1 Methods

Model Description

Let $D$, $C$, and $P$ be the total number of persons, constructs, and phenotypes respectively. Let $W$ represent our complete dataset of discrete observations. We may subdivide $W$ into observations associated with each person: $W = \{W_1, ..., W_D\}$. Furthermore, each person’s observations, $W_d$, may be subdivided by construct: $W_d = \{W_{d1}, ..., W_{dc}\}$. Finally, each $W_{dc}$ may be further decomposed into a set of individual observations: $W_{dc} = \{w_{dc1}, ..., w_{dc(N_{dc})}\}$, where $N_{dc}$ is the total number of observations of construct $c$ associated with person $d$.

MC3M assumes the existence of a set of phenotypes, $\Phi = \{\phi_1, ..., \phi_P\}$, each of which has a component dedicated to each construct: $\phi_p = \{\phi_{p1}, ..., \phi_{PC}\}$. Phenotypes are modeled as a set of $C$ independent discrete probability vectors. Each of these vectors is a discrete probability distribution over the corresponding construct’s survey response tokens. These phenotype-token distributions are sampled from a set of Dirichlet distributions with fixed parameters, $\beta = \{\beta_1, ..., \beta_C\}$.

$$P(\Phi; \beta) = \prod_{p=1}^{P} \prod_{c=1}^{C} \text{Dir}(\phi_{pc}; \beta_c) \quad (1)$$

MC3M also assumes a set of person-specific distributions over phenotypes or person-phenotype distributions: $\Theta = \{\theta_1, ..., \theta_D\}$. These distributions are modeled as samples from a single Dirichlet distribution with fixed parameter $\alpha$.

$$P(\Theta; \alpha) = \prod_{d=1}^{D} \text{Dir}(\theta_d; \alpha) \quad (2)$$

In MC3M, the observations in $W$ are assumed to be generated by interactions among the probability vectors in $\Theta$ and $\Phi$. To illustrate, consider the observations for a single person, $d'$ and construct $c'$. Each individual observation $w_{d'c'n}$ in $W_{d'c'} = \{w_{d'c'1}, ..., w_{d'c'(N_{dc'})}\}$ is modeled as a sample from the $c'$th component of one of the phenotypes in $\Phi$. The identity of this phenotype is obtained by sampling a phenotype assignment, $z_{d'c'n}$ from $\theta_{d'}$, where $z_{d'c'n} \in \{1, ..., P\}$. Thus, MC3M assumes that each observation in $W$ is obtained by first sampling an assignment from a person-phenotype distribution in $\Theta$, picking out the assigned phenotype from $\Phi$, and finally sampling the observation from one of the phenotype-token distributions. This allows us to write out the conditional probabilities of all the assignments, $Z$, and observations, $W$, as follows.

$$P(Z|\Theta) = \prod_{c=1}^{C} \prod_{d=1}^{D} \prod_{n=1}^{N_{dc}} \theta_d(z_{cdn}) \quad P(W|\Phi) = \prod_{c=1}^{C} \prod_{d=1}^{D} \prod_{n=1}^{N_{dc}} \phi(z_{cdn})c(w_{cdn}) \quad (3)$$

Taken together, the probability distributions in equations 1, 2, and 3 fully specify the generative model for MC3M.

Inference

Here we describe the inference algorithm we implement to obtain estimates of the variables $\Theta$ and $\Phi$ given our training data, $W$.

We adopt a Bayesian inference approach, and seek to obtain posterior estimates of $\Theta$ and $\Phi$. Unfortunately, these posteriors require we estimate the marginal likelihood of our data, which is intractable. Therefore, we make use of Markov chain Monte Carlo approximate inference methods, which do not necessitate estimation of this marginal likelihood. Specifically, we implement a collapsed Gibbs sampling algorithm for our model.

Briefly, we integrate out of the model’s joint distribution the latent variables $\Theta$ and $\Phi$. We then iteratively sample each of the the assignment variables in $Z$ from its complete conditional distribution which is the distribution of the assignment variable conditioned on the values of all other remaining variables in the model. We repeat this sampling procedure until observing convergence in the model’s
collapsed likelihood. At this point, we may recover the values of $\Theta$ and $\Phi$ as Monte Carlo estimates of their expectations with respect to the collapsed likelihood.

**Held-out Likelihood Estimation**

Our model has several fixed parameters including $\alpha$, $\beta$, and $P$. Each of these plays a role in determining how well our model fits the training set and generalizes to held-out data. In this work, we focus on optimizing the value of $P$, and hold $\alpha$ and $\beta$ fixed throughout all our experiments.

To identify the optimal value for $P$, we chose to estimate the posterior likelihood of a held-out dataset for a range of values of $P$. To do so, for each value of $P$ ran our inference algorithm over three independent runs. We then utilized a Chib-style estimator described in [1] and [2] to estimate the held-out likelihood for each model. We took the optimal value of $P$ as that which yielded the highest average held-out likelihood.

**Predictive Modeling**

We use MC3M’s person-specific phenotype distributions, $\Theta = \{\theta_1, ..., \theta_D\}$, as feature vectors for a binary prediction problem targeting elevated weight status. We choose elastic net (EN) logistic regression as our model for this task. Our goal was to use the model’s coefficients to determine which phenotypes were significant positive or negative predictors for elevated weight status.

EN logistic regression has several hyperparameters including the regularization strength, $\lambda$, and the L1/L2 elastic net mixing parameter, $\gamma$. Note that with $\gamma = 0.0$ only the L2 penalty is used; with $\gamma = 1.0$ only the L1 penalty is used. Otherwise, the model uses a mixture of penalties. We optimize $\lambda$ and $\gamma$ using 10-fold cross validation and grid search: $\lambda \in \{0.01, 0.5, 0.1, 1.0, 5.0, 10.0\}$, $\gamma \in \{0.0, 0.1, 0.2, ..., 0.9, 1.0\}$. The areas under the ROC (AUROC) and Precision-Recall (AUPRC) curves were used as target metrics. For each hyperparameter setting, we generate a point estimate for each metric and bootstrapped 95% confidence intervals (CIs). Bootstrapped metric distributions were generated by pooling true labels and label predictions over all held-out folds, sampling from this pool with replacement, and calculating the metric for each sample (10 thousand samples for each hyperparameter setting). A wide range of parameter values yielded similar performance. Therefore, we selected the values yielding the highest AUPRC and second highest AUROC: $\lambda = 10.0$, $\gamma = 0.8$. These values were used in all subsequent experiments.

We used the following bootstrapping procedure to identify significant coefficients. First, we sampled instances with replacement from the training set such that the number of positive and negative cases was preserved. Next, we trained an EN logistic regression model, and recorded the fitted coefficient values. This process was repeated for 10 thousand iterations yielding bootstrap distributions for each coefficient. We used these distributions to define bootstrapped 95% CIs. Significant coefficient were taken as those whose 95% CIs were non-overlapping with the null value, 0.

**References**

[1] Iain Murray and Russ R Salakhutdinov. Evaluating probabilities under high-dimensional latent variable models. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, Advances in Neural Information Processing Systems, volume 21. Curran Associates, Inc., 2009.

[2] Hanna M. Wallach, Iain Murray, Ruslan Salakhutdinov, and David Mimno. Evaluation methods for topic models. In Proceedings of the 26th Annual International Conference on Machine Learning, pages 1105–1112. Association for Computing Machinery, New York, NY, USA, June 2009.