Alpha optimization formula, simulation and real data analysis

2019-10-08

Contents

Purity calculation and alpha optimization

More informaiton please see the other pdf file.

Simulation

Data generation

Simulation Setting

Sample size

- 200 subjects in each group

Angle between Γ_1 and Γ_3

$0, 30, 60, 90, 120, 150, 180$ degrees.

Dimensions

p $= 4$

Parameters

- Each subject has 7 time points: $2 \times 200 \times 7 = 2800$
- True $\alpha = [0.5, 0.5, 0.5, 0.5]_4$
- $\beta_{pbo} = [0, 3, 0.9]', \beta_{drg} = [0, 3.1, 1]'$
- $\Gamma_{drg} = [0, 1, 0]'$, angle between the two Γ lines can be 0, 30, 60, 90, 120, 150, and 180 degrees.
- $|\Gamma_{drg}| = |\Gamma_{pbo}| = 1$

LME estimation

Renew:

- Use Lagrange multiplier to calculate the purity with restriction.
- Fit one LME to make the $D_1 = D_2$

If we want to set two $\hat{D}$ to be the same in two groups to make the model more stable (less variance), we could fit the following LME model:

$$Y_i = S(\beta_i + b_i + \Gamma(\alpha_i' x_i)) + \epsilon = S(\beta_i + \Gamma(\alpha_i' x_i)) + Sb_i + \epsilon$$

$$\begin{aligned} Y =& S(\beta_1 + \Gamma_1(\alpha' x))trt + S(\beta_2 + \Gamma_2(\alpha' x)) + Sb + \epsilon \\ =& S\big(\beta_1 trt + \beta_2 + \Gamma_1(\alpha' x)trt + \Gamma_2(\alpha' x)\big) + Sb + \epsilon \end{aligned}$$

That is,

- $\beta_{drg} = \beta_1 + \beta_2$, $\beta_{pbo} = \beta_2$
- $\Gamma_{drg} = \Gamma_1 + \Gamma_2$, $\Gamma_{pbo} = \Gamma_2$
- $D = D_1 = D_2$

For example:

Fit the model with one D

```
fit = lmer(y ~ tt + I(tt^2) +
               W + W * tt + W * I(tt^2) +
               trt * tt + trt *  I(tt^2) +
               trt * W * tt + trt * W * I(tt^2) +
               (tt + I(tt^2)|subj),
               data = dat_try, REML = FALSE)
summary(fit)$coefficient
```

```
##                   Estimate Std. Error    t value
## (Intercept)   18.45562018 1.02759256 17.9600563
## tt            -1.42921411 0.58851727 -2.4284998
## I(tt^2)        0.10956748 0.06803916  1.6103590
## W              0.91008250 1.32269030  0.6880541
## trt            0.10117127 0.66734489  0.1516027
## tt:W           0.74812405 0.75736632  0.9877968
## I(tt^2):W     -0.08328161 0.08753071 -0.9514559
## tt:trt        -0.43901264 0.38206665 -1.1490473
## I(tt^2):trt    0.03136069 0.04416680  0.7100513
## W:trt         -0.25010190 0.80665218 -0.3100492
```

```
## tt:W:trt      -0.50402961 0.46231424 -1.0902316
## I(tt^2):W:trt  0.05941113 0.05342908  1.1119623
```

Results:

```
fit_cov = as.matrix(fixef(fit))

    beta1 = fit_cov[c("(Intercept)", "tt", "I(tt^2)"),] +
      fit_cov[c("trt", "tt:trt", "I(tt^2):trt"),] * 2
    beta2 = fit_cov[c("(Intercept)", "tt", "I(tt^2)"),] +
      fit_cov[c("trt", "tt:trt", "I(tt^2):trt"),]

    gamma1 = fit_cov[c("W","tt:W", "I(tt^2):W"),] +
      fit_cov[c("W:trt", "tt:W:trt", "I(tt^2):W:trt"),] * 2
    gamma2 = fit_cov[c("W","tt:W", "I(tt^2):W"),] +
      fit_cov[c("W:trt", "tt:W:trt", "I(tt^2):W:trt"),]
```

```
  beta1
```

```
## (Intercept)          tt     I(tt^2)
##  18.6579627  -2.3072394   0.1722889
```

```
as.matrix(VarCorr(fit)$subj)[1:3, 1:3]
```

```
##             (Intercept)         tt     I(tt^2)
## (Intercept)   8.8805335  1.9034257 -0.22562589
## tt            1.9034257  2.4155154 -0.24581374
## I(tt^2)      -0.2256259 -0.2458137  0.02911379
```

Fit the model with two Ds

```
  dat_pbo_est = dat_try[dat_try$trt == 1, ]
  dat_drg_est = dat_try[dat_try$trt == 2, ]

  fit_pbo_est = lmer(y ~ tt + I(tt^2) + W + W * tt +
                                W * I(tt^2) + (tt+I(tt^2)|subj),
                              data = dat_pbo_est, REML = FALSE)
  fit_drg_est = lmer(y ~ tt + I(tt^2) + W + W * tt +
                                W * I(tt^2) + (tt+I(tt^2)|subj),
                              data = dat_drg_est, REML = FALSE)
```

```
  beta1 = as.matrix(fixef(fit_drg_est))[1:3]
  D1 = as.matrix(VarCorr(fit_drg_est)$subj)[1:3, 1:3]
  D2 = as.matrix(VarCorr(fit_pbo_est)$subj)[1:3, 1:3]
  beta1
```

```
## [1] 18.657981 -2.306764  0.172161
```

```
  D1
```

```
##             (Intercept)         tt     I(tt^2)
## (Intercept)   8.8801087  2.7590253 -0.36004432
## tt            2.7590253  1.0313445 -0.10484244
## I(tt^2)      -0.3600443 -0.1048424  0.01488122
```

```
  D2
```

```
##             (Intercept)         tt     I(tt^2)
## (Intercept)   9.2703539  0.9234027 -0.08391710
## tt            0.9234027  3.8175860 -0.39147387
```

```
## I(tt^2)       -0.0839171 -0.3914739  0.04423488
```

Result

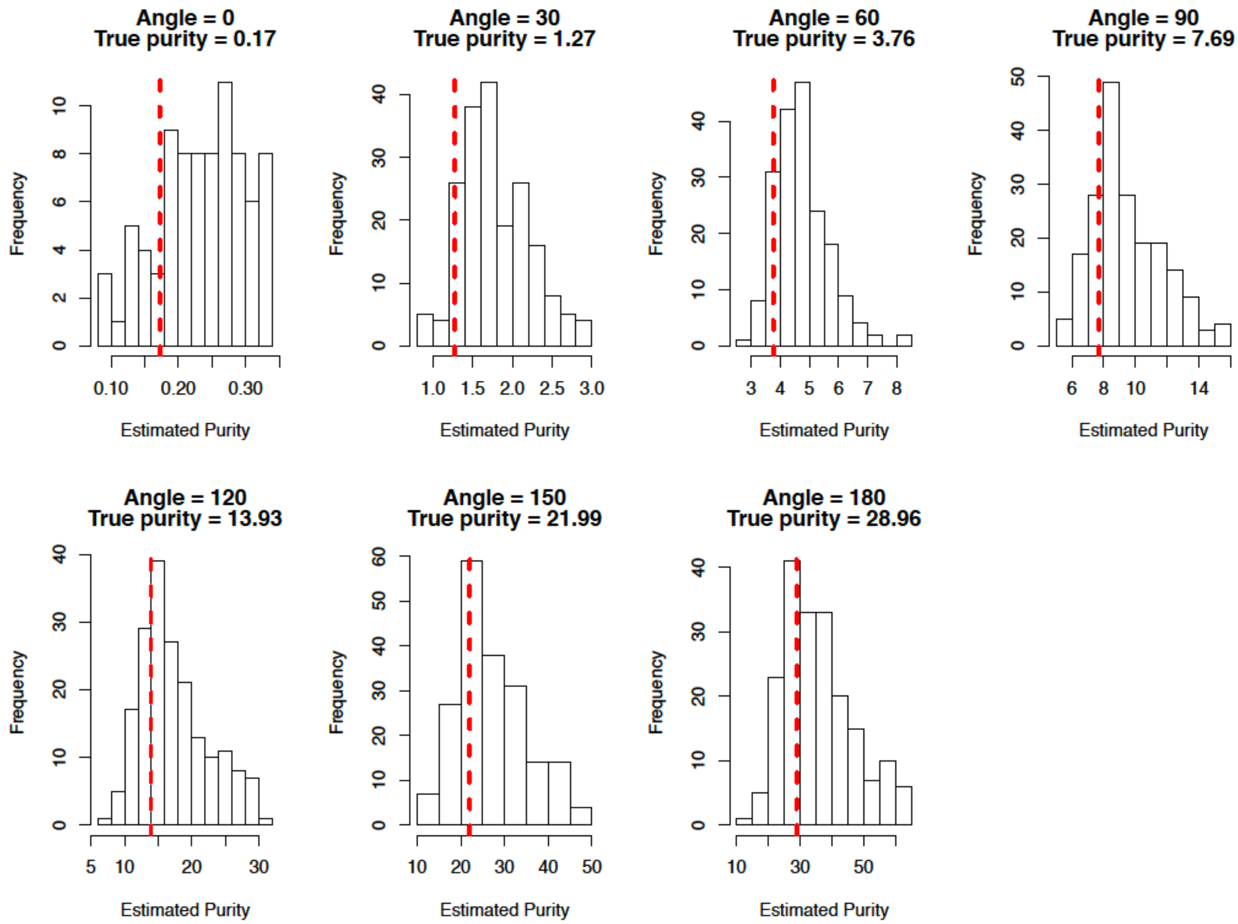
angels	cos	true_KL	KL	KL2
0	1	0.17	0.24	0.20
30	1	1.27	1.78	1.59
60	1	3.76	4.75	4.50
90	1	7.69	9.49	9.38
120	1	13.93	17.39	16.94
150	1	21.99	27.04	26.68
180	1	28.96	35.75	35.25

Where

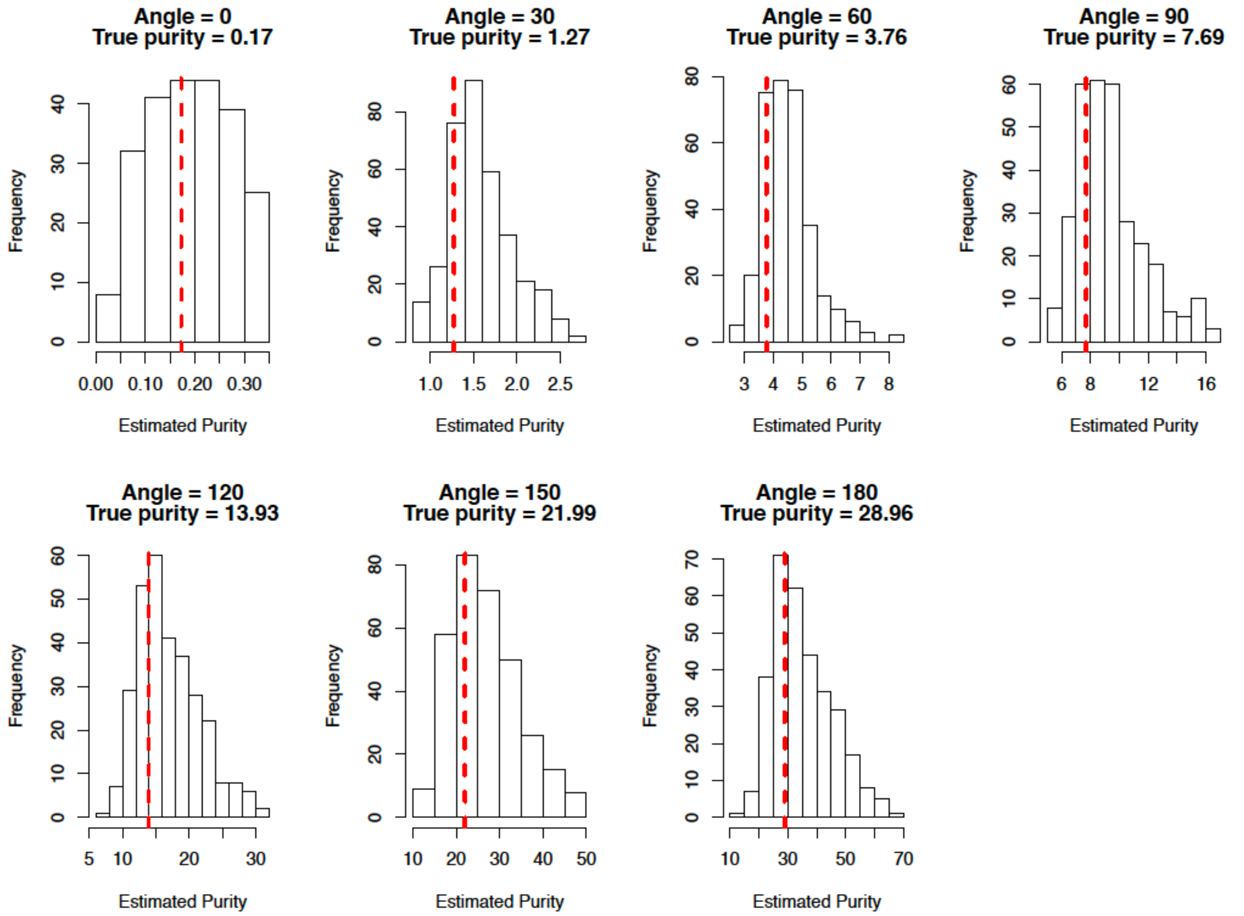
- angles: the angle between Γ_1 and Γ_2
- cos: the cosine similiarity.
- true_KL: the true purity
- KL: the mean estimated purity with two D matrix estimation
- KL2: the mean estimated purity with one D matrix estimation

Histograms

- Estimated purity with two D matrix estimation



- Estimated purity with one D matrix estimation



Purity with different number of covariaes.

I also simulate a dataset with the same parameters settings, while

- sample size for each group is 1000
- Angle between Γ_1 and Γ_2 is 60
- p = 10

With different number of covariates added in the model, will the purity have difference? Will the purity increase?

With one covariate included:

Covariate name	Purity with 2D	Purity with 1D
X1	30.61	0.39
X2	30.78	0.47
X3	30.62	0.43
X4	30.59	0.38
X5	30.59	0.35
X6	30.84	0.55
X7	30.59	0.41
X8	30.74	0.38
X9	30.68	0.50

Covariate name	Purity with 2D	Purity with 1D
X10	30.71	0.39

With more covariates included:

	Two Cov	Four Cov	Eight Cov	Ten Cov
Two D	30.72	31.34	34.24	41.46
One D	0.92	2.40	12.11	41.30

Embarc data analysis

Covariates

We would like to focus on the continuous variables first. Therefore, the covarites with level larger than 5 are included.

The covariates names are:

- "age_evaluation", "hamd17_baseline", "dur_MDE", "age_MDE", "axis2", "anger_attack" , "anxious"

As well as the behavior covariates:

Covariate name	Description
w0_4165	A not B Interference Reaction Time in negative trials
w0_4167	A not B Interference Reaction Time in non-negative trials
w0_4163	A not B Interference Reaction Time in all trials
w0_4162	A not B Itotal number of correct trials
w0_4169	Median Reaction time for correct trials in the Choice reaction time task
w0_1844	Number of valid recalled words in the Word Fluency task
w0_1916	Flanker Accuracy, an Accuracy effect is a measure of interference effects; Higher scores are indicative of increased interference effects (i.e., reduced cognitive control).
w0_1915	Flanker Reaction Time, a measure of interference effects; Higher scores are indicative of increased interference effects (i.e., reduced cognitive control).
w0_1920	Accuracy effect, it measures post-conflict behavioral adjustments; Higher values indicate better cognitive control

Purity with one covaraite

covariates	Purity with 2D	Purity with 1D
age_evaluation	1.602	0.651
hamd17_baseline	0.377	0.000
dur_MDE	1.352	0.166
age_MDE	1.710	0.450
axis2	1.402	0.304
anger_attack	1.188	0.152
anxious	0.423	0.251
w0_4165	1.253	0.238

covariates	Purity with 2D	Purity with 1D
w0_4167	0.000	0.257
w0_4163	0.000	0.179
w0_4162	1.543	0.634
w0_4169	1.214	0.000
w0_1844	1.232	0.167
w0_1916	1.959	0.350
w0_1915	2.130	0.375
w0_1920	1.267	0.193

Purity with two covaraites

Then randomly select two covariates, i.e, "w0_1916" and "w0_1915", the purity is:

With two D, D_1, D_2

```
## [1] -2.358486
```

With one D

```
## [1] -0.6529398
```

Purity with selected three covaraites

Then randomly select three covariates, i.e, i.e. "w0_4163" "w0_1844" "w0_1915", the purity is:

With two D, D_1, D_2

```
## [1] -2.273696
```

With one D

```
## [1] -0.6653502
```

Purity with all covariates

With two D, D_1, D_2

```
## [1] -2.13217
```

With one D

```
## [1] -3.175601
```