Examples of regression estimators with different rho functions

Matias Salibian-Barrera

2017-12-04

Here are some examples illustrating the use of `lmrobdetMM()` and `lmrobdetDCML()` with different choices of rho functions (bisquare, optimal and modified optimal).

First, we load the library:

```
library(RobStatTM)
```

The first example uses the `coleman` data available in the `robustbase` package. By default `lmrobdetMM` uses a bisquare rho tuned to 85% asymptotic efficiency for Gaussian errors. The default breakdown point is 50%.

```
data(coleman, package = "robustbase")
coef(m2 <- lmrobdetMM(Y ~ ., data = coleman))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 29.48394602 -1.65635168  0.08326375  0.66565464  1.17784872 -4.00376846
```

```
coef(m0 <- lm(Y ~ ., data = coleman))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 19.94856503 -1.79333327  0.04360156  0.55576006  1.11016839 -1.81092194
```

To use the optimal or modified optimal rho (also with default asymptotic efficiency of 85%), we use

```
coef(m4 <- lmrobdetMM(Y ~ ., data = coleman, control = lmrobdet.control(family = "modified.optimal")))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 29.75770013 -1.69734324  0.08511131  0.66615071  1.18378682 -4.06634517
```

```
coef(m5 <- lmrobdetMM(Y ~ ., data = coleman, control = lmrobdet.control(family = "optimal")))
```

```
## Warning in lmrob.fit(X, y, control, init = S.init, mf = mf): M-step did NOT
## converge. Returning unconverged lM-estimate
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 30.45276762 -1.69449992  0.09009071  0.67811958  1.21315201 -4.34422411
```

The algorithm failed to converge when using the `optimal` rho.

DCML

The DCML estimator is "between" the MM one and the LS one, as expected:

```
coef(m1 <- lmrobdetDCML(Y ~ ., data = coleman))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
##  25.3586308  -1.7156144   0.0661046   0.6181107   1.1485680  -3.0550719
```

```
coef(m2)
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 29.48394602 -1.65635168  0.08326375  0.66565464  1.17784872 -4.00376846
```

```
coef(m0)
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 19.94856503 -1.79333327  0.04360156  0.55576006  1.11016839 -1.81092194
```

Similarly, when we use a non-bisquare rho function:

```
coef(m10 <- lmrobdetDCML(Y ~ ., data = coleman, control = lmrobdet.control(family = "modified.optimal"))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 25.59439462 -1.73808443  0.06749327  0.61929745  1.15254084 -3.10907260
```

```
coef(m4)
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 29.75770013 -1.69734324  0.08511131  0.66615071  1.18378682 -4.06634517
```

```
coef(m0)
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 19.94856503 -1.79333327  0.04360156  0.55576006  1.11016839 -1.81092194
```

Unintuitive behaviour of `modified.optimal`

A behaviour that appears to be unintuitive is that as we increase the desired asymptotic efficiency, the estimators computed with the optimal / modified optimal rho do not seem to get closer to the least squares one.

When using the bisquare family of rho, as we increase the desired asymptotic efficiency, the MM-estimator approaches the least squares one. For example

```
coef(m2.eff <- lmrobdetMM(Y ~ ., data = coleman, control = lmrobdet.control(efficiency = 0.9999)))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 20.49300018 -1.78830691  0.04578737  0.56128534  1.11334086 -1.93145829
```

```
coef(m0)
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 19.94856503 -1.79333327  0.04360156  0.55576006  1.11016839 -1.81092194
```

Something similar happens for the `DCML` estimator based on an MM-estimator computed with a `bisquare` rho function, which seems to be identical to the least squares estimator:

```
coef(m3 <- lmrobdetDCML(Y ~ ., data = coleman, control = lmrobdet.control(efficiency = 0.9999)))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 19.94856503 -1.79333327  0.04360156  0.55576006  1.11016839 -1.81092194
```

However, if we use a `modified.optimal` rho, the results are very different:

```
coef(m7 <- lmrobdetMM(Y ~ ., data = coleman, control = lmrobdet.control(family = "modified.optimal",
    efficiency = 0.9999)))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 34.25477283 -1.61792045  0.08544125  0.67387777  1.11029441 -4.56713985
```

```
coef(m6 <- lmrobdetDCML(Y ~ ., data = coleman, control = lmrobdet.control(family = "modified.optimal",
    efficiency = 0.9999)))
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
```

```
## 27.82362671 -1.69677471  0.06663283  0.62077968  1.11023776 -3.32812247
```

```
coef(m0)
```

```
## (Intercept)     salaryP    fatherWc     sstatus   teacherSc   motherLev
## 19.94856503 -1.79333327  0.04360156  0.55576006  1.11016839 -1.81092194
```