Supplementary Methods

Genetic Disruption of WASHC4 Drives Endo-lysosomal Dysfunction and Cognitive-Movement Impairments in Mice and Humans

Jamie Courtland[1]*, Tyler W. A. Bradshaw[1]*, Greg Waitt[2], Erik J. Soderblom[2,3], Tricia Ho[2], Anna Rajab[4], Ricardo Vancini[5], Il Hwan Kim[2†], Ting Huang[6], Olga Vitek[6], Scott H. Soderling[3]

Author correspondence:
jlc123@duke.edu (JC); tyler.w.bradshaw@duke.edu (TWAB); greg.waitt@duke.edu (GW); erik.soderblom@duke.edu (EJB); tricia.ho@duke.edu (TH); drannarajab@gmail.com (DR); ricardo.vancini@duke.edu (RV); ikim9@uthsc.edu (IK); huang.tin@northeastern.edu (TH); o.vitek@northeastern.edu (OV); scott.soderling@duke.edu (SHS)

*These authors contributed equally to this work.

Present address:
†Department of Anatomy and Neurobiology, University of Tennessee Health Science Center, Memphis, TN 38163, USA

[1]Department of Neurobiology, Duke University School of Medicine, Durham, NC 27710, USA; [2]Proteomics and Metabolomics Shared Resource, Duke University School of Medicine, Durham, NC 27710, USA; [3]Department of Cell Biology, Duke University School of Medicine, Durham, NC 27710, USA; [4]Burjeel Hospital, VPS Healthcare, Muscat, Oman; [5]Department of Pathology, Duke University School of Medicine, Durham, NC 27710, USA; [6]Khoury College of Computer Sciences, Northeaster University, Boston, MA 02115, USA

Summary

In the review of this manuscript, significant concerns were raised by the reviewers about the validity of our statistical approach to perform protein- and module-level inference from our **WASH-iBioID** and **SWIP-TMT** proteomics datasets. Our previous statistical approach was dependent upon the R package `edgeR` to evaluate differential protein abundance. `edgeR` utilizes a negative binomial generalized linear model (NB GLM) framework, originally developed for analysis of read counts data generated in RNA-seq transcriptomics experiments. Previously, we failed to fully consider the validity of the NB GLM model used by `edgeR` for proteomics data. In response to this critique, we explore the goodness-of-fit of the NB GLM model for our `SWIP-TMT` data, and find evidence of a lack-of-fit. Thus, we revise our statistical approach and reanalyze our data making use of the recently published tool `MSstatsTMT`. `MSstatsTMT` uses a linear mixed-model (LMM) framework to model major sources of variation in a proteomics experiment. We extend the LMM framework used by `MSstatsTMT` to re-evaluate both protein- and module-level statistical comparisions. Despite evidence of a lack-of-fit for the NB GLM method used by `edgeR`, we find that the inferences we derived from our previous analysis are largely preserved in our reanalysis using `MSstatsTMT`.

Lack-of-fit of the Negative Binomial Model

Our previous approach is summarized as the 'Sum + IRS' method by Huang *et al.* (REF). Following protein summarization and Internal Reference Scaling (IRS) normalization, we applied `edgeR` to assess differential abundance of individual proteins and protein-groups or modules. The use of `edgeR` was based on work by Plubell and Khan, *et al.* (REFS) who describe IRS normalization and the use of `edgeR` for statistical testing in TMT mass spectrometry experiments. We failed however, to consider the overall adequacy of the NB GLM model for our TMT proteomics data.

Statisitical inference in `edgeR` is performed for each gene or protein in the dataset using a negative binomial framework in which the data are assumed to be adequately described by a NB distribution parameterized by a dispersion parameter, ϕ. Practically, the dispersion parameter accounts for the observed mean-variance relationship in proteomics and transcriptomics data.

As signal intensity in protein mass spectrometry is fundamentally related to the number of ions generated from a ionized, fragmented protein, we incorrectly inferred that TMT mass spectrometry data can be modeled as negative binomial count data. Based on this assumption, we justified the use of `edgeR`. Here, we reconsider the overall adequacy of the `edgeR` NB GLM model for TMT mass spectrometry data.

To evaluate the overall adequacy of the `edgeR` model, we plot the residual protein deviance statistics of all proteins against their theoretical, normal quantiles in a quantile-quantile (QQ) plot (***Figure 1***). The QQ plot addresses the question of how similar the observed data are to the theoretical distribution given by a NB GLM fit. A linear relationship between the observed and theoretical values is an indicator of goodness-of-fit. Deviation from this linear trend is evidence of a lack-of-fit.

Following protein summarization and normalization with `MSstatsTMT`, the data were fit with a simple NB GLM of the form `Abundance ~ Mixture + Condition` using `edgeR's glmFit` function which fits a NB GLM model to each protein or gene (the sub-subplot summaries) in the data. The dispersion parameter ϕ can take several forms, and `edgeR` supports three different dispersion metrics: 'common', 'trended', and 'tagwise'. ***Figure 1*** illustrates the divergence of the observed deviance statistics from the theoretical distribution for our TMT data fit with the NB GLM model. These plots emphasize the overall lack-of-fit for proteomics data with the `edgeR` model.

Given our experimental design, `MSstatsTMT` fits an analagous LMM: `Abundance ~ Condition + (1|Mixture)`. The QQ plot in ***Figure 1*** indicates that the data are well described by `MSstatsTMT's` LMM framework, which does not depend upon the negative binomial assumption.

Reanalysis of SWIP^{P1019R} TMT Proteomics

Of note, most tools for analysis of protein mass spectromety data are derived from tools originally developed for analysis of genomics and transcriptomics data. An exception to this norm is `MSstatsTMT`, an extension of `MSstats` for analysis of TMT proteomics experiments.

`MSstatsTMT` utilizes a linear mixed-model framework. The strength of LMMs lies in their flexibility. In mixed-models, the response variable is taken to be a function of both fixed- and mixed-effects. Using LMMs we can untangle the variance attributable to the biological effect we are interested in from the experimental and biological covariates which mask this response.

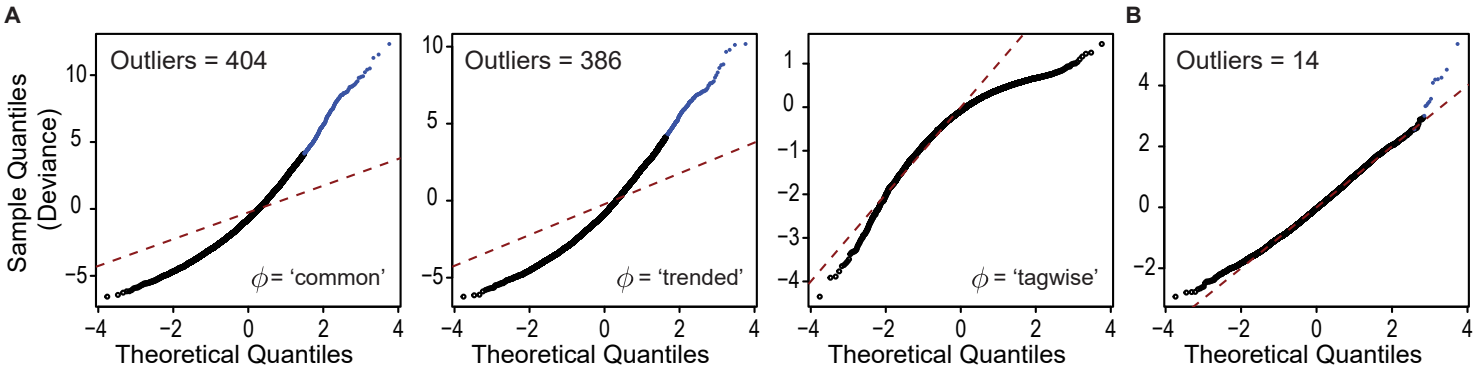

Figure 1. Goodness-of-fit of `edgeR` (A), and `MSstatsTMT` (B) statistical approaches. The overall adequacy of the linear models fit to the data were assessed by plotting the residual deviance for all proteins as a quantile-quantile plot (McCarthy *et al.*, (2012)). **(A)** For analysis with `edgeR`, The normalized protein data from `MSstatsTMT` were fit with a negative binomal generalized linear model (NBGLM) of the form: `Abundance ~ Mixture + Condition`. Where `Mixure` is an additive blocking factor that accounts for variablity between experiments. The NB framework used by edgeR utilizes a dispersion parameter to account for mean-variance relationships in the data. The dispersion parameter can take several forms. `edgeR` supports three dispersion models: 'common', 'trended', and 'tagwise'. However, when using `edgeR's` robust quasi-likelihood test methods, only global (i.e. 'common' or 'trended') dispersion metrics are appropriate (see `edgeR::glmQLFit's` documentation). We plot the protein-wise deviance from the data fit withe ach of the disperions parameters. Protein-wise deviance statistics were transformed to normality and plotted against theoretical normal quantiles using the `edgeR::gof` function. **(B)** For analysis with `MSstatsTMT`, the normalized protein data were fit with a linear mixed-effects model (LMM) of the form: `Abundance ~ 0 + Condition + (1|Mixture)`. Where `Mixture` represents the random effect of `Mixture`. The residual deviance and degrees of freedom were extracted from the fitted models, z-score normalized, and plotted as in (A). Proteins with a significantly poor fit are indicated as outliers in blue (Holm-adjusted P-value < 0.05).

If the set of possible levels of the covariate is fixed and reproducible then the factor is modeled as a fixed-effect parameter. In contrast, if the levels of an observation reflect a sampling of the set of all possible levels, then the covariate is modeled as a random effect. Random or mixed-effects represent categorical variables that reflect experimental or observational "units" in the data set (REF:Bates). As such, mixed-effect parameters account for the variation occurring among all of the lower level units of a particular upper level unit in the data (REF:Bates). For this reason, mixed-models may also be referred to as heirarchical models.

Huang *et al.* created `MSstatsTMT`, an R package for data normalization and hypothesis testing in multiplex TMT proteomics experiments. They outline a common vocabulary for describing the experimental design of TMT MS experiments. A TMT experiment consists of the analysis of `m = 1` ... `M` concatenations of isobarically labeled samples or `Mixtures`. This mixture is then analyzed by the mass spectrometer in a mass spectrometry `Run`. This mixture is often fractionated into multiple liquid chromotography `Fractions` to decrease sample complexity, and thereby increase the depth of proteome coverage. Within a mixture, each of the unique TMT channels is dedicated to the analysis of `c = 1` ... `C` individual biological or treatment `Conditions`. There may then be `b = 1` or more `B` biological replicates or `Subjects`. Finally, a single TMT mixture may be repeatedly analyzed in `t = 1` ... `T` technical replicate mass spectrometry runs.

The following equation is a LMM formula which describes a general TMT experiment composed of `M` mixtures, `T` technical replicates of mixture, `C` conditions, and `B` biological subjects. The abundance of a given protein, Y_{mcbt}, is then:

$$Y_{mcbt} = \mu + Mixture_m + TechRep(Mixture)_{m(t)} + Condition_c + Subject_b + \epsilon_{mcbt} \quad (1)$$

$$\begin{gathered} \sum_{c=1}^{C} Condition_c = 0 \\ Subject_{mcb} \overset{iid}{\sim} N(0, \sigma_S^2) \\ Mixture_m \overset{iid}{\sim} N(0, \sigma_M^2) \\ TechRep(Mixture)_{t(m)} \overset{iid}{\sim} N(0, \sigma_T^2) \\ \epsilon mtcb \overset{iid}{\sim} N(0, \sigma^2) \end{gathered} \quad (2)$$

The model's constraints distinguish fixed and random components of variation in the response. `Mixture` is a mixed-effect and represents the variation between

TMT mixtures. By definition mixed-effects are assumed to be independent and normally distributed (iid). `TechRep(Mixture)` represents random variation between replicate mass spectrometry runs. The term `Subject` cooresponds to each unique biological replicate and represents biological variation among the levels of the fixed-effect term `Condition`. The term ϵ_{mtcb}, is a random-effect representing both biological and technical variation, quantifying any remaining error.

If a component of the model is not estimable, then it is removed. For example, if there is no technical replication of mixture `(T=0)`, the model is reduced to:

$$Y_{mcbt} = \mu + Mixture_m + Condition_c + \epsilon_{mcb} \tag{3}$$

In the reduced model, biological variation among individual `Subjects` is captured by the term `Condition`, and is thus omitted.

Hypothesis Testing

`MSstatsTMT` performs protein-wise comparisons between `Conditions` of biological `Subjects` by a contrast of conditioned means obtained from fitting the data with a linear mixed-effects model expressing the major sources of variation in the experimental design.

Model-based testing of differential abundance between pairs of conditions is done by comparing the estimates obtained from the fit LMM. We are interested in testing the null hypothesis $H0 : l^T\beta = 0$. Kutzenova *et al.,* derive a test statistic for such contrasts (Kutzenova2017):

$$t = \frac{l^T\hat{\beta}}{\sqrt{l\sigma^2\hat{V}l^T}} \tag{4}$$

We obtain the model estimates $\hat{\beta}$, error σ^2, and variance-covariance matrix $\hat{V}$ from the fitted model. Together $\sigma^2 * \hat{V}$ is the scaled variance-covariance matrix describing the error estimates of the model's mixed-effect parameters. Given l^T, a vector of sum 1 specifying the positive and negative coefficients of the comparison, the numerator of the equation is then the fold change of a given comparison, and together the denominator represents the standard error of the contrast.

The degrees of freedom for the contrast are derived using the Satterthwaite moment of approximation method (Kutzenova2017). Finally, given the t-statistic, which is assumed to follow an approximate χ^2 distribution, and the degrees of freedom, a p-value is calculated. P-values for the protein-wise tests are adjusted

using the Benjamini-Hochberg FDR method (REF).

TMT Channel

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16
Mix1	WT-5K	WT-9K	WT-12K	WT-15K	WT-30K	WT-79K	WT-129K	QC1	MUT-5K	MUT-9K	MUT-12K	MUT-15K	MUT-30K	MUT-79K	MUT-129K	QC2
Mix2	WT-5K	WT-9K	WT-12K	WT-15K	WT-30K	WT-79K	WT-129K	QC1	MUT-5K	MUT-9K	MUT-12K	MUT-15K	MUT-30K	MUT-79K	MUT-129K	QC2
Mix3	WT-5K	WT-9K	WT-12K	WT-15K	WT-30K	WT-79K	WT-129K	QC1	MUT-5K	MUT-9K	MUT-12K	MUT-15K	MUT-30K	MUT-79K	MUT-129K	QC2

Figure 2. Experimental Design. We performed three 16-plex TMT experiments. Each TMT mixture is a concatenation of 16 labeled samples. In each experiment we analyzed 7 subcellular `BioFractions` prepared from the brain of a 'Control' or 'Mutant' mouse. In all we analyzed 3 `Subjects` from each Condition. Each `Mixture` includes two `Channels` dedicated to the analysis of a common quality control sample.

SWIP-TMT Proteomics Experimental Design

In our experiment, the fixed-effect term `Condition` represents the 14 combinations of `Genotype` and `BioFraction` obtained from subcellular fractionation of the brains of 'Control' and SWIP^{P1019R} 'Mutant' mice. We refer to these as `BioFractions` to distinguish them from a MS `Fraction`. Our TMT proteomics experimental design is summarized in ***Figure 2***.

In our experiment, each TMT mixture contains seven repeated measurements made from each biological `Subject`. To account for this source of intra-Subject variability, we should include the random-effect term `Subject` representing the random error within a subject. However, in our design `Mixture` is confounded with the term `Subject`. In each mixture we analyzed all `BioFractions` from a single Control and Mutant mouse. Thus we can choose to account for the effect of `Mixture` or `Subject`, but not both. We choose to account for variability of `Mixture` under the assumption that the effect of this experimental batch effect is greater than the variance attributable to the random variability inherent to repeated measurements of each subject. Thus we omit the term `Subject`. The reduced model is then the same as equation (EQ) when `Condition` is the interaction of `Genotype:BioFraction`.

Protein level comparisions

Using `MSstatsTMT` we assesssed two protein-level comparisons:

- 'intra-BioFraction' contrasts

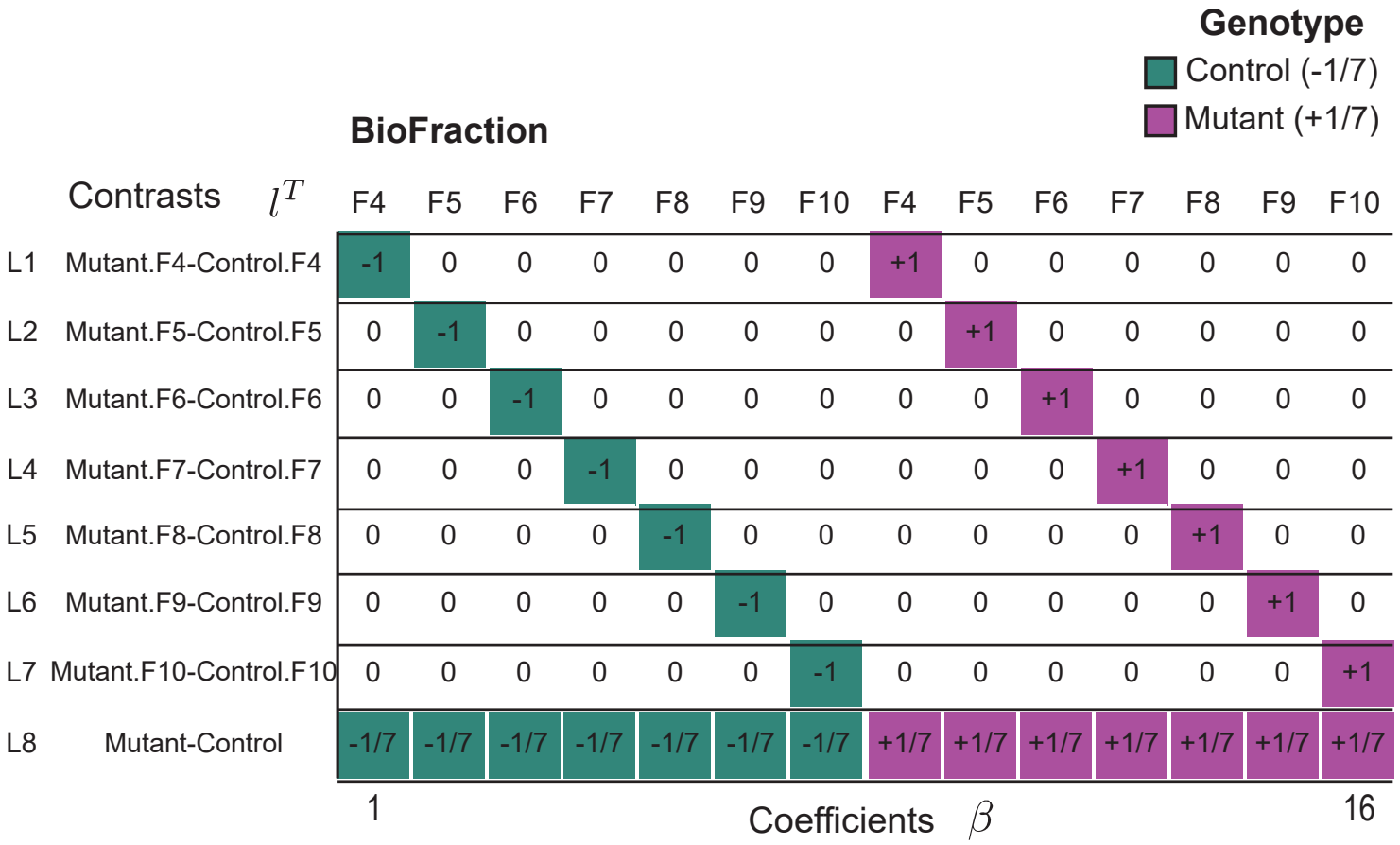

Figure 3. Statistical Comparisons. We assessed two types of contrasts. Each row of the matrix specifies a contrast between positive and negative coefficients in the mixed-effects model fit to each protein. Contrasts1-7 are 'intra-BioFraction' contrasts that specify the pairwise comparisons of Control and Mutant groups for a single fraction. In Contrast 8 we compare 'Mutant-Control' and asses the overall difference of 'Control' and 'Mutant' conditions. Each contrast is a vector of sum 1.

- 'Mutant-Control' contrast

'Intra-BioFraction' comparisons are the 7 pairwise comparisons of 'Control' and 'Mutant' protein abundance for each subcellular fractions. We also assessed differential abundance for the overall comparison between 'Control' and 'Mutant' groups. Each of these contrasts is represented by a vector, l^T, which specifies a comparison between coefficients in the LMM. ***Figure 3*** illustrates a matrix defining all 8 contrasts.

`MSstatsTMT` attempts to automatically parse the experimental design and fit an appropriate LMM for the experimental design. In order to understand and extend the function of `MSstatsTMT`, we extracted `MSstatsTMT's` core model-fitting and statistical testing steps and illustrate them here.

Following data preprocessing, summarization, and normalization, statistical

inference by `MSstatsTMT` can be summarized in two steps:

- Fit each protein with the appropriate LMM, and then
- given the fitted model, assess a contrast of interest.

At the core of the model fitting-step is the R package `lme4` which implements mixed-effects models with its function `lmer`. The package `lmerTest` extends `lme4's` functionality and enables the computation of Satterthwaite degrees of freedom.

```
# load dependencies
library(dplyr)
library(data.table)

#library(SwipProteomics)

# load the data
data(swip)
data(msstats_prot)

# formula to be fit to WASHC4, aka SWIP:
fx <- formula("Abundance ~ 0 + Genotype:BioFraction + (1|Mixture)")

# fit the LMM
fm <- lmerTest::lmer(fx, msstats_prot %>% filter(Protein == swip))

# examine the model's summary
summary(fm, ddf = "Satterthwaite")
```

Coefficient	Estimate	Std. Error	df	t value	p value
Mutant:F4	5.404300	0.121126	17.30594	44.61718	2.59e-19
Control:F4	6.710959	0.121126	17.30594	55.40477	6.26e-21
Mutant:F5	5.567441	0.121126	17.30594	45.96405	1.56e-19
Control:F5	6.945583	0.121126	17.30594	57.34180	3.47e-21
Mutant:F6	5.640188	0.121126	17.30594	46.56463	1.24e-19
Control:F6	7.240081	0.121126	17.30594	59.77313	1.7e-21
Mutant:F7	5.631680	0.121126	17.30594	46.49440	1.28e-19
Control:F7	7.321074	0.121126	17.30594	60.44180	1.4e-21
Mutant:F8	5.492772	0.121126	17.30594	45.34759	1.96e-19
Control:F8	7.129632	0.121126	17.30594	58.86129	2.21e-21
Mutant:F9	5.781022	0.121126	17.30594	47.72734	8.15e-20
Control:F9	6.954472	0.121126	17.30594	57.41518	3.39e-21
Mutant:F10	5.784403	0.121126	17.30594	47.75525	8.07e-20
Control:F10	7.618697	0.121126	17.30594	62.89894	7.04e-22

The model's estimates, β, represent our best estimate of the mean protein abundance in the 14 conditions of `Genotype:BioFraction`. To illustrate a comparison, we define a contrast comparing 'Mutant:F7' and 'Control:F7'. The function `lmerTestContrast` performs model-based comparisons of conditions defined by a contrast matrix. While the work done by this function is the same as the work done internally by `MSstatsTMT's groupComparisonsTMT` function, `lmerTestContrast` is more flexible. Provided the correct contrast, we easily assess the overall comparison between 'Mutant' and 'Control' groups.

```
# create a contrast
coeff <- lme4::fixef(fm)
contrast7 <- setNames(rep(0,length(coeff)), nm = names(coeff))
contrast7["GenotypeMutant:BioFractionF7"] <- +1 # positive coeff
contrast7["GenotypeControl:BioFractionF7"] <- -1 # negative coeff

# evaluate contrast
lmerTestContrast(fm, contrast7)
```

Contrast	log2FC	SE	Tstatistic	Pvalue	DF
Mutant:F7-Control:F7	-1.689	0.151	-11.153	2.09e-11	26

```
# use convenience function to contruct a contrast
lT <- getContrast(fm, "Mutant","Control")

# assess the comparison
lmerTestContrast(fm, lT)
```

Contrast	log2FC	SE	Tstatistic	Pvalue	DF
Mutant-Control	-1.517	0.057	-26.496	2.42e-20	26

Goodness-of-fit

It is useful to consider the goodness-of-fit of our LMM. A straight forward measure of a LMM's quality is the Nakagawa coefficient of determination (REF). Nakagawa's conditional R^2 is interpreted as the total variance explained by a LMM (R^2_{total}). The marginal R^2 is interpreted as the variance explained by the LMM's fixed-effects (R^2_{fixed}).

We implment Nakagawa's coeffficient of determination using the `r.squaredGLMM` function taken from the `MuMin` package (REF).

```
# assess gof with Nakagawa coefficient of determination
r.squaredGLMM.merMod(fm)

##            R2m       R2c
## [1,] 0.9353344 0.949433
```

We can see the total variation explained by the model, R^2_c, is 0.949. The variance explained by fixed-effects, `Genotype:BioFraction`, equates to 0.935 (R^2m). A vast majority of the variance is attributable to the fixed-effects. Only about 1.5% of the remaing variance is attributable to mixed-effects and the residual variance.

Module-level analysis

We wish to extend the LMM framework developed by `MSstatsTMT` to perform inference at the level of protein groups or modules. That is, for module-level comparisons, we are interested in the overall affect of Genotype on a group of proteins. Where modules are groups of covarying proteins which represent biological niches defined by proteins that localized together in subcellular space.

Here we hypothesize that the proteins within a module, which are a subset of the overall proteome, are a part of a common group, a module, with a common mean effect. Proteins within a module are correlated observations which we model as a mixed-effect. We take the stance that `Protein` is a mixed-effect in the view that we are primarily interested in making inference about the overall distribution of responses for a module rather than among its sublevels.

We model protein groups or modules by adding the mixed-effect term `Protein`

to the LMM equation (EQ):

$$Y_{mcbt} = \mu + Mixture_m + Condition_c + Protein_p + \epsilon_{mcb}$$
$$Protein_p \overset{iid}{\sim} N(0, \sigma_P^2) \tag{5}$$

The term `Protein` is associated with a variance component σ_P.

As a means of example, we demonstrate an ideal module, by fitting the LMM to the 5 WASH complex proteins. As before, we calculate the coefficient of determination for LMM's with the `r.squaredGLMM` function (REF).

```
# the module-level formula to be fit:
fx <- "Abundance ~0 + Genotype:BioFraction + (1|Mixture) + (1|Protein)"

# load WASH Complex proteins
data(washc_prots)

fit <- lmer(fx, msstats_prot %>% filter(Protein %in% washc_prots))

r.squaredGLMM.merMod(fit)

##             R2m       R2c
## [1,] 0.7620866 0.8928053
```

Again, we consider the variance explained by the model as a measure of its overall quality. Our model explains 89.0% of the total variance. The Fixed-effect term `Genotype:BioFraction` explains the majority of the variance ($R_m^2 = 0.762$). The remaining variance, 1.3%, is attributable to the combination of mixed-effects `Mixture` and `Protein` as well as the residual variance.

It is useful to consider the variation (σ^2) of the individual mixed-effect terms. These can be assessed with `lme4's VarCorr` function.

```
# calculate variance of mixed-effects
var_df <- as.data.frame(lme4::VarCorr(fit,comp="Variance"))
mixef_var <- setNames(var_df$vcov,nm=var_df$grp)

# as a percent of the total mixed-effect variance:
mixef_var/sum(mixef_var)

##    Protein    Mixture   Residual
## 0.52871411 0.02072397 0.45056191
```

The R package `variancePartition` enables us to calculate the percent variance explained by a LMM's parameters. To do so, it expects all terms to be mixed-effects. ***Figure 4***.

```
# load variancePartition
library(variancePartition)

# calculate partitioned variance
fx <- "Abundance ~ (1|Genotype) + (1|BioFraction) + (1|Mixture) + (1|Protein)
fit <- lmer(fx, data = msstats_prot %>% filter(Protein %in% washc_prots))

calcVarPart(fit)

## BioFraction    Genotype     Mixture     Protein   Residuals
## 0.032960635 0.822069159 0.002843637 0.074146798 0.067979772
```

We can see that the majority of the variance explained by the LMM fit to the WASH complex is attributable to `Genotype`. The mixed-effect terms `Protein` and `Mixture` account for a small fraction of the overall variance explained by the model.

As our overall goal is to identify groups or modules of proteins that strongly covary together, our clustering approach should maximize the variance explained by a module's fixed-effect parameters (Genotype + BioFraction) while minimizing the variance among its individual proteins. An ideal module is a perfect summary

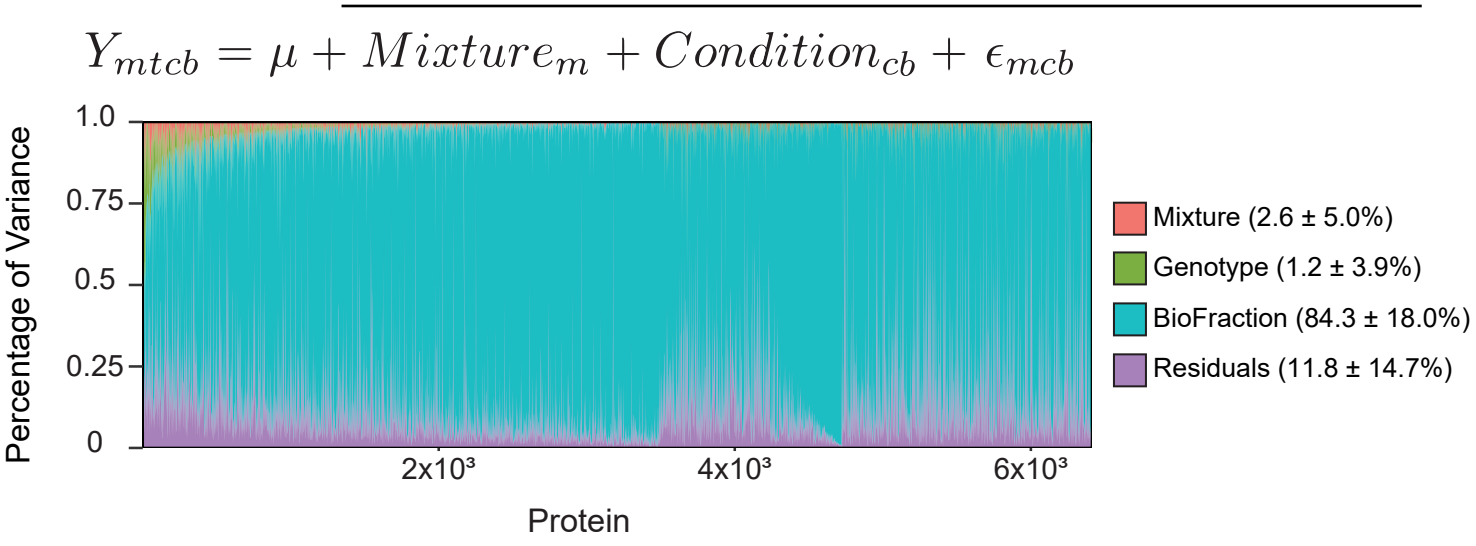

Figure 4. Analysis of Variance Components. The proportion of variance explained by Genotype, BioFraction, Mixture, and remaining residual error (subplot error) for all proteins. Note while the contribution of Mixture seems negligiable, its average for all proteins is approximately twice the average percent variance explained by Genotype. BioFraction explains the majority of the variance for all proteins. Analysis done with `variancePartition::calcVarPart`.

of its protein constituents, $PVE_{Protein} = 0$. We use this idea of a module's quality to supervise our clustering approach.

$$Quality_{Module} = \frac{PVE_{Genotype} + PVE_{BioFraction}}{PVE_{Protein}} \tag{6}$$

Network Construction

Using our **SWIP-TMT** dataset, we aim to identify modules or groups of proteins that covary together across subcellular space. Prior to building the co-variation network, other sources of variation should be removed. Although `MSstatsTMT` handles the batch effect inherent in experiments with multiple TMT mixtures, it is necessary to remove this effect prior to building the network. We removed the effect of `Mixture` using `limma::RemoveBatchEffect`. These adjusted data are used for network construction and plotting but not statistical modeling.

Prior to network construction, we removed protein models with poor fit ($R^2_{total} < 0.7$; n=791 proteins). Removing this noisey proteins facilitation module identification and improves overall module quality.

The final network was constructed using data from both 'Control' and 'Mutant' samples after adjusting for batch (Mixture). The final dataset included 42 samples and 6,119 proteins. The protein covariation network was build by calculating the Pearson correlation for all pairwise comparisons of proteins.

We performed network enhancement to remove biological noise from the network. This step is essential for module detection. Network enhancement reweights the network's edges and has the overall effect of making the network sparse. Conceptually this step is related to the soft-thresholding approach taken by WGCNA or WPCNA analysis workflows (REFS), but has the befinit of not assuming that the network has an overall scale free topology. Without reweighting or enhancing the network, most extant clustering algorithms fail to detect communities in the dataset. Network enhancment has the effect of making the network sparse and facilitates the identification of network structure (FIG).

(***Figure 6***)
(***Figure 5***)

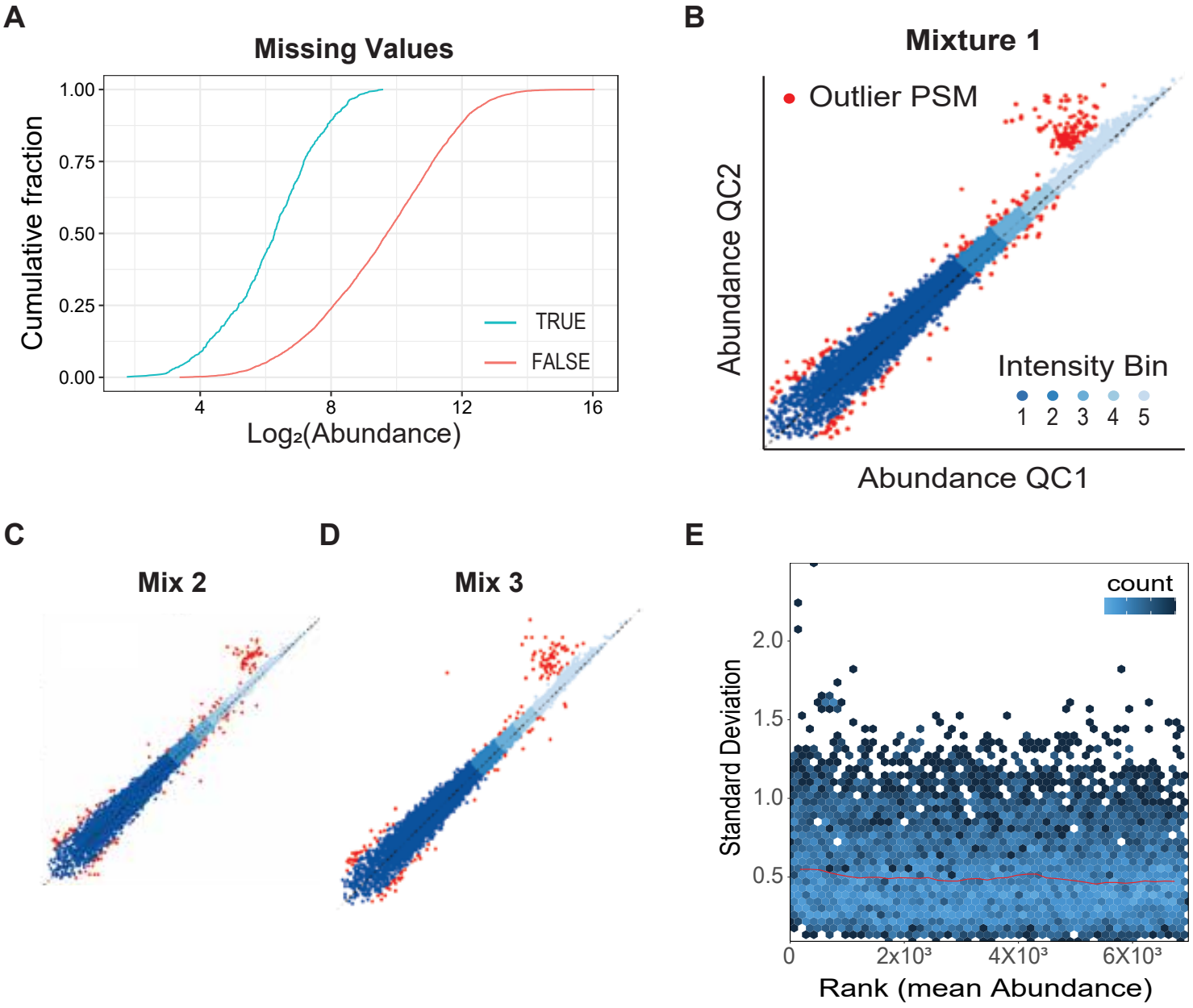

Figure 5. Missing value imputation and PSM outlier removal. A B C D

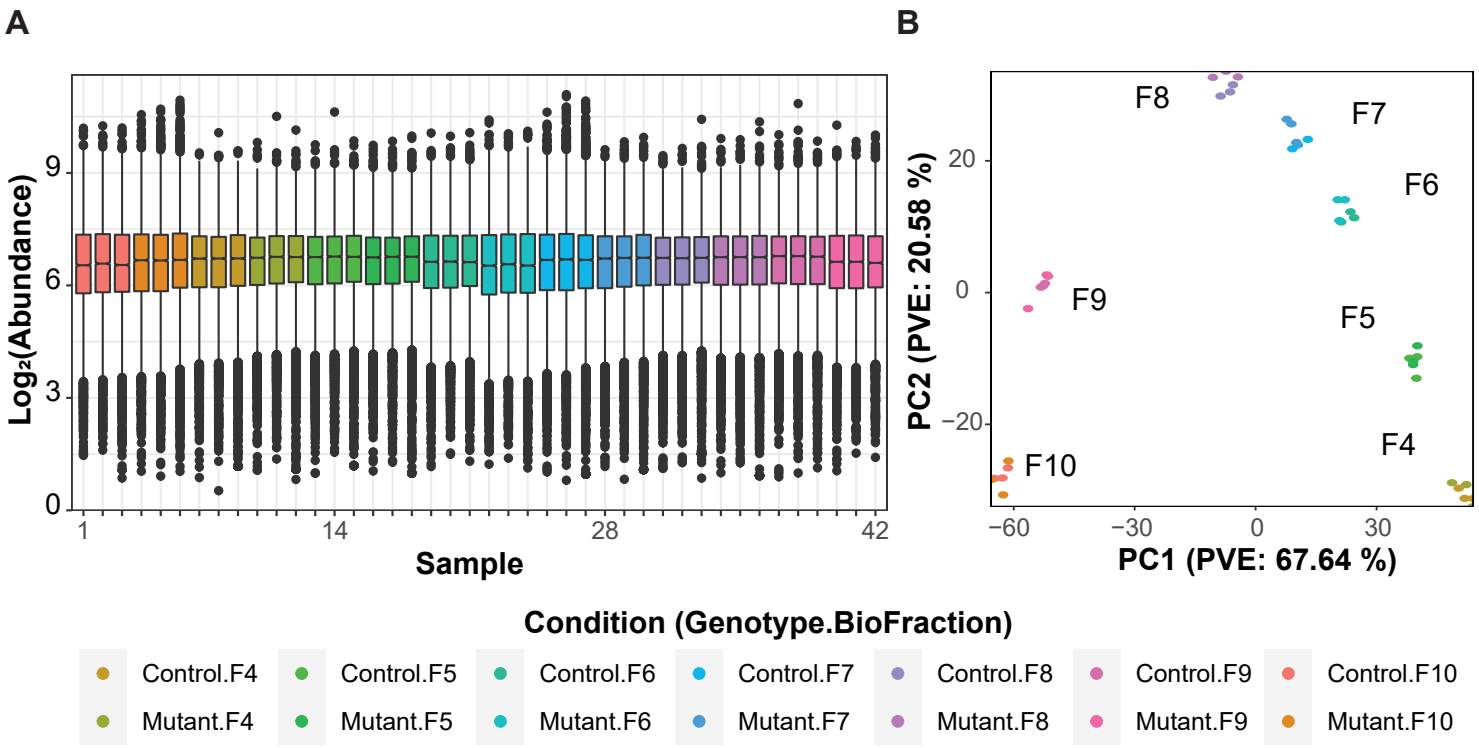

Figure 6. Data Normalization and PCA. A B