

ANALYSIS OF `lmrob()`

SKETCH: LEVEL 1 + DETAILS

- Get Tuning Parameters for `lmrob()` and Auxiliaries
 - `control = lmrob.control ()`
- Checks, Validations and Initializations
- Get the rank of QR decomposition (`rankQR`)
 - if (R version >= "3.1.0" Version)
 - `z0 = .lm.fit()` (Note: R package, called "stats")
 - `.C(C_Cdqr1s)`
 - else
 - `z0 = lm.fit()` (Note: R package, called "stats")
 - `.C(C_Cdqr1s)`
 - `rankQR` is a field of `z0` (`rankQR = z0$rankQR`)
- if (`rankQR > 0`)
 - Get the initial estimate
 - switch (`init`)
 - case "M-S": `init = lmrob.M.S()`
 - `lmrob.S()`
 - `lmrob.lar()`
 - `.psi.conv.cc()`
 - `.convSs()`
 - `.C(R_lmrob_M_S)`
 - `outlierStats()`
 - case "S": `init = lmrob.S()`
 - `.psi.conv.cc()`
 - `.psi2ipsi()`
 - `.convSs()`
 - `.C(R_lmrob_S)`
 - `lmrob.rweights()`
 - `outlierStats()`
 - Get the estimate `z` executing a MM-type estimator for regression
 - `z = lmrob.fit (... , init = init)`
 - switch (`method`)

- Case "D": `z = lmrob..D..fit()`
 - `lmrob.tau()`
 - `.lmrob.hat()`
 - `.psi.conv.cc()`
 - `.psi2ipsi()`
 - `.Mpsi()`
 - `lmrob.kappa()`
 - `lmrob.E()`
 - `Mwgt()`
 - `ghq()`
 - `.C(R_find_D_scale)`
 - `outlierStats()`
 - Case "M": `z = lmrob..M..fit()`
 - `.psi.conv.cc()`
 - `.psi2ipsi()`
 - `.C(R_lmrob_MM)`
 - `lmrob.rweights()`
 - `outlierStats()`
- `else (rankQR == 0)`
 - `set z of different way`
- `if (weights w is not null)`
 - `Update fields in the z list`
- `if (offset is not null)`
 - `Update other fields in the z list`
 - `z$MD = robMD ()`
 - `covMcd()`
 - `getDefCtrl()`
 - `h.alpha.n()`
 - `.MCDcons()`
 - `wgtFUN()`
 - `.lmrob.hat()`
- `return(z)`

Legend of colours

Green: routines and subroutines in R

Red: call to apis in C

Blue: Important cases