber" and the three outcomes for our predictive models (total fat, total lean, and VAT mass). Nonessential variables were dropped from the demographic and anthropometric dataset, excluding "IDNumber" and the predictor variables outlined in section 2.1.1. The anthropometric and demographic data (n = 519) contained more observations than the DXA data. The additional observations were 14 people who did not attend the laboratory visit for scans and 73 who were unable to receive X-ray scans (e.g., pregnant women, recent radiation exposure). Thus, when merging the DXA data with the demographic and anthropometric data, the sample size was reduced to only the number of participants who were able to receive their DXA scans with core measurements (total fat, lean, VAT mass). Merging these datasets was performed on "IDNumber", and the new dataset was called "body_dxa". One observation failed to merge as there was no matching ID in the demographic and anthropometric data set (n = 420).

The "body_dxa" data set had missingness in the circumference and skinfold measurements coded as -7777. These values were changed to NA for ease of coding. Furthermore, the skinfold measurements of subscapular, suprailiac, and abdomen had values > 67.0 mm. The presence of the greater than symbol automatically converted the measurements to categorical. These variables were converted back to continuous, or numeric, and those values were replaced with exactly 67.0 mm (n > 67.0mm: Sub. = 4, Sup. = 1, Abd. = 3). Converting these measures to 67.0 mm does

mean our models will slightly underestimate some of the higher predictions, however, this is preferred to removal of these data. The "body_dxa" data set had sex coded as (0 = Male, 1 = Female), which was recoded to just display the labels.

The last cleaning step was to check for missingness and remove those individuals prior to the analysis. The participants removed had at least one of the following measurements recorded as "NA": abdomen circumference, calf circumference, subscapular skinfold, abdomen skinfold, suprailiac skinfold, or VAT mass. The final sample size was n = 416. Prior to the predictive model building, the data were split into male and female subsets. A seed was set, and a random 75/25 training/testing split was created for each sex. The test and training data were only used for ridge regression models. All other predictive models were developed with bootstrapped samples of the complete data for each sex. The ridge regression models were not built using bootstrap samples due to computational limitations.

Cleaning 2010 Cohort Data

The 2010 cohort data was loaded as one data set, because DXA information was not collected on this group in the preliminary study (n = 3,102). The identification number for these data used a different nomenclature compared to the 2017-19 data set. An identification number from 2017-19 of "23" would be coded as "SG0023" in the 2010 data set. The characters "SG" and all leading zeros were removed from the identification numbers in the 2010 data set. The individuals that participated in the follow up study were removed from the 2010 data, based on matching identification. This removal was performed as the 2017-19 cohort have known mass values from their DXA scans, and they were used to develop the predictive models (n = 2,686). The 2010 data set had slightly different names for variables compared to the 2017-19 data set.

These variables were renamed to match the 2017-19 data set, as the models required the same variables names to be able to predict mass measurements. The final cleaning step for the 2010 data involved removing observations with missing values in at least one of the predictors, as they would not provide mass predictions from the models (n = 716). The cleaned 2010 cohort data contained 1,970 observations.

2.2 Statistical Analyses

The primary motivation for this research was to develop a nonparametric multivariate adaptive regression spline (MARS) model to accurately predict total fat, lean, and VAT mass in individuals from the 2010 Samoan cohort. The models described by Swinburn et al. (1999) estimated total fat mass in the Samoan population with ordinary least squares (OLS) regression. These models employed terms like BMI, or its components height and weight, which are convenient to measure. A more accurate model could be developed by including the additional terms from the anthropometric records in the 2010 and the 2017-19 follow up studies on the Samoan population (e.g., hip circumference, triceps skinfold). MARS, OLS, and alternatives like ridge regression all present valid approaches to building a predictive model with these data.

Prior to building the predictive models described above, the independent and dependent variables were assessed in various preliminary analyses. First, correlation between predictors was assessed via Pearson correlation matrix, to evaluate any potential collinearities. Subsequently, the Pearson correlation between each outcome and each predictor, by sex, was calculated to appraise their individual relationships. Given most variables were statistically different between sexes

(Table 1), sex stratification was applied to many analyses. Next, the distributions, mean, and standard deviation of total fat, lean, and VAT mass were graphed by sex.

After these analyses, the data were used, according to the methods described in section 2.2.1, to develop each respective predictive model for the three mass outcomes (i.e., total fat, lean, VAT mass) for each sex. The predictive accuracy of the selected models from each technique were evaluated by calculating the adjusted R^2, root mean squared error (RMSE), and the mean absolute error (MAE). Popular model selection metrics like Akaike information criterion (AIC) or Bayesian information criterion (BIC) were omitted due to their reliance on likelihood, which does not apply to the nonparametric MARS technique.

2.2.1 Mass Prediction Models

MARS

Multivariate adaptive regression splines (MARS), as described by Friedman, uses aspects of recursive partitioning and additive modeling to create a continuous nonlinear function that identifies variable interactions (Jerome H. Friedman, 1991). The goal of a MARS is to approximate the function $f(X)$ that describes the relationship between Y and X, in $Y = f(X)$. MARS estimates $f(X)$ as a qth degree polynomial spline function $\hat{f}_q(x)$, where the range of x values are divided into K+1 regions, split by K "knot" points (J. H. Friedman & Roosen, 1995).

A MARS model (Equation 1) is comprised of an intercept (β_0), coefficients (β_m), and basis functions ($h_m(X)$). A basis, or hinge function[3], takes the form of $(x-t)_+$and $(t-x)_+$, which are defined as max $(0, x-t)$ and max$(0, t-x)$, respectively, at a knot point of t. All observed values for all X_j are considered for a knot point of a basis function; the set of possible basis functions, C, is described in Equation 2, where N is sample size and p is the number of predictors. An $h_m(X)$ can be a single term from set C or a product of multiple terms from set C that are already included in the model. The product of multiple hinge functions is how the model incorporates interactions and polynomials terms. The β_m for a given $h_m(X)$ is estimated by minimizing the residual sum of squares (Hastie, Tibshirani, & Friedman, 2009). M is the number of terms, which is commonly constrained by the researcher.

$$\boldsymbol{f(X) = \beta_0 + \sum_{m=1}^{M} \beta_m h_m(X)}$$

Equation 1 MARS Function

$$C = [(X_j - t)_+, (t - X_j)_+]_{\substack{t \in (x_{1j}, x_{2j}, \ldots, x_{Nj}) \\ j=1,2,\ldots,p}}$$

Equation 2 Set of Possible Basis Functions

The model building process for MARS follows a forward and backward stepwise selection. The forward phase aims to overfit the model by including many basis functions. During the forward phase, the basis functions from set C that achieved the largest decrease in sum of squares residual error are added to the model iteratively, along with their corresponding coefficient

[3] The hinge function name is indicative of the joint-like nature of a basis function (symmetric lines emanating from a knot point); the hinge or basis function terms can be used interchangeably.

estimates (Boehmke & Greenwell, 2020). These additions continue until a constraint is reached, usually the number of model terms M. In addition to the term limit, the researcher can constrain the highest order q to reduce the size of the possible set of basis function. MARS programs also have built in conditions to terminate the forward phase if it becomes computationally inefficient to continue (e.g., attained R^2 of 0.999, or new terms change $R^2 < 0.001$).

Once the forward phase is terminated, the backward phase begins iteratively removing terms whose absence yields the lowest increase in residual squared error. While the forward phase adds basis functions in pairs at a knot point (e.g., $2.9 * \max(0, x - 5) + 1.7 * \max(0, 5 - x)$) the backwards phase can eliminate just one side of the pair. Each removal yields an estimated model $\widehat{f_\lambda}$ with λ number of terms. The removal of terms ends when the intercept is the only term remaining. The model subset from the backward phase that minimizes generalized cross-validation (GCV) is selected as the best model (Hastie et al., 2009). GCV (Equation 3) evaluates the fit of the $\widehat{f_\lambda}$ model on training data while penalizing model complexity (J. H. Friedman & Roosen, 1995). The effective number of terms, $M(\lambda)$, is a function of the linearly independent basis functions in the model (r), and the K knots penalized by a term c (typically, c = 3)[4].

$$GCV(\lambda) = \frac{\sum_{i=1}^{N}\left(y_i - \widehat{f_\lambda}(x_i)\right)^2}{\left(1 - \frac{M(\lambda)}{N}\right)^2}, M(\lambda) = r + cK$$

Equation 3 Generalized Cross-Validation Formula

[4] Mathematical simulations have found it optimal to penalize the model with three parameters for a knot point. This is also commonly adjusted down to a penalty of two to make the process more additive (Hastie et al., 2009).

The R packages “earth” and “caret” were used to automate the MARS model building for this thesis (Kuhn, 2020; Milborrow, 2020). The “train” function was used to build each MARS model for the three outcomes for both sexes. For each sex, this function was given the outcome of interest and the covariates. The method specified for this function was “bagEarth”, which incorporates bootstrap aggregation into the model building process. The function was set for B=50, which means the MARS model was fit 50 times, with each iteration using a bootstrap sample of the original data. The bootstrap samples were of equal size to the original data and were drawn randomly with replacement from the original data. The coefficients and model statistics are averaged from these 50 iterations. The training control for tuning the hyperparameters was a 10-fold cross-validation procedure. Finally, a matrix was created to specify combinations for the hyperparameters, M term limit and q order. The 30x2 matrix was created to match all possible degrees from 1-3, with the sequence of potential term limits that spanned 2-100 with a desired length of 10.

The “bagEarth” method begins by setting a training data set from the original data. The forward phase begins with the creation of MARS models with 10 fold cross-validation to tune the optimal degree and term limit, using the training set. The optimal combination of degree and term limit was chosen by RMSE by default. With the best model chosen from cross validation, 50 bootstrap samples of the training set are used to prune the terms and fit the model. The pruning and fitting follow the backward phase procedure, where the lowest GCV specifies the best subset of terms. The final model from this phase is fit using least squares regression to estimate the coefficients and relevant statistics (RMSE, MAE, R^2). The pruning and fitting are repeated for the 50 bootstrap samples, with the coefficient estimates and statistics averaged from these trials.

Linear (OLS) Regression

Linear regression was also used to approximate the function $f(X)$ that describes the relationship between Y and X, in the equation $Y = f(X)$. The form of a linear model is shown in Equation 4. To fit this linear model to a sample of size N, coefficient estimates are calculated by minimizing the residual sum of squares, shown in Equation 5 (Hastie et al., 2009). Unlike MARS, linear regression does not inherently select variables. When there is no knowledge of which variables to include in a predictive linear model, a valid variable selection technique must be applied prior to model fitting.

$$f(x) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j$$

Equation 4 Linear Regression Model

$$RSS(\beta) = \sum_{i=1}^{N} (y_i - x^T{}_i \beta)^2$$

Equation 5 Residual Sum of Squares

Swinburn et al. (1999) had previously developed models that predicted total fat mass in Polynesians, thus we had prior knowledge of covariates to include. However, these models only incorporated BMI, height, weight, sex, and age. We have access to various anthropometric data (e.g., hip circumference, triceps skinfold) in addition to the variables utilized by Swinburn. Thus, our linear models had two motivations for development; (1) to replicate the relationships seen in other research (Swinburn et al., 1999), and (2) to build models with comparable covariates to the MARS models.

Prior to building our own models, it was crucial that we establish some comparisons based on previous research in a population similar to Samoa. Swinburn et al. (1999) described two sex stratified models and one model with sex as a predictor, that all estimated total fat mass in

Polynesians. The sex stratified models, called Swinburn BMI in this thesis, utilized BMI as the sole predictor. The model without sex stratification, called Swinburn HWSA in this thesis, included height, weight, sex, and age (HWSA) as predictors.

In order to make linear models that were comparable to the MARS models, a backward stepwise variable selection was used to choose predictor variables. These models were considered comparable in the sense that they incorporated automated variable selection with the same set of predictors as MARS. The model building process was performed in R via the "step" function. The step function begins by creating a full model, and proceeds to drop one term per step. If the term dropped in that step reduces AIC, then it is permanently removed from the model and the process repeats on the smaller model. This continues until deleting any term from the model does not lower AIC. This process was repeated for all three outcomes and for both sexes. The resulting model contains the selected predictor variables.

The model fit statistics and coefficients for the Swinburn BMI, Swinburn HWSA, and the stepwise selected models were estimated with bootstrap resampling. In R, a function was created to fit each model and then capture coefficient estimates, RMSE, MAE, and adjusted R^2. From there, a seed was set, and the "boot" function (boot package; (Canty & Ripley, 2021)) was used to resample the data, record the captured statistics, and repeat 1000 times. The bootstrap averaged coefficient estimates, RMSE, MAE, and adjusted R^2 were recorded for each proposed model: the three Swinburn models, and the six models spanning each mass measurement for both sexes.

Ridge Regression

Ridge regression is a modeling technique for data that are affected by multicollinearity. This is of particular interest as many predictors in this research are body measurements which are

generally proportional (e.g., hip and abdomen circumference). Ridge regression coefficient estimation, like linear regression, is based on minimizing residual sum of squares. However, ridge regression penalizes coefficient size during estimation (Hastie et al., 2009). The ridge regression coefficient estimation, shown in Equation 6, can be penalized and shrunk towards zero by the parameter λ.

$$\hat{\beta} = \min \left[\sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right]$$

Equation 6 Ridge Regression Coefficient Estimation

The R package "glmnet" (J. Friedman, Hastie, & Tibshirani, 2010) was used to fit ridge regression models that predict the three mass outcomes for both sexes. The function "cv.glmnet" was used to determine optimal lambdas for each model of the three mass outcomes for both sexes. This function was given the set of predictor variables along with the desired outcome, from the training data sets. The model was specified as a ridge regression by setting "alpha=0", rather than "alpha=1" for lasso regression. The lambdas evaluated were the vector of 10^a where a was the sequence from 3 to -2 by -0.1, which gives a wide range for the potential optimal lambda. With these inputs, "cv.glmnet" runs a 10-fold cross-validation procedure for the specified model. The optimal lambda was obtained from the output of this function; the optimal lambda is defined as the lambda value associated with the lowest mean cross-validated error. The model fit was also stored from the output. The "predict" function was then used to calculate the predicted outcomes by inputting the fitted model saved in the last step, as well as the optimal lambda and the testing data set. These predicted output values and the true measured values were used to calculate adjusted R^2, RMSE, and MAE for the model.

2.2.2 Mass Imputation

Because predicting mass via MARS models was the primary goal of this thesis, the best MARS model was used to impute all three outcomes in both sexes for the 2010 cohort. None of the other models (linear or ridge regression) were used to impute mass measurements, despite potentially outperforming the prediction accuracy of a MARS model on the 2017-19 cohort. The "predict" function in R was used to predict total fat, lean, and VAT mass. This function takes the resulting MARS model for a given outcome and sex, along with the 2010 cohort data, and computes the expected outcome for each observation. The predicted total fat, lean, and VAT mass values were stored in new columns in the 2010 cohort data set. Summary statistics (min, max, mean, median, Q1, and Q3) for the distribution of the predicted outcomes were calculated for each sex. The distributions of the predicted outcomes were also graphed by sex, along with the mean and two standard deviation bars.

After the graphs and summary statistics were created, further data cleaning was required. The graphs and summary statistics showed that both males and females of the 2010 cohort had several negative VAT mass estimations. To resolve this issue, the estimated VAT mass values for both sexes in the 2010 cohort were truncated to the minimum values from each sex in the 2017-19 cohort.

During this stage of the analysis, the principal component of ancestry data and kinship matrix were loaded into the project. The kinship matrix and principal components utilized the "SG0000"convention for identification number. The same process of "SG" and leading zero removal was applied to the identification numbers of these data. The first three principal components (PC1, PC2, PC3) were subset and merged into the 2010 cohort data by identification

number. The resulting data set contained all necessary covariates for the genetic mixed effects model; kinship matrix is referenced independently from data.

2.2.3 Genetic Mixed Effects Model

The imputed mass measurements of the 2010 Samoan cohort, in isolation, do not provide much information towards understanding the role of genetics in body mass composition of the Samoan population. However, the relationship between these imputed mass measurements and the *CREBRF* minor A allele could help elucidate the mechanisms behind the association of this allele and BMI (Minster et al., 2016). While simple linear regression models could be used to measure the relationships between the mass measurements and *CREBRF* A allele, they would fail the assumption of independence between participants. This is due to the genetic relation, whether distant or recent, of the population being sampled. Standard practice in GWAS is to account for the distant relatedness of participants using principal component analysis (PCA). Furthermore, it is common to address the recent relatedness of participants using a kinship matrix as a random subject effect in a linear mixed model (Hoffman, 2013). The proposed genetic mixed effect models for this thesis follow a similar design to the models in the research performed by Carlson et al. (2020), studying the 2017-19 Samoan cohort.

The proposed model, shown in Equation 7, where Y is one of total fat, lean, or VAT mass. These models, like the MARS models, were stratified by sex. Age and Age^2 were included as fixed effects to account for linear and nonlinear effects of age on the outcomes. The models adjusted for the distant relation with three principal components as fixed effects, and recent relation with the random effect of ζ_i for participant *i*. The *CREBRF* allele A was the covariate of interest

in this model. The A allele term was treated as a continuous variable, where every A allele in a person's genotype counted as 1 unit. This follows the additive genetic modeling approach where disease risk is r for an AG genotype, but 2r for AA genotype (Lewis, 2002).

$$Y = \beta_0 + \boldsymbol{\beta_1 A\ Allele} + \beta_2 Age + \beta_3 Age^2 + \beta_4 PC_1 + \beta_5 PC_2 + \beta_6 PC_3 + \zeta_i + \varepsilon_{ij}$$

Equation 7 Genetic Mixed Effects Model for Mass Outcomes

These models were fit in R using the "lmekin" function of the "coxme" package (Therneau, 2020). The "lmekin" function was given the formula for the model of interest. The data were subset for the desired sex, and the "method" was set to "ML" for maximum likelihood. The "varlist" parameter was set to the kinship matrix multiplied by two, for reasons (need this). After all the following fields were populated, each model was fit. The coefficient estimate for the A allele term was stored, in addition to the standard error, p value, and 95% confidence interval.

3.0 Results

3.1 Sample Descriptions

The 2017-19 re-recruitment sample of the Samoan population consisted of 416 individuals after data cleaning. Males and females from the 2017-19 cohort had significant differences in every variable, except for weight, calf circumference, and CREBRF genotype (Table 1 2017-19 Cohort by Sex). The 2010 sample of the Samoan population after data cleaning resulted in a cohort of 1,970 individuals. The total in the 2010 sample accounts for the removal of individuals who were resampled in the follow up study. Table 2 compares the predictor variables for these two cohorts. All predictors were significantly different between the 2010 and 2017-19 samples, except for sex.

Table 1 2017-19 Cohort by Sex

	Male (N=187)	Female (N=229)	Total (N=416)	p
Age (yrs)				0.0391
Mean (SD)	52.496 (10.097)	50.512 (9.442)	51.404 (9.780)	
Range	30.680 - 72.695	30.850 - 71.929	30.680 - 72.695	
Genotype				0.7422
GG	85 (45.5%)	101 (44.1%)	186 (44.7%)	
AA	34 (18.2%)	37 (16.2%)	71 (17.1%)	
AG	68 (36.4%)	91 (39.7%)	159 (38.2%)	
Height (cm)				< 0.0011
Mean (SD)	172.331 (6.235)	161.966 (5.695)	166.625 (7.867)	
Range	157.800 - 189.500	147.650 - 177.400	147.650 - 189.500	
Weight (kg)				0.6521
Mean (SD)	98.326 (19.872)	97.438 (20.040)	97.837 (19.945)	
Range	54.700 - 162.200	45.800 - 192.750	45.800 - 192.750	
Abdomen Circum. (cm)				< 0.0011
Mean (SD)	108.646 (15.117)	114.995 (13.796)	112.141 (14.731)	
Range	76.450 - 153.650	79.850 - 178.600	76.450 - 178.600	
Hip Circum. (cm)				< 0.0011
Mean (SD)	109.678 (11.100)	119.872 (13.417)	115.290 (13.413)	

Range	87.850 - 162.350	86.400 - 181.200	86.400 - 181.200	
Calf Circum. (cm)				0.1231
Mean (SD)	38.792 (4.606)	38.086 (4.668)	38.403 (4.648)	
Range	21.950 - 52.550	24.200 - 50.550	21.950 - 52.550	
Triceps Skinfold (mm)				< 0.0011
Mean (SD)	28.703 (12.306)	35.089 (10.787)	32.219 (11.913)	
Range	5.000 - 60.000	8.000 - 63.000	5.000 - 63.000	
Forearm Skinfold (mm)				< 0.0011
Mean (SD)	13.598 (6.613)	19.009 (8.227)	16.576 (8.003)	
Range	3.000 - 35.500	3.500 - 41.500	3.000 - 41.500	
Subscapular Skinfold (mm)				< 0.0011
Mean (SD)	33.842 (13.997)	38.806 (11.064)	36.575 (12.695)	
Range	8.000 - 67.000	16.000 - 67.000	8.000 - 67.000	
Suprailiac Skinfold(mm)				< 0.0011
Mean (SD)	30.818 (13.308)	35.541 (11.473)	33.418 (12.539)	
Range	6.000 - 67.000	15.500 - 65.000	6.000 - 67.000	
Abdomen Skinfold (mm)				0.0021
Mean (SD)	33.791 (13.211)	37.692 (12.296)	35.938 (12.847)	
Range	5.900 - 65.000	15.000 - 67.000	5.900 - 67.000	

Total Lean Mass (g)				< 0.0011
Mean (SD)	65481.227 (9040.059)	51748.768 (7826.628)	57921.772 (10819.356)	
Range	40277.444 - 93765.749	27005.032 - 80860.575	27005.032 - 93765.749	
Total Fat Mass (g)				< 0.0011
Mean (SD)	29449.096 (12263.625)	42598.723 (12764.781)	36687.713 (14135.625)	
Range	5572.082 - 69207.741	16372.703 - 97199.197	5572.082 - 97199.197	
VAT Mass (g)				< 0.0011
Mean (SD)	1892.846 (1088.462)	1412.541 (608.199)	1628.447 (889.632)	
Range	120.546 - 5701.593	85.812 - 3131.137	85.812 - 5701.593	

1. Two sample t test 2. Pearson's Chi-sq test

Table 2 2010 and 2017-19 Cohort Comparison

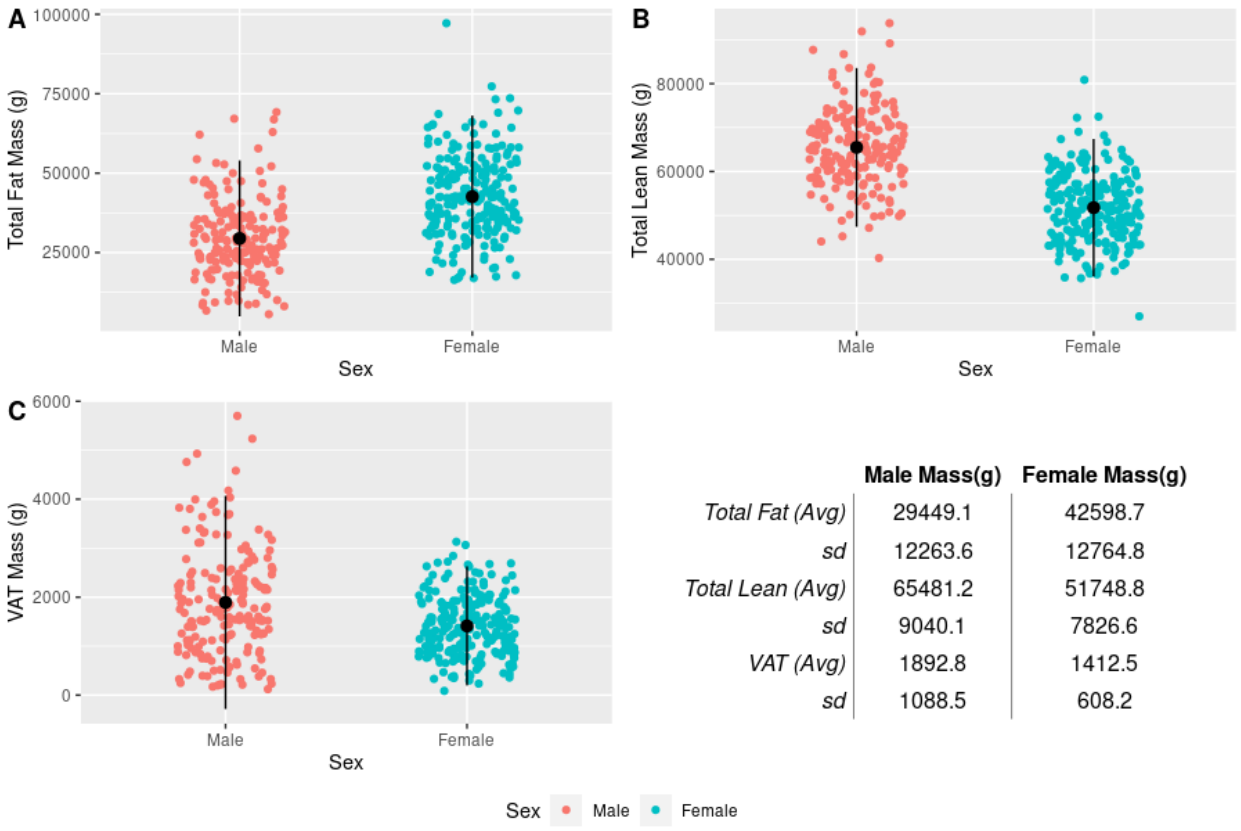
	2010 Cohort (N=1970)	2017-19 Cohort (N=416)	Total (N=2386)	p
Age (yrs)				< 0.001[1]
Mean (SD)	44.264 (11.573)	51.404 (9.780)	45.509 (11.600)	
Range	23.040 - 70.130	30.680 - 72.695	23.040 - 72.695	
Sex				0.669[2]
Female	1107 (56.2%)	229 (55.0%)	1336 (56.0%)	
Male	863 (43.8%)	187 (45.0%)	1050 (44.0%)	
Genotype				< 0.001[2]
N-Miss	4	0	4	
AA	117 (6.0%)	71 (17.1%)	188 (7.9%)	
AG	744 (37.8%)	159 (38.2%)	903 (37.9%)	
GG	1105 (56.2%)	186 (44.7%)	1291 (54.2%)	
Weight (kg)				< 0.001[1]
Mean (SD)	85.142 (15.535)	97.837 (19.945)	87.355 (17.078)	
Range	37.400 - 152.300	45.800 - 192.750	37.400 - 192.750	
Height (cm)				0.013[1]
Mean (SD)	165.569 (7.877)	166.625 (7.867)	165.753 (7.884)	
Range	140.100 - 188.200	147.650 - 189.500	140.100 - 189.500	
Abdomen Circum. (cm)				< 0.001[1]
Mean (SD)	100.670 (12.620)	112.141 (14.731)	102.670 (13.718)	
Range	58.800 - 144.700	76.450 - 178.600	58.800 - 178.600	
Hip Circum. (cm)				< 0.001[1]
Mean (SD)	106.924 (10.117)	115.290 (13.413)	108.383 (11.220)	
Range	78.500 - 152.300	86.400 - 181.200	78.500 - 181.200	
Calf Circum. (cm)				< 0.001[1]
Mean (SD)	40.782 (3.774)	38.403 (4.648)	40.367 (4.041)	
Range	27.900 - 54.500	21.950 - 52.550	21.950 - 54.500	
Triceps Skinfold (mm)				< 0.001[1]

Mean (SD)	27.113 (11.866)	32.219 (11.913)	28.004 (12.029)	
Range	5.000 - 60.500	5.000 - 63.000	5.000 - 63.000	
Forearm Skinfold (mm)				< 0.001[1]
Mean (SD)	10.776 (5.185)	16.576 (8.003)	11.788 (6.179)	
Range	2.000 - 42.500	3.000 - 41.500	2.000 - 42.500	
Subscapular Skinfold (mm)				< 0.001[1]
Mean (SD)	31.284 (11.755)	36.575 (12.695)	32.207 (12.090)	
Range	5.000 - 65.000	8.000 - 67.000	5.000 - 67.000	
Abdomen Skinfold (mm)				0.007[1]
Mean (SD)	34.170 (11.954)	35.938 (12.847)	34.478 (12.130)	
Range	6.000 - 64.000	5.900 - 67.000	5.900 - 67.000	
Suprailiac Skinfold(mm)				< 0.001[1]
Mean (SD)	28.594 (13.123)	33.418 (12.539)	29.435 (13.149)	
Range	4.000 - 61.500	6.000 - 67.000	4.000 - 67.000	

1. Two sample t test 2. Pearson's Chi-sq test

3.2 2017-19 Cohort: Graphical Summary of Mass Outcomes by Sex

The outcomes of total fat, lean, and VAT mass were graphed by sex and displayed in Figure 1. Females tended to have a larger total fat mass compared to males (mean female = 42,598.7g [sd = 12,764.8g], mean male = 29,449.1g [sd = 12,263.6g]). Males tended to have a larger total lean mass compared to females (mean male = 65,481.2g [sd = 9,040.1g], mean female = 51,748.8g [sd = 7,826.6]). Males had a marginally larger mean VAT mass compared to females (mean male = 1,892.8g [sd = 1,088.5g], mean female = 1,412.5g [sd = 608.2g]).



	Male Mass(g)	Female Mass(g)
Total Fat (Avg)	29449.1	42598.7
sd	12263.6	12764.8
Total Lean (Avg)	65481.2	51748.8
sd	9040.1	7826.6
VAT (Avg)	1892.8	1412.5
sd	1088.5	608.2

Figure 1 2017-19 Cohort Total Fat, Lean, and VAT Mass by Sex (mean ± 2 sd)

3.3 2017-19 Cohort: Correlation

3.3.1 Correlation Matrix

The correlation between pairs of continuous predictor variables were assessed in Figure 2. The largest positive correlation was seen between abdomen and hip circumferences (r = .87). The largest negative correlation was between age and triceps skinfold (r = -0.29). The lowest absolute correlation was between height and abdomen circumference (r = -0.03).

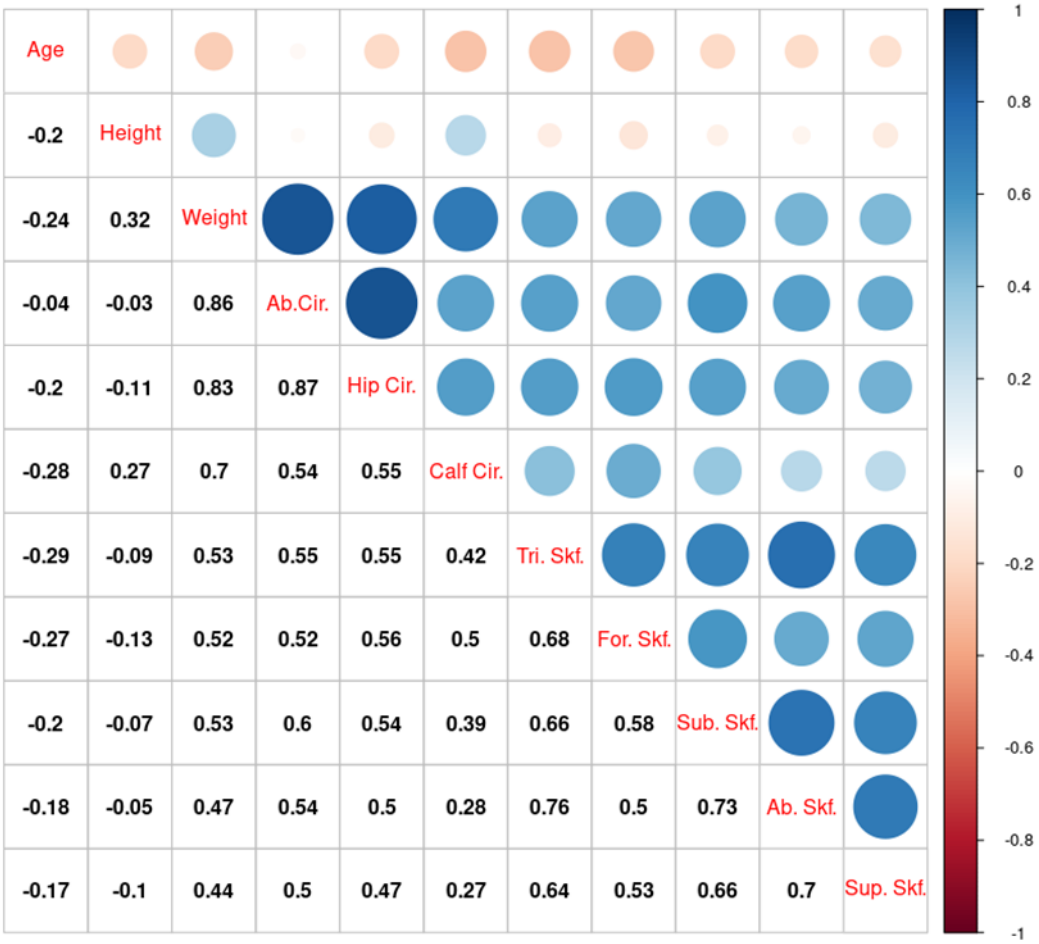


Figure 2 Correlation Between Predictor Variables

3.3.2 Total Fat Mass Correlation with Predictors

In order to examine the relationship between the continuous predictor variables and total fat mass, each predictor was graphed against total fat mass on a scatter plot. Each relationship was stratified by sex, and the correlation coefficients were calculated (Figure 3). Age in males was the only predictor that had a nonsignificant correlation; all other relationships for both sexes were significant (2e-16 < p < 0.0084, Figure 3). Age in both sexes showed a negative correlation with total fat mass; all other predictors for both sexes were positively associated with total fat mass (Figure 3).

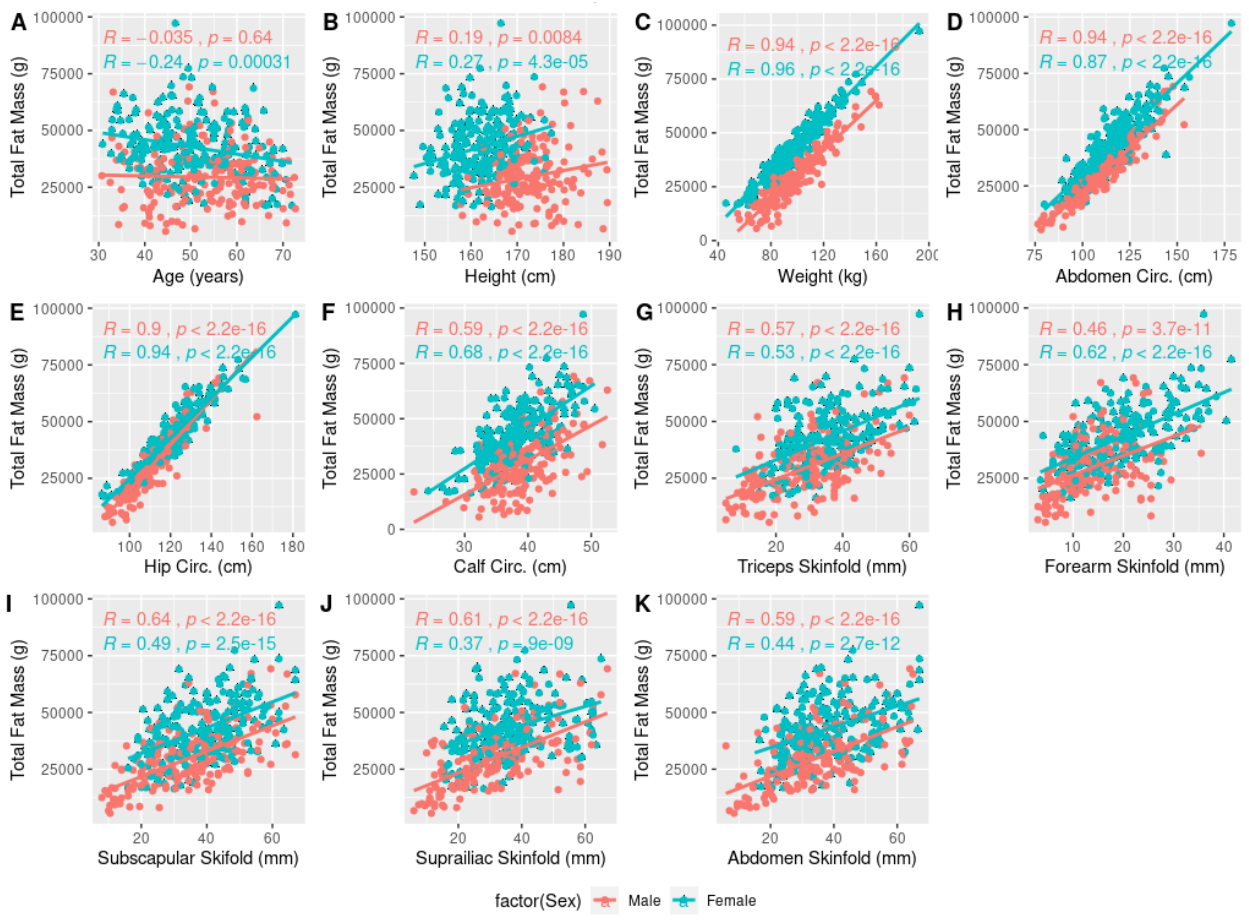


Figure 3 Total Fat Mass Correlation with Continuous Predictors by Sex

3.3.3 Total Lean Mass Correlation with Predictors

Similar to section 3.3.2, scatterplots and correlation coefficients were graphed and calculated to assess the relationships between the predictor variables and the outcome of total lean mass by sex. All the variables for both sexes were significantly correlated with total lean mass (2e-16 < p < 4.7e-6, Figure 4). Age in both sexes had a negative correlation with total lean mass,all other predictors were postively associated with total lean mass (Figure 4).

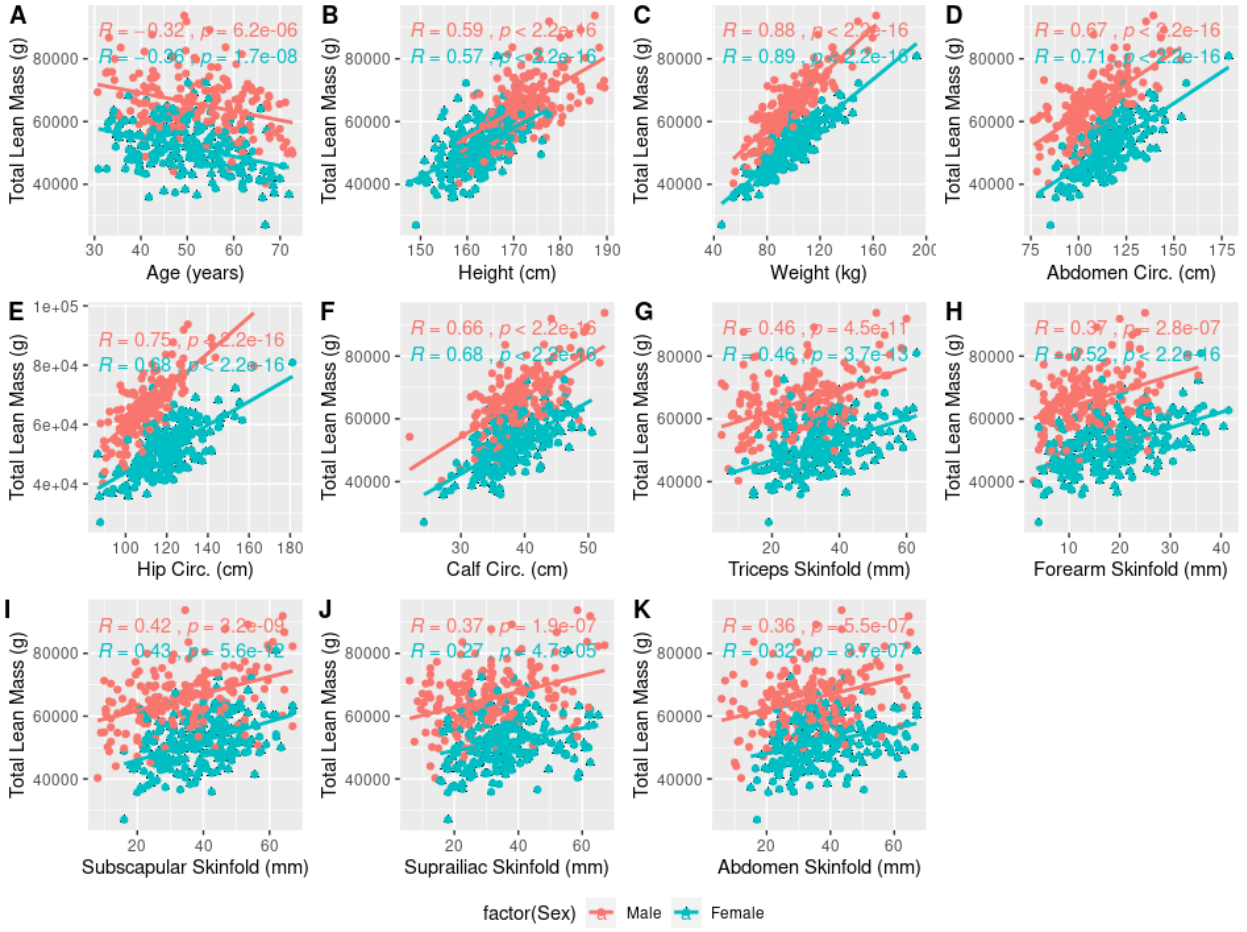


Figure 4 Total Lean Mass Correlation with Continuous Predictors by Sex

3.3.4 VAT Mass Correlation with Predictors

Finally, the relationship between the predictors and VAT mass by sex was assessed using the same scatterplot and correlation coefficient procedure. The correlation between age and VAT mass in females was not significant ($p = 0.52$, Figure 5). Age was positively correlated with VAT mass; however, this was only significant in the males ($r = 0.19$, $p = 0.0099$, Figure 5). Age was previously found to be negatively correlated with both total fat and total lean mass for both sexes (Figure 3, Figure 4). The correlations of height and VAT mass in both males and females were not significant (Figure 5). All other predictors were significantly correlated with VAT mass for both sexes (Figure 5). Notably, the correlations between predictors and VAT mass across sex were drastically different (Figure 5), which was not seen in total fat or lean mass (Figure 3, Figure 4). In general, males had stronger positive correlations between each predictor and VAT mass than females (Figure 5).

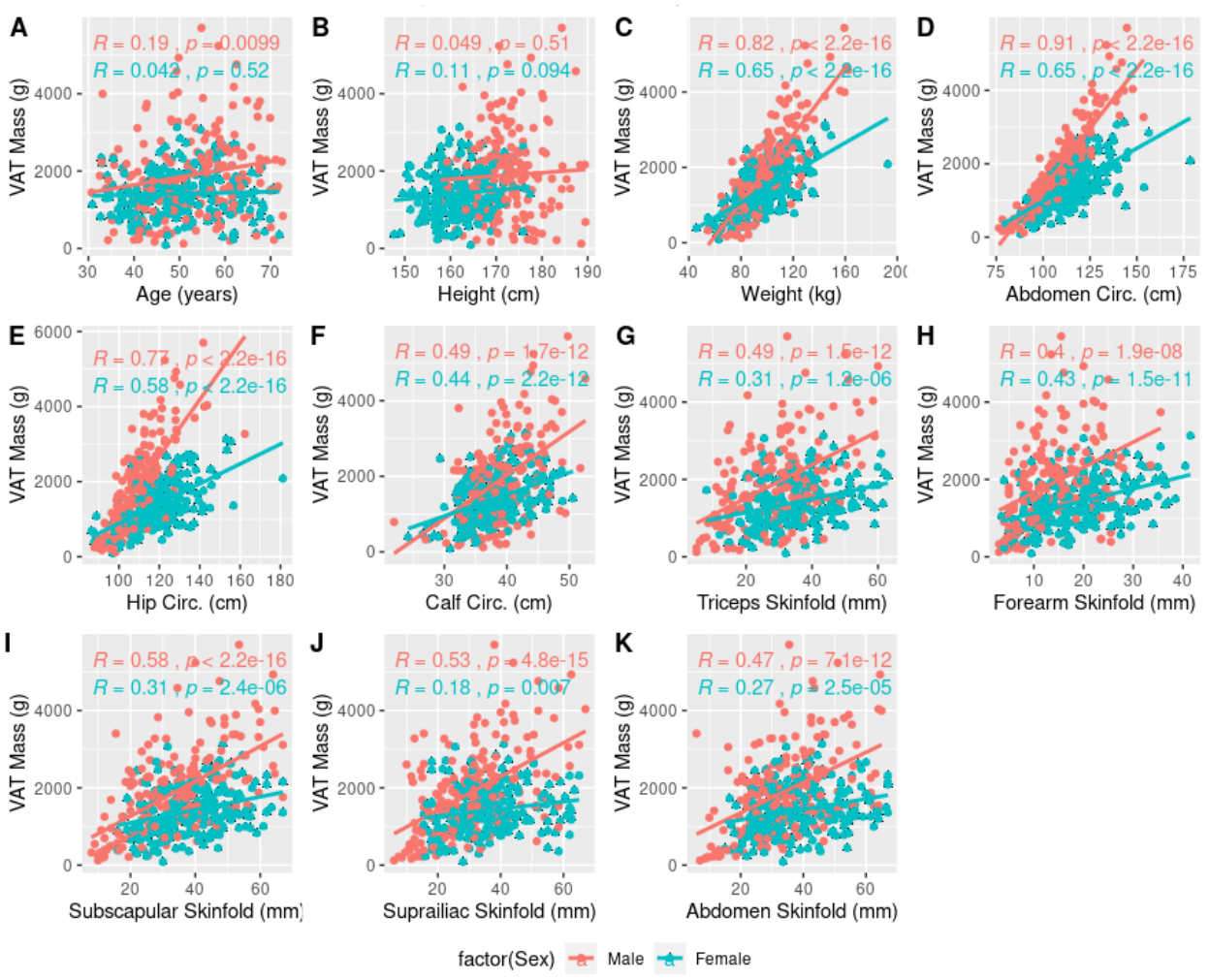


Figure 5 VAT Mass Correlation with Continuous Predictors by Sex

3.4 MARS Models

After preliminary analyses of the dependent and independent variables, the MARS predictive models were constructed and validated as described in section 2.2.1. The following subsections address the results from the model building process for the outcomes of total fat, lean, and VAT mass in both males and females. To assess predictive accuracy of the models, the cross validated RMSE, MAE, and R^2 values were calculated and displayed in tables. Finally, partial

dependency plots for the covariates were created for the best MARS model. The partial dependency plots show the predicted response as the given covariate(s) changes, while holding other covariates at their median value. For first degree terms, the plots have the predicted outcome on the y axis and the given covariate on the x axis. For second degree terms, or interactions between first degree terms, the plots display the two covariates on the x and y axes, and the predicted outcome on the z axis (vertical).

3.4.1 Total Fat Mass

Male

The final male MARS model for total fat mass utilized quadratic terms (degree) and a maximum of 12 terms (nprune) after backwards elimination (Table 3). The final model had the lowest RMSE, 3,072.64g, yet only the second highest R^2, at 0.943 (Table 3). A third-degree MARS model had comparable statistics to the final model, but it had a higher RMSE and was less parsimonious, with 67 terms allowed after backwards elimination (Table 3). The partial dependency plot in Figure 6 illustrates that out of the first-degree terms of the final model, weight, abdomen circumference, and hip circumference have the greatest impact on the prediction of total fat mass in males. Furthermore, Figure 6 shows that interactions were important for predicting total fat mass in males only if they included at least one of weight, abdomen circumference, or hip circumference.

Table 3 Male Total Fat Mass Best MARS Model

degree	nprune	RMSE	R^2	MAE	RMSE SD	R^2 SD	MAE SD
2	**12**	**3072.64**	**0.943**	**2524.77**	**419.43**	**0.022**	**374.93**
3	67	3082.71	0.944	2538.50	322.67	0.015	307.55

*Best model in **bold**

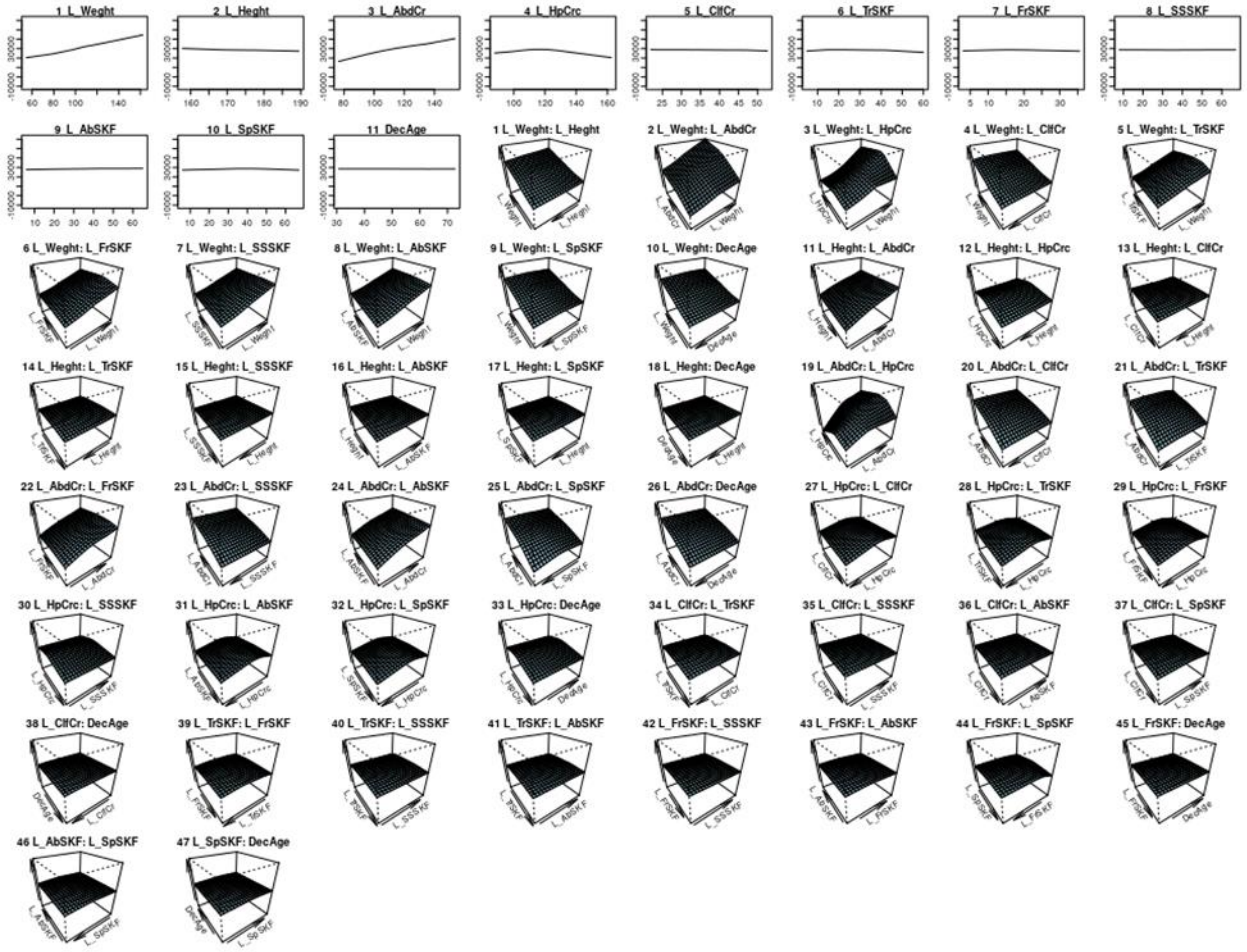


Figure 6 Partial Dependence of Male Total Fat Mass MARS Model Predictors

Female

The final female MARS model for total fat mass applied first-degree terms and a maximum of 12 terms (nprune) after backwards elimination (Table 4). The final model had the lowest RMSE, at 2,852.46g, and the highest R^2 at 0.951 (Table 4). There was one second-degree model that had similar metrics to the final model, however, it was slightly less accurate in all categories (Table 4). Figure 7 indicates that weight and hip had the largest impact in predicting total fat mass in females. There were small effects on predicted total fat mass for the lower most values of height, and the

upper most of abdomen circumference (Figure 7). Forearm skinfold and age demonstrated very small changes in predicted total fat mass in females, in comparison to the effects of the aforementioned terms (Figure 7).

Table 4 Female Total Fat Mass Best MARS Model

degree	nprune	RMSE	R^2	MAE	RMSE SD	R^2 SD	MAE SD
1	**12**	**2852.46**	**0.951**	**2240.36**	**356.37**	**0.016**	**324.03**
2	12	2860.72	0.951	2285.98	357.49	0.016	326.67

*Best model in **bold**

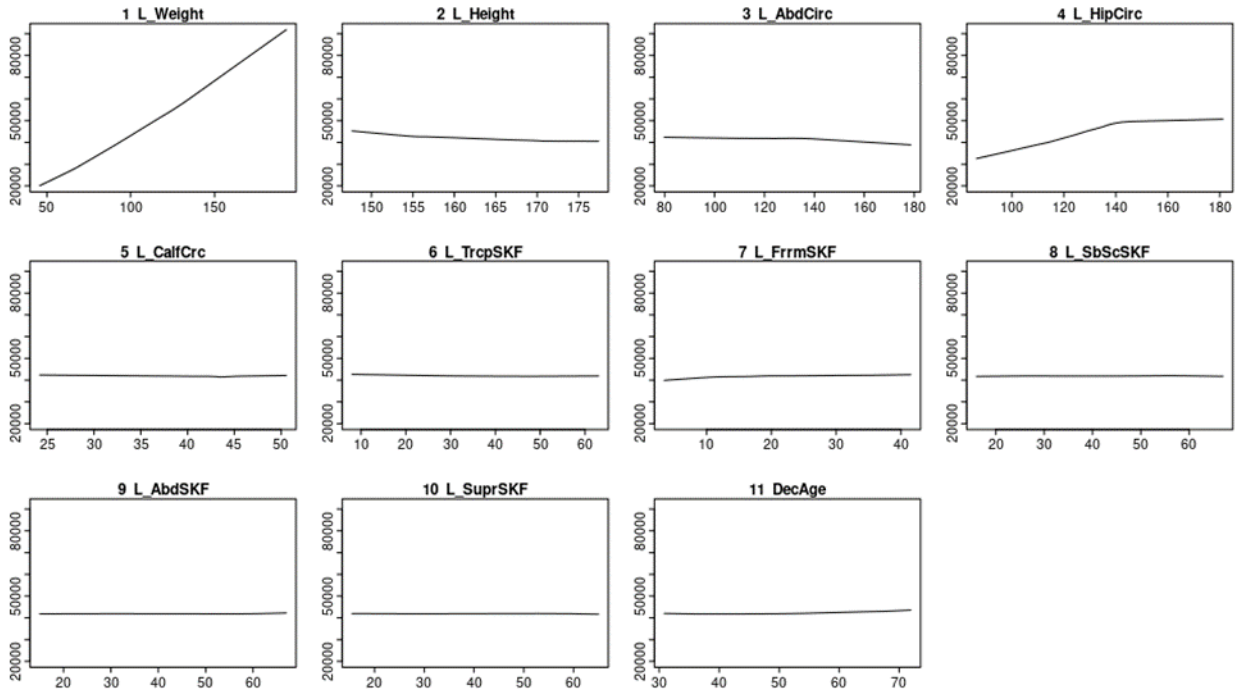


Figure 7 Partial Dependence of Female Total Fat Mass MARS Model Predictors

3.4.2 Total Lean Mass

Male

The final male MARS model for total lean mass used cubic terms (degree) and permitted a maximum of 23 terms (nprune) after backwards elimination (Table 5). This final model had the

lowest RMSE, 3,276.13g, and the highest R^2 at 0.876 (Table 5). The second best performing model in terms of RMSE was also third-degree, allowed 12 terms after pruning, and had the lowest MAE and R^2, at 2,656.26g and 0.875, respectively (Table 5). The third best model in terms of RMSE was second-degree and allowed 100 terms post-pruning. This model had tied for the highest R^2, 0.876, and had the second lowest MAE, 2,658.04g (Table 5). Despite the narrow margins of difference in some of the metrics, the third-degree model with 23 terms allowed after pruning was chosen as the final model based on RMSE (procedure in section 2.2.1). Figure 8 reveals that the most important first-degree terms for predicting total lean mass in males are weight and abdomen circumference, followed by hip circumference and triceps skinfold. The second-degree interactions important to predicting total lean mass included at least one of the previously mentioned first-degree terms, with the exception of triceps skinfold. Interactions between triceps skinfold and forearm, subscapular, suprailiac, or abdomen skinfolds (all seemingly unimportant as first-degree terms) did not greatly impact the predicted total lean mass (Figure 8).

Table 5 Male Total Lean Mass Best MARS Model

degree	nprune	RMSE	R^2	MAE	RMSE SD	R^2 SD	MAE SD
3	**23**	**3276.13**	**0.876**	**2679.35**	**358.96**	**0.037**	**310.55**
3	12	3281.21	0.875	2656.26	395.38	0.037	327.33
2	100	3285.79	0.876	2658.04	356.65	0.031	244.70

*Best model in **bold**

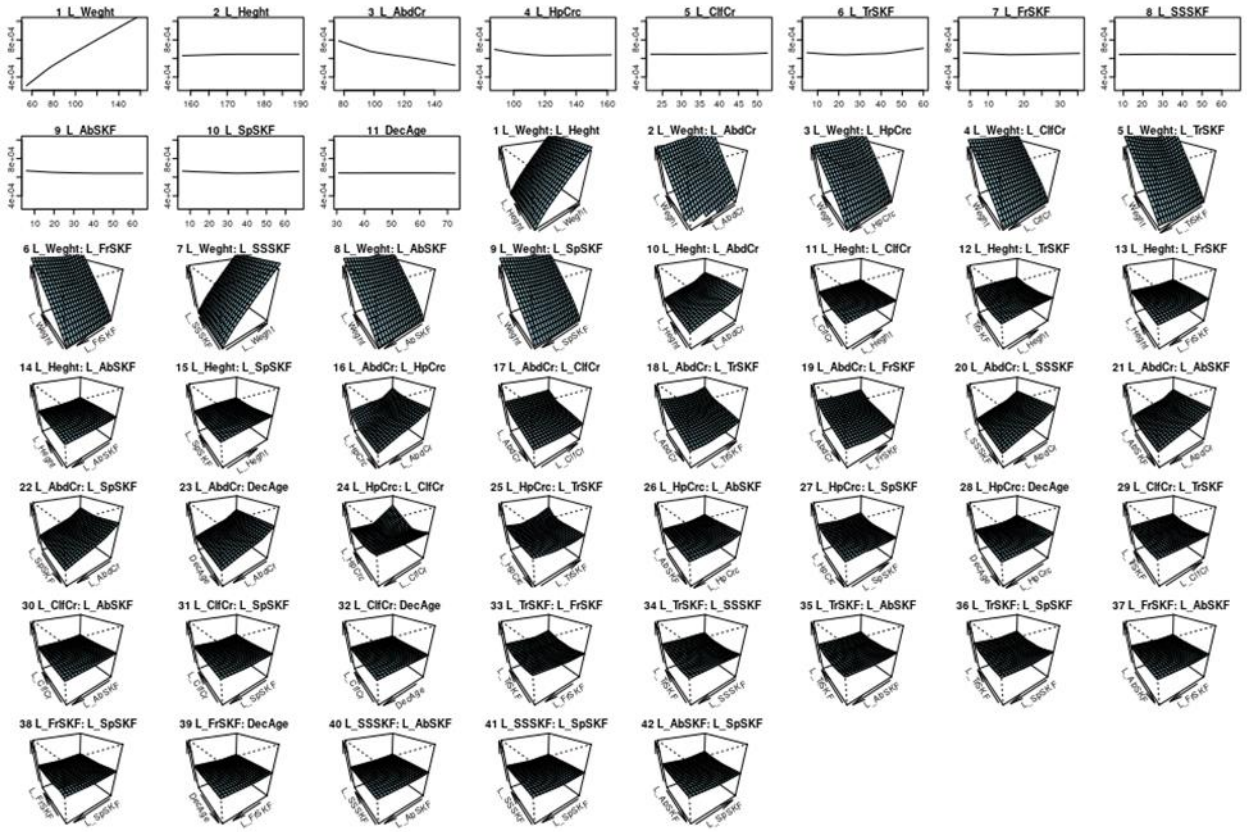


Figure 8 Partial Dependence of Male Total Lean Mass MARS Model Predictors

Female

The final female MARS model for total lean mass utilized cubic terms (degree) and a maximum of 12 terms (nrpune) after backwards elimination (Table 6). The RMSE for the best model was only just lower than the second best model, measured at 2,965.14g and 2,965.75g, respectively (Table 6). The best model also had a marginally higher R^2 at 0.867, compared to the second best model, at 0.865 (Table 6). Of note, the second best model, which was third-degree and had 89 terms allowed post-pruning, had the lowest MAE, 2,351.20g (Table 6). Figure 9 indicates the first-degree terms of weight and hip circumference were the most important with respect to predicting total lean mass in females. The only second-degree interactions that

demonstrated an effect on predicting lean mass included at least one of weight and hip circumference (Figure 9).

Table 6 Female Total Lean Mass Best MARS Model

degree	nprune	RMSE	R^2	MAE	RMSE SD	R^2 SD	MAE SD
3	**12**	**2965.14**	**0.867**	**2351.20**	**240.61**	**0.048**	**199.24**
3	89	2965.75	0.865	2321.72	275.27	0.047	205.84

*Best model in **bold**

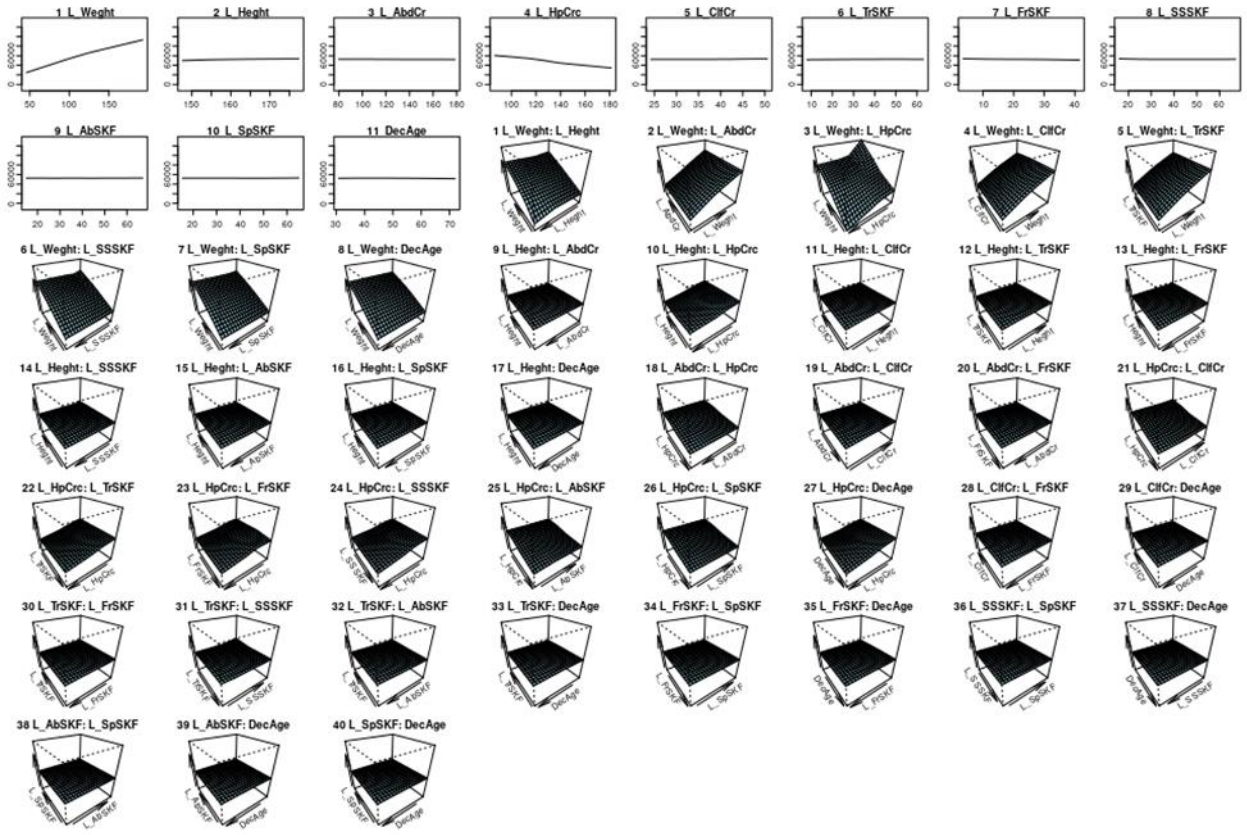


Figure 9 Partial Dependence of Female Total Lean Mass MARS Model Predictors

3.4.3 VAT Mass

Male

The final male MARS VAT mass model used cubic (degree) terms and only allowed 2 terms (nprune) after backwards elimination (Table 7). The final model had the lowest RMSE, 441.7g (Table 7). The second best model in terms of RMSE was first-degree, and also only allowed 2 terms after backwards elimination. This second choice model had a marginally higher RMSE (444.8g), but it performed almost as well as the final model in all metrics (Table 7). The male VAT mass model was the only MARS model that did not select all possible predictors (Table 9). Figure 10 demonstrates the final model's unilateral dependence on abdomen circumference for VAT mass prediction.

Table 7 Male VAT Mass Best MARS Model

degree	nprune	RMSE	R^2	MAE	RMSE SD	R^2 SD	MAE SD
3	**2**	**441.7**	**0.845**	**332.70**	**112.39**	**0.054**	**69.81**
1	2	444.8	0.843	333.92	111.32	0.052	69.34

*Best model in **bold**

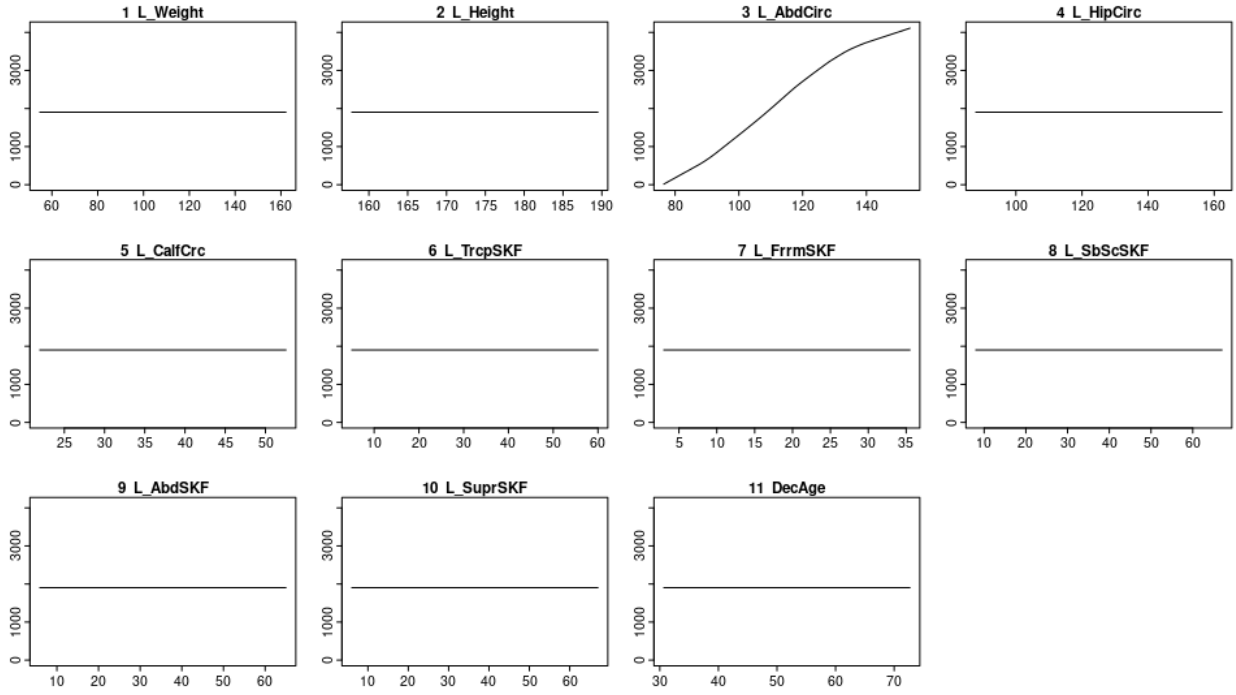


Figure 10 Partial Dependence of Male VAT Mass MARS Model Predictors

Female

The final female MARS VAT mass model used quadratic terms (degree) and a maximum of 34 terms (nprune) after backwards elimination (Table 8). The second choice model had a lower MAE, by less than 1g, and had an approximately equal R^2 to the final model (Table 8). However, the final model was selected based on the lowest RMSE, 432.9g, which was just over a gram lower than the next best choice (Table 8). Figure 11 demonstrates the high dependence on weight when predicting female VAT mass with this model. Other important terms include height, abdomen circumference, and hip circumference. The second-degree interactions do not reveal any new important terms for predicting VAT Mass; the interactions that appear important involve at least one of weight, height, abdomen circumference, and hip circumference (Figure 11).

The interaction of abdomen circumference and hip circumference produced a highly irregular, and extreme change in comparison to other interactions (Figure 11). This result was attributed to participant 2472, who had consistently high measurements compared to the rest of the females (e.g., 2472's Ab. Circ. = 178.6 cm, 2nd highest = 156.45 cm). Removal of this participant partially reduced the extreme interaction seen between abdomen and hip circumference, but it did not greatly impact the model's predictive accuracy (model w/o 2472: RMSE = 430.6g, R^2 = 0.498).

Table 8 Female VAT Mass Best Model

degree	nprune	RMSE	R^2	MAE	RMSE SD	R^2 SD	MAE SD
2	**34**	**432.86**	**0.504**	**361.59**	**52.82**	**0.142**	**55.05**
3	12	434.33	0.499	360.96	61.50	0.148	58.08

*Best model in **bold**

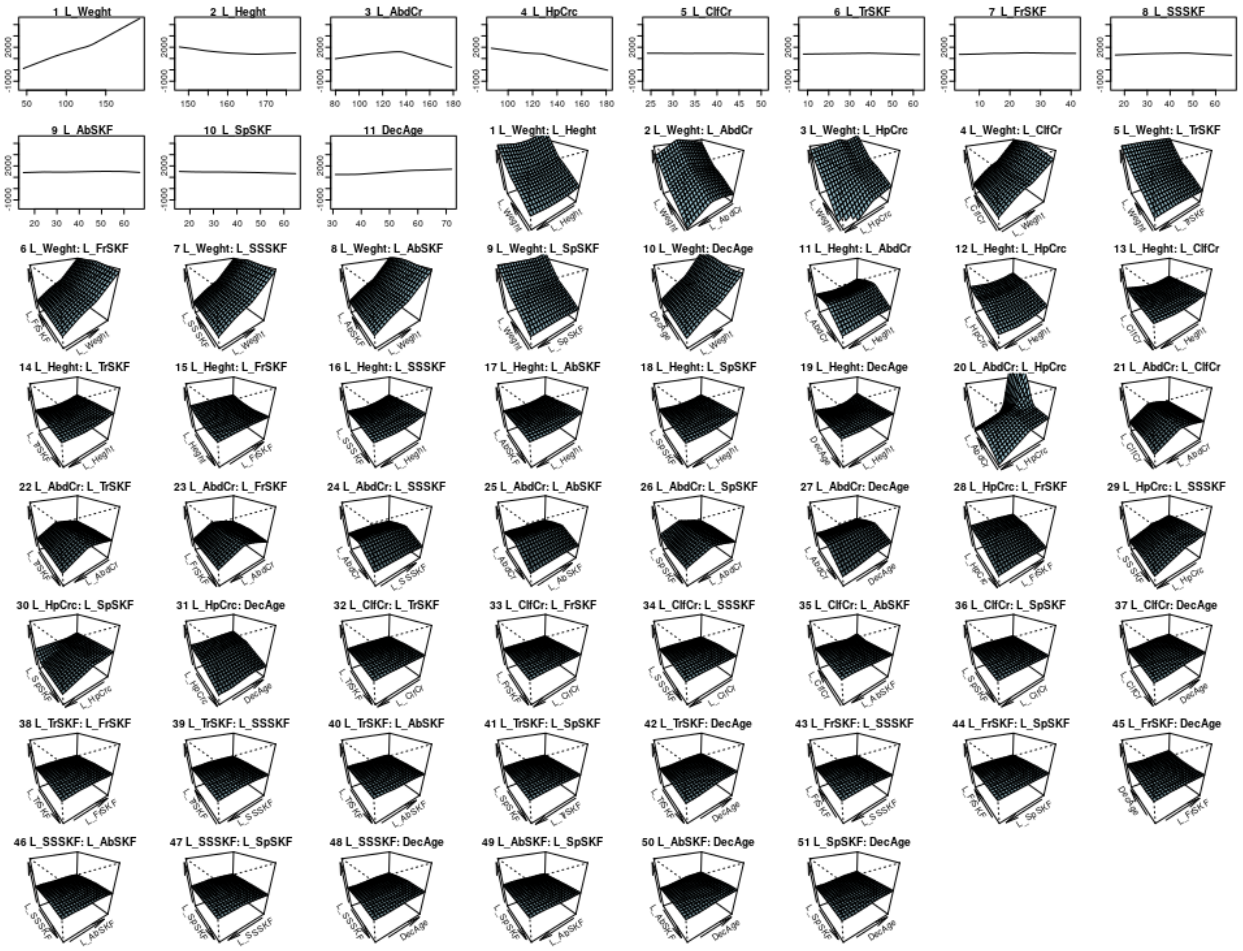


Figure 11 Partial Dependence of Female VAT Mass MARS Model Predictors

3.5 Model Performance

Model performance metrics and predictor selection information are displayed in Table 9. MARS models performed nearly as well, and in some cases, better than the other comparison models for all three outcomes. Multiple linear regression models presented comparable or better statistics than the respective MARS models. Ridge regression models were rarely the best

performer in all outcomes compared to the other techniques. The models developed from the research performed by Swinburn et al. (1996) performed markedly worse in predicting fat mass in comparison to the potential alternatives. The worst model in terms of RMSE and MAE was the Swinburn regression model that utilized height, weight, sex, and age, to predict total fat mass (Swinburn HWSA in table). There was not much difference between male and female models for the same outcome. The only notable difference in performance across sexes was for the female VAT mass models, which had R^2 values near 0.50 compared to males at roughly 0.80 (Table 9).

Table 9 Model Performance and Predictor Comparison

	Model Metrics					**Circumferences**			**Skinfolds**							
Male	RMSE	MAE	R^2	Wgt.	Hgt.	Abd.	Hip	Calf	Tri.	For.	Sub.	Sup.	Abd.	Age	BMI	Sex
Total Fat Mass																
MARS	**3072.6**	**2524.8**	**0.943**	X	X	X	X	X	X	X	X	X	X	X		
Regression	3132.6	2599.7	0.934	X	X	X		X					X			
Swinburn (BMI)	4190.3	3368.8	0.883												X	
Ridge	3680.0	2954.5	0.898	X	X	X	X	X	X	X	X	X	X	X		
Total Lean Mass																
MARS	3276.1	**2679.3**	0.876	X	X	X	X	X	X	X	X	X	X	X		
Regression	**3244.4**	2681.2	**0.934**	X	X	X		X	X			X	X			
Ridge	4169.5	3464.4	0.696	X	X	X	X	X	X	X	X	X	X	X		
VAT Mass																
MARS	441.7	332.7	0.845			X										
Regression	**405.3**	**318.2**	**0.868**	X	X	X	X							X		
Ridge	436.6	343.1	0.798	X	X	X	X	X	X	X	X	X	X	X		
Female	RMSE	MAE	R^2	Wgt.	Hgt.	Abd.	Hip	Calf	Tri.	For.	Sub.	Sup.	Abd.	Age	BMI	Sex
Total Fat Mass																
MARS	2852.5	2240.4	0.951	X	X	X	X	X	X	X	X	X	X	X		
Regression	**2767.5**	**2186.1**	**0.953**	X	X		X			X				X		
Swinburn (BMI)	4080.5	3166.3	0.893												X	
Ridge	2776.3	2198.3	0.937	X	X	X	X	X	X	X	X	X	X	X		
Total Lean Mass																
MARS	2965.1	2351.2	0.867	X	X	X	X	X	X	X	X	X	X	X		
Regression	**2722.9**	**2119.9**	**0.954**	X	X		X	X		X				X		
Ridge	2884.8	2317.0	0.808	X	X	X	X	X	X	X	X	X	X	X		
VAT Mass																
MARS	432.9	361.6	0.504	X	X	X	X	X	X	X	X	X	X	X		
Regression	420.8	343.8	**0.520**	X	X	X	X							X		
Ridge	**388.1**	**313.6**	0.399	X	X	X	X	X	X	X	X	X	X	X		
Swinburn HWSA	RMSE	MAE	R^2	Wgt.	Hgt.	Abd.	Hip	Calf	Tri.	For.	Sub.	Sup.	Abd.	Age	BMI	Sex
Regression	14820.2	12041.4	0.946	X	X									X		X

*Metrics in **bold** are the highest for R^2 and lowest for RMSE and MAE for an outcome; X denotes variable was selected for the model

3.5.1 Total Fat Mass

The Swinburn BMI models had the worst performance metrics for both sexes. The Swinburn HWSA model that utilized height, weight, sex, and age to predict total fat mass had a high R^2 (0.946, Table 9), however, the RMSE and MAE for this model were the worst out of all potential models tested.

For males, the MARS model had the lowest RMSE and MAE (3072.6g and 2524.8g, respectively) and the highest R^2 (0.943, Table 9). The male MARS model selected all eleven of the possible predictors[5]. The linear regression model had comparable statistics to the MARS model but was ultimately worse in all three metrics.

For females, the linear regression presented the most optimal RMSE, MAE, and R^2 out of all the possible models. This regression model included weight, height, hip circumference, forearm skinfold, and age as predictors. The female MARS model for total fat produced the third best RMSE and MAE, which were only off from the best by less than 100g. The female MARS model had the second best R^2, at 0.951, and included all eleven of the possible predictors (Table 9).

3.5.2 Total Lean Mass

For predicting total lean mass in males, the linear regression model had the lowest RMSE (3244.4g) followed by the MARS model (3276.1, Table 9). The linear regression model posted

[5] This excludes sex and BMI, which were only used when recreating the Swinburn models.

the highest R^2 , at 0.934, but the MARS model had the lowest MAE, at 2679.3g (Table 9). The MARS model selected all eleven predictors, while the linear regression was comprised of all but hip circumference, forearm skinfold, subscapular skinfold, and age.

For females, the linear regression model performed the best at predicting total lean mass. The linear model had the lowest RMSE and MAE, and the highest R^2 (Table 9). The MARS model had the third lowest RMSE at 2965.1g, compared to the linear regression model's RMSE of 2722.9g. The MARS model selected all eleven predictors, while the linear model excluded triceps skinfold, subscapular skinfold, suprailiac skinfold, and abdomen skinfold (Table 9).

3.5.3 VAT Mass

The male linear regression model for predicting VAT mass had the lowest RMSE and MAE, and the highest R^2 . The male MARS model RMSE was slightly above that for linear regression, at 441.7g and 405.3g, respectively (Table 9). The linear regression model selected the predictors of height, weight, abdomen circumference, hip circumference, and age. Notably, the MARS model only employed abdomen circumference for predicting VAT mass in males.

For females, the ridge regression presented the lowest RMSE and MAE out of all three VAT mass models. The ridge regression model had an RMSE of 388.1g, compared to 420.8g and 432.9g of the linear and MARS models respectively (Table 9). The ridge regression model had the worst R^2, at 0.399, while the linear regression model had the best, at 0.520 (Table 9). The ridge and MARS models utilized all eleven predictors, while the linear model only included weight, height, abdomen circumference, hip circumference, and age.

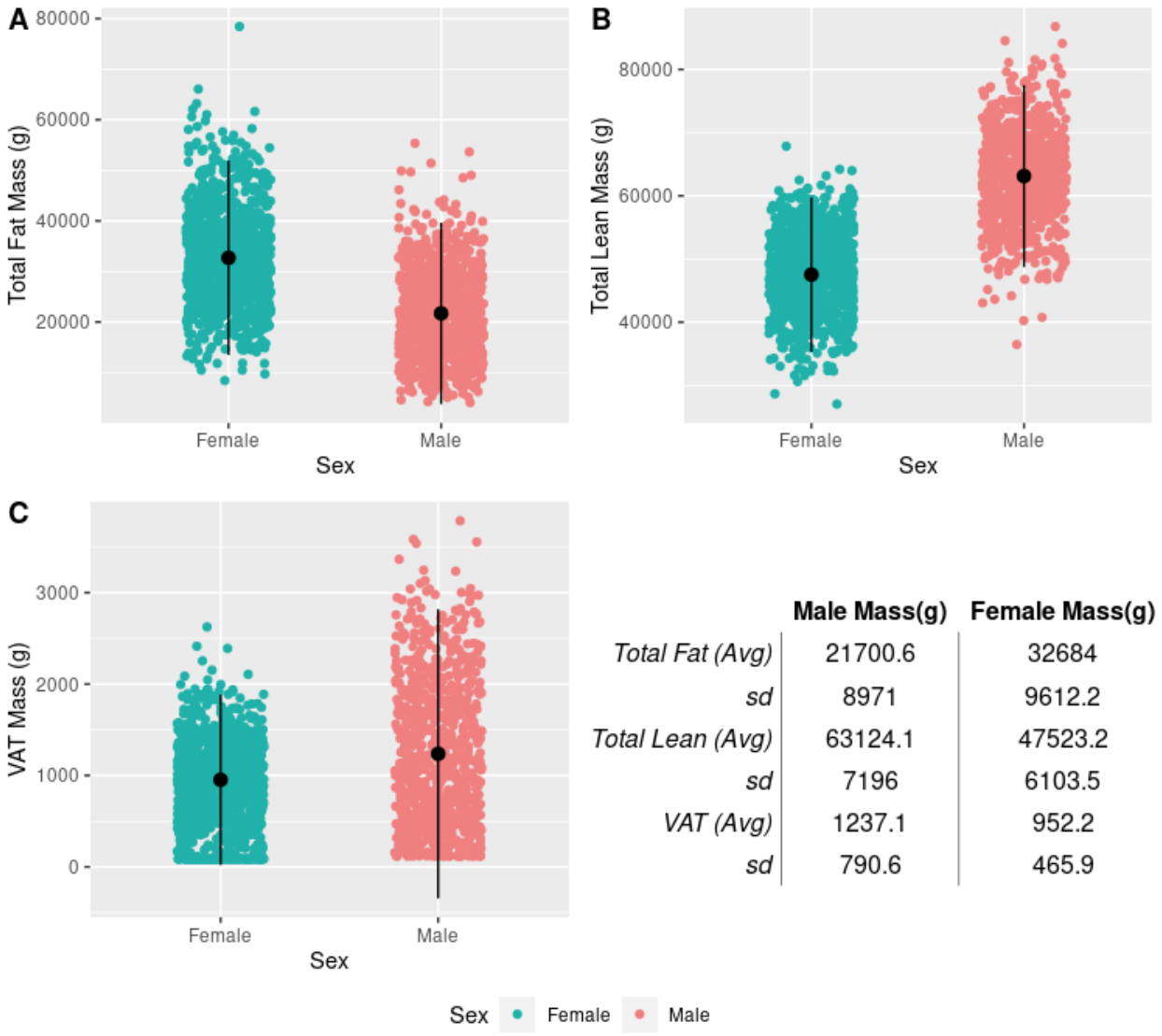
3.6 Imputed Mass Measurements

The MARS imputed total fat, lean, and VAT mass measurements for the 2010 cohort are summarized in Table 10. The males from the 2010 cohort had an imputed mean total fat mass of 21,701g, while the females had a mean of 32,684g (Table 10). The females of the 2010 cohort had an imputed mean total lean mass of 47,523g, compared to the male's mean of 63,124g. The imputed mean male VAT mass for the 2010 cohort was 1237.1g, while the female mean was 952.2g. Figure 12 displays the distribution of each of these imputed outcomes for the 2010 cohort stratified by sex. The imputed outcome distributions retain the same sex-based relationships found in the 2017-19 cohort. females tend to have a higher fat mass and a lower lean mass than males, VAT mass reaches higher for males (Figure 12). Predicted VAT mass values were truncated to the minimum value observed in the 2017-19 cohort for each sex (min. male VAT mass = 120.5g, min. female VAT mass = 85.8g, Table 10).

Table 10 2010 Cohort Mass Imputations by Sex and 2017-19 Cohort Comparisons

	Males		Females	
Total Fat Mass (g)	*2010 Cohort Imputations*	*2017-19 Cohort*	*2010 Cohort Imputations*	*2017-19 Cohort*
Min	4,048	5,572	8,408	16,373
Q1	14,554	22,203	26,203	34,377
Med	21,061	28,089	31,980	41,920
Mean	21,701	29,449	32,684	42,599
Q3	28,147	36,265	38,333	50,891
Max	55,352	69,208	78,441	97,199
Total Lean Mass (g)	*2010 Imputations*	*2017-19*	*2010 Imputations*	*2017-19*
Min	36,467	40,277	27,042	27,005
Q1	58,585	60,032	43,400	46,368
Med	63,236	65,042	47,492	51,467
Mean	63,124	65,481	47,523	51,749
Q3	67,645	70,558	51,777	57,133
Max	86,792	93,766	67,846	80,861
VAT Mass (g)	*2010 Imputations*	*2017-19*	*2010 Imputations*	*2017-19*
Min	120.5*	120.5	85.8*	85.8
Q1	551.6	1,122.0	624.3	957.8
Med	1,152.9	1,772.1	992.5	1,362.7
Mean	1,237.1	1,892.8	952.2	1,412.5
Q3	1,824.1	2,458.7	1,285.8	1,858.1
Max	3,785.7	5,701.6	2,624.9	3,131.1

*2010 Cohort VAT mass imputations were truncated to minimum values for each sex in 2017-19 cohort



	Male Mass(g)	Female Mass(g)
Total Fat (Avg)	21700.6	32684
sd	8971	9612.2
Total Lean (Avg)	63124.1	47523.2
sd	7196	6103.5
VAT (Avg)	1237.1	952.2
sd	790.6	465.9

Figure 12 2010 Cohort Imputed Mass Outcome by Sex (mean ± 2 sd)

3.7 Genetic Mixed Effects Models

The three outcomes of total fat, lean, and VAT mass were sex stratified and regressed following the formula in Equation 8. Of interest were the estimated coefficients for the minor (A)

allele of rs373863828. Adjustments were made to account for the fixed effects of age, age^2, three principal components of ancestry, and the random subject effect via a kinship matrix.

$$Y = \beta_0 + \boldsymbol{\beta_1 A\ Allele} + \beta_2 Age + \beta_3 Age^2 + \beta_4 PC_1 + \beta_5 PC_2 + \beta_6 PC_3 + \zeta_i + \varepsilon_{ij}$$

Equation 8 Genetic Mixed Effects Model for Mass Outcomes

3.7.1 *CREBRF* A Allele

Both males and females showed significant positive associations between the A allele of rs373863828, and the outcomes of total fat, lean, and VAT mass ($p < 0.01$, Table 11). In males, each copy of the A allele was associated with an average total fat mass of 1,552.2g (95% CI [578.0-2,524.4g]), and females an average of 1,673.5g (95% CI [762.8-2,584.1g], Table 11). The average total lean mass in males was 1,634.9g (95% CI [860.6-2,409.3g]) for each copy of the A allele, while average total lean mass in females was 1050.0g per copy of the A allele (95% CI [485.9-1,614.0g], Table 11). Average VAT mass was also lower in females, at 70.7g (95% CI [29.6-111.8g]) per copy of the A allele, compared to the males' average of 115.74g per copy of the A allele (95% CI [32.5-199.0g], Table 11).

Table 11 CREBRF A Allele Coefficients

Male (n = 857)	**β**	**SE**	**p**	**95% L**	**95% U**
Total Fat Mass	1552.17	496.03	0.001753	579.96	2524.38
Total Lean Mass	1634.90	395.08	0.000035	860.55	2409.26
VAT Mass	115.74	42.46	0.006417	32.51	198.96
Female (n = 1100)	**β**	**SE**	**p**	**95% L**	**95% U**
Total Fat Mass	1673.45	464.60	0.000316	762.83	2584.07
Total Lean Mass	1049.95	287.78	0.000264	485.91	1613.99
VAT Mass	70.70	20.95	0.000741	29.63	111.77

4.0 Discussion

4.1 Pre-Model Building Analyses

Both the 2010 and 2017-19 cohorts had participants removed for missingness in either the predictors or outcome variables (2017-19 n miss = 5, 2010 n miss[6] = 716, section 2.1.2). Although no extensive analyses were performed to assess the makeup of those removed, it was assumed that missingness was mainly due to randomness in both the 2010 and 2017-19 cohorts.

When assessing the correlation between the continuous predictors in the 2017-19 cohort, weight, hip circumference, and abdomen circumference were found to have strong associations with one another ($r > 0.80$, Figure 2). Post hoc assessment of variance inflation factor (VIF) for the covariates of the linear models revealed that the weight, hip circumference, and abdomen circumference terms had high VIFs, ranging from 5.4 to 15.6. Each model was built with a valid variable selection technique, but combinations of these terms were often present in the final model. The inclusion of collinear terms was justified as the objective of the thesis was to accurately predict the three mass outcomes; their inclusion should not affect accuracy of prediction on new data, presuming the new data holds similar patterns of correlation between weight, hip, and abdomen measurements. However, future research should consider the benefits of removing some of these redundant terms, like improved model interpretability.

[6] The 716 individuals removed from the 2010 data does not include the removal of those who were recruited for the follow up study in 2017-19.

The correlations between total fat mass and the continuous predictors by sex (Figure 3) had the same directionality (all positive, except for age) for both sexes. Across sex, most predictors had a similar relationship with total fat mass, except for age; the age variable was not found to be significantly correlated to total fat mass in males ($r = -0.035$, $p = 0.64$, Figure 3), but it was significantly correlated with total fat in females ($r = -0.24$, $p = 0.00031$, Figure 3). The correlations between total lean mass and the continuous predictors by sex (Figure 4) had the same directionality (all positive, except for age) for both sexes. The relationships between total lean mass and predictors across sex were very similar, with the most notable difference seen in forearm skinfold (male: $r = 0.37$, $p < 0.0001$, female: $r = 0.52$, $p < 0.0001$, Figure 4).

When assessing correlation between VAT mass and the continuous predictors by sex (Figure 5), we saw all positive relationships with no discordance between sex. Notably, age had inverse effects on total fat and lean mass (negative), compared to VAT mass (positive). Although the relationships had the same directionality across sex, males tended to have stronger correlations between VAT mass and the predictors (male r > female r, excluding height and forearm skinfold, Figure 5). This was most evident in the correlations between VAT mass and weight, abdomen circumference, and hip circumference, which were all relatively strong in males yet only moderately strong in females (male: $r = 0.82, 0.91, 0.77$, respectively, female: $r = 0.65, 0.65, 0.58$, respectively, Figure 5). These sex-based differences in relation to body composition, particularly with VAT mass, necessitate further research.

4.2 Model Comparisons

The primary objective of this thesis was to develop accurate MARS models that utilize demographic and anthropometric data to predict total fat, lean, and VAT mass. To evaluate the accuracy of these models, comparison models were made using linear regression and ridge regression techniques. Prior to analysis, MARS was thought to be the optimal approach as its piecewise spline components would be able to incorporate sections of predictors, and work with higher amounts of predictors compared to the alternatives. However, our analyses showed that MARS was only the optimal model when predicting total fat mass in males.

The male MARS total fat mass model had the best RMSE, MAE, and R^2 compared to the alternatives (Table 9). In all other outcomes for both sexes, MARS was not unanimously the best model. The linear regression models were the best option for all other outcomes in both sexes, with the exception of the total fat mass model in males (MARS) and the VAT mass model in females (ridge regression, Table 9). It is important to note that, generally speaking, the MARS, linear regression, and ridge models performed similarly; differences in the metrics were arguably marginal in most cases, and few models were clearly performing better than their alternatives (Table 9). The regression models that mirrored those outlined by Swinburn et al. (1999) were the only models that performed arguably worse than alternatives (Table 9).

The male and female total fat mass models informed by the research of Swinburn et al. (1999), were worse than the best model by 1,000g or more in RMSE and MAE (Table 9). The other total fat mass models for male and female were only off a few 100g in RMSE or MAE from the best models (Table 9). Furthermore, the Swinburn HWSA model proved to be the worst, as its RMSE and MAE were the highest of all the techniques (Table 9). While the linear regression, ridge regression, and MARS models were rather comparable in most cases, the Swinburn based

models were consistently outperformed. These results suggest that Swinburn's total fat mass prediction models, which are some of the only apparent Polynesian based models, are inadequate if one has the means to measure predictors other than BMI, sex, and age.

No single modeling technique was most optimal for predicting total fat, lean, and VAT mass in males and females. The best total fat mass model for males was MARS, while it was linear regression in females (Table 9). The best total lean mass model in both sexes appeared to be linear regression (Table 9). The best VAT mass model was linear regression in males, and ridge regression in females (Table 9). However, the "best" model nomination is only being given based on which model had aggregated the lowest RMSE and MAE, or the highest R^2; the differences in these metrics was often very close when comparing models (Table 9).

The results suggest that linear regression would be the best option if one must pick a technique that would be optimal for predicting all three outcomes in both sexes. The linear regression models were the best in four of six scenarios, spanning the three outcomes for both sexes (Table 9). The linear models were built rather naively, utilizing automated backwards elimination (section 2.2.1) instead of a more epidemiology or biology-informed approach. The predictive accuracy of these models could potentially be improved with a stepwise approach to variable selection, at the cost of biased coefficients. On the other hand, MARS was relatively close to the performance of the linear regression models in the four scenarios (Table 9).

The MARS building procedure could have been adjusted to potentially improve predictive accuracy when compared to alternatives like linear or ridge regression. The MARS models only tested up to 100 maximum number of terms after backwards elimination. The maximum number could have been increased to 200 or more, which may be less parsimonious, but also may improve accuracy of predictions. Furthermore, MARS can be manually adjusted to force a predictor to

enter the model linearly rather than as a basis function. Predictors will be automatically added linearly if the MARS algorithm finds the minimum value to be the best knot point (Milborrow, 2020). However, there were certain predictors that were highly linearly related to the outcomes of total fat, lean, and VAT mass (e.g., total fat mass ~ hip circ., abd. circ., weight, Figure 3). Forcing MARS to add linear predictors such as these into the model may improve predictive accuracy in comparison to alternative linear or ridge regression models.

4.3 2010 Cohort Mass Imputations

This thesis opted to employ the MARS models to impute mass measurements in the 2010 cohort. Although linear regression seemingly edged out MARS on various occasions, the primary objective was to assess the MARS modeling procedure as a predictive tool. In the future, utilization of an appropriate linear model, or even an alternative model such as a neural network may be more optimal. Nonetheless, MARS performed adequately at predicting total fat and lean mass. This can be noted in Figure 12, where sex-based trends in predicted total fat or lean mass were analogous to what was seen in the 2017-19 cohort (Figure 1). Furthermore, Table 10 shows that the imputed total fat and lean mass measures for the 2010 cohort were fairly consistent with the 2017-19, considering the nearly 10-year difference between cohorts.

The imputation of VAT mass was rather inadequate for both sexes of the 2010 cohort. The MARS models for VAT mass estimated multiple negative VAT mass values in the 2010 cohort. The decision was made to truncate the estimated VAT mass in the 2010 cohort to the minimum values measured in the 2017-19 cohort by sex (male min. VAT mass = 120.5g, female , min. VAT mass = 85.8g, Table 10). This decision also corrected imputations that were lower than the

minimum observed in 2017-19, but not implausible (i.e., non-negative and non-zero). This was a simple and inelegant solution that allowed further analyses to be performed with more reasonable VAT mass estimates. Future research will need to address the low VAT mass estimates by improving overall model accuracy, or through a more elegant constraint system. The low VAT mass estimates could also be an artifact of the near decade separation between model building data (2017-19) and the imputation data (2010). While the lower ranges of VAT mass estimations in both sexes from 2010 had issues, the middle and upper ends were reasonably similar to the 2017-19 cohort (Table 10). For the purposes of this thesis, after truncation of VAT estimates, all imputed mass measurements were used in the genetic mixed models.

4.4 *CREBRF* and Total Fat, Lean, and VAT Mass

The A allele of *CREBRF* variant rs373863828 was positively associated with all three mass outcomes for both sexes (Table 11). Previously, Minster et al. (2016) found that the this allele was associated with an higher BMI yet lower odds of type 2 diabetes. One might consider that this paradoxical relationship could be due to higher lean mass solely since additional lean mass would not confer risk of developing type 2 diabetes. Alternatively, if this allele was linked to a decrease in VAT mass, one might expect to see a reduction in odds of developing diabetes. Excess VAT mass is linked to increased insulin resistance and impaired glucose metabolism (Ritchie & Connell, 2007), thus reducing this mass may protect against developing diabetes. However, the results of the genetic mixed models indicated that each copy of the A allele was associated with higher average mass in all three measurements.

Despite uniformly higher average total fat, lean, and VAT mass for each A allele of rs373863828, there are still potentially informative results when comparing the effects across the sexes. While males had similarly higher averages in total fat mass and total lean mass per copy of the A allele (1,552.17g/A, 1,634.90g/A respectively, Table 11), in females the effect of rs373863828 was larger for total fat mass than for total lean mass (1,673.45g/A , 1,049.95g/A respectively, Table 11). These results are not consistent with the hypothesis that *CREBRF* is associated with greater lean mass, explaining the higher average BMI yet lower odds of type 2 diabetes. If females with a copy of the A allele have larger average fat mass than lean mass, then the relationship between rs373863828 and lean mass may not fully explain its protective effect on type 2 diabetes. Future research will have to address the potential sex differences in the effect of *CREBRF* and its paradoxical relationship with BMI and diabetes.

Previous hypotheses that attempt to explicate the *CREBRF* minor A allele's paradoxical relationship with BMI and type 2 diabetes have generally involved specifying fat mass depot. While higher lean mass would confer a higher BMI without risk of developing diabetes, our results in both sexes showed that average difference in total fat mass per A allele were similar to or more than total lean mass. Furthermore, while having less VAT mass would explain the lowered odds of type 2 diabetes, our results show that both sexes also had higher average VAT mass per copy of the A allele. Considering the lack of clarity on the paradoxical nature of *CREBRF* variant rs373863828, future research may prove more successful by examining potential epigenetic factors.

Appendix A Analysis R Script

```
#' ---
#' title: "Thesis Work"
#' author: "Greg Procario"
#' date: "12/28/2020"
#' output: html_document
#' code_folding: hide
#' ---
#'
## ----setup, include=FALSE-----------------------------------------------------------------------------------------------------------------------------------------------------------------
knitr::opts_chunk$set(echo = TRUE)


#' # Libraries
#'
## ------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
knitr::purl("/home/grp20/grp20_explore.Rmd","/home/grp20/Procario_Thesis.Rmd", documentation = 2)
```

```
library(tidyverse)
library(readxl)
library(dplyr)
library(ggplot2)
# Fitting MARS models
library(earth)
# For tuning process
library(caret)
# For variable importance
# VIP didn't end up working with Bagging
library(vip)
# For variable relationships partial dep
library(pdp)
# For plot details
library(ggforce)
# For colors
library(RColorBrewer)
# For correlation plots
library(corrplot)
library(ggpubr)
# For tables
library(arsenal)
library(kableExtra)
```

```
# For bootstrapping
library(boot)
# For Mixed Models
library(coxme)
# For model assessments and printing results
# I don't think I used these in the final cut
library(pander)
library(olsrr)
# For nicer plots
library(gridExtra)
library(ggrepel)
# For VIF calculations
library(car)


#'
#' # Data
#'
## ------------------------------------------------------------------------------------------------------------
-

# Checking excel sheet names
excel_sheets("/home/shared_data/samoa/2018_Samoa_Phenotypes/2017-
2019_Adiposity_Study_Data_Files/All_Data_2020_11_10.xlsx")
```

```
# Reading in both Antrhopometric data and Demographic data
body_data <- read_excel("/home/shared_data/samoa/2018_Samoa_Phenotypes/2017-2019_Adiposity_Study_Data_Files/All_Data_2020_11_10.xlsx",
                        sheet = "Anthropometrics")

demo_data <- read_excel("/home/shared_data/samoa/2018_Samoa_Phenotypes/2017-2019_Adiposity_Study_Data_Files/All_Data_2020_11_10.xlsx",
                        sheet = "Demographic")

# Reading in DXA data
dxa_data <- read_excel("/home/shared_data/samoa/2018_Samoa_Phenotypes/2017-2019_Adiposity_Study_Data_Files/Anthropometry_and_DXA/Soifua Manuia DXA Data January 28 2020.xlsx")

head(body_data)

head(demo_data)

head(dxa_data)

#'
```

```
#' # Data Cleaning I
#'
## ------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------
-
# DXA2 removes rows that are duplicates, using only the "Scantouse==1" & "Corescan==1" versions of the repeated measurements

dxa2_data <-  dxa_data[which(dxa_data$Scantouse==1),]

dxa2_data <- dxa2_data[which(dxa2_data$Corescan==1),]



dxa2_data <- dxa2_data[
            c("IDNumber",
              "TotalFatMass",
              "TotalLeanMass",
              "VATMass"
              )
]
```

```
# Subset anthro measurements and useful characteristics
body2_data <- body_data[
                c("IDNumber",
                  "DecAge",
                  "Age_Group",
                  "Sex",
                  "Genotype_code",
                  "L_Height",
                  "L_Weight",
                  "L_AbdCirc",
                  "L_HipCirc",
                  "L_CalfCirc",
                  "L_TricepSKF",
                  "L_ForearmSKF",
                  "L_SubScapSKF",
                  "L_AbdSKF",
                  "L_SupraSKF"
                 )
]

# Merge anthro and dxa by ID number
```

```
body_dxa <- merge(body2_data,dxa2_data,by="IDNumber")

# Check for duplicate rows, although they should have been removed in prior steps
n_occur <- data.frame(table(body_dxa$IDNumber))
table(n_occur[,2]!=1)

# Make sex categorical
body_dxa$Sex = as.factor(body_dxa$Sex)
body_dxa$Genotype_code = as.factor(body_dxa$Genotype_code)

#'
#' # Data Cleaning II: electric boogaloo
#'
## ------------------------------------------------------------------------------------------------------------
-

summary(body_dxa)

table(body_dxa$L_AbdSKF)
table(body_dxa$L_SubScapSKF)
table(body_dxa$L_SupraSKF)
```

```
# This code is NA checking/max measurement checking
# the three vars below were character vectors because they used >67.0 to denote maximum measurements

# max_subscap<-body_dxa[body_dxa$L_SubScapSKF == ">67.0",]
# max_abdskf<-body_dxa[body_dxa$L_AbdSKF == ">67.0",]
# max_supra<-body_dxa[body_dxa$L_SupraSKF == ">67.0",]
#null_subscap <- body_dxa[body_dxa$L_SubScapSKF == "NULL",]
#null_abdskf <- body_dxa[body_dxa$L_AbdSKF == "NULL",]
#null_supra <- body_dxa[body_dxa$L_SupraSKF == "NULL",]

# replace >67.0 with 67.0, as per Dr. Carlson's request
body_dxa$L_SubScapSKF[body_dxa$L_SubScapSKF == ">67.0"] <- "67.0"
body_dxa$L_SupraSKF[body_dxa$L_SupraSKF == ">67.0"] <- "67.0"
body_dxa$L_AbdSKF[body_dxa$L_AbdSKF == ">67.0"] <- "67.0"


# Quick assessment of continuous x variables
summary(body_dxa$L_Height)

# AbdCirc has a -7777 val, which is supposed to be NA
summary(body_dxa$L_AbdCirc)
```

```
# This replaces all -7777 to NA's
body_dxa[body_dxa == -7777] <- NA

summary(body_dxa$L_HipCirc)

summary(body_dxa$L_CalfCirc)

summary(body_dxa$L_TricepSKF)

summary(body_dxa$L_ForearmSKF)

# SubScap was a character, converted to numeric now that all values are numbers (no more
>67.0)
summary(body_dxa$L_SubScapSKF)
body_dxa$L_SubScapSKF <- as.numeric(body_dxa$L_SubScapSKF)
summary(body_dxa$L_SubScapSKF)

# AbdSkf was also character
summary(body_dxa$L_AbdSKF)
body_dxa$L_AbdSKF <- as.numeric(body_dxa$L_AbdSKF)
summary(body_dxa$L_AbdSKF)

# Supra was also character
```

```
summary(body_dxa$L_SupraSKF)

body_dxa$L_SupraSKF <- as.numeric(body_dxa$L_SupraSKF)

summary(body_dxa$L_SupraSKF)


summary(body_dxa$DecAge)


table(body_dxa$Age_Group)


table(body_dxa$Sex)


table(body_dxa$Genotype_code)









summary(body_dxa)

body_dxa[is.na(body_dxa)]


# Recoding Sex

body_dxa$Sex <- recode(body_dxa$Sex,

                       "0" = "Male",

                       "1" = "Female")
```

```
#'
#' # SexGeno Variable [Did not use]
#'
## ------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------
-
#Creating a Var that combines sex and genotype
# body_dxa$SexGeno <- NA
#
# body_dxa$SexGeno[body_dxa$Sex == 1 & body_dxa$Genotype_code == 0] <- "Female
GG"
# body_dxa$SexGeno[body_dxa$Sex == 1 & body_dxa$Genotype_code == 1] <- "Female
AA"
# body_dxa$SexGeno[body_dxa$Sex == 1 & body_dxa$Genotype_code == 2] <- "Female
AG"
#
# body_dxa$SexGeno[body_dxa$Sex == 0 & body_dxa$Genotype_code == 0] <- "Male
GG"
# body_dxa$SexGeno[body_dxa$Sex == 0 & body_dxa$Genotype_code == 1] <- "Male
AA"
```

```
# body_dxa$SexGeno[body_dxa$Sex == 0 & body_dxa$Genotype_code == 2] <- "Male
AG"


#table(body_dxa$SexGeno)



body_dxa$Genotype_code <- recode(body_dxa$Genotype_code,
                    "0" = "GG",
                    "1" = "AA",
                    "2" = "AG")


#'
#' # Missingness in DXA
#'
## ------------------------------------------------------------------------------------------------------------
-

#Check and save Missingness
missing_dxa <- body_dxa[!complete.cases(body_dxa),]
missing_dxa
# Remove missings, they will cause issue with MARS
body_dxa <- body_dxa[complete.cases(body_dxa),]
```

```
#'
#'
#' # Basic Descriptive Plots and Tables for DXA sample
#'
#' ## Total Fat Mass by Sexgeno
## ------------------------------------------------------------------------------------------------------------
-
# Sexgeno
#
# tfm_sexgen_plot <- ggplot(body_dxa, aes(x = SexGeno, y = TotalFatMass))
# tfm_sexgen_plot + geom_jitter(aes(color = SexGeno),
#                  position=position_jitter(0.2)) +
#     stat_summary(fun.data=mean_sdl,
#                  geom="pointrange",
#                  color="black") +
#     labs(title = "Total Body Fat Mass by Sex and Genotype",
#          x = "Sex and Genotype",
#          y = "Total Fat Mass (g)")

# Sex

tfm_sex_plot <- ggplot(body_dxa, aes(x = Sex, y = TotalFatMass))
```

```
tfm_sex_plot2 <- tfm_sex_plot + geom_jitter(aes(color = Sex),
                position=position_jitter(0.2)) +
   stat_summary(fun.data=mean_sdl,
                geom="pointrange",
                color="black") +
   labs(
       x = "Sex",
       y = "Total Fat Mass (g)")
tfm_sex_plot2


# Geno
# tfm_geno_plot <- ggplot(body_dxa, aes(x = Genotype_code, y = TotalFatMass))
# tfm_geno_plot + geom_jitter(aes(color = Genotype_code),
#                 position=position_jitter(0.2)) +
#    stat_summary(fun.data=mean_sdl,
#                 geom="pointrange",
#                 color="black") +
#    labs(title = "Total Body Fat Mass by Genotype",
#         x = "Genotype",
#         y = "Total Fat Mass (g)")



#'
```

```
#' ## Total Lean Mass by SexGeno
## ------------------------------------------------------------------------------------------------------------
-

# SexGeno
# tlm_sexgen_plot <- ggplot(body_dxa, aes(x = SexGeno, y = TotalFatMass))
# tlm_sexgen_plot + geom_jitter(aes(color = SexGeno),
#                  position=position_jitter(0.2)) +
#     stat_summary(fun.data=mean_sdl,
#                  geom="pointrange",
#                  color="black") +
#     labs(title = "Total Body Lean Mass by Sex and Genotype",
#          x = "Sex and Genotype",
#          y = "Total Lean Mass (g)",
#          caption = "Sex and Genotype")

# Sex

tlm_sex_plot <- ggplot(body_dxa, aes(x = Sex, y = TotalLeanMass))
tlm_sex_plot2 <- tlm_sex_plot + geom_jitter(aes(color = Sex),
                 position=position_jitter(0.2)) +
    stat_summary(fun.data=mean_sdl,
                 geom="pointrange",
```

```
                color="black") +
    labs(
        x = "Sex",
        y = "Total Lean Mass (g)")
tlm_sex_plot2
# Geno

# tlm_geno_plot <- ggplot(body_dxa, aes(x = Genotype_code, y = TotalLeanMass))
# tlm_geno_plot + geom_jitter(aes(color = Genotype_code),
#                 position=position_jitter(0.2)) +
#     stat_summary(fun.data=mean_sdl,
#                 geom="pointrange",
#                 color="black") +
#     labs(title = "Total Body Lean Mass by Genotype",
#         x = "Genotype",
#         y = "Total Fat Mass (g)")

#'
#' ## Visceral Mass
#'
## ----------------------------------------------------------------------------------------------------
```

```
# Sexgeno
# vat_sexgen_plot <- ggplot(body_dxa, aes(x = SexGeno, y = VATMass))
# vat_sexgen_plot + geom_jitter(aes(color = SexGeno),
#                  position=position_jitter(0.2)) +
#     stat_summary(fun.data=mean_sdl,
#                  geom="pointrange",
#                  color="black") +
#     labs(title = "Visceral Adipose Tissue Mass by Sex and Genotype",
#          x = "Sex and Genotype",
#          y = "VAT Mass (g)")
# Sex

vat_sex_plot <- ggplot(body_dxa, aes(x = Sex, y = VATMass))
vat_sex_plot2 <- vat_sex_plot + geom_jitter(aes(color = Sex),
                 position=position_jitter(0.2)) +
   stat_summary(fun.data=mean_sdl,
                geom="pointrange",
                color="black") +
   labs(
        x = "Sex",
        y = "VAT Mass (g)")
vat_sex_plot2
# Geno
```

```
# vat_geno_plot <- ggplot(body_dxa, aes(x = Genotype_code, y = VATMass))
# vat_geno_plot + geom_jitter(aes(color = Genotype_code),
#                  position=position_jitter(0.2)) +
#     stat_summary(fun.data=mean_sdl,
#                  geom="pointrange",
#                  color="black") +
#     labs(title = "Visceral Adipose Tissue Mass by Genotype",
#          x = "Genotype",
#          y = "VAT Mass (g)")




# Compute descriptive statistics by groups
avgsd <- function(data){
   avg<-round(mean(data),1)
   sd<-round(sd(data),1)
return(c(avg,sd))
}



mfat_mu<- subset(body_dxa$TotalFatMass, body_dxa$Sex=="Male") %>%
```

```
 avgsd

mfat_mu

ffat_mu <- subset(body_dxa$TotalFatMass, body_dxa$Sex=="Female") %>% avgsd


mlean_mu <-subset(body_dxa$TotalLeanMass, body_dxa$Sex=="Male") %>% avgsd


flean_mu <- subset(body_dxa$TotalLeanMass, body_dxa$Sex=="Female") %>% avgsd


mvat_mu <- subset(body_dxa$VATMass, body_dxa$Sex=="Male") %>% avgsd


fvat_mu <- subset(body_dxa$VATMass, body_dxa$Sex=="Female") %>% avgsd




desc_table <- data.frame("Male Mass(g)"=
                           c(mfat_mu,mlean_mu,mvat_mu),
                         "Female Mass(g)"=
                           c(ffat_mu,flean_mu,fvat_mu),
                          check.names = F
                           )
```

```
# Summary table plot
stable.p.1 <- ggtexttable(desc_table2, rows = c("Total Fat (Avg)","sd","Total Lean (Avg)",
"sd", "VAT (Avg)","sd"),
                          theme = ttheme("minimal", base_size = 1))

# plot everything
outcome_plots <- ggarrange(tfm_sex_plot2, tlm_sex_plot2, vat_sex_plot2, stable.p.1,
          labels=c("A","B","C"),
          common.legend = TRUE, legend = "bottom")
outcome_plots

annotate_figure(outcome_plots, top="Mass Outcomes by Sex, 2017-19 Cohort")



#'
#' ## Correlation Plots
## ----fig.width=10,fig.height=9-------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------
# save correlation for quant vars
corr_dxa <- cor(body_dxa[,c(2,6:15)], method = "pearson")
```

```
corr_dxa
colnames(corr_dxa) <- c("Age",
                        "Height",
                        "Weight",
                        "Ab.Cir.",
                        "Hip Cir.",
                        "Calf Cir.",
                        "Tri. Skf.",
                        "For. Skf.",
                        "Sub. Skf.",
                        "Ab. Skf.",
                        "Sup. Skf."
                        )
# Plot Correlation matrix
corrplot(corr_dxa, type="upper", order="hclust",
         col=brewer.pal(n=8, name="RdBu"))
cplot <- corrplot.mixed(corr_dxa, lower.col = "black")
cplot
min(abs(cplot))



#'
#' ## Total Fat Mass Corr. Plots: Age, Height, Weight
```

```
#'
## ---- figure.width=16, figure.height=16----------------------------------------------------
-------------------------------------------------------------------------------------------------
---------
# Scatter plots broken down by sex with pearson corrs.
# TFM by Age, weight, height
tfm_age <- ggplot(data = body_dxa, aes(x = DecAge, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Age (years)",
            y = "Total Fat Mass (g)") +
           stat_cor(aes(color = Sex))
tfm_age



# Height
tfm_height <- ggplot(data = body_dxa, aes(x = L_Height, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
```

```
        geom_smooth(method = "lm",
                    se = F,
                    aes(color = factor(Sex))) +
        labs(
         x = "Height (cm)",
         y = "Total Fat Mass (g)") +
        stat_cor(aes(color = Sex))
tfm_height



# Weight
tfm_weight <- ggplot(data = body_dxa, aes(x = L_Weight, y = TotalFatMass)) +
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
                    se = F,
                    aes(color = factor(Sex))) +
        labs(
         x = "Weight (kg)",
         y = "Total Fat Mass (g)") +
        stat_cor(aes(color = Sex))
tfm_weight
```

```
#'
#' ## Total Fat Mass Corr. Plots: Abdom., Hip, Calf Circum.
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-
# Abdominal circ
tfm_abd <- ggplot(data = body_dxa, aes(x = L_AbdCirc, y = TotalFatMass)) +
    geom_point(aes(shape = factor(Sex))) +
    geom_point(aes(color = factor(Sex))) +
    geom_smooth(method = "lm",
        se = F,
        aes(color = factor(Sex))) +
   labs(
    x = "Abdomen Circ. (cm)",
    y = "Total Fat Mass (g)") +
   stat_cor(aes(color = Sex))
```

```
tfm_abd


# Hip Circ
tfm_hip <- ggplot(data = body_dxa, aes(x = L_HipCirc, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Hip Circ. (cm)",
            y = "Total Fat Mass (g)") +
           stat_cor(aes(color = Sex))
tfm_hip


# Calf Circ
tfm_calf <- ggplot(data = body_dxa, aes(x = L_CalfCirc, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
```

```
            x = "Calf Circ. (cm)",
            y = "Total Fat Mass (g)") +
          stat_cor(aes(color = Sex))
tfm_calf



#'
#' ## Total Fat Mass Corr. Plots: Tricep, Forearm, Subscap, Supra SKFs
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-
# Tricep skf
tfm_tri <- ggplot(data = body_dxa, aes(x = L_TricepSKF, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
          labs(
            x = "Triceps Skinfold (mm)",
            y = "Total Fat Mass (g)") +
          stat_cor(aes(color = Sex))
```

```
tfm_tri


# Forearm skf
tfm_fore <- ggplot(data = body_dxa, aes(x = L_ForearmSKF, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Forearm Skinfold (mm)",
            y = "Total Fat Mass (g)") +
           stat_cor(aes(color = Sex))
tfm_fore


# Subscap skf
tfm_sub <- ggplot(data = body_dxa, aes(x = L_SubScapSKF, y = TotalFatMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
```

```
        x = "Subscapular Skifold (mm)",
        y = "Total Fat Mass (g)") +
      stat_cor(aes(color = Sex))
tfm_sub

# Supra skf

tfm_sup <- ggplot(data = body_dxa, aes(x = L_SupraSKF, y = TotalFatMass)) +
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
      labs(
        x = "Suprailiac Skinfold (mm)",
        y = "Total Fat Mass (g)") +
      stat_cor(aes(color = Sex))
tfm_sup

# Abd SKF

tfm_abs <- ggplot(data = body_dxa, aes(x = L_AbdSKF, y = TotalFatMass)) +
        geom_point(aes(shape = factor(Sex))) +
```

```
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Abdomen Skinfold (mm)",
        y = "Total Fat Mass (g)") +
        stat_cor(aes(color = Sex))
    tfm_abs

    # Arrange all plots
    tfm_plots <- ggarrange(tfm_age, tfm_height, tfm_weight, tfm_abd, tfm_hip,
tfm_calf,tfm_tri, tfm_fore, tfm_sub, tfm_sup , tfm_abs,
        labels=c("A","B","C","D","E","F","G","H","I","J","K"),
        common.legend = TRUE, legend = "bottom")

    annotate_figure(tfm_plots, top="Total Fat Mass by Continuous Predictors")

    #'
    #'
    #' ## Total Lean Mass Corr. Plots: Age, Height, Weight
    #'
```

```
## ------------------------------------------------------------------------------------------------------------------
-
```

```
# Scatter plots broken down by sex with pearson corrs.
# TLM by Age, weight, height


tlm_age <- ggplot(data = body_dxa, aes(x = DecAge, y = TotalLeanMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                 se = F,
                 aes(color = factor(Sex))) +
           labs(
            x = "Age (years)",
            y = "Total Lean Mass (g)") +
           stat_cor(aes(color = Sex))
tlm_age



# Height
tlm_height <- ggplot(data = body_dxa, aes(x = L_Height, y = TotalLeanMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
```

```
        geom_smooth(method = "lm",
                    se = F,
                    aes(color = factor(Sex))) +
        labs(
         x = "Height (cm)",
         y = "Total Lean Mass (g)") +
        stat_cor(aes(color = Sex))
tlm_height


# Weight
tlm_weight <- ggplot(data = body_dxa, aes(x = L_Weight, y = TotalLeanMass)) +
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
                    se = F,
                    aes(color = factor(Sex))) +
        labs(
         x = "Weight (kg)",
         y = "Total Lean Mass (g)") +
        stat_cor(aes(color = Sex))
tlm_weight
```

```
#'
#' ## Total Lean Mass Corr. Plots: Abdom., Hip, Calf Circum.
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-
# Abdominal circ
tlm_abd <- ggplot(data = body_dxa, aes(x = L_AbdCirc, y = TotalLeanMass)) +
    geom_point(aes(shape = factor(Sex))) +
    geom_point(aes(color = factor(Sex))) +
    geom_smooth(method = "lm",
        se = F,
        aes(color = factor(Sex))) +
   labs(
    x = "Abdomen Circ. (cm)",
    y = "Total Lean Mass (g)") +
   stat_cor(aes(color = Sex))
tlm_abd

# Hip Circ
tlm_hip <- ggplot(data = body_dxa, aes(x = L_HipCirc, y = TotalLeanMass)) +
    geom_point(aes(shape = factor(Sex))) +
```

```
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Hip Circ. (cm)",
        y = "Total Lean Mass (g)") +
        stat_cor(aes(color = Sex))
tlm_hip


# Calf Circ
tlm_calf <- ggplot(data = body_dxa, aes(x = L_CalfCirc, y = TotalLeanMass)) +
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Calf Circ. (cm)",
        y = "Total Lean Mass (g)") +
        stat_cor(aes(color = Sex))
tlm_calf
```

```
#'
#' ## Total Lean Mass Corr. Plots: Tricep, Forearm, Subscap, Supra SKFs
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-
# Tricep skf
tlm_tri <- ggplot(data = body_dxa, aes(x = L_TricepSKF, y = TotalLeanMass)) +
    geom_point(aes(shape = factor(Sex))) +
    geom_point(aes(color = factor(Sex))) +
    geom_smooth(method = "lm",
        se = F,
        aes(color = factor(Sex))) +
   labs(
    x = "Triceps Skinfold (mm)",
    y = "Total Lean Mass (g)") +
   stat_cor(aes(color = Sex))
tlm_tri

# Forearm skf
tlm_fore <- ggplot(data = body_dxa, aes(x = L_ForearmSKF, y = TotalLeanMass)) +
    geom_point(aes(shape = factor(Sex))) +
```

```
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Forearm Skinfold (mm)",
        y = "Total Lean Mass (g)") +
        stat_cor(aes(color = Sex))
tlm_fore


# Subscap skf
tlm_sub <- ggplot(data = body_dxa, aes(x = L_SubScapSKF, y = TotalLeanMass)) +
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Subscapular Skinfold (mm)",
        y = "Total Lean Mass (g)") +
        stat_cor(aes(color = Sex))
tlm_sub
```

```
# Supra skf


tlm_sup <- ggplot(data = body_dxa, aes(x = L_SupraSKF, y = TotalLeanMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Suprailiac Skinfold (mm)",
            y = "Total Lean Mass (g)") +
           stat_cor(aes(color = Sex))
tlm_sup
# Abd skf
tlm_abs <- ggplot(data = body_dxa, aes(x = L_AbdSKF, y = TotalLeanMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Abdomen Skinfold (mm)",
            y = "Total Lean Mass (g)") +
```

```
            stat_cor(aes(color = Sex))
        tlm_abs



        # Arrange all plots
        tlm_plots <- ggarrange(tlm_age, tlm_height, tlm_weight, tlm_abd, tlm_hip,
tlm_calf,tlm_tri, tlm_fore, tlm_sub, tlm_sup , tlm_abs,
                labels=c("A","B","C","D","E","F","G","H","I","J","K"),
                common.legend = TRUE, legend = "bottom")

        annotate_figure(tlm_plots, top="Total Lean Mass by Continuous Predictors")



        #'
        #' ## VAT Mass Corr. Plots: Age, Height, Weight
        #'
        ## ----------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------
-
        # Scatter plots broken down by sex with pearson corrs.
        # VAT by Age, weight, height

        vat_age <- ggplot(data = body_dxa, aes(x = DecAge, y = VATMass)) +
```

```
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Age (years)",
        y = "VAT Mass (g)") +
        stat_cor(aes(color = Sex))
vat_age


# Height
vat_height <- ggplot(data = body_dxa, aes(x = L_Height, y = VATMass)) +
        geom_point(aes(shape = factor(Sex))) +
        geom_point(aes(color = factor(Sex))) +
        geom_smooth(method = "lm",
            se = F,
            aes(color = factor(Sex))) +
        labs(
        x = "Height (cm)",
        y = "VAT Mass (g)") +
        stat_cor(aes(color = Sex))
```

```
vat_height



# Weight
vat_weight <- ggplot(data = body_dxa, aes(x = L_Weight, y = VATMass)) +
              geom_point(aes(shape = factor(Sex))) +
              geom_point(aes(color = factor(Sex))) +
              geom_smooth(method = "lm",
                    se = F,
                    aes(color = factor(Sex))) +
             labs(
              x = "Weight (kg)",
              y = "VAT Mass (g)") +
             stat_cor(aes(color = Sex))
vat_weight



#'
#' ## VAT Mass Corr. Plots: Abdom., Hip, Calf Circum.
#'
## ----------------------------------------------------------------------------------------------------
```

```
# Abdominal circ
vat_abd <- ggplot(data = body_dxa, aes(x = L_AbdCirc, y = VATMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Abdomen Circ. (cm)",
            y = "VAT Mass (g)") +
           stat_cor(aes(color = Sex))
vat_abd


# Hip Circ
vat_hip <- ggplot(data = body_dxa, aes(x = L_HipCirc, y = VATMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Hip Circ. (cm)",
            y = "VAT Mass (g)") +
```

```
            stat_cor(aes(color = Sex))
        vat_hip


        # Calf Circ
        vat_calf <- ggplot(data = body_dxa, aes(x = L_CalfCirc, y = VATMass)) +
             geom_point(aes(shape = factor(Sex))) +
             geom_point(aes(color = factor(Sex))) +
             geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
            labs(
             x = "Calf Circ. (cm)",
             y = "VAT Mass (g)") +
            stat_cor(aes(color = Sex))
        vat_calf



        #'
        #' ## VAT Mass Corr. Plots: Tricep, Forearm, Subscap, Supra SKFs
        #'
        ## ----------------------------------------------------------------------------------------------------
-
```

```
# Tricep skf
vat_tri <- ggplot(data = body_dxa, aes(x = L_TricepSKF, y = VATMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Triceps Skinfold (mm)",
            y = "VAT Mass (g)") +
           stat_cor(aes(color = Sex))
vat_tri


# Forearm skf
vat_fore <- ggplot(data = body_dxa, aes(x = L_ForearmSKF, y = VATMass)) +
            geom_point(aes(shape = factor(Sex))) +
            geom_point(aes(color = factor(Sex))) +
            geom_smooth(method = "lm",
                  se = F,
                  aes(color = factor(Sex))) +
           labs(
            x = "Forearm Skinfold (mm)",
            y = "VAT Mass (g)") +
```

```
        stat_cor(aes(color = Sex))
vat_fore


# Subscap skf
vat_sub <- ggplot(data = body_dxa, aes(x = L_SubScapSKF, y = VATMass)) +
         geom_point(aes(shape = factor(Sex))) +
         geom_point(aes(color = factor(Sex))) +
         geom_smooth(method = "lm",
              se = F,
              aes(color = factor(Sex))) +
        labs(
         x = "Subscapular Skinfold (mm)",
         y = "VAT Mass (g)") +
        stat_cor(aes(color = Sex))
vat_sub


# Supra skf


vat_sup <- ggplot(data = body_dxa, aes(x = L_SupraSKF, y = VATMass)) +
         geom_point(aes(shape = factor(Sex))) +
         geom_point(aes(color = factor(Sex))) +
         geom_smooth(method = "lm",
              se = F,
```

```
                aes(color = factor(Sex))) +
            labs(
             x = "Suprailiac Skinfold (mm)",
             y = "VAT Mass (g)") +
            stat_cor(aes(color = Sex))
        vat_sup

        # Abd skf
        vat_abs <- ggplot(data = body_dxa, aes(x = L_AbdSKF, y = VATMass)) +
             geom_point(aes(shape = factor(Sex))) +
             geom_point(aes(color = factor(Sex))) +
             geom_smooth(method = "lm",
                 se = F,
                 aes(color = factor(Sex))) +
            labs(
             x = "Abdomen Skinfold (mm)",
             y = "VAT Mass (g)") +
            stat_cor(aes(color = Sex))
        vat_abs

        # Arrange all plots
        vat_plots <- ggarrange(vat_age, vat_height, vat_weight, vat_abd, vat_hip, vat_calf,vat_tri,
vat_fore, vat_sub, vat_sup , vat_abs,
```

```
        labels=c("A","B","C","D","E","F","G","H","I","J","K"),
        common.legend = TRUE, legend = "bottom")

    annotate_figure(vat_plots, top="Visceral Adipose Tissue (VAT) Mass by Continuous Predictors")



    #'
    #' ## Descriptive Tables
    #'
    ## ----results="asis"-----------------------------------------------------------------------------------------------------------------

    # Sex stratified stats
    table_one <- tableby(Sex ~ DecAge +
                         Genotype_code +
                         L_Height +
                         L_Weight +
                         L_AbdCirc +
                         L_HipCirc +
                         L_CalfCirc +
                         L_TricepSKF +
                         L_ForearmSKF +
```

```
                L_SubScapSKF +
                L_SupraSKF +
                L_AbdSKF +
                TotalLeanMass +
                TotalFatMass +
                VATMass,
              data = body_dxa)

labels <- list(DecAge = "Age (yrs)",
               Genotype_code = "Genotype",
               L_Height = "Height (cm)",
               L_Weight = "Weight (kg)",
               L_AbdCirc = "Abdomen Circum. (cm)" ,
               L_HipCirc = "Hip Circum. (cm)",
               L_CalfCirc = "Calf Circum. (cm)",
               L_TricepSKF = "Tricep Skin Fold (mm)",
               L_ForearmSKF = "Forearm Skin Fold (mm)",
               L_SubScapSKF = "Subscapular Skin Fold (mm)",
               L_SupraSKF = "Suprailiac Skin Fold(mm)",
               L_AbdSKF = "Abdomen Skin Fold (mm)",
               TotalLeanMass = "Total Lean Mass (g)",
               TotalFatMass = "Total Fat Mass (g)",
               VATMass = "VAT Mass (g)")
```

```
summary(table_one,
        labelTranslations = labels,
        pfootnote = TRUE)



# Sex and Genotype
table_two <- tableby(SexGeno ~ DecAge +
                     L_Height +
                     L_AbdCirc +
                     L_HipCirc +
                     L_CalfCirc +
                     L_TricepSKF +
                     L_ForearmSKF +
                     L_SubScapSKF +
                     L_SupraSKF +
                     TotalLeanMass +
                     TotalFatMass +
                     VATMass,
                   data = body_dxa)
summary(table_two,
        labelTranslations = labels,
        pfootnote = TRUE)
```

```
#'
#' # TEST Model I [Not Using this]
#'
## ------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
-
# Checking to see if the package works with data
mars1 <- earth(TotalFatMass ~
                 L_Height +
                 L_AbdCirc +
                 L_HipCirc +
                 L_CalfCirc +
                 L_TricepSKF +
                 L_ForearmSKF +
                 L_SubScapSKF +
                 L_AbdSKF +
                 L_SupraSKF +
                 Sex +
                 Genotype_code +
                 SexGeno +
                 DecAge,
```

```
            data = body_dxa)


# Do NAs need to be dropped?
bodydxa_noNA <- body_dxa %>% drop_na()


# NAs need to be dropped, I don't know why I named this dmars1
dmars1 <- earth(TotalFatMass ~
                  L_Height +
                  L_AbdCirc +
                  L_HipCirc +
                  L_CalfCirc +
                  L_TricepSKF +
                  L_ForearmSKF +
                  L_SubScapSKF +
                  L_AbdSKF +
                  L_SupraSKF +
                  Sex +
                  Genotype_code +
                  SexGeno +
                  DecAge,
                data = bodydxa_noNA)


# Model summary
```

```
print(dmars1)


# Coefficients
summary(dmars1) %>% .$coefficients


# Plots
plot(dmars1, which = 1)



#'
#' # TEST Tuning Models and CV [Not Using This]
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-

# Creating a grid for tuning
# Grid contains the degree of interactions, and number of terms to be used in model
# Looking for optimal combination to minimize prediction error
hyper_grid <- expand.grid(
  degree = 1:3,
  nprune = seq(2, 100, length.out = 10) %>% floor()
)
```

```
smp_size <- floor(0.75 * nrow(bodydxa_noNA))

set.seed(1212021)

train_ind <- sample(seq_len(nrow(bodydxa_noNA)), size = smp_size)

train <- bodydxa_noNA[train_ind, ]
test <- bodydxa_noNA[-train_ind, ]

cv_mars <- train(
    x = subset(train,
        select = c(
        L_Height,
        L_AbdCirc,
        L_HipCirc,
        L_CalfCirc,
        L_TricepSKF,
        L_ForearmSKF,
        L_SubScapSKF,
        L_AbdSKF,
        L_SupraSKF,
        Sex,
        Genotype_code,
```

```
        SexGeno,
        DecAge
        )),
    y = train$TotalFatMass,
    method = "earth",
    metric = "RMSE",
    trControl = trainControl(method = "cv",
                             number = 10),
    tuneGrid = hyper_grid
)

cv_mars

cv_mars$bestTune

cv_mars$results %>%
    filter(nprune == cv_mars$bestTune$nprune, degree == cv_mars$bestTune$degree)


cv_mars$finalModel %>%
  coef() %>%
  broom::tidy() %>%
  filter(stringr::str_detect(names, "\\*"))
```

```
#'
#' # TEST Plots [Not Using This]
#'
## ------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-
ggplot(cv_mars)

cv_mars$resample



# variable importance plots
p1 <- vip(cv_mars, num_features = 40, geom = "point", value = "gcv") + ggtitle("GCV")
p2 <- vip(cv_mars, num_features = 40, geom = "point", value = "rss") + ggtitle("RSS")

gridExtra::grid.arrange(p1, p2, ncol = 2)




#'
#'
```

```
#' # TEST Model TLM and VAT [Not Using This]
#'
## ------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------
-

#TLM
set.seed(1212021)


cv_tlm_mars <- train(
   x = subset(train,
      select = c(
      L_Height,
      L_AbdCirc,
      L_HipCirc,
      L_CalfCirc,
      L_TricepSKF,
      L_ForearmSKF,
      L_SubScapSKF,
      L_AbdSKF,
      L_SupraSKF,
      Sex,
      Genotype_code,
      SexGeno,
```

```
        DecAge
        )),
    y = train$TotalLeanMass,
    method = "earth",
    metric = "RMSE",
    trControl = trainControl(method = "cv",
                             number = 10),
    tuneGrid = hyper_grid
)

cv_tlm_mars$bestTune

cv_tlm_mars$results %>%
    filter(nprune == cv_mars$bestTune$nprune, degree == cv_mars$bestTune$degree)



cv_tlm_mars$finalModel %>%
  coef() %>%
  broom::tidy() %>%
  filter(stringr::str_detect(names, "\\*"))

# VAT
```

```
cv_vat_mars <- train(
    x = subset(train,
        select = c(
        L_Height,
        L_AbdCirc,
        L_HipCirc,
        L_CalfCirc,
        L_TricepSKF,
        L_ForearmSKF,
        L_SubScapSKF,
        L_AbdSKF,
        L_SupraSKF,
        Sex,
        Genotype_code,
        SexGeno,
        DecAge
        )),
    y = train$VATMass,
    method = "earth",
    metric = "RMSE",
    trControl = trainControl(method = "cv",
                             number = 10),
    tuneGrid = hyper_grid
```

```
)

cv_vat_mars$bestTune

cv_vat_mars$results %>%
   filter(nprune == cv_mars$bestTune$nprune, degree == cv_mars$bestTune$degree)



cv_vat_mars$finalModel %>%
  coef() %>%
  broom::tidy() %>%
  filter(stringr::str_detect(names, "\\*"))

#'
#'
#' # RMSE and MAE Functions
#'
## ------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------
-

# NEED to figure out how to appropriately test these models
# Created Root Mean Sq Error function
RMSE = function(m, o){
```

```
  sqrt(mean((m - o)^2))
}

test_resid <- as.data.frame(predict(cv_mars$finalModel, test))
colnames(test_resid)[1] <- "prediction"

# Mean Abs error
RMSE(test_resid$prediction, test$TotalFatMass)

mae <- function(x,y)
  {
  mean(abs(x-y))
  }



#'
#'
#'
#' # Stratified Models
#'
```

```
## ------------------------------------------------------------------------------------------------------------
-
# Stratify dxa data by sex
male_dxa <- body_dxa[which(body_dxa$Sex == "Male"),]

female_dxa <- body_dxa[which(body_dxa$Sex == "Female"),]



# BMI calculator, since bmi was not in the DXA dataset, I probably dropped it by accident in the first steps
# Height was in cm for our study, so I multiplied the bmi equation by 10000, since its supposed to be m^2
male_dxa$bmi <- round(10000*male_dxa$L_Weight/(male_dxa$L_Height^2),1)
female_dxa$bmi <- round(10000*female_dxa$L_Weight/(female_dxa$L_Height^2),1)



set.seed(1212021)

# making test and train splits
# Don't know if I'm using these splits

male_n <- floor(0.75 * nrow(male_dxa))
```

```
female_n <- floor(0.75 * nrow(female_dxa))

train_ind <- sample(seq_len(nrow(male_dxa)), size = male_n)

train_male <- male_dxa[train_ind, ]
test_male <- male_dxa[-train_ind, ]

train_ind <- sample(seq_len(nrow(female_dxa)), size = female_n)

train_female <- female_dxa[train_ind, ]
test_female <- female_dxa[-train_ind, ]

#'
#' # Male Fat Mass Bagged MARS Model
#'
## ---- fig.width=10,fig.height=8-----------------------------------------------------------------
------------------------------------------------------------------------------------------------------------------
-------

set.seed(2102021)

# Model built for Males predicting TFM
```

```
male_fat_mars <- train( x = subset(male_dxa,
        select = c(
        L_Weight,
        L_Height,
        L_AbdCirc,
        L_HipCirc,
        L_CalfCirc,
        L_TricepSKF,
        L_ForearmSKF,
        L_SubScapSKF,
        L_AbdSKF,
        L_SupraSKF,
        DecAge
        )),
    y = male_dxa$TotalFatMass,
    method = 'bagEarth',
    trControl = trainControl(method = "cv",
                             number = 10),
    tuneGrid = hyper_grid,
    keepX=F,
    B = 50)

# This code pulls the equation for the model *Warning* It is usually very long
```

```
#format(male_fat_mars$finalModel)

male_fat_mars$bestTune

# Use this to pull best model, and create plots

summary(male_fat_mars$finalModel)

male_fat_mars$results


male_fat_mars$results %>%

   filter(nprune == male_fat_mars$bestTune$nprune,

          degree == male_fat_mars$bestTune$degree) %>%

           kable(allign='center')


male_fat_mars


# Plots

ggplot(male_fat_mars)


show(male_fat_mars)



# This is akin to a partial dependence plot

# It displays:degree1 variables in additive (non interaction) terms.

# degree2 variables appearing together in interaction terms
```

```
# degree1 plot is generated by plotting the predicted response as the variable changes.
# degree2 plot is generated by plotting the predicted response as two variables are changed
# In both background variables are held fixed at their median values
plotmo(male_fat_mars)



# RMSE and MAE on orignial male sample
test_resid <- as.data.frame(predict(male_fat_mars$finalModel, male_dxa))
colnames(test_resid)[1] <- "prediction"

RMSE(test_resid$prediction, male_dxa$TotalFatMass)
mae(test_resid$prediction, male_dxa$TotalFatMass)




#'
#' # Male Lean Mass Bagged MARS Model
#'
## ---- fig.width=10,fig.height=8----------------------------------------------------------------
```

```
set.seed(2102021)
# Model built for Males predicting TLM
male_lean_mars <- train( x = subset(male_dxa,
      select = c(
      L_Weight,
      L_Height,
      L_AbdCirc,
      L_HipCirc,
      L_CalfCirc,
      L_TricepSKF,
      L_ForearmSKF,
      L_SubScapSKF,
      L_AbdSKF,
      L_SupraSKF,
      DecAge
      )),
   y = male_dxa$TotalLeanMass,
   method = 'bagEarth',
   trControl = trainControl(method = "cv",
                            number = 10),
   tuneGrid = hyper_grid,
   keepX=F,
   B = 50)
```

```
# This code pulls the equation for the model *Warning* It is usually very long
#format(male_lean_mars$finalModel)

male_lean_mars$bestTune
# Use this to pull best model, and create plots
male_lean_mars$finalModel

male_lean_mars$results %>%
    filter(nprune == male_lean_mars$bestTune$nprune,
           degree == male_lean_mars$bestTune$degree) %>%
            kable(allign='center')
male_lean_mars$results

# Plots
ggplot(male_lean_mars)

male_lean_mars$resample

show(male_lean_mars)
# variable importance plots
plotmo(male_lean_mars)
```

```
#'
#' # Male VAT Mass Bagged MARS Model
#'
## ----  fig.width=10,fig.height=8----------------------------------------------------------------
-------------------------------------------------------------------------------------------------
-------

set.seed(2102021)
# Model built for Males predicting vat
male_vat_mars <- train( x = subset(male_dxa,
    select = c(
    L_Weight,
    L_Height,
    L_AbdCirc,
    L_HipCirc,
    L_CalfCirc,
    L_TricepSKF,
    L_ForearmSKF,
```

```
      L_SubScapSKF,
      L_AbdSKF,
      L_SupraSKF,
      DecAge
      )),
   y = male_dxa$VATMass,
   method = 'bagEarth',
   trControl = trainControl(method = "cv",
                            number = 10),
   tuneGrid = hyper_grid,
   keepX=F,
   B = 50)

# This code pulls the equation for the model *Warning* It is usually very long
#format(male_vat_mars$finalModel)

male_vat_mars$bestTune
# Use this to pull best model, and create plots
male_vat_mars$finalModel

male_vat_mars$results %>%
   filter(nprune == male_vat_mars$bestTune$nprune,
          degree == male_vat_mars$bestTune$degree) %>%
```

```
            kable(allign='center')
        male_vat_mars$results
        # Plots
        ggplot(male_vat_mars)


        male_vat_mars$resample


        show(male_vat_mars)
        # variable importance plots
        plotmo(male_vat_mars)




        #'
        #' # Female Fat Mass Bagged MARS Model
        #'
        ## ----  fig.width=10,fig.height=8----------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------
-------
        # Find row of participant 2472, who had very extreme measurements
        which(female_dxa$IDNumber==2472)
        # Make a data frame without the extreme measurements
```

```
#female_dxa_no2472 <- female_dxa[-197,]


set.seed(2102021)
# Model built for females predicting TFM
female_fat_mars <- train( x = subset(female_dxa,
      select = c(
      L_Weight,
      L_Height,
      L_AbdCirc,
      L_HipCirc,
      L_CalfCirc,
      L_TricepSKF,
      L_ForearmSKF,
      L_SubScapSKF,
      L_AbdSKF,
      L_SupraSKF,
      DecAge
      )),
   y = female_dxa$TotalFatMass,
   method = 'bagEarth',
   trControl = trainControl(method = "cv",
                            number = 10),
   tuneGrid = hyper_grid,
```

```
   keepX=F,
   B = 50)



female_fat_mars$results %>%
   filter(nprune == female_fat_mars$bestTune$nprune,
          degree == female_fat_mars$bestTune$degree) %>%
           kable(allign='center')
female_fat_mars$results

# This code pulls the equation for the model *Warning* It is usually very long
#format(female_fat_mars$finalModel)

ggplot(female_fat_mars)

plotmo(female_fat_mars)

# Run the MARS without extreme observations using the no2472 data
#female_fat_mars2 <- train( x = subset(female_dxa_no2472,
       #select = c(
       #L_Height,
       #L_AbdCirc,
       #L_HipCirc,
```

```
      #L_CalfCirc,
      #L_TricepSKF,
      #L_ForearmSKF,
      #L_SubScapSKF,
      #L_AbdSKF,
      #L_SupraSKF,
      #DecAge
      #)),
   #y = female_dxa_no2472$TotalFatMass,
   #method = 'bagEarth',
   #trControl = trainControl(method = "cv",
                        #number = 10),
   #tuneGrid = hyper_grid,
   #keepX=F,
   #B = 50)



# female_fat_mars2$results %>%
#    filter(nprune == female_fat_mars2$bestTune$nprune,
#           degree == female_fat_mars2$bestTune$degree) %>%
#           kable(allign='center')
```

```
#ggplot(female_fat_mars2)



# Removing the extreme data point did not affect the model very much
# We will just leave this person in the dataset

#plotmo(female_fat_mars2)



#'
#' # Female Lean Mass Bagged MARS Model
#'
## ----  fig.width=10,fig.height=8----------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------

set.seed(2102021)
# Model built for females predicting TLM
female_lean_mars <- train( x = subset(female_dxa,
      select = c(
      L_Weight,
      L_Height,
      L_AbdCirc,
```

```
        L_HipCirc,
        L_CalfCirc,
        L_TricepSKF,
        L_ForearmSKF,
        L_SubScapSKF,
        L_AbdSKF,
        L_SupraSKF,
        DecAge
        )),
    y = female_dxa$TotalLeanMass,
    method = 'bagEarth',
    trControl = trainControl(method = "cv",
                             number = 10),
    tuneGrid = hyper_grid,
    keepX=F,
    B = 50)
# This code pulls the equation for the model *Warning* It is usually very long
#format(female_lean_mars$finalModel)
female_lean_mars$results %>%
    filter(nprune == female_lean_mars$bestTune$nprune,
           degree == female_lean_mars$bestTune$degree) %>%
            kable(allign='center')
```

```
female_lean_mars$results


ggplot(female_lean_mars)


plotmo(female_lean_mars)


vip(female_lean_mars)



#'
#' # Female VAT Mass Bagged MARS Model
#'
## ---- fig.width=10,fig.height=8----------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------

set.seed(2102021)
# Model built for Males predicting TFM
female_vat_mars <- train( x = subset(female_dxa,
      select = c(
      L_Weight,
      L_Height,
      L_AbdCirc,
```

```
        L_HipCirc,
        L_CalfCirc,
        L_TricepSKF,
        L_ForearmSKF,
        L_SubScapSKF,
        L_AbdSKF,
        L_SupraSKF,
        DecAge
        )),
    y = female_dxa$VATMass,
    method = 'bagEarth',
    trControl = trainControl(method = "cv",
                             number = 10,
                             summaryFunction=defaultSummary),
    tuneGrid = hyper_grid,
    keepX=F,
    B = 50)
# This code pulls the equation for the model *Warning* It is usually very long
#format(female_vat_mars$finalModel)

female_vat_mars$results %>%
    filter(nprune == female_vat_mars$bestTune$nprune,
           degree == female_vat_mars$bestTune$degree) %>%
```

```
        kable(allign='center')

female_vat_mars$results



ggplot(female_vat_mars)

plotmo(female_vat_mars)



# Run again for no 2472

female_vat_mars2 <- train( x = subset(female_dxa_no2472,
      select = c(
      L_Weight,
      L_Height,
      L_AbdCirc,
      L_HipCirc,
      L_CalfCirc,
      L_TricepSKF,
      L_ForearmSKF,
      L_SubScapSKF,
      L_AbdSKF,
```

```
        L_SupraSKF,
        DecAge
        )),
    y = female_dxa_no2472$VATMass,
    method = 'bagEarth',
    trControl = trainControl(method = "cv",
                             number = 10,
                             summaryFunction=defaultSummary),
    tuneGrid = hyper_grid,
    keepX=F,
    B = 50)

female_vat_mars2$results %>%
    filter(nprune == female_vat_mars2$bestTune$nprune,
           degree == female_vat_mars2$bestTune$degree) %>%
            kable(allign='center')

#'
#' # Regular Regression: Fat Models
#'
## ------------------------------------------------------------------------------------------------------
```

```
# Step performs both forward and backwards selection using AIC to decide inclusion/exclusion
# I did not ask for any interactions here

set.seed(3072021)

male_fat_lin <- step(lm(TotalFatMass ~ L_Weight +
    L_Height +
    L_AbdCirc +
    L_HipCirc +
    L_CalfCirc +
    L_TricepSKF +
    L_ForearmSKF +
    L_SubScapSKF +
    L_AbdSKF +
    L_SupraSKF +
    DecAge,
    data = male_dxa))

# Check which variables were selected
male_fat_lin$coefficients
```

```
# Make a function that finds the coefficents, adj R sq, RMSE, MAE of the linear regression using the variables from above
# This function will be used to find the statistics in the boot loop below
fit_mfat = function(data,index){
model = lm(TotalFatMass ~ L_AbdCirc + L_Height + L_Weight + L_AbdSKF + L_CalfCirc,
        data = data, subset = index)
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, male_dxa), male_dxa$TotalFatMass),
  MAE = mae(predict(model, male_dxa), male_dxa$TotalFatMass))
}

# bootstrap these stats 1000 times
male_fat_linb <- boot(male_dxa, fit_mfat, 1000)
male_fat_linb



# Female  model

female_fat_lin <- step(lm(TotalFatMass ~ L_Weight +
      L_Height +
```

```
        L_AbdCirc +
        L_HipCirc +
        L_CalfCirc +
        L_TricepSKF +
        L_ForearmSKF +
        L_SubScapSKF +
        L_AbdSKF +
        L_SupraSKF +
        DecAge,
        data = female_dxa))

female_fat_lin$coefficients

# Make a function that finds the coefficents, adj R sq, RMSE, MAE of the linear regression using the variables from above
# This function will be used to find the statistics in the boot loop below
fit_ffat = function(data,index){
model = lm(TotalFatMass ~ L_Height + L_Weight + L_HipCirc + DecAge + L_ForearmSKF,
          data = data, subset = index)
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, female_dxa), female_dxa$TotalFatMass),
```

```
  MAE = mae(predict(model, female_dxa), female_dxa$TotalFatMass))
}
# bootstrap these coefs 1000 times
boot(female_dxa, fit_ffat, 1000)




#'
#' # Regular Regression: Lean Models
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-
set.seed(3072021)
# Step performs both forward and backwards selection using AIC to decide
inclusion/exclusion
# I did not ask for any interactions here
male_lean_lin <- step(lm(TotalLeanMass ~ L_Weight +
    L_Height +
    L_AbdCirc +
    L_HipCirc +
    L_CalfCirc +
```

```
        L_TricepSKF +
        L_ForearmSKF +
        L_SubScapSKF +
        L_AbdSKF +
        L_SupraSKF +
        DecAge,
        data = male_dxa))
male_lean_lin$coefficients

# Make a function that finds the coefficents, adj R sq, RMSE, MAE of the linear regression using the variables from above
# This function will be used to find the statistics in the boot loop below
fit_mlean = function(data,index){
model = lm(TotalLeanMass ~ L_Height + L_Weight + L_AbdCirc + L_CalfCirc + L_SupraSKF + L_TricepSKF + L_AbdSKF,
        data = data, subset = index)
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, male_dxa), male_dxa$TotalLeanMass),
  MAE = mae(predict(model, male_dxa), male_dxa$TotalLeanMass))
}

# bootstrap these coefs 1000 times
```

```
male_lean_linb <- boot(male_dxa, fit_mlean, 1000)
male_lean_linb



female_lean_lin <- step(lm(TotalLeanMass ~ L_Weight +
    L_Height +
    L_AbdCirc +
    L_HipCirc +
    L_CalfCirc +
    L_TricepSKF +
    L_ForearmSKF +
    L_SubScapSKF +
    L_AbdSKF +
    L_SupraSKF +
    DecAge,
    data = female_dxa))

female_lean_lin$coefficients

# Make a function that finds the coefficents, adj R sq, RMSE, MAE of the linear regression using the variables from above
# This function will be used to find the statistics in the boot loop below
fit_flean = function(data,index){
```

```
        model = lm(TotalLeanMass ~ L_Height + L_Weight + L_HipCirc + L_CalfCirc +
L_ForearmSKF + DecAge,
                   data = data, subset = index)
        c(coef(model),
          rsq = summary(model)$adj.r.squared,
          rmse = RMSE(predict(model, female_dxa), female_dxa$TotalLeanMass),
          MAE = mae(predict(model, female_dxa),female_dxa$TotalLeanMass))
        }
        # bootstrap these coefs 1000 times
        female_lean_linb <- boot(female_dxa, fit_flean, 1000)
        female_lean_linb



        #'
        #' # Regular Regression: VAT Models
        #'
        ## ------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------
-
        set.seed(3072021)
        # Step performs both forward and backwards selection using AIC to decide
inclusion/exclusion
        # I did not ask for any interactions here
```

```
male_vat_lin <- step(lm(VATMass ~ L_Weight +
        L_Height +
        L_AbdCirc +
        L_HipCirc +
        L_CalfCirc +
        L_TricepSKF +
        L_ForearmSKF +
        L_SubScapSKF +
        L_AbdSKF +
        L_SupraSKF +
        DecAge,
        data = male_dxa))

male_vat_lin$coefficients



# Make a function that finds the coefficents, adj R sq, RMSE, MAE of the linear regression using the variables from above
# This function will be used to find the statistics in the boot loop below
fit_mvat = function(data,index){
model = lm(VATMass ~ L_Height + L_Weight + L_HipCirc + L_AbdCirc + DecAge,
          data = data, subset = index)
c(coef(model),
```

```
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, male_dxa), male_dxa$VATMass),
  MAE = mae(predict(model, male_dxa), male_dxa$VATMass))
}

# bootstrap these coefs 1000 times
male_vat_linb <- boot(male_dxa,fit_mvat, 1000)
male_vat_linb

female_vat_lin <- step(lm(VATMass ~ L_Weight +
       L_Height +
       L_AbdCirc +
       L_HipCirc +
       L_CalfCirc +
       L_TricepSKF +
       L_ForearmSKF +
       L_SubScapSKF +
       L_AbdSKF +
       L_SupraSKF +
       DecAge,
       data = female_dxa))

female_vat_lin$coefficients
```

```
# Make a function that finds the coefficent of the linear regression using the variables from
above
fit_fvat = function(data,index){
model = lm(VATMass ~ L_Height + L_Weight + L_HipCirc + L_AbdCirc + DecAge,
        data = data, subset = index)
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, female_dxa), female_dxa$VATMass),
  MAE = mae(predict(model, female_dxa), female_dxa$VATMass))
}
# bootstrap these coefs 1000 times
female_lean_linb <- boot(female_dxa, fit_fvat, 1000)
female_lean_linb




#'
#' # Swinburn Models
#'
## ----------------------------------------------------------------------------------------------------
-
```

```
# Don't Know if we are using BMI models
# Samoan Male Fat Mass Model, fat measured in kg
#  Fat = 1.81 * BMI - 32.21
# Function below takes bmi and outputs total fat in grams for males (grams were used in our dxa study)
Swin_male_bmi <- function(bmi){
  (1.81*bmi - 32.21)*1000
}
swin_bmi <- lm(TotalFatMass ~ bmi, male_dxa)
summary(swin_bmi)

# We decided to recreate the models with our data, rather than using the fixed coefficients from Swinburns paper
fit_mswin = function(data,index){
model = lm(TotalFatMass ~ bmi,
          data = data, subset = index)
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, male_dxa), male_dxa$TotalFatMass),
  MAE = mae(predict(model, male_dxa), male_dxa$TotalFatMass))
}
set.seed(3072021)
```

```
boot(male_dxa,fit_mswin,1000)
# Coefficient of bmi for boot recreation  is 1.9 compared to 1.81, very similar



# Female  model


# Samoan Female Fat Mass Model
#  Fat = 1.69 * BMI - 20.41
# Function below takes bmi and outputs total fat in grams for females
Swin_female_bmi <- function(bmi){
  (1.69*bmi - 20.41)*1000
}


female_dxa$swin_bmi_fat <- Swin_female_bmi(female_dxa$bmi)


Swin_female_lin <- lm(swin_bmi_fat ~ bmi, female_dxa)
summary(Swin_female_lin)


# We decided to recreate the models with our data, rather than using the fixed coefficients
from Swinburns paper
fit_fswin = function(data,index){
model = lm(TotalFatMass ~ bmi,
        data = data, subset = index)
```

```
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, female_dxa), female_dxa$TotalFatMass),
  MAE = mae(predict(model, female_dxa), female_dxa$TotalFatMass))
}
set.seed(3072021)
boot(female_dxa,fit_fswin,1000)
# Coeff of bmi for boot recreation is 1.72 compared to original 1.69, very similar

# Height and Weight (Sex: 1=male, 0=female)
# Fat = 35.98 + 0.64 * weight - 10.51 * sex - 0.35 * height + 0.05 * age
# Function below takes height weight sex and age, and outputs total fat in grams
Swin_hwsa <- function(weight, sex, height, age){
  1000*(35.98 + 0.64*weight -10.51*sex -0.35*height + 0.05*age)
}

body_dxa$swin_fat <- with(body_dxa,Swin_hwsa(L_Weight, as.numeric(Sex), L_Height,
DecAge))


Swin_lin <- lm(swin_fat ~ L_Weight +
                 as.factor(Sex) +
                 L_Height +
```

```
                DecAge,
                body_dxa)
summary(Swin_lin)



fit_swin = function(data,index){
model = lm(TotalFatMass ~ L_Weight +
             as.factor(Sex) +
             L_Height +
             DecAge,
           data=data,subset=index)
c(coef(model),
  rsq = summary(model)$adj.r.squared,
  rmse = RMSE(predict(model, body_dxa), body_dxa$TotalFatMass),
  MAE = mae(predict(model, body_dxa), body_dxa$TotalFatMass))
}



set.seed(3072021)
boot(body_dxa,fit_swin,1000)
```

```
#'
#'
#' # Ridge: Total Fat Models
#'
## ------------------------------------------------------------------------------------------------------------
-
set.seed(03092021)
?seq()

lambdas <- 10^seq(3, -2, by = -.1)
lambdas
# Bagging.lasso(x=subset(train_male,
#                        select = c(L_Weight,
#                                   L_Height,
#                                   L_AbdCirc,
#                                   L_HipCirc,
#                                   L_CalfCirc,
#                                   L_TricepSKF,
#                                   L_ForearmSKF,
#                                   L_SubScapSKF,
#                                   L_AbdSKF,
```

```
#                               L_SupraSKF,
#                               DecAge
#                              )
#                     ),
#               y=male_dxa$TotalFatMass,
# family=c("gaussian"),
# M=2,
# predictor.importance=TRUE,
# trimmed=FALSE,
# seed=0123)


male_ridge_fat <- cv.glmnet(x = as.matrix(subset(train_male,
                            select = c(L_Weight,
                                       L_Height,
                                       L_AbdCirc,
                                       L_HipCirc,
                                       L_CalfCirc,
                                       L_TricepSKF,
                                       L_ForearmSKF,
                                       L_SubScapSKF,
                                       L_AbdSKF,
                                       L_SupraSKF,
                                       DecAge
```

```
                    ))),
                y = train_male$TotalFatMass,
                alpha=0,
                lambdas = lambdas)

coef(male_ridge_fat)


opt_lambda <- male_ridge_fat$lambda.min
fit <- male_ridge_fat$glmnet.fit

y_predicted <- predict(fit, s = opt_lambda, newx = as.matrix(subset(test_male,
                    select = c(L_Weight,
                            L_Height,
                            L_AbdCirc,
                            L_HipCirc,
                            L_CalfCirc,
                            L_TricepSKF,
                            L_ForearmSKF,
                            L_SubScapSKF,
                            L_AbdSKF,
                            L_SupraSKF,
                            DecAge
```

```
                                ))))

# Sum of Squares Total and Error
sst <- sum((test_male$TotalFatMass - mean(test_male$TotalFatMass))^2)
sse <- sum((y_predicted - test_male$TotalFatMass)^2)

# R squared
rsq <- 1 - sse / sst
rsq
# Adj. R sq, with the 11 predictors
adj.rsq <- rsq - (1-rsq)*(11/(nrow(test_male)-11-1))
adj.rsq

RMSE(y_predicted, test_male$TotalFatMass)
mae(y_predicted, test_male$TotalFatMass)



# Female
female_ridge_fat <- cv.glmnet(x = as.matrix(subset(train_female,
                        select = c(L_Weight,
                                   L_Height,
                                   L_AbdCirc,
```

```
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
                    ))),
                y = train_female$TotalFatMass,
                alpha=0,
                lambdas = lambdas)

opt_lambda <- female_ridge_fat$lambda.min
fit <- female_ridge_fat$glmnet.fit

coef(female_ridge_fat)


y_predicted <- predict(fit, s = opt_lambda, newx = as.matrix(subset(test_female,
                    select = c(L_Weight,
                    L_Height,
                    L_AbdCirc,
```

```
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
                    ))))

# Sum of Squares Total and Error
sst <- sum((test_female$TotalFatMass - mean(test_female$TotalFatMass))^2)
sse <- sum((y_predicted - test_female$TotalFatMass)^2)

# R squared
rsq <- 1 - sse / sst
rsq

adj.rsq <- rsq - (1-rsq)*(11/(nrow(test_female)-11-1))
adj.rsq

RMSE(y_predicted, test_female$TotalFatMass)
mae(y_predicted, test_female$TotalFatMass)
```

```
#'
#' # Ridge: Total Lean Mass Models
#'
## ------------------------------------------------------------------------------------------------------------
-

set.seed(03092021)


male_ridge_lean <- cv.glmnet(x = as.matrix(subset(train_male,
                                                  select = c(L_Weight,
                                                             L_Height,
                                                             L_AbdCirc,
                                                             L_HipCirc,
                                                             L_CalfCirc,
                                                             L_TricepSKF,
                                                             L_ForearmSKF,
                                                             L_SubScapSKF,
                                                             L_AbdSKF,
                                                             L_SupraSKF,
```

```
                    DecAge
                    ))),
                y = train_male$TotalLeanMass,
                alpha=0,
                lambdas = lambdas)
coef(male_ridge_lean)

opt_lambda <- male_ridge_lean$lambda.min
fit <- male_ridge_lean$glmnet.fit


y_predicted <- predict(fit, s = opt_lambda, newx = as.matrix(subset(test_male,
                    select = c(L_Weight,
                    L_Height,
                    L_AbdCirc,
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
```

```
                                ))))

# Sum of Squares Total and Error
sst <- sum((test_male$TotalLeanMass - mean(test_male$TotalLeanMass))^2)
sse <- sum((y_predicted - test_male$TotalLeanMass)^2)

# R squared
rsq <- 1 - sse / sst
rsq

adj.rsq <- rsq - (1-rsq)*(11/(nrow(test_male)-11-1))
adj.rsq


RMSE(y_predicted, test_male$TotalLeanMass)
mae(y_predicted, test_male$TotalLeanMass)



# Female
female_ridge_lean <- cv.glmnet(x = as.matrix(subset(train_female,
                        select = c(L_Weight,
                                   L_Height,
```

```
                    L_AbdCirc,
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
                    ))),
                y = train_female$TotalLeanMass,
                alpha=0,
                lambdas = lambdas)

coef(female_ridge_lean)

opt_lambda <- female_ridge_lean$lambda.min
fit <- female_ridge_lean$glmnet.fit


y_predicted <- predict(fit, s = opt_lambda, newx = as.matrix(subset(test_female,
                    select = c(L_Weight,
                    L_Height,
```

```
                    L_AbdCirc,
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
                    ))))

# Sum of Squares Total and Error
sst <- sum((test_female$TotalLeanMass - mean(test_female$TotalLeanMass))^2)
sse <- sum((y_predicted - test_female$TotalLeanMass)^2)

# R squared
rsq <- 1 - sse / sst
rsq


adj.rsq <- rsq - (1-rsq)*(11/(nrow(test_female)-11-1))
adj.rsq
```

```
RMSE(y_predicted, test_female$TotalLeanMass)

mae(y_predicted, test_female$TotalLeanMass)



#'

#' # Ridge: VAT Mass Models

#'

## ------------------------------------------------------------------------------------------------------------

-

set.seed(03092021)


male_ridge_vat <- cv.glmnet(x = as.matrix(subset(train_male,

                            select = c(L_Weight,

                                       L_Height,

                                       L_AbdCirc,

                                       L_HipCirc,

                                       L_CalfCirc,

                                       L_TricepSKF,

                                       L_ForearmSKF,

                                       L_SubScapSKF,

                                       L_AbdSKF,

                                       L_SupraSKF,
```

```
                    DecAge
                    ))),
                y = train_male$VATMass,
                alpha=0,
                lambdas = lambdas)

coef(male_ridge_vat)

opt_lambda <- male_ridge_vat$lambda.min
fit <- male_ridge_vat$glmnet.fit

y_predicted <- predict(fit, s = opt_lambda, newx = as.matrix(subset(test_male,
                    select = c(L_Weight,
                    L_Height,
                    L_AbdCirc,
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
```

```
                        DecAge
                        ))))

# Sum of Squares Total and Error
sst <- sum((test_male$VATMass - mean(test_male$VATMass))^2)
sse <- sum((y_predicted - test_male$VATMass)^2)

# R squared
rsq <- 1 - sse / sst
rsq

adj.rsq <- rsq - (1-rsq)*(11/(nrow(test_male)-11-1))
adj.rsq

RMSE(y_predicted, test_male$VATMass)
mae(y_predicted, test_male$VATMass)


# Female
female_ridge_vat <- cv.glmnet(x = as.matrix(subset(train_female,
                    select = c(L_Weight,
                        L_Height,
                        L_AbdCirc,
```

```
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
                    ))),
                y = train_female$VATMass,
                alpha=0,
                lambdas = lambdas)

coef(female_ridge_vat)

opt_lambda <- female_ridge_vat$lambda.min
fit <- female_ridge_vat$glmnet.fit


y_predicted <- predict(fit, s = opt_lambda, newx = as.matrix(subset(test_female,
                    select = c(L_Weight,
                    L_Height,
                    L_AbdCirc,
```

```
                    L_HipCirc,
                    L_CalfCirc,
                    L_TricepSKF,
                    L_ForearmSKF,
                    L_SubScapSKF,
                    L_AbdSKF,
                    L_SupraSKF,
                    DecAge
                    ))))

# Sum of Squares Total and Error
sst <- sum((test_female$VATMass - mean(test_female$VATMass))^2)
sse <- sum((y_predicted - test_female$VATMass)^2)

# R squared
rsq <- 1 - sse / sst
rsq

adj.rsq <- rsq - (1-rsq)*(11/(nrow(test_female)-11-1))
adj.rsq

RMSE(y_predicted, test_female$VATMass)
mae(y_predicted, test_female$VATMass)
```

```
#'
#'
#' # Samoan 2010 Loading and Cleaning
#'
## ------------------------------------------------------------------------------------------------------------------
-

#load("//home//shared_data//samoa//Samoan_data//Samoan_Discovery_Phenotypes//Samoan_Discovery_Phenotype_v3_2020-01-13.RData")

load("//home//grp20//phenotype_models_jenna_16Feb2021.RData")

samoa_2010 <- phenotypes.gwas
# Remove the SG and leading zeros from ID to match the ID from DXA data set
samoa_2010$SUBJECT_ID <- str_remove(samoa_2010$SUBJECT_ID, "SG")
samoa_2010$SUBJECT_ID <- str_remove(samoa_2010$SUBJECT_ID, "^0+")

# Remove those individuals from the 2017-19 DXA study
samoa_2010 <- samoa_2010[!(samoa_2010$SUBJECT_ID %in% body_dxa$IDNumber),]
```

```
# Check for pregnant women, because their body measures would not be accurate
# table(samoa_2010$Pregnant)

# Check their gender var
table(samoa_2010$Gender)
# Create a sex var that is equivalent to the Sex from 2017-19 study
samoa_2010$Sex <- recode(samoa_2010$Gender,
                         "1" = "Male",
                         "2" = "Female")

# Make a smaller data frame of just the predictors, I will use these to compare to the 2017-19 sample
samoa_2010_pred <- samoa_2010[c("Dec_Age",
                                "Sex",
                                "rs373863828",
                                "Weight",
                                "Height",
                                "Abd_Circ",
                                "Hip_Circ",
                                "Calf_Circ",
                                "Tri_Skf",
                                "For_Skf",
                                "Sub_Skf",
```

```
                        "Abd_Skf",
                        "Sup_Skf")]

# Make an identical data set from 2017-19
samoa_dxa_pred <- body_dxa[c("DecAge",
                        "Sex",
                        "Genotype_code",
                        "L_Weight",
                        "L_Height",
                        "L_AbdCirc",
                        "L_HipCirc",
                        "L_CalfCirc",
                        "L_TricepSKF",
                        "L_ForearmSKF",
                        "L_SubScapSKF",
                        "L_AbdSKF",
                        "L_SupraSKF")]

samoa_dxa_pred <- samoa_dxa_pred %>%
                rename(
                        Dec_Age = DecAge,
                        rs373863828 = Genotype_code,
                        Weight = L_Weight,
```

```
                    Height = L_Height,
                    Abd_Circ = L_AbdCirc,
                    Hip_Circ = L_HipCirc,
                    Calf_Circ = L_CalfCirc,
                    Tri_Skf = L_TricepSKF,
                    For_Skf = L_ForearmSKF,
                    Sub_Skf = L_SubScapSKF,
                    Abd_Skf = L_AbdSKF,
                    Sup_Skf = L_SupraSKF
                    )
# Create indicator var for sample
samoa_dxa_pred$samp_2010 <- "2017-19 Cohort"
samoa_2010_pred$samp_2010 <- "2010 Cohort"



samoa_pred <- rbind(samoa_2010_pred, samoa_dxa_pred)




#'
#' # Samoa 2010 Descriptives [not used, these are prior to more cleaning]
```

```
#'
## ------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
-


# Table for 2010 data, minus the 2017-19 individuals, broken down by sex
table_three <- tableby(Sex ~ Dec_Age +
                         rs373863828 +
                         Weight +
                         Height +
                         Abd_Circ +
                         Hip_Circ +
                         Calf_Circ +
                         Tri_Skf +
                         For_Skf +
                         Sub_Skf +
                         Abd_Skf +
                         Sup_Skf,
                       data = samoa_2010)


labels3 <- list(DecAge = "Age (yrs)",
                rs373863828 = "Genotype",
```

```
            Weight = "Weight (kg)",
            Height = "Height (cm)",
            Abd_Circ = "Abdomen Circum. (cm)" ,
            Hip_Circ = "Hip Circum. (cm)",
            Calf_Circ = "Calf Circum. (cm)",
            Tri_Skf = "Tricep Skin Fold (mm)",
            For_Skf = "Forearm Skin Fold (mm)",
            Sub_Skf = "Subscapular Skin Fold (mm)",
            Abd_Skf = "Abdomen Skin Fold (mm)",
            Sup_Skf = "Suprailiac Skin Fold(mm)")

summary(table_three,
      labelTranslations = labels3,
      pfootnote = TRUE)

# Table comparing 2010 and 2017-19 samples
table_four <- tableby(samp_2010 ~ Dec_Age +
                  Sex +
                  rs373863828 +
                  Weight +
                  Height +
                  Abd_Circ +
                  Hip_Circ +
```

```
                Calf_Circ +
                Tri_Skf +
                For_Skf +
                Sub_Skf +
                Abd_Skf +
                Sup_Skf,
              data = samoa_pred)


labels4 <- list(DecAge = "Age (yrs)",
                Sex = "Sex",
                rs373863828 = "Genotype",
                Weight = "Weight (kg)",
                Height = "Height (cm)",
                Abd_Circ = "Abdomen Circum. (cm)" ,
                Hip_Circ = "Hip Circum. (cm)",
                Calf_Circ = "Calf Circum. (cm)",
                Tri_Skf = "Tricep Skin Fold (mm)",
                For_Skf = "Forearm Skin Fold (mm)",
                Sub_Skf = "Subscapular Skin Fold (mm)",
                Abd_Skf = "Abdomen Skin Fold (mm)",
                Sup_Skf = "Suprailiac Skin Fold(mm)")
summary(table_four,
```

```
        labelTranslations = labels4,
        pfootnote = TRUE)




        #'
        #' # 2010 Sample cleaning
        #'
        ## ------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------
-
        # subsetting samoa 2010 for the essential vars: demographic info, predictors


        samoa_sub <- samoa_2010[c("SUBJECT_ID",
                                  "Dec_Age",
                                  "Sex",
                                  "rs373863828",
                                  "Weight",
                                  "Height",
                                  "Abd_Circ",
```

```
                "Hip_Circ",
                "Calf_Circ",
                "Tri_Skf",
                "For_Skf",
                "Sub_Skf",
                "Abd_Skf",
                "Sup_Skf")
                ]

# Rename the predictors to match the naming from the datasets that built the mars models
# Mars needs to see exatcly the same names for it predict the outcome
samoa_sub <- samoa_sub %>%
            rename(
                DecAge = Dec_Age,
                Genotype_code = rs373863828,
                L_Weight = Weight,
                L_Height = Height,
                L_AbdCirc = Abd_Circ,
                L_HipCirc = Hip_Circ,
                L_CalfCirc = Calf_Circ,
                L_TricepSKF = Tri_Skf,
                L_ForearmSKF = For_Skf,
                L_SubScapSKF = Sub_Skf,
```

```
                    L_AbdSKF = Abd_Skf,
                    L_SupraSKF = Sup_Skf
                    )

    # Create empty columns for outcomes to be predicted
    samoa_sub$TotalFatMass  <- NA
    samoa_sub$TotalLeanMass  <- NA
    samoa_sub$VATMass  <- NA

    # Create empty column for a allele, I may have missed this in the original data set
    # This is for the additive genetic model, it needs to be a contin var with the count being number of A alleles

    # I code a new variable A allele whic counts the A alleles, this is unnecessary as the original data set had both a genotype and an A allele var
    samoa_sub$A_allele <- NA
    samoa_sub$A_allele <- recode(samoa_sub$Genotype_code,
                    "GG" = 0,
                    "AA" = 2,
                    "AG" = 1)

    # Need to remove missingness in the predictors for MARS to work
    # Check and save Missingness in predictors
```

```
missing_pred_2010 <- samoa_sub[!complete.cases(samoa_sub[c("L_Weight",
                    "L_Height",
                    "L_AbdCirc",
                    "L_HipCirc",
                    "L_CalfCirc",
                    "L_TricepSKF",
                    "L_ForearmSKF",
                    "L_SubScapSKF",
                    "L_AbdSKF",
                    "L_SupraSKF")]),]
missing_pred_2010
# 716 entries removed for not having completed values in all predictors


# Remove missings, they will cause issue with MARS
samoa_sub <- samoa_sub[complete.cases(samoa_sub[c("L_Weight",
                    "L_Height",
                    "L_AbdCirc",
                    "L_HipCirc",
                    "L_CalfCirc",
                    "L_TricepSKF",
                    "L_ForearmSKF",
                    "L_SubScapSKF",
```

```
                        "L_AbdSKF",
                        "L_SupraSKF")]),]

        summary(samoa_sub[c("DecAge",
                        "L_Weight",
                        "L_Height",
                        "L_AbdCirc",
                        "L_HipCirc",
                        "L_CalfCirc",
                        "L_TricepSKF",
                        "L_ForearmSKF",
                        "L_SubScapSKF",
                        "L_AbdSKF",
                        "L_SupraSKF")])

        #'
        #'
        #' # Predicting values with MARS Models
        #'
        ## ------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------
-
        # stratifying by sex so each model can be run seprately, I will rejoin these later
```

```
samoa_male <- samoa_sub[samoa_sub$Sex=="Male",]
samoa_female <- samoa_sub[samoa_sub$Sex=="Female",]

# Fat Predictions
# Note: the order of the variables in the list needs to match the order for the MARS model
samoa_male$TotalFatMass <- predict(male_fat_mars$finalModel, samoa_male[c(
                                    "L_Weight",
                                    "L_Height",
                                    "L_AbdCirc",
                                    "L_HipCirc",
                                    "L_CalfCirc",
                                    "L_TricepSKF",
                                    "L_ForearmSKF",
                                    "L_SubScapSKF",
                                    "L_AbdSKF",
                                    "L_SupraSKF",
                                    "DecAge")]
                                   )

samoa_female$TotalFatMass <- predict(female_fat_mars$finalModel, samoa_female[c(
                                    "L_Weight",
                                    "L_Height",
```

```
                                "L_AbdCirc",
                                "L_HipCirc",
                                "L_CalfCirc",
                                "L_TricepSKF",
                                "L_ForearmSKF",
                                "L_SubScapSKF",
                                "L_AbdSKF",
                                "L_SupraSKF",
                                "DecAge")]
                              )



# Lean
samoa_male$TotalLeanMass <- predict(male_lean_mars$finalModel, samoa_male[c(
                                "L_Weight",
                                "L_Height",
                                "L_AbdCirc",
                                "L_HipCirc",
                                "L_CalfCirc",
                                "L_TricepSKF",
                                "L_ForearmSKF",
                                "L_SubScapSKF",
```

```
                                    "L_AbdSKF",
                                    "L_SupraSKF",
                                    "DecAge")]
                                  )


        samoa_female$TotalLeanMass <- predict(female_lean_mars$finalModel,
samoa_female[c(
                                    "L_Weight",
                                    "L_Height",
                                    "L_AbdCirc",
                                    "L_HipCirc",
                                    "L_CalfCirc",
                                    "L_TricepSKF",
                                    "L_ForearmSKF",
                                    "L_SubScapSKF",
                                    "L_AbdSKF",
                                    "L_SupraSKF",
                                    "DecAge")]
                                  )

        # VAT
        samoa_male$VATMass <- predict(male_vat_mars$finalModel, samoa_male[c(
```

```
                "L_Weight",
                "L_Height",
                "L_AbdCirc",
                "L_HipCirc",
                "L_CalfCirc",
                "L_TricepSKF",
                "L_ForearmSKF",
                "L_SubScapSKF",
                "L_AbdSKF",
                "L_SupraSKF",
                "DecAge")]
            )

samoa_female$VATMass <- predict(female_vat_mars$finalModel, samoa_female[c(
                "L_Weight",
                "L_Height",
                "L_AbdCirc",
                "L_HipCirc",
                "L_CalfCirc",
                "L_TricepSKF",
                "L_ForearmSKF",
                "L_SubScapSKF",
                "L_AbdSKF",
```

```
                                  "L_SupraSKF",
                                  "DecAge")]
                                )



        # Rejoin sexes with imputed measurements into one data frame, called samoa_mars

        samoa_mars <- as.data.frame(rbind(samoa_female,samoa_male))

        #'
        #' # 2010 Samoa Cleaned Tables
        #'
        ## ------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------
-
        # Redoing tables from above, since we removed missing cases from the 2010 data set
        # start by resubsetting the 2010 for predictors

        samoa_2010_pred <- samoa_mars[c("DecAge",
                                        "Sex",
                                        "Genotype_code",
```

```
                        "L_Weight",
                        "L_Height",
                        "L_AbdCirc",
                         "L_HipCirc",
                         "L_CalfCirc",
                         "L_TricepSKF",
                         "L_ForearmSKF",
                         "L_SubScapSKF",
                         "L_AbdSKF",
                         "L_SupraSKF")]
# Add column for sample identifier
samoa_2010_pred$samp_2010 <- "2010 Cohort"
# Repeat for 2017-19 data
samoa_dxa_pred <- body_dxa[c("DecAge",
                        "Sex",
                        "Genotype_code",
                        "L_Weight",
                        "L_Height",
                        "L_AbdCirc",
                         "L_HipCirc",
                         "L_CalfCirc",
                         "L_TricepSKF",
                         "L_ForearmSKF",
```

```
                    "L_SubScapSKF",
                    "L_AbdSKF",
                    "L_SupraSKF")]
samoa_dxa_pred$samp_2010 <- "2017-19 Cohort"

samoa_pred <- rbind(samoa_2010_pred, samoa_dxa_pred)



# Table comparing 2010 and 2017-19 samples
table_four <- tableby(samp_2010 ~ DecAge +
                    Sex +
                    Genotype_code +
                    L_Weight +
                    L_Height +
                    L_AbdCirc +
                    L_HipCirc +
                    L_CalfCirc +
                    L_TricepSKF +
                    L_ForearmSKF +
                    L_SubScapSKF +
                    L_AbdSKF +
                    L_SupraSKF,
```

```
                data = samoa_pred)



labels4 <- list(DecAge = "Age (yrs)",
                Sex = "Sex",
                Genotype_code = "Genotype",
                L_Weight = "Weight (kg)",
                L_Height = "Height (cm)",
                L_AbdCirc = "Abdomen Circum. (cm)" ,
                L_HipCirc = "Hip Circum. (cm)",
                L_CalfCirc = "Calf Circum. (cm)",
                L_TricepSKF = "Tricep Skin Fold (mm)",
                L_ForearmSKF = "Forearm Skin Fold (mm)",
                L_SubScapSkf = "Subscapular Skin Fold (mm)",
                L_AbdSKF = "Abdomen Skin Fold (mm)",
                L_SupraSKF = "Suprailiac Skin Fold(mm)")
summary(table_four,
        labelTranslations = labels4,
        pfootnote = TRUE)



#'
```

```
#'
#'
#' # Low Predicted VAT Measurements
#'
## ------------------------------------------------------------------------------------------------------------
-
# Subset anyone with a negative predicted VATmass, since this number should never be below zero (or zero for that matter)
low_vat <- samoa_mars[samoa_mars$VATMass < 0,]

# Some prelim analysis on whether these people are really lean, or young, etc.
table(low_vat$Sex)

summary(low_vat$L_Weight)

meanfun <- function(data, i){
  d <- data[i, ]
  return(mean(d))
}

boot_weight <- boot(low_vat[, "L_Weight", drop = FALSE], meanfun, R=5000)
boot_weight$t
```

```
boot.ci(boot_weight, conf=0.95, type="bca")


hist(boot_weight$t)
hist(samoa_mars$L_Weight)


summary(low_vat$DecAge)
summary(female_dxa$DecAge)


summary(low_vat$L_Height)
# How do we treat these individuals?
# Capped  the lowest possible VAT measure as the lowest measured in the 2017-19 for respective sex
min_male <- min(male_dxa$VATMass)
min_female <- min(female_dxa$VATMass)


samoa_mars$VATMass[samoa_mars$VATMass < min_male & samoa_mars$Sex == "Male"] <- min_male
samoa_mars$VATMass[samoa_mars$VATMass < min_female & samoa_mars$Sex == "Female"] <- min_female


#plot the dist of vat mass by sex after change
vat_fix_plot <- ggplot(samoa_mars, aes(x = Sex, y = VATMass)) +
  geom_jitter(aes(color = Sex),
```

```
                position=position_jitter(0.2)) +
    stat_summary(fun.data=mean_sdl,
                 geom="pointrange",
                 color="black") +
    labs(
        x = "Sex",
        y = "VAT Mass (g)") +
  scale_color_manual(values=c("lightseagreen","lightcoral"))


vat_fix_plot


summary(samoa_mars$VATMass[samoa_mars$Sex=="Female"])


summary(samoa_mars$VATMass[samoa_mars$Sex=="Male"])


#'
#' # Predicted Mass Plots
#'
## ------------------------------------------------------------------------------------------------------------
-


tfm_2010_fat <- ggplot(samoa_mars, aes(x = Sex, y = TotalFatMass)) +
```

```
  geom_jitter(aes(color = Sex),
                position=position_jitter(0.2)) +
    stat_summary(fun.data=mean_sdl,
                 geom="pointrange",
                 color="black") +
    labs(
        x = "Sex",
        y = "Total Fat Mass (g)")+
  scale_color_manual(values=c("lightseagreen", "lightcoral"))
tfm_2010_fat




tlm_2010_lean <- ggplot(samoa_mars, aes(x = Sex, y = TotalLeanMass)) +
  geom_jitter(aes(color = Sex),
                position=position_jitter(0.2)) +
    stat_summary(fun.data=mean_sdl,
                 geom="pointrange",
                 color="black") +
    labs(
        x = "Sex",
        y = "Total Lean Mass (g)")+
```

```
  scale_color_manual(values=c("lightseagreen", "lightcoral"))

tlm_2010_lean


# Compute descriptive statistics by groups
avgsd <- function(data){
   avg<-round(mean(data),1)
   sd<-round(sd(data),1)
return(c(avg,sd))
}


mfat_mu_2010 <- subset(samoa_mars$TotalFatMass, samoa_mars$Sex=="Male") %>%
 avgsd
mfat_mu_2010
ffat_mu_2010 <- subset(samoa_mars$TotalFatMass, samoa_mars$Sex=="Female") %>%
avgsd

mlean_mu_2010 <-subset(samoa_mars$TotalLeanMass, samoa_mars$Sex=="Male")
%>% avgsd
```

```
flean_mu_2010 <- subset(samoa_mars$TotalLeanMass, samoa_mars$Sex=="Female")
%>% avgsd

mvat_mu_2010 <- subset(samoa_mars$VATMass, samoa_mars$Sex=="Male") %>%
avgsd

fvat_mu_2010 <- subset(samoa_mars$VATMass, samoa_mars$Sex=="Female") %>%
avgsd



desc_table2 <- data.frame("Male Mass(g)"=
                            c(mfat_mu_2010,mlean_mu_2010,mvat_mu_2010),
                          "Female Mass(g)"=
                            c(ffat_mu_2010,flean_mu_2010,fvat_mu_2010),
                          check.names = F
                           )



# Summary table plot, medium orange theme
```

```
stable.p <- ggtexttable(desc_table2, rows = c("Total Fat (Avg)","sd","Total Lean (Avg)",
"sd", "VAT (Avg)","sd"),
                        theme = ttheme("minimal", base_size = 1))



outcome_plots2 <- ggarrange(tfm_2010_fat, tlm_2010_lean, vat_fix_plot, stable.p,
          labels=c("A","B","C"),
          common.legend = TRUE, legend = "bottom")
outcome_plots2


annotate_figure(outcome_plots2, top="Imputed Mass Outcomes by Sex, 2010 Cohort")




#'
#'
#' # Regression Prep.
#'
## ----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
-
# Loading kinship information and principle components for mixed effects model
```

```
ibs <- get(load("/home/shared_data/samoa/samoan-phewas/kinship-matrix-12Jul2018.RData"))
pca <- get(load("/home/shared_data/samoa/samoan-phewas/samoa-hg38-pcair-round2_pcair.RData"))

# Reformat subject id in row and col of matrix ibs
rownames(ibs)<- str_remove(rownames(ibs), "SG")
rownames(ibs) <- str_remove(rownames(ibs), "^0+")

colnames(ibs)<- str_remove(colnames(ibs), "SG")
colnames(ibs) <- str_remove(colnames(ibs), "^0+")

# Laod PCAs as data frame
pcs <- as.data.frame(pca$vectors)
# Make Sample Id column
pcs$sample.id <- row.names(pcs)
# reformat sample id to match Subject ID from 2010 data set
pcs$sample.id<- str_remove(pcs$sample.id, "SG")
pcs$sample.id <- str_remove(pcs$sample.id, "^0+")
# Rename sample id to Subject ID so we can merge data on the ID
pcs <- pcs %>%
  rename(
    SUBJECT_ID = sample.id
```

```
    )
    # Merge first 3 Pcs by ID
    samoa_mars <- left_join(samoa_mars, pcs[,c("V1","V2","V3","SUBJECT_ID")], by="SUBJECT_ID")
    # Rename PCs to PC1, PC2, PC3
    names(samoa_mars)[names(samoa_mars) %in% c("V1","V2","V3")] <- c("PC1","PC2","PC3")

    head(samoa_mars)



    #'
    #'
    #' # Total Fat Genetic Regression
    #'
    ## ------------------------------------------------------------------------------------------------------------------------------------------------

    # Males
    male_fat_me <- lmekin(TotalFatMass ~ A_allele + DecAge + I(DecAge^2) + PC1 + PC2 + PC3 + (1|SUBJECT_ID),
                          data=subset(samoa_mars, samoa_mars$Sex=="Male"),
```

```
                varlist=ibs*2,
                method="ML")

print(male_fat_me) # p-value printed is rounded


beta <- male_fat_me$coefficients$fixed[2]

se <- sqrt(male_fat_me$var[2,2])

results_mfat <- data.frame(N=male_fat_me$n, Beta=beta, SE=se,
P=round(2*pnorm(abs(beta/se), lower.tail=FALSE), 12), CI.L=round(beta - 1.96*se,8),
CI.U=round(beta + 1.96*se,8))
results_mfat


# Females
female_fat_me <- lmekin(TotalFatMass ~ A_allele + DecAge + I(DecAge^2) + PC1 + PC2
+ PC3 + (1|SUBJECT_ID),
                data=subset(samoa_mars, samoa_mars$Sex=="Female"),
                varlist=ibs*2,
                method="ML")
```

```
print(female_fat_me) # p-value printed is rounded



beta <- female_fat_me$coefficients$fixed[2]

se <- sqrt(female_fat_me$var[2,2])

results_ffat <- data.frame(N=female_fat_me$n, Beta=beta, SE=se,
P=round(2*pnorm(abs(beta/se), lower.tail=FALSE), 12), CI.L=round(beta - 1.96*se,8),
CI.U=round(beta + 1.96*se,8))
results_ffat



#'
#' # Total Lean Genetic Regression
#'
## ------------------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------------------
-
# Males
```

```
male_lean_me <- lmekin(TotalLeanMass ~ A_allele + DecAge + I(DecAge^2) + PC1 + PC2 + PC3 + (1|SUBJECT_ID),
                       data=subset(samoa_mars, samoa_mars$Sex=="Male"),
                       varlist=ibs*2,
                       method="ML")

print(male_lean_me) # p-value printed is rounded



beta <- male_lean_me$coefficients$fixed[2]

se <- sqrt(male_lean_me$var[2,2])

results_mlean <- data.frame(N=male_lean_me$n, Beta=beta, SE=se, P=round(2*pnorm(abs(beta/se), lower.tail=FALSE), 12), CI.L=round(beta - 1.96*se,8), CI.U=round(beta + 1.96*se,8))
results_mlean


# Females
female_lean_me <- lmekin(TotalLeanMass ~ A_allele + DecAge + I(DecAge^2) + PC1 + PC2 + PC3 + (1|SUBJECT_ID),
```

```
                data=subset(samoa_mars, samoa_mars$Sex=="Female"),
                varlist=ibs*2,
                method="ML")

print(female_lean_me) # p-value printed is rounded



beta <- female_lean_me$coefficients$fixed[2]

se <- sqrt(female_lean_me$var[2,2])

results_flean <- data.frame(N=female_lean_me$n, Beta=beta, SE=se,
P=round(2*pnorm(abs(beta/se), lower.tail=FALSE), 12), CI.L=round(beta - 1.96*se,8),
CI.U=round(beta + 1.96*se,8))
results_flean



#'
#'
#'
#' # VAT Genetic Regression
```

```
#'
## ------------------------------------------------------------------------------------------------------------------
-
# Males
male_vat_me <- lmekin(VATMass ~ A_allele + DecAge + I(DecAge^2) + PC1 + PC2 + PC3 + (1|SUBJECT_ID),
                      data=subset(samoa_mars, samoa_mars$Sex=="Male"),
                      varlist=ibs*2,
                      method="ML")

print(male_vat_me) # p-value printed is rounded



beta <- male_vat_me$coefficients$fixed[2]

se <- sqrt(male_vat_me$var[2,2])

results_mvat <- data.frame(N=male_vat_me$n, Beta=beta, SE=se, P=round(2*pnorm(abs(beta/se), lower.tail=FALSE), 12), CI.L=round(beta - 1.96*se,8), CI.U=round(beta + 1.96*se,8))
results_mvat
```

```
# Females
female_vat_me <- lmekin(VATMass ~ A_allele + DecAge + I(DecAge^2) + PC1 + PC2 + PC3 + (1|SUBJECT_ID),
                        data=subset(samoa_mars, samoa_mars$Sex=="Female"),
                        varlist=ibs*2,
                        method="ML")

print(female_vat_me) # p-value printed is rounded



beta <- female_vat_me$coefficients$fixed[2]

se <- sqrt(female_vat_me$var[2,2])

results_fvat <- data.frame(N=female_vat_me$n, Beta=beta, SE=se, P=round(2*pnorm(abs(beta/se), lower.tail=FALSE), 12), CI.L=round(beta - 1.96*se,8), CI.U=round(beta + 1.96*se,8))
results_fvat
```

```
#'
#' # Variable Inflation Factor From Models
#'
## ------------------------------------------------------------------------------------------------------------
-

vif(male_fat_lin)
# Weight VIF=12.9, AbC=9.1
vif(female_fat_lin)
# Weight=9.7 Hip=7.5
vif(male_lean_lin)
# Weight=13.9 AbC=9.4
vif(female_lean_lin)
# Weight=10.7 HipC=7.5
vif(male_vat_lin)
# Height=15.6 AbC=14.0 HipC=8.2
vif(female_vat_lin)
# Weight=11.8 AbC=5.4 Hip=7.7
```

Bibliography

Albanese, C. V., Diessel, E., & Genant, H. K. (2003). Clinical Applications of Body Composition Measurements Using DXA. *Journal of Clinical Densitometry, 6*(2), 75-85. doi:https://doi.org/10.1385/JCD:6:2:75

Arslanian, K. J., Fidow, U. T., Atanoa, T., Unasa-Apelu, F., Naseri, T., Wetzel, A. I., . . . Hawley, N. L. (2021). A missense variant in CREBRF, rs373863828, is associated with fat-free mass, not fat mass in Samoan infants. *International Journal of Obesity, 45*(1), 45-55. doi:10.1038/s41366-020-00659-4

Boehmke, B., & Greenwell, B. M. (2020). *Hands-on machine learning with R*: Taylor & Francis.

Canty, A., & Ripley, B. (2021). *boot: Bootstrap Functions (Originally by Angelo Canty for S)*.

Carlson, J. C., Rosenthal, S. L., Russell, E. M., Hawley, N. L., Sun, G., Cheng, H., . . . Minster, R. L. (2020). A missense variant in CREBRF is associated with taller stature in Samoans. *American Journal of Human Biology, 32*(6), e23414. doi:https://doi.org/10.1002/ajhb.23414

Cheryl D. Fryar, M. S. P. H., Margaret D. Carroll, M.S.P.H., and Cynthia L. Ogden, Ph.D. (2016, July 2016). Prevalence of Overweight, Obesity, and Extreme Obesity Among Adults Aged 20 and Over: United States, 1960–1962 Through 2013–2014. Retrieved from https://www.cdc.gov/nchs/data/hestat/obesity_adult_13_14/obesity_adult_13_14.htm

Cichosz, S. L., Rasmussen, N. H., Vestergaard, P., & Hejlesen, O. (2020). Precise Prediction of Total Body Lean and Fat Mass From Anthropometric and Demographic Data: Development and Validation of Neural Network Models. *Journal of Diabetes Science and Technology, 0*(0), 1932296820971348. doi:10.1177/1932296820971348

Fielding, R. A., Vellas, B., Evans, W. J., Bhasin, S., Morley, J. E., Newman, A. B., . . . Zamboni, M. (2011). Sarcopenia: an undiagnosed condition in older adults. Current consensus definition: prevalence, etiology, and consequences. International working group on sarcopenia. *J Am Med Dir Assoc, 12*(4), 249-256. doi:10.1016/j.jamda.2011.01.003

Frankenfield, D. C., Rowe, W. A., Cooney, R. N., Smith, J. S., & Becker, D. (2001). Limits of body mass index to detect obesity and predict body composition. *Nutrition, 17*(1), 26-30. doi:https://doi.org/10.1016/S0899-9007(00)00471-8

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software, 33*(1), 1-22. Retrieved from https://www.jstatsoft.org/v33/i01/

Friedman, J. H. (1991). Multivariate Adaptive Regression Splines. *The Annals of Statistics, 19*(1), 1-67. Retrieved from http://www.jstor.org/stable/2241837

Friedman, J. H., & Roosen, C. B. (1995). An introduction to multivariate adaptive regression splines. *Stat Methods Med Res, 4*(3), 197-217. doi:10.1177/096228029500400303

Gogarten, S. M., Sofer, T., Chen, H., Yu, C., Brody, J. A., Thornton, T. A., . . . Conomos, M. P. (2019). Genetic association testing using the GENESIS R/Bioconductor package. *Bioinformatics, 35*(24), 5346-5348. doi:10.1093/bioinformatics/btz567

Hanson, R. L., Safabakhsh, S., Curtis, J. M., Hsueh, W.-C., Jones, L. I., Aflague, T. F., . . . Nelson, R. G. (2019). Association of CREBRF variants with obesity and diabetes in Pacific

Islanders from Guam and Saipan. *Diabetologia, 62*(9), 1647-1652. doi:10.1007/s00125-019-4932-z

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning : data mining, inference, and prediction* (2nd ed.). New York, NY: Springer.

Hawley, N. L., Minster, R. L., Weeks, D. E., Viali, S., Reupena, M. S., Sun, G., . . . McGarvey, S. T. (2014). Prevalence of adiposity and associated cardiometabolic risk factors in the Samoan genome-wide association study. *Am J Hum Biol, 26*(4), 491-501. doi:10.1002/ajhb.22553

Hawley, N. L., Pomer, A., Rivara, A. C., Rosenthal, S. L., Duckham, R. L., Carlson, J. C., . . . McGarvey, S. T. (2020). Exploring the Paradoxical Relationship of a Creb 3 Regulatory Factor Missense Variant With Body Mass Index and Diabetes Among Samoans: Protocol for the Soifua Manuia (Good Health) Observational Cohort Study. *JMIR Res Protoc, 9*(7), e17329. doi:10.2196/17329

Hoffman, G. E. (2013). Correcting for Population Structure and Kinship Using the Linear Mixed Model: Theory and Extensions. *PLOS ONE, 8*(10), e75707. doi:10.1371/journal.pone.0075707

Ibrahim, M. M. (2010). Subcutaneous and visceral adipose tissue: structural and functional differences. *Obesity Reviews, 11*(1), 11-18. doi:https://doi.org/10.1111/j.1467-789X.2009.00623.x

Kopelman, P. G. (2000). Obesity as a medical problem. *Nature, 404*(6778), 635-643. doi:10.1038/35007508

Krishnan, M., Major, T. J., Topless, R. K., Dewes, O., Yu, L., Thompson, J. M. D., . . . Merriman, T. R. (2018). Discordant association of the CREBRF rs373863828 A allele with increased BMI and protection from type 2 diabetes in Māori and Pacific (Polynesian) people living in Aotearoa/New Zealand. *Diabetologia, 61*(7), 1603-1613. doi:10.1007/s00125-018-4623-1

Kuhn, M. (2020). caret: Classification and Regression Training (Version R package 6.0-86). Retrieved from https://CRAN.R-project.org/package=caret

Lebovitz, H. E. (2001). Insulin resistance: definition and consequences. *Exp Clin Endocrinol Diabetes, 109 Suppl 2*, S135-148. doi:10.1055/s-2001-18576

Lee, D. H., Keum, N., Hu, F. B., Orav, E. J., Rimm, E. B., Sun, Q., . . . Giovannucci, E. L. (2017). Development and validation of anthropometric prediction equations for lean body mass, fat mass and percent fat in adults using the National Health and Nutrition Examination Survey (NHANES) 1999–2006. *British Journal of Nutrition, 118*(10), 858-866. doi:10.1017/S0007114517002665

Lewis, C. M. (2002). Genetic association studies: Design, analysis and interpretation. *Briefings in Bioinformatics, 3*(2), 146-153. doi:10.1093/bib/3.2.146

Lin, S., Naseri, T., Linhart, C., Morrell, S., Taylor, R., McGarvey, S. T., . . . Zimmet, P. (2017). Trends in diabetes and obesity in Samoa over 35 years, 1978–2013. *Diabetic Medicine, 34*(5), 654-661. doi:https://doi.org/10.1111/dme.13197

Manichaikul, A., Mychaleckyj, J. C., Rich, S. S., Daly, K., Sale, M., & Chen, W.-M. (2010). Robust relationship inference in genome-wide association studies. *Bioinformatics, 26*(22), 2867-2873. doi:10.1093/bioinformatics/btq559

Milborrow, S. (2020). earth: Multivariate Adaptive Regression Splines (Version R package version 5.3.0): Derived from mda:mars by Trevor Hastie and Rob Tibshirani Uses Alan

Miller's Fortran utilities with Thomas Lumley's leaps wrapper,. Retrieved from https://CRAN.R-project.org/package=earth

Minster, R. L., Hawley, N. L., Su, C.-T., Sun, G., Kershaw, E. E., Cheng, H., . . . McGarvey, S. T. (2016). A thrifty variant in CREBRF strongly influences body mass index in Samoans. *Nature Genetics, 48*(9), 1049-1054. doi:10.1038/ng.3620

Motoshima, H., Wu, X., Sinha, M. K., Hardy, V. E., Rosato, E. L., Barbot, D. J., . . . Goldstein, B. J. (2002). Differential regulation of adiponectin secretion from cultured human omental and subcutaneous adipocytes: effects of insulin and rosiglitazone. *J Clin Endocrinol Metab, 87*(12), 5662-5667. doi:10.1210/jc.2002-020635

O'Donovan, G., Owen, A., Kearney, E. M., Jones, D. W., Nevill, A. M., Woolf-May, K., & Bird, S. R. (2005). Cardiovascular disease risk factors in habitual exercisers, lean sedentary men and abdominally obese sedentary men. *International Journal of Obesity, 29*(9), 1063-1069. doi:10.1038/sj.ijo.0803004

Prakash, J., Mittal, B., Awasthi, S., Agarwal, C. G., & Srivastava, N. (2013). Hypoadiponectinemia in obesity: association with insulin resistance. *Indian journal of clinical biochemistry : IJCB, 28*(2), 158-163. doi:10.1007/s12291-012-0246-3

Ritchie, S. A., & Connell, J. M. C. (2007). The link between abdominal obesity, metabolic syndrome and cardiovascular disease. *Nutrition, Metabolism and Cardiovascular Diseases, 17*(4), 319-326. doi:https://doi.org/10.1016/j.numecd.2006.07.005

Romero-Corral, A., Lopez-Jimenez, F., Sierra-Johnson, J., & Somers, V. K. (2008). Differentiating between body fat and lean mass—how should we measure obesity? *Nature Clinical Practice Endocrinology & Metabolism, 4*(6), 322-323. doi:10.1038/ncpendmet0809

Samoa-WHO: Country Cooperation Strategy 2018-2022. (2017). Retrieved from

Shuster, A., Patlas, M., Pinthus, J. H., & Mourtzakis, M. (2012). The clinical importance of visceral adiposity: a critical review of methods for visceral adipose tissue analysis. *The British Journal of Radiology, 85*(1009), 1-10. doi:10.1259/bjr/38447238

Swinburn, B. A., Craig, P. L., Daniel, R., Dent, D. P., & Strauss, B. J. (1996). Body composition differences between Polynesians and Caucasians assessed by bioelectrical impedance. *International journal of obesity and related metabolic disorders : journal of the International Association for the Study of Obesity, 20*(10), 889-894. Retrieved from http://europepmc.org/abstract/MED/8910091

Swinburn, B. A., Ley, S. J., Carmichael, H. E., & Plank, L. D. (1999). Body size and composition in Polynesians. *Int J Obes Relat Metab Disord, 23*(11), 1178-1183. doi:10.1038/sj.ijo.0801053

Therneau, T. M. (2020). coxme: Mixed Effects Cox Models (Version R package version 2.2-16). Retrieved from https://CRAN.R-project.org/package=coxme

WHO. (2010). Pacific islanders pay heavy price for abandoning traditional diet. *Bull World Health Organ, 88*(7), 484-485. doi:10.2471/BLT.10.010710