Trait selection for matrix factorization

Ashton Omdahl

January 2022

1 Problem

In using latent factor analysis to examine shared genetic patterns in traits (i.e. across GWAS data), the choice of which traits to evaluate is important. Indeed, as demonstrated in the Tanigawa (), Udler (2018) and Burren (2020) papers, the choice of traits can make a tremendous difference on the outcome and focus of your factorization. The question is, which GWAS traits should be included? In Udler (2018), domain knowledge drove the selection of traits. In Tanigawa (), as many GWAS traits as possible were used. In order to focus in on a narrow set of traits to explore a disease etiology of interest, picking the right GWAS traits is important.

In this problem formulation, suppose we already have some set of M traits/diseases that we know are relevant (our prior knowledge), centered on some disease of interest. We then have a set of L additional GWAS studies with phenotypes that could be relevant to our disease space of interest, but may or may not be. We wish to determine which ones will be worth including, and which ones won't, in trying to understand the genetic etiology of our disease. (this is a very abstract objective function, I know. I'm not sure how to make it concrete without being too limiting here).

2 Existing Methods

This problems falls under a sort of unsupervised feature selection problem, which has been examined in the literature over the past few years. A few interesting papers on that

- Review paper: https://dl.acm.org/doi/pdf/10.1145/3136625
- Feature selection for multi-cluster data: ttp://people.cs.uchicago.edu/ xiaofei/SIGKDD2010-Cai.pdf
- In the biology setting: https://www.frontiersin.org/articles/10.3389/fgene.2020.603808/full
- Mixed spectral feature selection https://reader.elsevier.com/reader/sd/pii/S0031320317302923?token=F6 east-1originCreation=20220121131453

While I'm still investigating these, the my problem is unique in that we aren't choosing features totally *de novo*- we know what basis of information we are looking for, and a set of phenotypes that are already relevant (a prior). While I've only taken a cursory look, of the methods I have seen they don't accomodate a prior currently.

3 Proposed method

This proposal uses the chosen traits as a starting point, and keeps the selection process in the matrix factorization framework.
Let $X = UV^T$, where X is our current $N \times M$ data of GWAS summary statistics, U our loadings (SNP contributions to latent factors, $N \times K$) and V our factors (trait assignment to latent factors, $M \times K$). Let

$$x_{n,m} \sim N(u_n^T v_m, \sigma_x^2)$$

$$u_{n,k} \sim N(0, \sigma_u^2)$$

$$v_{t,k} \sim N(0, 1/\alpha_k)$$

such that the vector v_k has a shared precision under an automatic relevance determining (ARD) prior. The procedure is as follows

1. Perform factor analysis on your initial set of M traits as above, learning the distribution of latent factors v_k and their corresponding precisions.

2. For each additional GWAS trait T_l for $l \in \{1...L\}$ (T_l is an $M \times 1$ vector), do

   (a) Project GWAS T_l onto learned loadings: $\hat{V}_l = U^T T_l$

   (b) Calculate the probability for entries as extreme as those in $\hat{V}_l$ or more extreme (i.e. a p-value for each), as in $\phi(v_{l,k}) = \int_{|v_{l,k}|}^{\infty} p(v_{t,k}) dv$

   (c) Determine if this distribution of p-values corresponds to a $unif[0, 1]$ distribution (probably with a KS test), and store the test statistic. (The intuition here is that we want just a few cells that are heavily loaded, but the rest to be pretty small. So we would hope that in the case of a relevant trait with a few highly loaded factors, the distribution would be significantly non-uniform, i.e. inflated)

3. Generate a null distribution for the test statistics gathered in 2(c) above by randomly sampling from each V_k distribution, calculating the corresponding p-values, and testing for uniformity as in 2c.

4. Determine p-values p_l for true tests statistics (2c) based on this null distribution.

5. Select which traits to add to the original M set of traits. This may be done in a few different ways

- Select traits corresponding to $p_l < \alpha$ for specified α. This may be calibrated using real data (i.e. the Udler dataset of relevant traits) or by some kind of simulation.
- Select the trait with the smallest $p_l < \alpha$, and then repeat steps 1 and 2 on the new traits and repeat this process until no more traits are added.

Okay, cool bro. Problem here is that using that probability would favor traits that are all just 0,0,0,0, since that is the mean. Our distribution here is over factors, whereas the distribution of v is over traits. Huh.