Mixed Model or Pooled Analysis of Clustered Data

2022-02-21

Contents

1 The Generalized Gauss-Markov Theorem

Let

$$Y = X\beta + \epsilon$$

where

1. Y is a $n \times 1$ vector of observations, 2, X is a $n \times p$ matrix of observed values and X is of full column rank,
2. β is an unobserved $p \times 1$ vector, and
3. ϵ is an unobserved $n \times 1$ random vector, where we know that $\mathrm{E}(\epsilon) = 0$ and $\mathrm{Var}(\epsilon) = c\Sigma$, with c a possibly unknown positive constant, and Σ a known $n \times n$ positive definite matrix.

Consider the problem of estimating, η, a linear combination of β given by $\eta = L\beta$ where L is a given $h \times p$ matrix.

The conclusion of the theorem is that the best (minimum variance) estimator of η among estimators that are:

1. linear: of the form $\hat{\eta} = AY$ where A is a $h \times n$ matrix, and
2. unbiased: $\mathrm{E}(\hat{\eta}) = \eta$ for all values of β,

is:

$$\hat{\eta} = L\hat{\beta}$$

where

$$\hat{\beta} = (X'\Sigma^{-1}X)^{-1}(X'\Sigma^{-1}Y)$$

This estimator is called the **BLUE** of η. Its variance is

$$\mathrm{Var}(\hat{\eta}) = L(X'\Sigma^{-1}X)^{-1}L'$$

Many important practices in statistics devolve from this theorem. You can use it to prove many special cases.

1.1 *The sample mean is the BLUE of the population mean*

For example, if $Y_1, ..., Y_n$ are uncorrelated each with mean μ and variance σ^2 (known or unknown), then the BLUE of μ is $\bar{Y}$. To prove this using the GGM, consider what happens if you let X be a column of 1's and $c\Sigma = \sigma^2 I$ where I is the $n \times n$ identity matrix.

1.2 *Combining estimators with weights proportional to inverse variances*

Here is one of the most important consequences of the GGM:

Let $\hat{\beta}_1, ..., \hat{\beta}_k$ be k uncorrelated unbiased estimators of the same unknown parameter β. Suppose that it is known that

$$\text{Var}(\hat{\beta}_i) = \Sigma_i$$

where $\Sigma_1, ..., \Sigma_k$ are positive-definite matrices that are known or known up to a common constant factor.

Then the BLUE of β is the weighted average of the $\hat{\beta}_i$s with weights proportional to the inverse of their variances, i.e.

$$\hat{\beta} = \left\{ \sum_{i=1}^{k} \Sigma_i^{-1} \right\}^{-1} \left\{ \sum_{i=1}^{k} \Sigma_i^{-1} \hat{\beta}_i \right\}$$

with variance:

$$\text{Var}(\hat{\beta}) = \left\{ \sum_{i=1}^{k} \Sigma_i^{-1} \right\}^{-1}$$

Exercise: Prove this using the GGM.

Hint: Let X be a vertical stack of identity matrices and Σ be a block-diagonal matrix formed with matrices Σ_i in the diagonal blocks.

2 *One-way ANOVA with random effects*

We will see what the GGM tells us about different ways of estimating the mean of a population when we have a clustered sample.

We will consider the simplest possible use of mixed models: one-way analysis of variance with random effects, where observations on a variable Y are obtained in clusters and where the goal is to estimate the overall mean of the population from which the Ys originated.

Let's call the cluster variable I with values $i = 1, ..., K$.

We'll use notation that is consistent with the notation for the general mixed model, although the notation is unnecessarily complex for this simple example.

Let the overall population mean be γ_{00} and let the 'true' mean for cluster i be:

$$\beta_{0i} = \gamma_{00} + u_{0i}, \quad i = 1, ..., K$$

where

$$u_{0i} \sim \ iid\ N(0, g_{00})$$

In each cluster, the observations, Y_{ij}, are generated as:

$$Y_{ij} = \beta_{0i} + \epsilon_{ij}, \ \ \epsilon_{ij} \sim \ iid\ N(0, \sigma^2)$$

where $i = 1, ..., K$ and, for each i the index j denotes individuals in the ith cluster with $j = 1, ..., n_i$, where n_i is the sample size in the ith cluster.

Let N denote the total number of observations:

$$N = \sum_{i=1}^{K} n_i$$

Our goal is to perform inference for the grand population mean of Y, γ_{00}. We would like to find an efficient (small standard error given the data) unbiased estimator and we would like to report an honest estimate of the standard error.

In practice the within-cluster variance, σ^2, and the between-cluster variance, γ_{00}, are unknown and need to be estimated, but we will consider what could be done if we knew their values. In applications, we use estimates of these parameters.

2.1 Competing estimators

We will consider 4 plausible estimators of γ_{00}:

1. The pooled approach using

$$\bar{Y} = \frac{\sum_{i,j} Y_{ij}}{N}$$

2. The mean of means approach using

$$\bar{Y}_M = \frac{\sum_i \bar{Y}_i}{K}$$

3. The mean of means weighted by sample size

$$\bar{Y}_W = \frac{\sum_i n_i \bar{Y}_i}{\sum_i n_i}$$

4. The mean of means weighted by inverse variance

$$\bar{Y}_{IV} = \frac{\sum_i v_i^{-1} \bar{Y}_i}{\sum_i v_i^{-1}}$$

where

$$v_i = \text{Var}(\bar{Y}_i) = g_{00} + \frac{\sigma^2}{n_i}$$

Note that $\bar{Y}_{IV}$ is the estimate produced with a mixed model analysis.

Exercises:

1. Show that $\bar{Y} = \bar{Y}_W$, so 1 and 3 are actually the same.
2. Show that all three estimators are unbiased estimators of γ_{00}.
3. Show that, under the assumptions above for the way the Y_{ij}s are generated, $\text{Var}(\bar{Y}_i - \gamma_{00}) = g_{00} + \frac{\sigma^2}{n_i}$.
4. Show that, under the assumptions above for the way the Y_{ij}s are generated, $\text{Var}(\bar{Y}_i - \gamma_{00}) = g_{00} + \frac{\sigma^2}{n_i}$.
5. Find an expression for the variance of $\bar{Y} - \gamma_{00} = \bar{Y}_W - \gamma_{00}$.
6. Find an expression for the variance of $\bar{Y}_M - \gamma_{00}$.
7. Find an expression for the variance of $\bar{Y}_I V - \gamma_{00}$. Hint: Use the GGM theorem and you won't need to do any calculation.
8. What argument can you give for the proposition that $\bar{Y}_{IV}$ is a 'better' estimator than the others?
9. Why is using weights proportional to

$$\left(g_{00} + \frac{\sigma^2}{n_i}\right)^{-1}$$

the same as using weights proportional to

$$\left(1 + \frac{\sigma^2}{n_i g_{00}}\right)^{-1}$$

which is also the same as using weights proportional to

$$n_i \left(1 + \frac{n_i g_{00}}{\sigma^2}\right)^{-1}$$

10. What happens to the weights defining $\bar{Y}_{IV}$ as the ratio $\frac{g_{00}}{\sigma^2} \to 0$?
11. What happens to the weights defining $\bar{Y}_{IV}$ as the ratio $\frac{g_{00}}{\sigma^2} \to \infty$?

3 *To Do:*

1. Show how much the estimated variance of $\bar{Y}$ assuming independence is smaller than its correct variance taking clustering into account. By how much? How does it depends on n_is and g_{00}/σ^2?
2. Plot variances and relative variances of the three estimators as a function of n_is and g_{00}/σ^2
3. Plot the estimated variances under the assumptions that are the basis of each estimator.

3.1 *Efficiency: Comparison of SEs*

First we will see what happens when cluster sizes show moderate variation.

We will take a sample from 10 clusters in which the number of observations in each cluster, $n_i \sim \text{Poisson}(10)$.

Since the relative SE depends only on the ratio g_{00}/σ^2, we will take $\sigma = 1$ and let g_{00} vary from 0 to 10.

1. Generate sample sizes:

```
set.seed(23153)
ns <- rpois(10,10)
ns
```

```
##  [1] 10 12  8 12 11 10 12  8  7  4
```

```
sum(ns)
```

```
## [1] 94
```

2. Write a function to compute variances of estimators:

A function that finds the variance of a linear combination of $\bar{Y}_i$ as fuction of a vector of n_is and $g = g_{00}$ when $\sigma = 1$.

```
var_est <- function(g, n, w) {
  # g: variance between relative to variance within
  # n: vector of sample sizes
  # w: vector of relative weights defining estimator as linear comb. of cluster means
  # note that var_est is not vectorized wrt g
  sum(w^2 * (g + 1/n )) / sum(w)^2
}
```

3. Find SEs and ratios on a sequence of values for g

```
dd <- expand.grid(
  g = seq(0,10, .01)
)
dd <- within(
  dd,
  {

    # Efficiency: True SEs

    se__Y_bar <- sqrt(sapply(g, var_est, ns, ns))  # weights are proportional to sample sizes
    se__Y_mean <- sqrt(sapply(g, var_est, ns, 0*ns + 1)) # weights are constant
    se__Y_IV <- sqrt(
```

```
      sapply(g, function(g) var_est(g, ns, (g + 1/ns)^(-1)))
    )

    # Relative Efficiency: True SE / Best SE

    rel_se__Y_bar <- se__Y_bar/ se__Y_IV
    rel_se__Y_mean <- se__Y_mean/ se__Y_IV

    # Honesty: RMS of Reported SE

    reported_se__Y_bar <- sqrt((g+1)/sum(ns))
    rel_reported_se__Y_bar <- sqrt((g+1)/sum(ns))/se__Y_bar
  }
)
library(latticeExtra)
```

```
## Loading required package: lattice
```

```
library(spida2)
gd(lty=c(1,2,4), lwd = 2)
# xyplot(se__Y_bar + se__Y_mean + se__Y_IV ~ g, dd, type ='l',
#        auto.key = list(space = 'right', lines = T, points =F))
# xyplot(se__Y_mean + se__Y_IV ~ g, dd, type ='l',
#        auto.key = list(space = 'right', lines = T, points =F))
# xyplot(se__Y_mean + se__Y_IV ~ g, dd, type ='l',
#        auto.key = list(space = 'right', lines = T, points =F))
# xyplot(rel_se__Y_mean + rel_se__Y_bar ~ g, dd, type ='l',
#        auto.key = list(space = 'right', lines = T, points =F))


rat1 <- function(g) {
   var_est(g, ns, ns)/var_est(g, ns, (g + 1/ns)^(-1)  )
}
rat2 <- function(g) {
  var_est(g, ns, 0*ns + 1)/var_est(g, ns, (g + 1/ns)^(-1)  )
}
rat1(0)
```

```
## [1] 1
```

```
rat1(1)
```

```
## [1] 1.054225
```

```
rat1(10)
```

```
## [1] 1.068633
```

```
rat2(0)
```

```
## [1] 1.11274
```

```
rat2(1)
```

```
## [1] 1.001714
```

```
rat2(10)
```

```
## [1] 1.000022
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:spida2':
##
##     Null
```

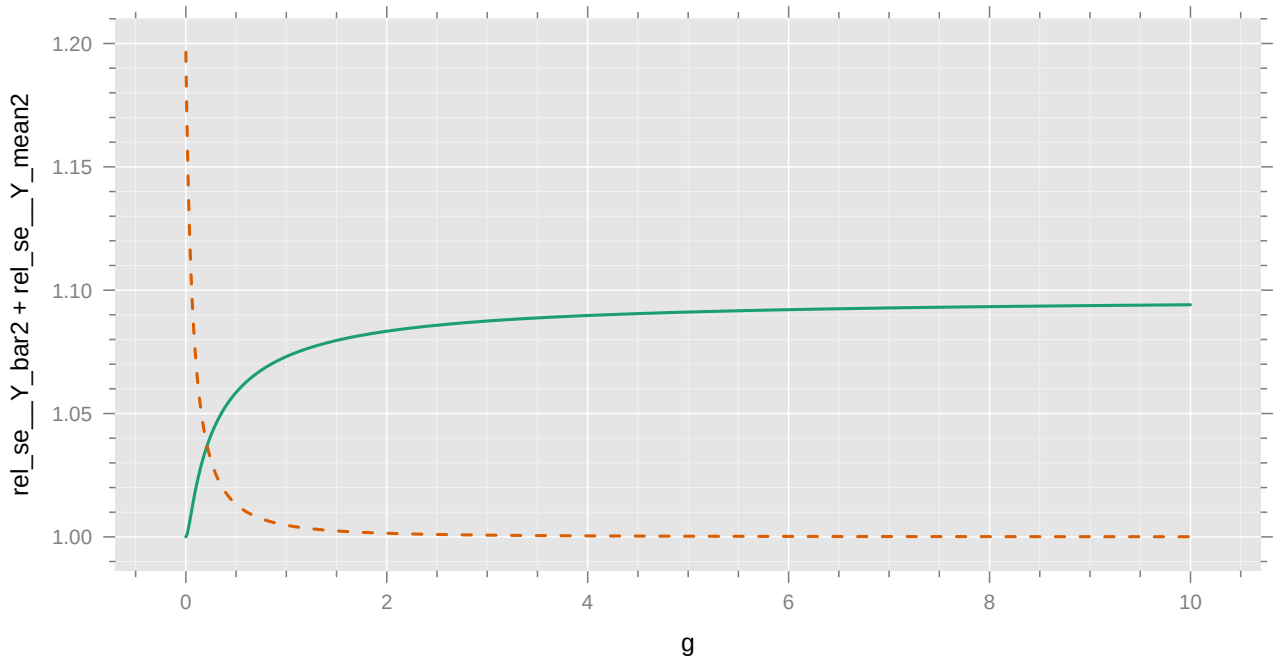
```
?rnegbin
```

```
rng <- function(n, m, v) {
  # negative binomial using mean and variance
  stopifnot(v >= m)
  if(v==m) rpois(n,m)
  else rnegbin(n, m, m^2/(v-m))
}
```

Let's generate a sample with greater variance in n's

```
nsgv <- rng(10, 9, 20) + 1

dd <- within(
  dd,
  {

    # Efficiency: True SEs

    se__Y_bar2 <- sqrt(sapply(g, var_est, nsgv, nsgv))  # weights are proportional to sample sizes
    se__Y_mean2 <- sqrt(sapply(g, var_est, nsgv, 0*nsgv + 1)) # weights are constant
    se__Y_IV2 <- sqrt(
      sapply(g, function(g) var_est(g, nsgv, (g + 1/nsgv)^(-1)))
    )

    # Relative Efficiency: True SE / Best SE
```

```
    rel_se__Y_bar2 <- se__Y_bar2/ se__Y_IV2
    rel_se__Y_mean2 <- se__Y_mean2/ se__Y_IV2

    # Honesty: RMS of Reported SE

    reported_se__Y_bar2 <- sqrt((g+1)/sum(nsgv))
    rel_reported_se__Y_bar2 <- sqrt((g+1)/sum(nsgv))/se__Y_bar2
  }
)

xyplot(rel_se__Y_bar2 + rel_se__Y_mean2 ~ g, dd, type = 'l')
```



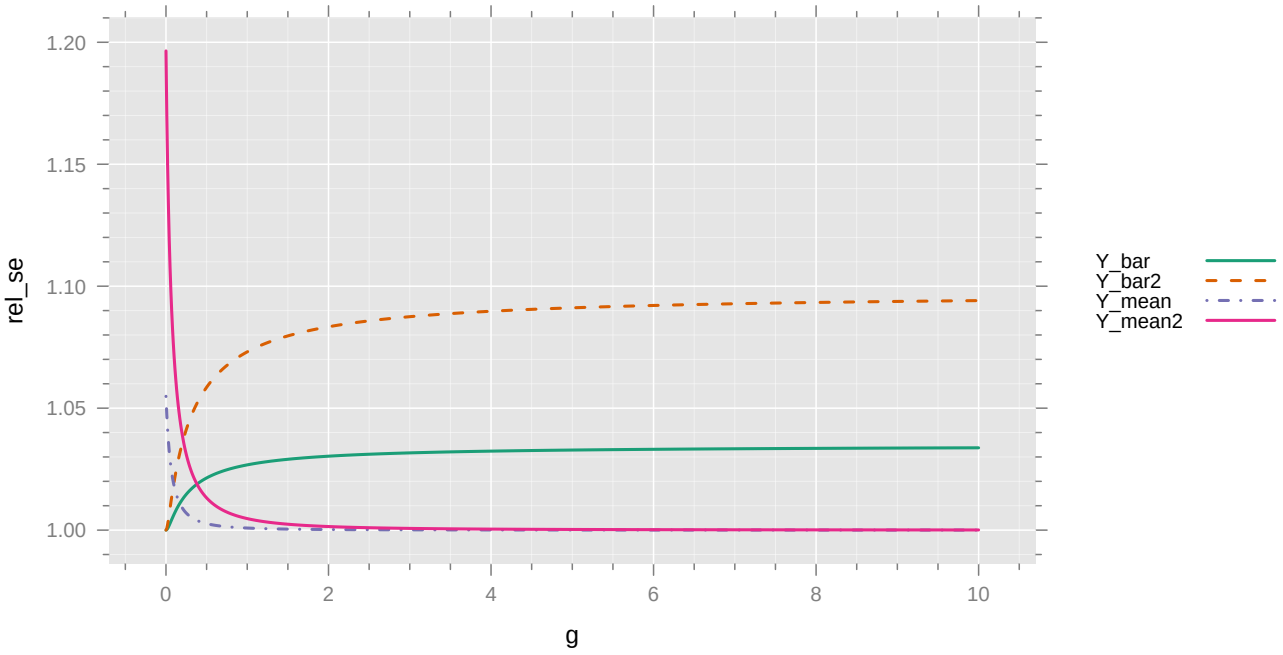
Why the fancy names?

```
dl <- tolong(dd, sep= '__')
head(dl)
```

```
##            g  time rel_reported_se reported_se   rel_se        se id
## 1.Y_bar 0.00 Y_bar       1.0000000   0.1031421 1.000000 0.1031421  1
## 2.Y_bar 0.01 Y_bar       0.9579402   0.1036566 1.000191 0.1082078  2
## 3.Y_bar 0.02 Y_bar       0.9214644   0.1041684 1.000654 0.1130466  3
## 4.Y_bar 0.03 Y_bar       0.8894618   0.1046778 1.001276 0.1176867  4
## 5.Y_bar 0.04 Y_bar       0.8611068   0.1051847 1.001986 0.1221506  5
## 6.Y_bar 0.05 Y_bar       0.8357714   0.1056892 1.002743 0.1264571  6
```

```
subset(dl, !is.na(rel_se)) %>%
```

```
  xyplot(rel_se ~ g, ., groups = time, type = 'l',
       auto.key = list(space = 'right', lines = T, points = F ))
```




```
subset(dl, !is.na(rel_se)) %>%
  xyplot(rel_se ~ g, ., groups = time, type = 'l',
       xlim = c(0,1),
       auto.key = list(space = 'right', lines = T, points = F ))

subset(dl, !is.na(rel_reported_se)) %>%
  xyplot(rel_reported_se ~ g, ., groups = time, type = 'l',
         # xlim = c(0,1),
         auto.key = list(space = 'right', lines = T, points = F ))
```

This is the end for now

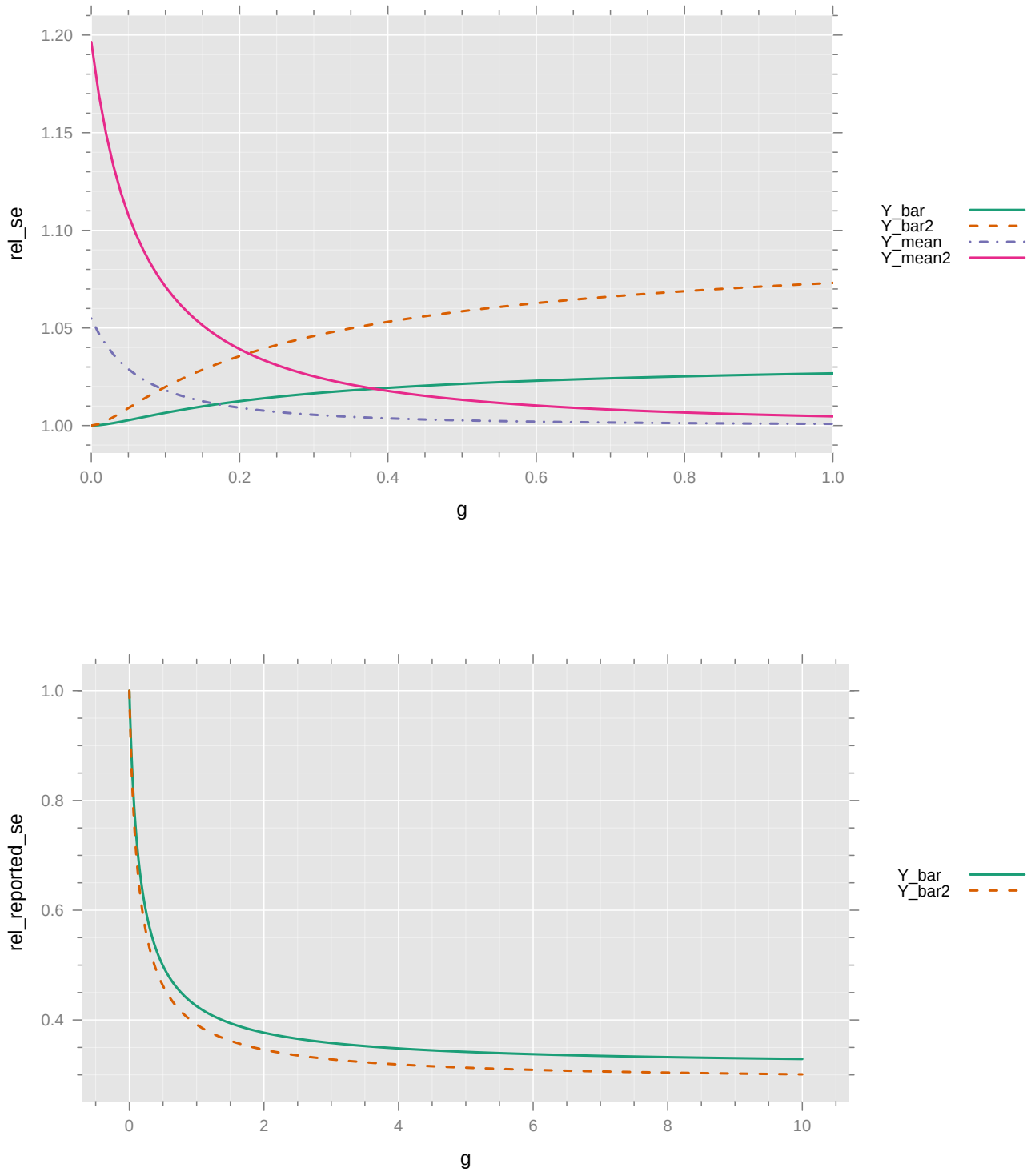


Figure 1: Standard error reported by pooled analysis compared with true standard error.