Distribution Learning ppt

Wei Ying, Yiming Li

November 29, 2023

Overview

Problem Statement

Suppose in the reference site, we observe n i.i.d. samples $(X_i,\ T_i,\ \boldsymbol{G}_i)$, where the genotypes of the i-th subject on a segment of DNA of interest $\boldsymbol{G}_i \in \mathbb{R}^p$, the gene expression level of the i-th subject in the RNA sequence data $X_i \in \mathbb{R}^1$, and the phenotypes of the i-th subject in a GWAS data $T_i \in \mathbb{R}^1$.

Problem Statement

Assuming conditional independence of $\boldsymbol{T}$ and $\boldsymbol{X}_{\boldsymbol{G}}$ given $\boldsymbol{G}$, we have

$$\begin{aligned}\mathbb{E}[t|x] &= \frac{\int\int tf\left(t \mid x, \boldsymbol{G}\right) f\left(x \mid \boldsymbol{G}\right) f(\boldsymbol{G}) d\boldsymbol{G} dt}{\int_{\boldsymbol{G}} f\left(x \mid \boldsymbol{G}\right) f(\boldsymbol{G}) d\boldsymbol{G}} \\ &= \frac{\int\int tf\left(x \mid \boldsymbol{G}\right) f(t, \boldsymbol{G}) d\boldsymbol{G} dt}{\int_{\boldsymbol{G}} f\left(x \mid \boldsymbol{G}\right) f(\boldsymbol{G}) d\boldsymbol{G}} \\ &\approx \frac{\sum_i T_i f\left(\boldsymbol{X}_{\boldsymbol{G}} = x \mid G_i\right)}{\sum_i f\left(\boldsymbol{X}_{\boldsymbol{G}} = x \mid G_i\right)}\end{aligned}$$

The key step in our proposed approach is to estimate the conditional density of $\boldsymbol{X}_{\boldsymbol{G}}|\boldsymbol{G}$.

Dimension Reduction

Since $\boldsymbol{G}$ is high-dimension, we consider the principal component to catch the main feature of $\boldsymbol{G}$.

$$\text{argmin}_{\boldsymbol{A},\boldsymbol{B},f}\text{Cost} = \|\boldsymbol{G} - \boldsymbol{A}\boldsymbol{B}^{\top}\| + L(\boldsymbol{X}, f(\boldsymbol{A})) + \Omega(\boldsymbol{A}, \boldsymbol{B})$$

where $\boldsymbol{B}$ defines the feature of SNPs and $\boldsymbol{A}$ is individual mutation loading on that factor or the projection of genotype data on these factors. The first part is the decomposition of the aimed matrix $\boldsymbol{G}$. The second part is the measurement of the similarity of our proposed distribution $f(\boldsymbol{A})$ to the real $\boldsymbol{X}$.

Gaussian Mixture

A natural way is to consider the Gaussian mixture for the conditional distribution of $\boldsymbol{X} \mid \boldsymbol{G}$. If the genotypes can be approximated by D features, i.e., $\boldsymbol{A}_{n \times D}$, it makes sense to assume the gene expression is also a mixture of D Gaussian distributions, whose means and variances depend on $\boldsymbol{A}_d, d = 1, 2, ..., D$. So we could assume

$$\boldsymbol{X} \mid \boldsymbol{G} \sim \sum_{d=1}^{D} \pi_d \Psi \left(\boldsymbol{\mu}_d \left(\boldsymbol{A}_d \right), \boldsymbol{\sigma}_d \left(\boldsymbol{A}_d \right) \right)$$

Basis Spline

In theory, any continuous function can be approximated by the combination of basis functions with a sufficient number of components. We hope $\boldsymbol{\mu}_d\,(A_d)$ can be well approximated by a combination of K basis functions:

$$\boldsymbol{A}_d = (A_{1d}, A_{2d}, ..., A_{nd}), \boldsymbol{\mu}_d = (\mu_{1d}, \mu_{2d}, ..., \mu_{nd}), \boldsymbol{\sigma}_d = (\sigma_{1d}, \sigma_{2d}, ..., \sigma_{nd})$$

$$\mu_{id} = \sum_{q=1}^{K} \alpha_q B_q(A_{id})$$

We also hope σ_{id} is the MLE estimation, i.e., $\sigma_{id} = (\sum_{i=1}^{n}(X_i - \mu_{id}))^{\frac{1}{2}}$

One Dimension Case

When $\boldsymbol{X}$ is a n by 1 matrix. The Optimization question is:

$$\mathrm{C} = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{p}(g_{ij} - a_i b_j)^2 + \sum_{i=1}^{n}\log\sigma_i + \frac{(x_i - \mu_i)^2}{2\sigma_i^2}$$

$$\text{subject to} \quad \|B_1\|_2 = \sqrt{\sum_{j}^{p} b_j^2} = 1$$

$$\text{where} \quad \sigma_i = (\sum_{i=1}^{n}(X_i - \mu_i))^{\frac{1}{2}}, \quad \mu_i = \sum_{q=1}^{K}\alpha_q B_q(a_i)$$

One Dimension Case

$$a_i = \frac{\sum_{j=1}^{p} g_{ij} b_j - \frac{(\mu_i - x_i)\mu_i'}{\sigma_i^2}}{\sum_{j=1}^{p} b_j^2}, \quad b_j = \frac{\sum_{i=1}^{n} a_i g_{ij}}{\sum_{i=1}^{n} a_i^2}, \quad \boldsymbol{\alpha} = \mathrm{E}[\frac{\boldsymbol{X}}{\boldsymbol{\sigma}} | \frac{B_{q=1:K}}{\boldsymbol{\sigma}}]$$

$\boldsymbol{B}$ should be sparse, we consider the standard Lasso penalization on b_j and get the following solutions:

$$b_j^{lasso} = \mathrm{sgn}(b_j^*)(|b_j^*| - \frac{\lambda}{\sum_{i=1}^{n} a_i^2})^+, \quad \text{where } b_j^* = \frac{\sum_{i=1}^{n} a_i g_{ij}}{\sum_{i=1}^{n} a_i^2}$$

Additive Model

Since the genotype data is high-dimensional, the computation time would be hugely prolonged if we always refresh the previous updated value and rebuild the model. One possible solution might be residual regression.
In our case, We fix the previously updated coefficients, $\boldsymbol{A}_{\text{pre}}, \boldsymbol{B}_{\text{pre}}, \boldsymbol{\alpha}_{\text{pre}}$ where pre represents previous updated value. Then we only update the residual of $\boldsymbol{G}$: $\boldsymbol{G}_{\text{res}} = \boldsymbol{G}_{\text{pre}} - \boldsymbol{A}_{\text{pre}}\boldsymbol{B}_{\text{pre}}^{\top}$ where res represents residual.
To include as much information on genotype data $\boldsymbol{G}$ as possible, we require $\boldsymbol{B}_{\text{new}}$ to be orthogonal with all the previous $\boldsymbol{B}_{\text{pre}}$. We realize this by multiplying $\boldsymbol{B}_{\text{new}}$ by a matrix $\boldsymbol{I}_p - \boldsymbol{B}_{\text{pre}}\left(\boldsymbol{B}_{\text{pre}}^{\top}\boldsymbol{B}_{\text{pre}}\right)^{-1}\boldsymbol{B}_{\text{pre}}^{\top}$ to project $\boldsymbol{B}_{\text{new}}$ on the orthogonal space of $\boldsymbol{B}_{\text{pre}}$.

Algorithm for One Dimension Case

S1: Start with D $= 1$

- s1 update a_i, b_j, α_q for i $= 1$:n, j $= 1$:p, q $= 1$:K
- s2 calculate Γ matrix for Gaussian mixture model
- s3 compare where the 2-norms of difference for $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\alpha}$ are smaller than the prespecified value. If yes, calculate the cost and move forward to **Step 2** D $=$ D$+1$. If not, go back to s1

S2: $\boldsymbol{G} = \boldsymbol{G} - \boldsymbol{A}\boldsymbol{B}^\top$

- ss1 update a_i, b_j, α_q for i $= 1$:n, j $= 1$:p, q $= 1$:K. Multiply $\boldsymbol{B}_{\text{new}}$ by $\boldsymbol{I}_p - \boldsymbol{B}\left(\boldsymbol{B}^\top \boldsymbol{B}\right)^{-1} \boldsymbol{B}^\top$.
- ss2 calculate Γ matrix for Gaussian mixture model
- ss3 compare where the 2-norms of difference for $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\alpha}$ are smaller than the prespecified value. If not, go back to ss1. If yes, calculate the cost and move forward to ss7.
- ss4 If new cost decrease more than 10%, repeat **Step 2**, D $=$ D$+1$. If not, Stop

One Dimension With Multiple Response

When $\boldsymbol{X}$ is n by M, M$>$ 1

$$\mathrm{C} = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{p}(g_{ij} - a_i b_j)^2 + \sum_{m=1}^{M}(\sum_{i=1}^{n}\log\sigma_i^m + \frac{(x_i^m - \mu_i^m - \boldsymbol{\eta}_m \boldsymbol{x}_i^{-m})^2}{2(\sigma_i^m)^2})$$

$$\text{subject to} \quad \|B_1\|_2 = \sqrt{\sum_{j}^{p} b_j^2} = 1$$

$$\text{where} \quad \sigma_i^m = (\sum_{i=1}^{n}(x_i^m - \mu_i^m))^{\frac{1}{2}}, \quad \mu_i^m = \sum_{q=1}^{K}\alpha_q^m B_q(a_i)$$

$$\boldsymbol{\eta}_m = (\eta_m^1, ..., \eta_m^{m-1}, \eta_m^{m+1}, \eta_m^M)$$

Solution

$$a_i^m = \frac{\sum_{j=1}^{p} g_{ij} b_j - \sum_{m=1}^{M} \frac{(\mu_i^m + \boldsymbol{\eta}_m \boldsymbol{x}_i^{-m} - x_i^m)\mu_i^{m'}}{\sigma_i^{m2}}}{\sum_{j=1}^{p} b_j^2}$$

$$b_j^m = \frac{\sum_{i=1}^{n} a_i g_{ij}}{\sum_{i=1}^{n} a_i^2}$$

$$\boldsymbol{\alpha}^m = \mathrm{E}[\frac{\boldsymbol{X}^m - \boldsymbol{\eta}_m \boldsymbol{X}^{-m}}{\boldsymbol{\sigma}^m} | \frac{B_{q=1:K}}{\boldsymbol{\sigma}^m}]$$

Solution

$$\Delta = \begin{pmatrix} 1 & \eta_{1,1}^{(d^*)} & \cdots & \eta_{1,m-2}^{(d^*)} & \eta_{1,m-1}^{(d^*)} \\ \eta_{2,1}^{(d^*)} & 1 & \cdots & \eta_{2,m-2}^{(d^*)} & \eta_{2,m-1}^{(d^*)} \\ & & \cdots & & \\ \eta_{m,1}^{(d^*)} & \eta_{m,2}^{(d^*)} & \cdots & \eta_{m,m-1}^{(d^*)} & 1 \end{pmatrix}$$

Δ is symmetric about the diagonal. Thus we have extra constraints when solving η.

Algorithm for One Dimension Case With Multiple Response

S1: Start with D $=1$

- s1 update $a_i^m, b_j^m, \alpha_q^m, \boldsymbol{\eta}^m$ for i $=$ 1:n, j $=$ 1:p, q $=$ 1:K, m $=$1:M
- s2 calculate Γ matrix for Gaussian mixture model
- s3 compare where the 2-norms of difference for $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\alpha}$ are smaller than the prespecified value. If yes, calculate the cost and move forward to **Step 2** D $=$ D+1. If not, go back to s1

S2: $\boldsymbol{G} = \boldsymbol{G} - \boldsymbol{A}\boldsymbol{B}^\top$

- ss1 update $a_i^m, b_j^m, \alpha_q^m, \boldsymbol{\eta}^m$ for i $=$ 1:n, j $=$ 1:p, q $=$ 1:K, m $=$1:M.

  Multiply $\boldsymbol{B}_{\text{new}}$ by $\boldsymbol{I}_p - \boldsymbol{B}\left(\boldsymbol{B}^\top \boldsymbol{B}\right)^{-1}\boldsymbol{B}^\top$.
- ss2 calculate Γ matrix for Gaussian mixture model
- ss3 compare where the 2-norms of difference for $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{\alpha}$ are smaller than the prespecified value. If not, go back to ss1. If yes, calculate the cost and move forward to ss7.
- ss4 If new cost decrease more than 10%, repeat **Step 2**, D $=$ D+1. If not, Stop

Simulation Study

Aim1: one dimension case, our proposed method is better than neural network and glm. WD qq plot.
Aim2: if we have multiple responses $\boldsymbol{X}$, We believe $\boldsymbol{X}$ are inner correlated. Thus, Conditional model(our) should be better than Marginal model(duplicate one response model.

Simulation Result

We use the whole blood sample data GTExXSNHG5. The number of individuals is 670, the number of snp is 3417. We set the causal.rate = 0.01. For the homogeneous model:

$$Y = \beta^\top \boldsymbol{X}_{\text{casual}} + \epsilon$$

For the heterogeneous model:

$$Y = \beta^\top \boldsymbol{X}_{\text{casual}} + (1 + c\beta^\top \boldsymbol{X}_{\text{casual}})\epsilon$$

We then normalize Y and remove the highly duplicate column X.

Single Result

We use 10 fold cross-validation to test the performance of three methods. WD is the abbreviation for Wasserstein distance.

$$W_1(\mu_1, \mu_2) = \int_{\mathbb{R}} \left| F_1(y) - F_2(y) \right| \mathrm{d}x$$

We use numerical approximation:

$$W_1(\mu_1, \mu_2) = \frac{1}{n} \sum_{i=1}^{n} |Q_1(y, \frac{i}{n}) - Q_2(y, \frac{i}{n})|$$

Where $Q(y, \frac{i}{n})$ is $\frac{i}{n}$-th quantile of y. WD is the result for our proposed method; WD2 is the result for the elastic net with optimal tuning parameter; WD3 is the result for neural network method with the same number of mixtures as our proposed method.

Single Result

For this homo case, we set the coefficient $\beta = 0.1$

```{r}
## abs()
mean(WD)
mean(WD2)
mean(WD3)

var(WD)
var(WD2)
var(WD3)
```

```
[1] 0.1723963
[1] 0.4081771
[1] 0.3144389
[1] 0.003727351
[1] 0.0004586147
[1] 0.03686145
```

Figure: homosingle

Single Result

For this hete case, we set the coefficient $\beta = 0.1$

```r
## abs()
mean(WD)
mean(WD2)
mean(WD3)

var(WD)
var(WD2)
var(WD3)
```

```
[1] 0.1723963
[1] 0.4081771
[1] 0.3144389
[1] 0.003727351
[1] 0.0004586147
[1] 0.03686145
```

Figure: hetesingle

Multiple Result

We also want to show when Y's are correlated with each other, the multiple-algorithm could work better than multiple single-algorithms.

$$\boldsymbol{Y} = \boldsymbol{\beta}^{\top}\boldsymbol{X} + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \Sigma)$$

The result of multiple-algorithm are simulated using the Markov chain

$$Y_i^t \sim \mathcal{N}(\mu + \boldsymbol{\eta}_i \boldsymbol{Y}_{-\boldsymbol{i}}^{(\boldsymbol{t-1})}, \sigma_i) i = 1, 2, ..., D \quad t = 1, 2, ..., T$$

Where i represents the position of dimension Y, and t represents the time-varying variable. And $\boldsymbol{Y}^0$ is some random initial values. The prediction is the mean of the latter sequence of $\{\boldsymbol{Y}^t\}\,|_{t \geq 0.75T}$

Notation

cm is the abbreviation for the MSE of the conditional model (multiple-version), mm is the abbreviation for the MSE of the marginal model (single-version). cmwd is the abbreviation for the Wasserstein distance of the conditional model (multiple-version), mmwd is the abbreviation for the Wasserstein distance of the marginal model (single-version).

Multiple Result

For homo case, Σ is the identity matrix. Conditional model vs marginal model (mse)

```{r}
cm1
cm2
cm3

(cm1 + cm2 + cm3 )/3
```

```
[1] 0.831271
[1] 0.6225814
[1] 0.5336311
[1] 0.6624945
```

```{r}
mm1
mm2
mm3

(mm1 + mm2 + mm3 )/3
```

```
[1] 0.8257151
[1] 0.8897963
[1] 0.8620893
[1] 0.8592002
```

Figure: homo

Multiple Result

Conditional model vs marginal model (Wasserstein distance)

```{r}
cmwd1
cmwd2
cmwd3

(cmwd1 + cmwd2 + cmwd3 )/3
```

```
[1] 0.4922632
[1] 0.2026474
[1] 0.1487395
[1] 0.2812167
```

```{r}
mmwd1
mmwd2
mmwd3

(mmwd1 + mmwd2 + mmwd3 )/3
```

```
[1] 0.4563343
[1] 0.2794005
[1] 0.3870225
[1] 0.3742525
```

Figure: homo

Multiple Result

For hete model,

$$\boldsymbol{Y} = \boldsymbol{\beta}^\top \boldsymbol{X} + (1 + c\boldsymbol{\beta}^\top \boldsymbol{X})\epsilon$$

We suppose $\boldsymbol{\beta}^\top = (0.1, 0.3, 0.5)$ and $\Sigma = \begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{pmatrix}$ and $\mathrm{D} = 3$.

1,2,3 represents the position of dimension.

Multiple Result

conditional model vs marginal model (mse)

```{r}
cm1
cm2
cm3

(cm1 + cm2 + cm3 )/3
```

```
[1] 0.8223781
[1] 0.7486099
[1] 0.7678286
[1] 0.7796055
```

```{r}
mm1
mm2
mm3

(mm1 + mm2 + mm3 )/3
```

```
[1] 0.7801134
[1] 0.7755223
[1] 0.8705353
[1] 0.8087237
```

Figure: hete

Multiple Result

conditional model vs marginal model (Wasserstein distance)

```r
cmwd1
cmwd2
cmwd3

(cmwd1 + cmwd2 + cmwd3 )/3
```

```
[1] 0.4737388
[1] 0.4178665
[1] 0.3876963
[1] 0.4264339
```

```r
mmwd1
mmwd2
mmwd3

(mmwd1 + mmwd2 + mmwd3 )/3
```

```
[1] 0.5586712
[1] 0.3910544
[1] 0.4188093
[1] 0.4561783
```

Figure: hete