penalized loglikelihood function

2020-06-22

We would like to compare the two criteria for α selection: (i) maximize the purity function; (ii) maximize the loglikelihood function.

Recall the purity fucntion

$$G_p(\alpha) = A_0 + A_1\hat{\mu}_x'\alpha + \frac{A_2}{2}[\alpha'\hat{\Sigma}_x\alpha + \alpha'\hat{\mu}_x\hat{\mu}_x'\alpha] - \lambda||\alpha||_1 \tag{1}$$

$$s.t.\ ||\alpha||_2^2 = 1$$

where

- $\lambda > 0$
- $A_0 = -q + \frac{1}{2}tr(\hat{D}_2^{-1}\hat{D}_1) + \frac{1}{2}tr(\hat{D}_1^{-1}\hat{D}_2) + \frac{1}{2}(\hat{\beta}_1 - \hat{\beta}_2)'(\hat{D}_1^{-1} + \hat{D}_2^{-1})(\hat{\beta}_1 - \hat{\beta}_2)$
- $A_1 = (\hat{\beta}_1 - \hat{\beta}_2)'(\hat{D}_1^{-1} + \hat{D}_2^{-1})(\hat{\Gamma}_1 - \hat{\Gamma}_2)$
- $A_2 = (\hat{\Gamma}_1 - \hat{\Gamma}_2)'(\hat{D}_1^{-1} + \hat{D}_2^{-1})(\hat{\Gamma}_1 - \hat{\Gamma}_2)$
- q is the dimension of D matrix.

For the optimization based on loglikelihood function, suppose the true model for outcome at treatment k is:

$$y_{ki} = X_i(\beta_k + b_{ki} + \Gamma_k(\alpha' x_i)) + \epsilon_{ki}, k \in \{1, 2\} \tag{2}$$

where $\epsilon_1 \sim N(0, \sigma_1^2), \epsilon_2 \sim N(0, \sigma_2^2)$ $b_{1i} \sim N(0, D_1), b_{2i} \sim N(0, D_2)$.

And then we know that the outcome follows multivariate normal distribution:

$$y_{ki} \sim MVN\big(X_i\beta_k + X_i\Gamma_k(\alpha' x_i), X_i D_k X_i' + \sigma_k^2 I\big)$$

And its pdf is

$$f(y_{ki}; \beta_k, \Gamma_k, D_k, \sigma_k^2) = \frac{1}{\sqrt{(2\pi)^7|\Sigma_{ki}|}} \exp\big(-\frac{1}{2}(y_{ki} - \mu_{ki})'\Sigma_k^{-1}(y_{ki} - \mu_{ki})\big)$$

where $\mu_{ki} = X_i\beta_k + X_i\Gamma_k(\alpha' x_i)$, $\Sigma_{ki}^2 = X_i D_k X_i' + \sigma_k^2 I$

Therefore, the likelihood function is

$$\begin{aligned}
&L(\beta_1, \beta_2, \Gamma_1, \Gamma_2, D_1, D_2, \alpha, \sigma_1^2, \sigma_2^2) \\
&= \text{ likelihoods in group 1} \times \text{ likelihoods in group 2} \\
&= \big(\prod_{i=1}^{n_1} \frac{1}{\sqrt{(2\pi)^7|\Sigma_{1i}|}} \exp\big(-\frac{1}{2}(y_{1i} - \mu_{1i})'\Sigma_k^{-1}(y_{1i} - \mu_{1i})\big)\big)\big(\prod_{i=1}^{n_2} \frac{1}{\sqrt{(2\pi)^7|\Sigma_{2i}|}} \exp\big(-\frac{1}{2}(y_{2i} - \mu_{2i})'\Sigma_k^{-1}(y_{2i} - \mu_{2i})\big)\big)
\end{aligned} \tag{3}$$

The log-likelihood function:

$$l(\beta_1, \beta_2, \Gamma_1, \Gamma_2, D_1, D_2, \alpha, \sigma_1^2, \sigma_2^2) = \log\big(L(\beta_1, \beta_2, \Gamma_1, \Gamma_2, D_1, D_2, \alpha, \sigma_1^2, \sigma_2^2)\big) \tag{4}$$

For variable selection via a lasso-type penalty, for a given λ, we can consider maximizing

$$\text{penalized loglikelihood function: } pl(\Theta) = l(\beta_1, \beta_2, \Gamma_1, \Gamma_2, D_1, D_2, \alpha, \sigma_1^2, \sigma_2^2) - \lambda||\alpha||_1 \tag{5}$$

To calculate the function value of equation (5), for a given α and λ, we may fit LME in equation (1) for drug group and placebo group separately, and then calculate $\hat{\beta}_1, \hat{\beta}_2, \hat{\Gamma}_1, \hat{\Gamma}_2, \hat{D}_1, D_2, \hat{\sigma}_1^2, \hat{\sigma}_2^2$. Plug in the estimated values in the penalized loglikelihood function $pl(\hat{\Theta})$ and get the penalized loglikelihood value.

To find the α that maximizes the equation (5), we may treat it as a non differenable function and apply the Nelder-Mead algorithm in the optim function in R.

Two criteria comparison

We conduct a simulation study to compare the two criteria: penalized purity function and penalized loglikelihood function.

Simulation setting

Parameters

- $p = 3$, which is the dimension of x, the predictors.
- $\alpha = [\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}]$
- $X = [1, t, t^2]$, $t = [0, 1, 2, 3, 4, 6, 8]$ is the design matrix for fixed effect and random effect
- $x \sim MVN(\mu_x, \Sigma_x)$, $\mu_x = \mathbf{0}_p$, Σ_x has diagonal equals to 1 and 0.5 everywhere else.
- $\beta_1 = \beta_2 = [1, 1, 0.2]$
- $\Gamma_1 = [1, 1, 0], \Gamma_2 = [1, \cos(\pi/3), \sin(\pi/3)]$
- $D_1 = D_2 = \begin{pmatrix} 1 & 0.3 & 0.1 \\ 0.3 & 1 & 0.5 \\ 0.1 & 0.5 & 1 \end{pmatrix}$
- $\epsilon_1, \epsilon_2 \sim N(0, 1^2)$
- $\lambda = 0.1, 1, 10, 100, 1000$

For each λ value, 100 different datasets were generated based on the above parameters. For each dataset, the $\hat{\alpha}_{pur}$ that maximizes the penalized purity function and the maximum of the penalized purity function values as well as the the $\hat{\alpha}_{log}$ that maximizes the penalized log likelihood function and the max penalized log likelihood function values are saved.

Function value vs λ

The figure 1 shows the relationship between maximum penalized purity values vs different λ values. The black points are the mean values of the maximumpenalized purity values across the 100 simulations. And the black bars present their standard deviations.

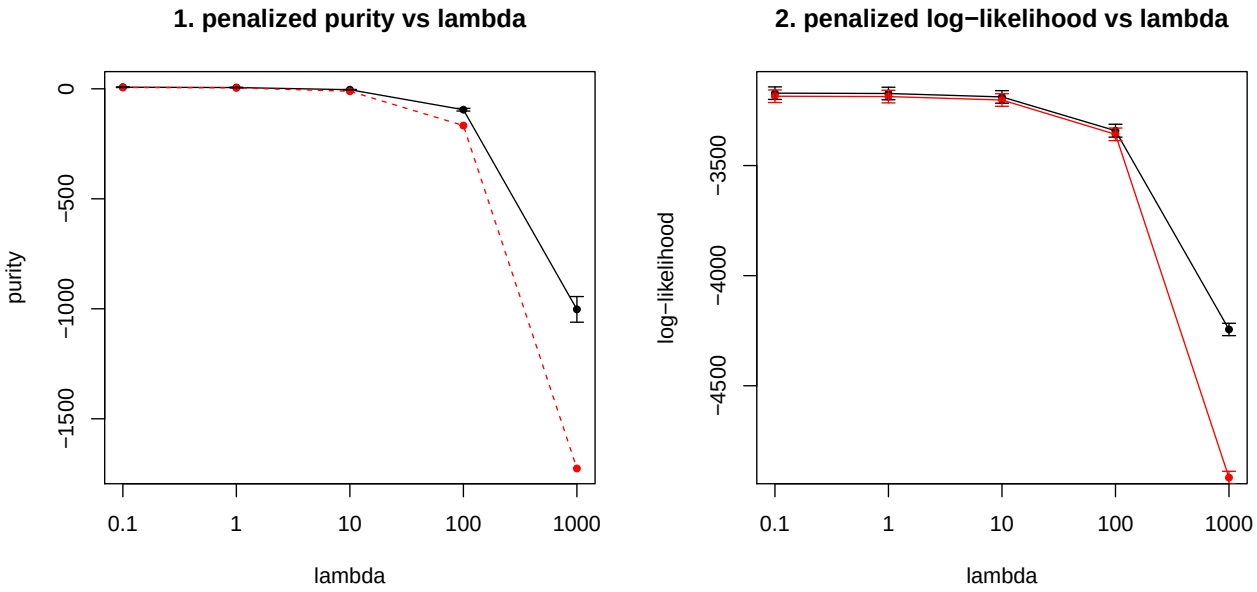
Since we know the true values of the $\beta, \Gamma, D, \sigma, \Sigma_x, \alpha$, we could calculate the true purity values, which are the red points in the plot.

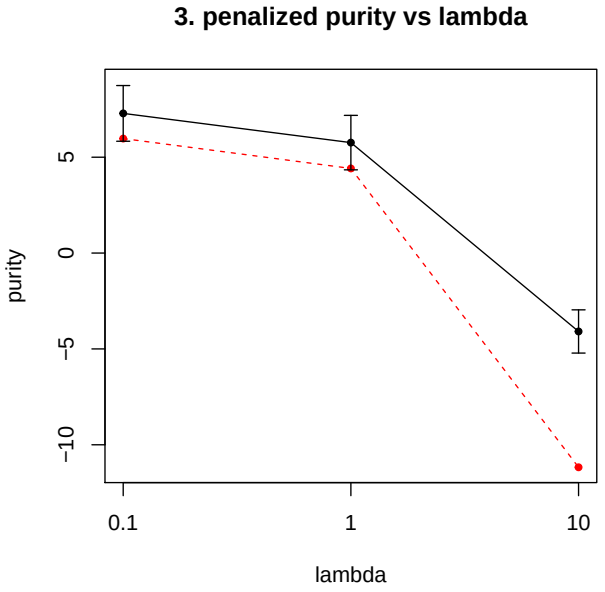
In figure 2, the maximum penalized log-likelihoods are calculated across the 100 simulations. The means are the black points and the black bars present their standard deviations.

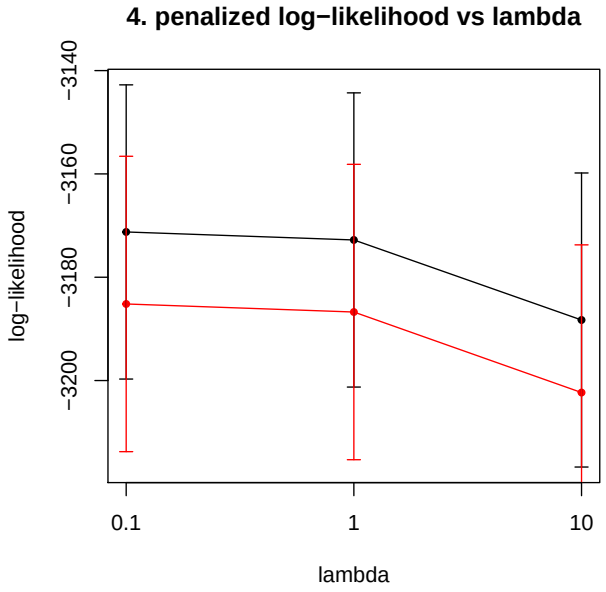
We may also calculate the euqation (5) with true $\beta, \Gamma, D, \sigma, \Sigma_x, \alpha$ values. The true values are shown as red points and bars.

From the plots, the loglikelihood method has a better estimation than the purity method.

- Note1: The true purity for a given λ is a fixed value. The true loglikelihood for a given λ can change, based on different input y_{ki} from different datasets. Since I generated 100 sets, the true loglikelihoods are a batch of numbers.
- Note2: Since when $\lambda = 1000$, the values are quite large, the plots 1 and 2 may not show the relationship well. I also drew plot 3 and 4, which $\lambda = 0.1, 1, 10$. And the λ that maximizes the penalized purity function probablity not the λ that maximizes the penazlied log-likelihood function. Since the purity is about 7, a λ about 10 can be a quite large parameter for it while the log-likelihood function has values about -3000, λ around 10 is quite small for it.







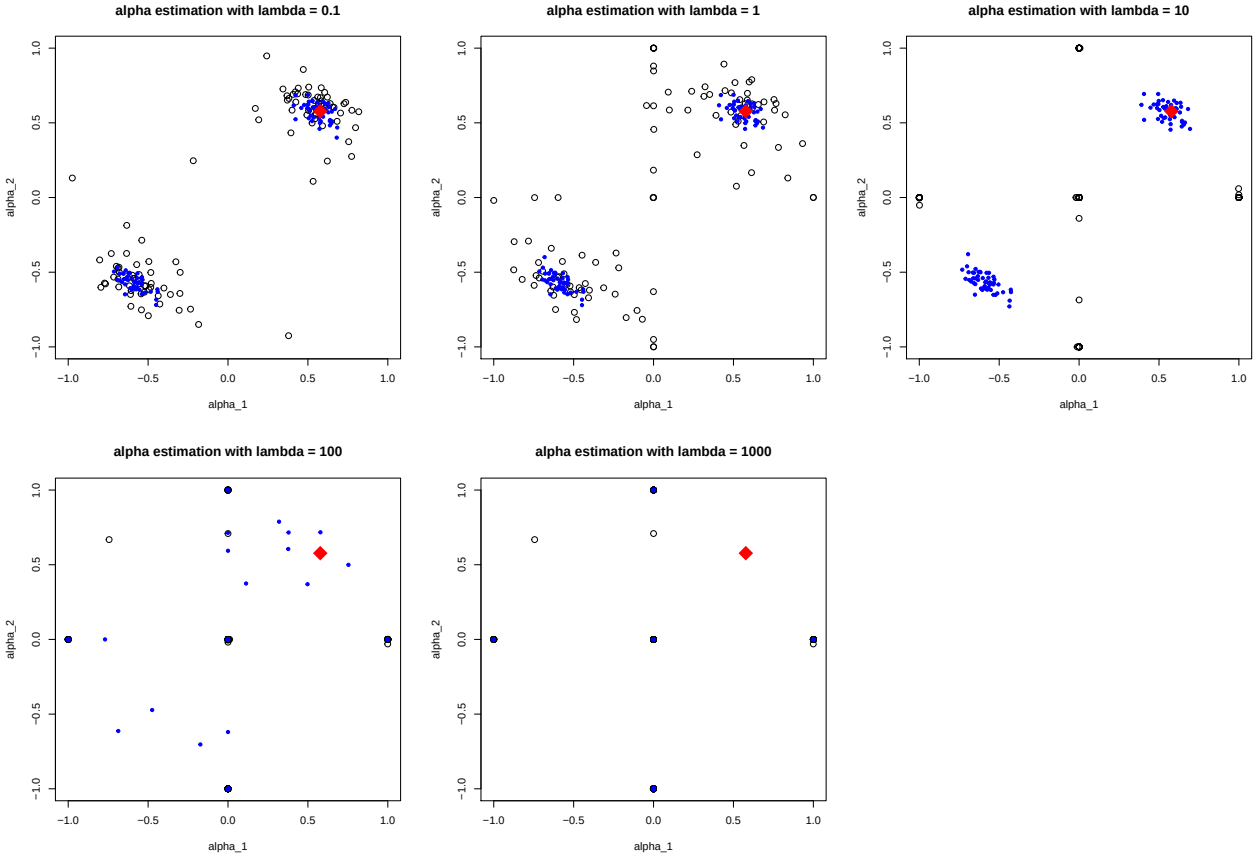
Estimation of α

The following 5 plots show the scatter plots of α vs $\lambda = 0.1, 1, 10, 100, 1000$. Since true α is set as $\alpha = (\alpha_1, \alpha_2, \alpha_3) = (\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}})$ with 3 dimensions and $||\alpha||_2 = 1$, therefore I just use α_1 and α_2 as the x-axis and y-axis.

The black points are the $\hat{\alpha}_{pur}$ estimated with purity function as the objective fucntion while the blue points are the $\hat{\alpha}_{log}$ estimated with the log likelihood functions. The red point shows the true α.

We can see that the blue points (log likelihood as the criterion) are more dense than the black points (purity as the criterion), i.e. the loglikelihood crierion estimation is a more stable.

When $\lambda = 10$, the black points only have 7 different values, i.e. one or two elements in α is estimated to be 0. As λ gets larger, the blue points also show that elements in α equal to 0. The black points are more sensitive to large λ values since the purity functions are much smaller than the absolue log likelihood values.



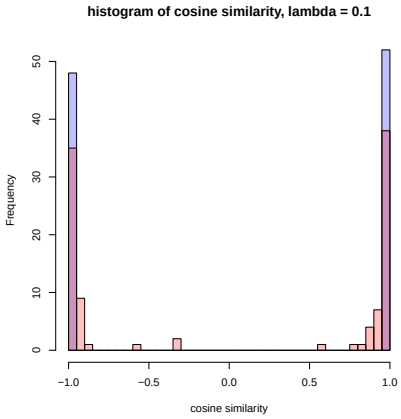
Histogram of cosine similarity

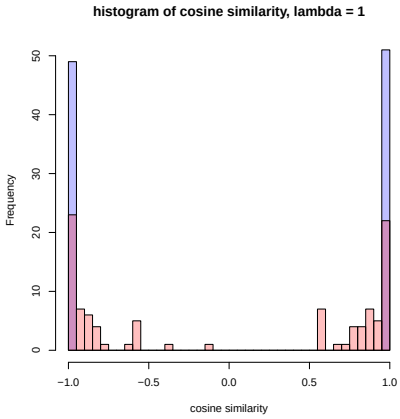
We may also calculate the consine similarity between the true α and each estimated $\hat{\alpha}_{pur}$ and $\hat{\alpha}_{log}$. The histogram of those cosine similarities are shown in the following plots.

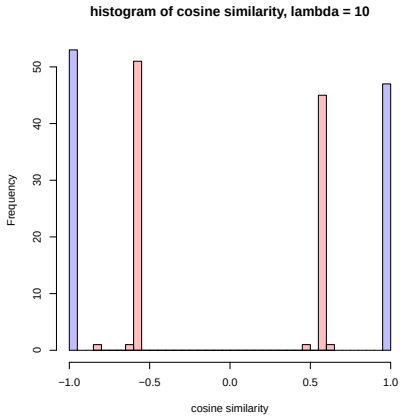
The blue bars present the cosine similarities for log likelihood optimization while the red bars present the cosine similarities for purity optimization.

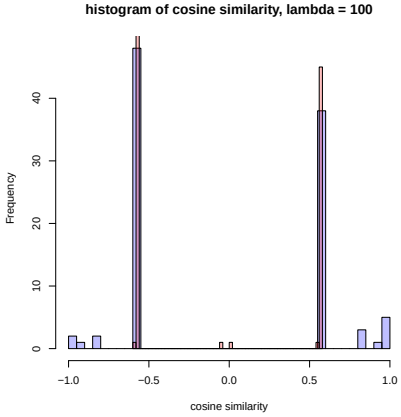
The blue bars have better similarities with true α, since the values are whether 1 or -1.

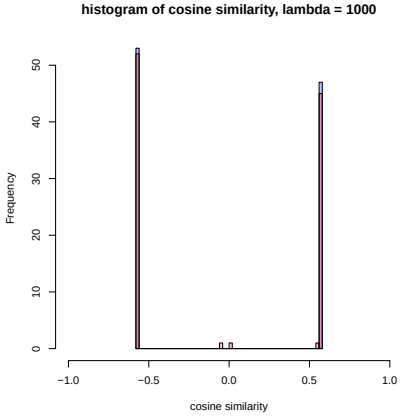
As λ gets very large, e.g. 1000, the elements in α are selected to be 0. Therefore, the cosine similarity gets worse.


histogram of cosine similarity, lambda = 0.1
Frequency
cosine similarity


histogram of cosine similarity, lambda = 1
Frequency
cosine similarity


histogram of cosine similarity, lambda = 10
Frequency
cosine similarity


histogram of cosine similarity, lambda = 100
Frequency
cosine similarity


histogram of cosine similarity, lambda = 1000
Frequency
cosine similarity