
```

      name: <unnamed>
      log: /Users/connorbrennan/Library/CloudStorage/OneDrive-TheUniversityof
> fChicago/mmb/output/stepwise_regressions/piq_timing_max_all.smcl
      log type: smcl
      opened on: 12 Sep 2025, 19:25:33
- capture ! rm ./output/stepwise_regressions/`depvar'`type'.pdf
= capture ! rm ./output/stepwise_regressions/piq_timing_max_all.pdf
- di "*****"
> *****"
*****"
> **
- di "Outcomes of bi-directional stepwise regressions with `depvar' (type = `t
> ype'))"
= di "Outcomes of bi-directional stepwise regressions with piq_timing_max (typ
> e = all)"
Outcomes of bi-directional stepwise regressions with piq_timing_max (type = al
> l)
- di "*****"
> *****"
*****"
> **
- ${reg_`depvar'`type'}
= glm piq_timing_max rule_g rule_itr estimated i.wlth#i.open cb_authors_ext
> ln_neq i.ntwrth#i.bnkcrcdit ntwrth open bnkcrcdit vint_early vint_mid if piq
> _timing_max < 99 , family(nbinomial) link(log) vce(robust) nolog

```

```

> ----- begin
> glm -----
- local version : di "version " string(_caller()) ", missing:"
- if _by() {
    local BY "`by' `_byvars' `_byrc0':"
  }
- local vceopts bootopts(noheader notable) jkopts(noheader notable) mark(EXP
> osure OFFset mu INIt T CLuster)
- `BY' _vce_parserun glm, `vceopts' robustok multivce : `0'
= _vce_parserun glm, bootopts(noheader notable) jkopts(noheader notable) ma
> rk(EXPosure OFFset mu INIt T CLuster) robustok multivce : piq_timing_max rul
> e_g rule_itr estimated i.wlth#i.open cb_authors_ext ln_neq i.ntwrth#i.bnkc
> rdit ntwrth open bnkcrcdit vint_early vint_mid if piq_timing_max < 99 , fam
> ily(nbinomial) link(log) vce(robust) nolog

```

```

> ----- begin _vce_parse
> run -----
- version 9
- local version : di "version " string(_caller()) ":"

```

```

- local bycall = _by()
- _on_colon_parse `0'
= _on_colon_parse glm, bootopts(noheader notable) jkopts(noheader notable)
> mark(EXPosure OFFset mu INIt T CLuster) robustok multivce : piq_timing_max
> rule_g rule_itr estimated i.wlth#i.open cb_authors_ext ln_neq i.ntwrth#i.b
> nkcrdit ntwrth open bnkcrdit vint_early vint_mid if piq_timing_max < 99 ,
> family(nbinomial) link(log) vce(robust) nolog

```

```

> ----- begin _on_colon_pa
> rse -----
- version 8.2
- sreturn local before ""
- sreturn local after ""
- gettoken before after : 0, parse(":") bind match(par) quotes
- if "`par'" != "" {
= if "" != "" {
local before "`('before')'"
}
- if ``"before"" == ":" {
= if `glm, bootopts(noheader notable) jkopts(noheader notable) mark(EXP
> osure OFFset mu INIt T CLuster) robustok multivce "" == ":" {
sreturn local after ``after'"
exit
}
- while ``COLON'" != ":" & ``after'" != "" {
= while ``'" != ":" & ``: piq_timing_max rule_g rule_itr estimated i.wl
> th#i.open cb_authors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit
> vint_early vint_mid if piq_timing_max < 99 , family(nbinomial) link(log) vce
> (robust) nolog'" != "" {
- gettoken COLON after : after, parse(":") bind match(par) quotes
- if "`par'" != "" {
= if "" != "" {
local before `before' (`COLON')
local COLON
}
- else if ``COLON'" != ":" {
= else if `:"'" != ":" {
local before ``before' `COLON'"
local COLON
}
- }
- while ``COLON'" != ":" & ``after'" != "" {
= while `:"'" != ":" & `` piq_timing_max rule_g rule_itr estimated i.wl
> th#i.open cb_authors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit
> vint_early vint_mid if piq_timing_max < 99 , family(nbinomial) link(log) vce
> (robust) nolog'" != "" {

```

```

    gettoken COLON after : after, parse(":") bind match(par) quotes
    if "`par'" != "" {
      local before `before' (`COLON')
      local COLON
    }
    else if ``"COLON"" != ":" {
      local before ``"before' `COLON""
      local COLON
    }
  }
  - if ``"COLON"" != ":" {
= if `":' != ":" {
  di as err "" found where ':' expected"
  exit 198
}
- sreturn local before ``"before'"
= sreturn local before `glm, bootopts(noheader notable) jkopts(noheader
> notable) mark(EXPosure OFFset mu INIt T CLuster) robustok multivce ""
- sreturn local after ``"after'"
= sreturn local after `piq_timing_max rule_g rule_itr estimated i.wlt
> h#i.open cb_authors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit v
> int_early vint_mid if piq_timing_max < 99 , family(nbinomial) link(log) vce(
> robust) nolog""

_____ end _on_colon_pa
> rse _____
- local ZERO ``"s(before)'"
= local ZERO `glm, bootopts(noheader notable) jkopts(noheader notable) ma
> rk(EXPosure OFFset mu INIt T CLuster) robustok multivce ""
- local 0 ``"s(after)'"
= local 0 `piq_timing_max rule_g rule_itr estimated i.wlth#i.open cb_au
> thors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit vint_early vint
> _mid if piq_timing_max < 99 , family(nbinomial) link(log) vce(robust) nolog"
> '
- quietly syntax [anything(equalok)] [if] [in] [fw aw pw iw] [, VCE(string
> asis) VCE1(string asis) LRMODEL * ]
- if ``"vce'" == "" {
= if `robust'" == "" {
  MYRESULT
  exit
}
- local AFTER ``"0'"
= local AFTER `piq_timing_max rule_g rule_itr estimated i.wlth#i.open c
> b_authors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit vint_early
> vint_mid if piq_timing_max < 99 , family(nbinomial) link(log) vce(robust) no
> log""

```

```

- local 0 ``ZERO''
= local 0 ``glm, bootopts(noheader notable) jkopts(noheader notable) mark(
> EXPosure OFFset mu INIt T CLuster) robustok multivce ''
- syntax name(name=caller id="calling command name") [, WTYPES(string) NOB
> OOTstrap NOJACKknife NOOTHERvce NOTEST PANELdata STdata bootopts(string asis
> ) jkopts(string asis) robustok multivce REQUIREDopts(namelist) NOEQLIST EQop
> ts(passthru) EQARGopts(passthru) noNEEDVARLIST COMMONopts(namelist) MARKopts
> (passthru) SMARKopts(passthru) RMCOLLopts(passthru) RMDCOLLopts(passthru) NU
> MDEPvars(passthru) NEEDEQUAL ALLDEPSMISSing PAREntheses UNPARFIRSTeq OR EQUA
> L UNEQUALFIRSTeq IGNORENOCONS ]
- if ``requiredopts'' != '' {
= if ``'' != '' {
tempname reqopts
.`reqopts' = .__optlist.new
foreach req of local requiredopts {
.`reqopts'.addopt `req', passthru
}
}
- if ``wtypes'' == '' {
= if '' == '' {
- local wtypes fw aw pw iw
- }
- if ``paneldata'' != '' {
= if '' != '' {
local panelopt "CLuster(varname) I(varname)"
}
- if ``stdata'' != '' {
= if '' != '' {
local stopt "CLuster(varname) SHared(varname)"
}
- local typelist svy jackknife bootstrap
- if ``stdata'' == '' & replay() & ! _by() {
= if '' == '' & replay() & ! _by() {
local ecmd `e(cmd)'
local ecmdname `e(cmdname)'
if ``ecmd'' == '' {
error 301
}
else if ``ecmd'' != ``caller'' & ``ecmdname'' == ``caller'' & `:list
> ecmd in typelist' {
`ecmd' `0'
MYRESULT exit
}
MYRESULT
exit
}

```

```

- gettoken svy typelist : typelist
- if "`reqopts'" != "" {
= if "" != "" {
    local REQOPTLIST ``'.`reqopts'.dumpoptions''
    local REQOPTNAMES ``'.`reqopts'.dumpnames''
  }
- local 0 ```AFTER''
= local 0 `` piq_timing_max rule_g rule_itr estimated i.wlth#i.open cb_au
> thors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit vint_early vint
> _mid if piq_timing_max < 99 , family(nbinomial) link(log) vce(robust) nolog"
> '
- quietly syntax [anything(equalok)] [if/] [in] [`wtypes'] [, VCE(string a
> sis) VCE1(string asis) Robust NODOTS `panelopt' `stopt' `REQOPTLIST' * ]
= quietly syntax [anything(equalok)] [if/] [in] [fw aw pw iw] [, VCE(strin
> g asis) VCE1(string asis) Robust NODOTS * ]
- if `:list vce == vce1' {
= if 0 {
    local vce1
  }
- if ```vce1'' != "" & "`multivce" == "" {
= if ``'' != "" & "multivce" == "" {
    di as err "option vce() specified multiple times"
    exit 198
  }
- if "`robust'" != "" & "`robustok" == "" {
= if "" != "" & "robustok" == "" {
    di as err "options vce() and robust may not be combined"
    exit 198
  }
- if ```if'' != "" {
= if ``"piq_timing_max < 99 "' != "" {
- local if ``if (`if)''
= local if ``if (piq_timing_max < 99 )''
- }
- if "`robust'" != "" {
= if "" != "" {
    local options ```robust' `options''
  }
- if "`weight'" != "" {
= if "" != "" {
    local wgt [`weight'`exp']
  }
- if ```options'' != "" {
= if ``"family(nbinomial) link(log) nolog"' != "" {
- local opts ```:list retok options''
= local opts ``"family(nbinomial) link(log) nolog"

```

```

- }
- gettoken uvcetype vceopts : vce, parse(" ,")
- capture unabVCetype, `uvcetype'
= capture unabVCetype, robust

```

```

> _____ begin _vce_parserun.unabVCet
> ype _____
- syntax [, B00Tstrap BStrap JACKknife JACKNife JKnife Robust * ]
- if "`bootstrap" != "" {
= if "" != "" {
    local bootstrap bootstrap
  }
- if "`jknife" != "" | "`jackknife" != "" {
= if "" != "" | "" != "" {
    local jackknife jackknife
  }
- local vcetype ``jackknife' `bootstrap' `robust' `options'
= local vcetype ``robust'
- sreturn local vcetype ``:list retok vcetype'
= sreturn local vcetype `robust'

```

```

> _____ end _vce_parserun.unabVCet
> ype _____
- if c(rc) {
    di as err "invalid vce() option"
    exit c(rc)
  }
- local vcetype `s(vcetype)'
= local vcetype robust
- if "`lrmodel'" != "" {
= if "" != "" {
    _check_lrmodel `caller', vcetype(`vcetype')
  }
- if ``vce1'" != "" & "`multivce" != "" {
= if `"' != "" & "multivce" != "" {
    gettoken uvceltype vcelopts : vce1, parse(" ,")
    capture unabVCetype, `uvceltype'
    if c(rc) {
      di as err "invalid vce() option"
      exit c(rc)
    }
    local vceltype `s(vcetype)'
    if `:list vceltype in typelist' | `:list vcetype in typelist' {
      di as err "options vce(`vce') and vce(`vce1') cannot be combined"
      exit 198
    }
  }

```

```

MYRESULT
exit
}
- if _caller() >= 17 {
- if "`vcetype'" == "cluster" {
= if "robust" == "cluster" {
  gettoken vceopts vcelopts: vceopts, parse(",")
  if `:list sizeof vceopts' > 1 {
    local typelist "`typelist' cluster_multiway"
    local vcetype cluster_multiway
    if `:list sizeof vcelopts' > 0 {
      gettoken comma vcelopts: vcelopts, parse(",")
      local vcelopts: list retokenize vcelopts
      di as err "{p}VCE option " "{bf:`vcelopts'} not allowed{p_end}"
      exit 198
    }
    if !inlist("`caller'", "areg", "regress", "xtreg") {
      di as err "{p}multiway cluster VCE " "not allowed{p_end}"
      exit 198
    }
  }
  MYRESULT
  exit
}
}
- }
- if ! `:list vcetype in typelist' {
= if ! 0 {
- if "`vcetype'" != "robust" & "`noothervce'" == "noothervce" {
= if "robust" != "robust" & "" == "noothervce" {
  di as err "option vce(`uvcetype') not allowed"
  exit 198
}
- MYRESULT
}

> _____ begin _vce_parserun.MYRES
> ULT _____
  - args exit
  - sreturn local exit ``exit'""
  = sreturn local exit `""'

> _____ end _vce_parserun.MYRES
> ULT _____
  - exit

> _____ end _vce_parse
> run _____

```

```

- if "`s(exit)'" != "" {
= if "" != "" {
  version 10: ereturn local cmdline `"glm `0'"
  if e(vf) != 1 {
    tempname V
    matrix `V' = e(vf)*e(V)
    version 9: ereturn repost V = `V'
  }
  version 9: ereturn local clustvar ``e(cluster)'"
  version 9: ereturn local cluster
  quietly syntax [anything] [fw aw iw pw] [if] [in] [, noHEADer noTABLE Leve
> l(passthru) eform * ]
  if ! `:length local level' {
    if ``s(level)'" != "" {
      local level `"level(`s(level))'"
    }
  }
  _get_diopts diopts options, `options'
  version 7: Display, `header' `table' `level' `eform' `diopts'
  exit
}
- version 7, missing
- if replay() {
  if _by() {
    error 190
  }
  if ``e(cmd)'" != "glm" {
    error 301
  }
  if ``e(predict)'" == "glm_p" {
    glm_6 `0'
    exit
  }
  Display `0'
  error `e(rc)'
  exit
}
- if _caller() < 7 {
  mac drop SGLM_*
  capture noi `BY' glm_6 `0'
  mac drop SGLM_*
  exit _rc
}
- mac drop SGLM_running SGLM_nonstan
- capture noisily `version' `BY' glm_7 `0'
= capture noisily version 18.5, missing: glm_7 piq_timing_max rule_g rule_i

```



```

> tr estimated i.wlth#i.open cb_authors_ext ln_neq i.ntwrth#i.bnkcrcdit ntwr
> th open bnkcrcdit vint_early vint_mid if piq_timing_max < 99 , family(nbinom
> ial) link(log) vce(robust) nolog

----- begin glm.gl
> m_7 -----
- version 7, missing
- local oldIC oldic
- if _caller() < 9 {
  local iteropt ITERate(integer 50)
}
- else local iteropt ITERate(integer `c(maxiter)`)
= else local iteropt ITERate(integer 300)
- local dltol -1
- syntax varlist(numeric fv ts) [fw aw pw iw] [if] [in] [, EForm FROM(stri
> ng) LLevel(cilevel) OFFset(varname numeric) LNOFFset(varname numeric) noCONSt
> ant Robust CLuster(varname) Family(string) Link(string) ASIS SEARCH IRLS ML
> LTOLerance(real `dltol`) SCore(string) NOLOg LOg noTABLE noDISPLAY FISHER(in
> teger -1) noDOTS `iteropt' TRAcE SCALe(string) estscale MU(varname) INIt(var
> name) T(string) noHEADer DISP(real 1) VFactor(real 1) EXPosure(varname numer
> ic) moptobj(passthru) * ]
  = syntax varlist(numeric fv ts) [fw aw pw iw] [if] [in] [, EForm FROM(stri
> ng) LLevel(cilevel) OFFset(varname numeric) LNOFFset(varname numeric) noCONSt
> ant Robust CLuster(varname) Family(string) Link(string) ASIS SEARCH IRLS ML
> LTOLerance(real -1) SCore(string) NOLOg LOg noTABLE noDISPLAY FISHER(integer
> -1) noDOTS ITERate(integer 300) TRAcE SCALe(string) estscale MU(varname) IN
> It(varname) T(string) noHEADer DISP(real 1) VFactor(real 1) EXPosure(varname
> numeric) moptobj(passthru) * ]
  - local FROM_ORIG : copy local from
  - local fvops = "`s(fvops)'" == "true" | _caller() >= 11
  = local fvops = "true" == "true" | _caller() >= 11
  - if `fvops' {
  = if 1 {
    - local VV : di "version " string(max(11,_caller())) ", missing:"
    - local MM e2
    - local negh negh
    - }
    - else {
      local VV "version 8.1:"
      local MM d2
    }
  - GetVCE, `irls' cluster(`cluster') `robust' `options'
  = GetVCE, cluster() vce(robust)

----- begin glm.Get
> VCE -----

```

```

- version 9
- syntax [fw aw iw pw] [, IRLS BSTRAP BREP(integer -1) JKNIFE JKNIFE1 NW
> EST(string) OIM OPG BHHH UNBiased VCE(string) VCE1(string) VCE2(string) CLus
> ter(varname) Robust * ]
- if "`weight'" != "" {
= if "" != "" {
    local wt [`weight'`exp']
  }
- local len : length local vce
- if "`vce'" == bsubstr("unbiased", 1, max(3,`len')) {
= if "robust" == bsubstr("unbiased", 1, max(3,6)) {
    local vce : copy local vce1
    local vce1 : copy local vce2
    local vce2
    local unbiased "unbiased"
  }
- local len : length local vce1
- if "`vce1'" == bsubstr("unbiased", 1, max(3,`len')) {
= if "" == bsubstr("unbiased", 1, max(3,0)) {
    local vce1 : copy local vce2
    local vce2
    local unbiased "unbiased"
  }
- local len : length local vce2
- if "`vce2'" == bsubstr("unbiased", 1, max(3,`len')) {
= if "" == bsubstr("unbiased", 1, max(3,0)) {
    local vce2
    local unbiased "unbiased"
  }
- if "`vce2'" != "" {
= if "" != "" {
    di as err "too many vce() options specified"
    exit 198
  }
- if ("`bhhh'" != "") local opg opg
= if ("") local opg opg
- local oldvce `bstrap' `jknife' `jknife1' `opg' `unbiased'
= local oldvce
- if ("`nwest'" != "") local oldvce ``oldvce' nwest(`newst')""
= if ("") local oldvce ``nwest()'""
- opts_exclusive ``oldvce'""
= opts_exclusive ""

```

```

> ----- begin opts_exclus

```

```

> ive

```

```

- version 8.2

```

```

- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- if "`irls'" == "" {
= if "" == "" {
- local allow eim
- }
- else {
  local allow eim oim
}
- local optlist EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIVE
- local argoptlist CLuster HAC
- _vce_parse, opt(`optlist') argopt(`argoptlist') pw(`allow') old : `wt'
> , vce(`vce') cluster(`cluster') `robust'
  = _vce_parse, opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIVE)
> argopt(CLuster HAC) pw(eim) old : , vce(robust) cluster()

> ----- begin _vce_pa
> rse -----
- version 10
- _on_colon_parse `0'
= _on_colon_parse , opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased
> NATIVE) argopt(CLuster HAC) pw(eim) old : , vce(robust) cluster()

> ----- begin _on_colon_pa
> rse -----
- version 8.2
- sreturn local before ""
- sreturn local after ""
- gettoken before after : 0, parse(":") bind match(par) quotes
- if "`par'" != "" {
= if "" != "" {
  local before `("before")'
}

```

```

- if ``before''' == ":" {
= if `", opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIVE)
> argopt(CLuster HAC) pw(eim) old '' == ":" {
    sreturn local after ``after'''
    exit
  }
- while ``COLON''' != ":" & ``after''' != "" {
= while `''' != ":" & `":", vce(robust) cluster() '' != "" {
- gettoken COLON after : after, parse(":") bind match(par) quotes
- if "par'" != "" {
= if "" != "" {
    local before `before' (`COLON')
    local COLON
  }
- else if ``COLON''' != ":" {
= else if `":'" != ":" {
    local before ``before' `COLON'''
    local COLON
  }
- }
- while ``COLON''' != ":" & ``after''' != "" {
= while `":'" != ":" & `", vce(robust) cluster() '' != "" {
    gettoken COLON after : after, parse(":") bind match(par) quotes
    if "par'" != "" {
      local before `before' (`COLON')
      local COLON
    }
    else if ``COLON''' != ":" {
      local before ``before' `COLON'''
      local COLON
    }
  }
- if ``COLON''' != ":" {
= if `":'" != ":" {
    di as err "" found where ':' expected"
    exit 198
  }
- sreturn local before ``before'''
= sreturn local before `", opt(EIM OIM OPG Robust jackknife1 jknife1
> UNBiased NATIVE) argopt(CLuster HAC) pw(eim) old ''
- sreturn local after ``after'''
= sreturn local after `", vce(robust) cluster() ''

> _____ end _on_colon_pa
> rse _____
- local 0 ``s(before)'''

```

```

    = local 0 `", opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIV
> E) argopt(CLuster HAC) pw(eim) old ""
    - local user ``s(after)'''
    = local user `", vce(robust) cluster() ""
    - syntax [varname(default=none)], [ OPTlist(namelist) ARGOPTlist(namel
> ist) PWallowed(namelist) OLD_is_ok NOARGOPT_is_ok ]
    - local touse `varlist'
    = local touse
    - tempname o a
    - `o' = ._optlist.new
    = __000001 = ._optlist.new
    -class {
    -instance:
    -      array options    = {}
    -      array names      = {}
    -}
    - local optlist : list uniq optlist
    - foreach name of local optlist {
    - `o'.adopt `name'
    = __000001.adopt EIM
    _____ begin _optlist.add
> _____
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("EIM")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "EIM"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "eim"
    _____ end _optlist.add
> _____
> opt _____
    - }
    - `o'.adopt `name'
    = __000001.adopt OIM
    _____ begin _optlist.add
> _____
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]

```

```

- local mname = lower("`optspec'")
= local mname = lower("OIM")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "OIM"
- .names.Arrpush "`mname'"
= .names.Arrpush "oim"

```

```

> ----- end _optlist.add
> opt
- }
- .`o'.addopt `name'
= .__000001.addopt OPG

```

```

> ----- begin _optlist.add
> opt
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("OPG")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "OPG"
- .names.Arrpush "`mname'"
= .names.Arrpush "opg"

```

```

> ----- end _optlist.add
> opt
- }
- .`o'.addopt `name'
= .__000001.addopt Robust

```

```

> ----- begin _optlist.add
> opt
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("Robust")
- if "`passthru'" != "" {
= if "" != "" {

```

```

        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "Robust"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "robust"
}
> _____ end _optlist.add
> opt _____
    - }
    - .`o'.addopt `name'
    = .__000001.addopt jackknife1
}
> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("jackknife1")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "jackknife1"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "jackknife1"
}
> _____ end _optlist.add
> opt _____
    - }
    - .`o'.addopt `name'
    = .__000001.addopt jknife1
}
> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("jknife1")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "jknife1"
}

```

```

- .names.Arrpush "`mname'"
= .names.Arrpush "jknife1"
> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__000001.addopt UNBiased
> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("UNBiased")
- if "`passthru'" != "" {
= if "" != "" {
  local optspec `optspec'(passthru)
}
- .options.Arrpush "`optspec'"
= .options.Arrpush "UNBiased"
- .names.Arrpush "`mname'"
= .names.Arrpush "unbiased"
> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__000001.addopt NATIVE
> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("NATIVE")
- if "`passthru'" != "" {
= if "" != "" {
  local optspec `optspec'(passthru)
}
- .options.Arrpush "`optspec'"
= .options.Arrpush "NATIVE"
- .names.Arrpush "`mname'"
= .names.Arrpush "native"
> _____ end _optlist.add

```



```

> opt
- }
- .`a' = ._optlist.new
= __000002 = ._optlist.new
- local argoptlist : list uniq argoptlist
- foreach name of local argoptlist {
- .`a'.adopt `name'
= __000002.adopt CLuster

> _____ begin _optlist.add
> opt
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("CLuster")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "CLuster"
- .names.Arrpush "`mname'"
= .names.Arrpush "cluster"

> _____ end _optlist.add
> opt
- }
- .`a'.adopt `name'
= __000002.adopt HAC

> _____ begin _optlist.add
> opt
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("HAC")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "HAC"
- .names.Arrpush "`mname'"
= .names.Arrpush "hac"

> _____ end _optlist.add

```

```

> opt -----
- }
- local OPTNAMES ```.o'.dumpnames'''
-----

> ----- begin _optlist.dumpna
> mes -----
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/8 {
- local namelist `namelist' `.names['i']'
= local namelist eim
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1 unbiased
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1 unbiased nati
> ve
- }
- local namelist : list uniq namelist
- sreturn local namelist ```.namelist'''
= sreturn local namelist `eim oim opg robust jackknife1 jknife1 unb
> iased native''
- class exit ```.namelist'''
= class exit `eim oim opg robust jackknife1 jknife1 unbiased native
> ''
-----

> ----- end _optlist.dumpna
> mes -----
= local OPTNAMES `eim oim opg robust jackknife1 jknife1 unbiased nati
> ve''

```

```

- local ARGOPTNAMES ```.a'.dumpnames'''


---


> 

---

 begin _optlist.dumpna
> mes 

---


- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/2 {
- local namelist `namelist' `.names['i']'
= local namelist cluster
- }
- local namelist `namelist' `.names['i']'
= local namelist cluster hac
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist "cluster hac"
- class exit ``namelist'''
= class exit "cluster hac"


---


> 

---

 end _optlist.dumpna
> mes 

---


= local ARGOPTNAMES "cluster hac"
- local AND : list OPTNAMES & ARGOPTNAMES
- if `:list sizeof AND' {
= if 0 {
di as err "vcetype `:word 1 of `AND'' specified twice"
exit 198
}
- local pwallowed : list uniq pwallowed
- local pwlist robust cluster
- foreach n of local pwallowed {
- local lc = lower("`n'")
= local lc = lower("eim")
- local pwlist `pwlist' `lc'
= local pwlist robust cluster eim
- }
- if "`old_is_ok'" != "" {
= if "old_is_ok" != "" {
- local ROBUST robust
- if `:list ROBUST in OPTNAMES' {
= if 1 {
- local robust Robust
- }
- local CLUSTER cluster
- if `:list CLUSTER in ARGOPTNAMES' {
= if 1 {

```

```

- local cluster CCluster(varname)
- }
- }
- local 0 ``user'''
= local 0 `` , vce(robust) cluster() ''
- syntax [fw aw pw iw] [, VCE(string asis) `robust' `cluster' ]
= syntax [fw aw pw iw] [, VCE(string asis) Robust CCluster(varname) ]
- if ``vce''' != "" {
= if ``robust''' != "" {
- gettoken name args : vce, parse(" ,")
- local args : list retok args
- capture ParseVCE `o' `a' , `name'
= capture ParseVCE __000001 __000002 , robust

```

```

> _____ begin _vce_parse.Parse
> VCE _____
- gettoken 0 0 : 0, parse(" ,")
- gettoken A 0 : 0, parse(" ,")
- local OPTSPECS ```.0'.dumpoptions' `.`A'.dumpoptions'''

```

```

> _____ begin _optlist.dumpopti
> ons _____
- version 9
- forval i = 1/`.options.arnels' {
= forval i = 1/8 {
- local optlist `optlist' `.options['i']'
= local optlist EIM
- }
- local optlist `optlist' `.options['i']'
= local optlist EIM OIM
- }
- local optlist `optlist' `.options['i']'
= local optlist EIM OIM OPG
- }
- local optlist `optlist' `.options['i']'
= local optlist EIM OIM OPG Robust
- }
- local optlist `optlist' `.options['i']'
= local optlist EIM OIM OPG Robust jackknife1
- }
- local optlist `optlist' `.options['i']'
= local optlist EIM OIM OPG Robust jackknife1 jknife1
- }
- local optlist `optlist' `.options['i']'
= local optlist EIM OIM OPG Robust jackknife1 jknife1 UNBiased
- }

```

```

- local optlist `optlist' `.options['i']'
= local optlist EIM OIM OPG Robust jackknife1 jknife1 UNBiased NAT
> IVE
- }
- local optlist : list uniq optlist
- sreturn local optlist ```optlist'''
= sreturn local optlist `"EIM OIM OPG Robust jackknife1 jknife1 UN
> Biased NATIVE"'
- class exit ```optlist'''
= class exit `"EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATI
> VE"'
_____
> _____ end _optlist.dumpopti
> ons _____
_____
> _____ begin _optlist.dumpopti
> ons _____
- version 9
- forval i = 1/`.options.arnels' {
= forval i = 1/2 {
- local optlist `optlist' `.options['i']'
= local optlist CLuster
- }
- local optlist `optlist' `.options['i']'
= local optlist CLuster HAC
- }
- local optlist : list uniq optlist
- sreturn local optlist ```optlist'''
= sreturn local optlist `"CLuster HAC"'
- class exit ```optlist'''
= class exit `"CLuster HAC"'
_____
> _____ end _optlist.dumpopti
> ons _____
= local OPTSPECS `"EIM OIM OPG Robust jackknife1 jknife1 UNBiased NA
> TIVE CLuster HAC"'
- local OPTNAMES ```. `0'.dumpnames' `.`A'.dumpnames'''
_____
> _____ begin _optlist.dumpna
> mes _____
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/8 {
- local namelist `namelist' `.names['i']'
= local namelist eim
- }

```

```

- local namelist `namelist' `.names['i']'
= local namelist eim oim
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1 unbiased
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1 unbiased na
> tive
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist `eim oim opg robust jackknife1 jknife1 u
> nbiased native'''
- class exit ``namelist'''
= class exit `eim oim opg robust jackknife1 jknife1 unbiased nati
> ve'''
_____
> _____ end _optlist.dumpna
> mes _____
_____
> _____ begin _optlist.dumpna
> mes _____
- version 9
- forval i = 1/`.names.arrnels' {
= forval i = 1/2 {
- local namelist `namelist' `.names['i']'
= local namelist cluster
- }
- local namelist `namelist' `.names['i']'
= local namelist cluster hac
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''

```

```

        = sreturn local namelist `"cluster hac"
        - class exit `""namelist""
        = class exit `"cluster hac"
    }
}

> ----- end _optlist.dumpna
> mes -----
    = local OPTNAMES `"eim oim opg robust jackknife1 jknife1 unbiased na
> tive cluster hac"
    - syntax [, `OPTSPECS']
    = syntax [, EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIVE CL
> uster HAC]
    - foreach OPT of local OPTNAMES {
    - if `""`OPT`"" != "" {
    = if `""` != "" {
        return local vce `OPT'
        continue, break
    }
    - }
    - if `""`OPT`"" != "" {
    = if `""` != "" {
        return local vce `OPT'
        continue, break
    }
    - }
    - if `""`OPT`"" != "" {
    = if `""` != "" {
        return local vce `OPT'
        continue, break
    }
    - }
    - if `""`OPT`"" != "" {
    = if `"robust" != "" {
        - return local vce `OPT'
        = return local vce robust
        - continue, break
        }
    }
}

> ----- end _vce_parse.Parse
> VCE -----
    - if c(rc) {
        di as err "vcetype {bf:`name'} not allowed"
        exit 198
    }
    - local name `r(vce)'
    = local name robust

```

```

- if "`weight'" == "pweight" & !':list name in pwlist' {
= if "" == "pweight" & !1 {
  di as err "pweights are not allowed with vce(`name')"
  exit 198
}
- return add
- if ``"robust" != "" & !inlist("`name'", "", "robust", "eim", "oim", "cl
> uster") {
= if `""' != "" & !inlist("robust", "", "robust", "eim", "oim", "cluster")
> {
  opts_exclusive "`robust' vce(`name')"
}
- if "`name'" == "cluster" {
= if "robust" == "cluster" {
  if ``"args" == "" {
    di as err "invalid vce(cluster) option"
    exit 198
  }
  confirm var `args'
  unab args : `args'
  local markout `args'
  if "`cluster'" != "" {
    if !':list args == cluster' {
      di as err "options cluster() and vce(cluster) are in conflict"
      exit 100
    }
  }
}
- else if "`cluster'" != "" & !inlist("`name'", "eim", "oim") {
= else if "" != "" & !inlist("robust", "eim", "oim") {
  opts_exclusive "cluster() vce(`name')"
}
- if ``"args" != "" {
= if `""' != "" {
  if !':list name in ARGOPTNAMES' {
    di as err "invalid vce() option"
    exit 198
  }
  if "`name'" == "hc2" {
    confirm var `args'
    unab args : `args'
    local markout `args'
  }
  return local vceargs ``':list retok args''
}
- else if !':list name in ARGOPTNAMES' {

```



```

= else if 0 {
  if "`noargopt_is_ok'" == "" {
    di as err "invalid vce() option"
    exit 198
  }
}
- }
- else {
  if "`weight'" == "pweight" | "`robust'" != "" {
    local vce robust
  }
  if "`cluster'" != "" {
    local vce cluster
    local args `cluster'
    local markout `cluster'
  }
  return local vce ``vce'''
  return local vceargs ``args'''
}
- if "`return(vce)'" != "" {
= if "robust" != "" {
- if ``return(vceargs)''' != "" {
= if `''' != "" {
  local ss " "
}
- return local vceopt `"vce(`return(vce)''ss'`return(vceargs)')'"
= return local vceopt `"vce(robust)'"
- }
- if "`return(vce)'" == "cluster" {
= if "robust" == "cluster" {
  return local cluster `return(vceargs)'
}
- if inlist("`return(vce)'", "robust", "cluster") {
= if inlist("robust", "robust", "cluster") {
- return local robust robust
- }
- if "`touse'" != "" {
= if "" != "" {
  if "`markout'" != "" {
    markout `touse' `markout', strok
  }
}
}
}

> _____ end _vce_pa
> rse _____
- if "`r(vce)'" != "" {

```

```

= if "robust" != "" {
- if "`r(cluster)'" != "" {
= if "" != "" {
    local cluster `r(cluster)'
    local vce `r(vce)'
  }
- else local vce `r(vce)' `r(vceargs)'
= else local vce robust
- }
- _vce_parse, opt(`optlist') argopt(`argoptlist') old :, vce(`vce1')
= _vce_parse, opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIVE)
> argopt(CLuster HAC) old :, vce()


---


> 

---

 begin _vce_pa
> rse 

---


    - version 10
    - _on_colon_parse `0'
    = _on_colon_parse , opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased
> NATIVE) argopt(CLuster HAC) old :, vce()


---


> 

---

 begin _on_colon_pa
> rse 

---


    - version 8.2
    - sreturn local before ""
    - sreturn local after ""
    - gettoken before after : 0, parse(":") bind match(par) quotes
    - if "`par'" != "" {
    = if "" != "" {
        local before `("before")'
    }
    - if ``before' == ":" {
    = if `', opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIVE)
> argopt(CLuster HAC) old '' == ":" {
        sreturn local after ``after'
        exit
    }
    - while ``COLON' != ":" & ``after' != "" {
    = while ``' != ":" & `:', vce()' != "" {
    - gettoken COLON after : after, parse(":") bind match(par) quotes
    - if "`par'" != "" {
    = if "" != "" {
        local before `before' (`COLON')
        local COLON
    }
    - else if ``COLON' != ":" {
    = else if `:"' != ":" {

```

```

        local before ``before' `COLON'
        local COLON
    }
- }
- while ``COLON' != ":" & ``after' != "" {
= while `:"' != ":" & `, vce()' != "" {
    gettoken COLON after : after, parse(":") bind match(par) quotes
    if ``par' != "" {
        local before `before' (`COLON')
        local COLON
    }
    else if ``COLON' != ":" {
        local before ``before' `COLON'
        local COLON
    }
}
- if ``COLON' != ":" {
= if `:"' != ":" {
    di as err "" found where ':' expected"
    exit 198
}
- sreturn local before ``before'
= sreturn local before `', opt(EIM OIM OPG Robust jackknife1 jknife1
> UNBiased NATIV) argopt(CLuster HAC) old ''
- sreturn local after ``after'
= sreturn local after `', vce())

> _____ end _on_colon_pa
> rse _____
- local 0 ``s(before)'
= local 0 `', opt(EIM OIM OPG Robust jackknife1 jknife1 UNBiased NATIV
> E) argopt(CLuster HAC) old ''
- local user ``s(after)'
= local user `', vce()'
- syntax [varname(default=none)], [ OPTlist(namelist) ARGOPTlist(namel
> ist) PWallowed(namelist) OLD_is_ok NOARGOPT_is_ok ]
- local touse `varlist'
= local touse
- tempname o a
- .`o' = . _optlist.new
= .__000001 = . _optlist.new
-class {
-instance:
-         array options    = {}
-         array names      = {}
-}

```

```

- local optlist : list uniq optlist
- foreach name of local optlist {
-   .`o`.addopt `name'
=   .__000001.addopt EIM
}

> ----- begin _optlist.add
> opt -----
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("EIM")
- if "`passthru'" != "" {
= if "" != "" {
  local optspec `optspec'(passthru)
}
- .options.Arrpush "`optspec'"
= .options.Arrpush "EIM"
- .names.Arrpush "`mname'"
= .names.Arrpush "eim"
}

> ----- end _optlist.add
> opt -----
- }
- .`o`.addopt `name'
= .__000001.addopt OIM
}

> ----- begin _optlist.add
> opt -----
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("OIM")
- if "`passthru'" != "" {
= if "" != "" {
  local optspec `optspec'(passthru)
}
- .options.Arrpush "`optspec'"
= .options.Arrpush "OIM"
- .names.Arrpush "`mname'"
= .names.Arrpush "oim"
}

> ----- end _optlist.add
> opt -----
- }
- .`o`.addopt `name'
= .__000001.addopt OPG
}

```

```

> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("OPG")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "OPG"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "opg"
    _____

> _____ end _optlist.add
> opt _____
    - }
    - .`o'.addopt `name'
    = .__000001.addopt Robust
    _____

> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("Robust")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "Robust"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "robust"
    _____

> _____ end _optlist.add
> opt _____
    - }
    - .`o'.addopt `name'
    = .__000001.addopt jackknife1
    _____

> _____ begin _optlist.add
> opt _____
    - version 9

```

```

- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("jackknife1")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "jackknife1"
- .names.Arrpush "`mname'"
= .names.Arrpush "jackknife1"

```

```

> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__000001.addopt jknife1

```

```

> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("jknife1")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "jknife1"
- .names.Arrpush "`mname'"
= .names.Arrpush "jknife1"

```

```

> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__000001.addopt UNBiased

```

```

> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("UNBiased")
- if "`passthru'" != "" {

```

```

    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "UNBiased"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "unbiased"
    _____
> _____ end _optlist.add
> opt _____
    - }
    - .`o'.addopt `name'
    = .__000001.addopt NATIVE
    _____
> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("NATIVE")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "NATIVE"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "native"
    _____
> _____ end _optlist.add
> opt _____
    - }
    - .`a' = ._optlist.new
    = .__000002 = ._optlist.new
    - local argoptlist : list uniq argoptlist
    - foreach name of local argoptlist {
    - .`a'.addopt `name'
    = .__000002.addopt CLuster
    _____
> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("CLuster")
    - if "`passthru'" != "" {

```

```

= if "" != "" {
    local optspec `optspec'(passthru)
}
- .options.Arrpush "`optspec'"
= .options.Arrpush "CLuster"
- .names.Arrpush "`mname'"
= .names.Arrpush "cluster"

```

```

> ----- end _optlist.add
> opt
    - }
    - .`a'.addopt `name'
    = .__000002.addopt HAC

```

```

> ----- begin _optlist.add
> opt
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("HAC")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "HAC"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "hac"

```

```

> ----- end _optlist.add
> opt
    - }
    - local OPTNAMES ```.`o'.dumpnames''

```

```

> ----- begin _optlist.dumpna
> mes
    - version 9
    - forval i = 1/`.names.arnels' {
    = forval i = 1/8 {
    - local namelist `namelist' `.names['i']'
    = local namelist eim
    - }
    - local namelist `namelist' `.names['i']'
    = local namelist eim oim
    - }
    - local namelist `namelist' `.names['i']'

```



```

= local namelist eim oim opg
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1 unbiased
- }
- local namelist `namelist' `.names['i']'
= local namelist eim oim opg robust jackknife1 jknife1 unbiased nati
> ve
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist "eim oim opg robust jackknife1 jknife1 unb
> iased native"
- class exit ``namelist'''
= class exit "eim oim opg robust jackknife1 jknife1 unbiased native
> ""
_____
> _____ end _optlist.dumpna
> mes _____
= local OPTNAMES "eim oim opg robust jackknife1 jknife1 unbiased nati
> ve"
- local ARGOPTNAMES ```.a'.dumpnames'''
_____
> _____ begin _optlist.dumpna
> mes _____
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/2 {
- local namelist `namelist' `.names['i']'
= local namelist cluster
- }
- local namelist `namelist' `.names['i']'
= local namelist cluster hac
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist "cluster hac"

```

```

- class exit ``namelist'''
= class exit ``cluster hac''

> _____ end _optlist.dumpna
> mes _____
= local ARGOPTNAMES ``cluster hac''
- local AND : list OPTNAMES & ARGOPTNAMES
- if `:list sizeof AND' {
= if 0 {
di as err "vcetype ``:word 1 of `AND'' specified twice"
exit 198
}
- local pwallowed : list uniq pwallowed
- local pwlist robust cluster
- foreach n of local pwallowed {
local lc = lower("`n'")
local pwlist `pwlist' `lc'
}
- if "`old_is_ok'" != "" {
= if "old_is_ok" != "" {
- local ROBUST robust
- if `:list ROBUST in OPTNAMES' {
= if 1 {
- local robust Robust
- }
- local CLUSTER cluster
- if `:list CLUSTER in ARGOPTNAMES' {
= if 1 {
- local cluster CLuster(varname)
- }
- }
- local 0 ``user'''
= local 0 `", vce()''
- syntax [fw aw pw iw] [, VCE(string asis) `robust' `cluster' ]
= syntax [fw aw pw iw] [, VCE(string asis) Robust CLuster(varname) ]
- if ``vce''' != "" {
= if `''' != "" {
gettoken name args : vce, parse(" ,")
local args : list retok args
capture ParseVCE `o' `a' , `name'
if c(rc) {
di as err "vcetype {bf:`name'} not allowed"
exit 198
}
local name `r(vce)'
if "weight" == "pweight" & !`:list name in pwlist' {

```

```

di as err "pweights are not allowed with vce(`name')"
exit 198
}
return add
if ```robust`'' != "" & !inlist("`name`","", "robust", "eim", "oim", "cl
> uster") {
opts_exclusive "`robust' vce(`name')"
}
if "`name'" == "cluster" {
if ```args`'' == "" {
di as err "invalid vce(cluster) option"
exit 198
}
confirm var `args'
unab args : `args'
local markout `args'
if "`cluster'" != "" {
if !`:list args == cluster' {
di as err "options cluster() and vce(cluster) are in conflict"
exit 100
}
}
}
else if "`cluster'" != "" & !inlist("`name`","", "eim", "oim") {
opts_exclusive "cluster() vce(`name')"
}
if ```args`'' != "" {
if !`:list name in ARGOPTNAMES' {
di as err "invalid vce() option"
exit 198
}
}
if "`name'" == "hc2" {
confirm var `args'
unab args : `args'
local markout `args'
}
return local vceargs ```:list retok args`''
}
else if `:list name in ARGOPTNAMES' {
if "`noargopt_is_ok'" == "" {
di as err "invalid vce() option"
exit 198
}
}
}
}
- else {

```

```

- if "`weight'" == "pweight" | "`robust'" != "" {
= if "" == "pweight" | "" != "" {
    local vce robust
  }
- if "`cluster'" != "" {
= if "" != "" {
    local vce cluster
    local args `cluster'
    local markout `cluster'
  }
- return local vce ``vce'""
= return local vce `""'
- return local vceargs ``args'""
= return local vceargs `""'
- }
- if "`return(vce)'" != "" {
= if "" != "" {
    if ``return(vceargs)'" != "" {
      local ss " "
    }
    return local vceopt `"vce(`return(vce)'"ss'"return(vceargs)')'"
  }
- if "`return(vce)'" == "cluster" {
= if "" == "cluster" {
    return local cluster `return(vceargs)'
  }
- if inlist("`return(vce)'", "robust", "cluster") {
= if inlist("", "robust", "cluster") {
    return local robust robust
  }
- if "`touse'" != "" {
= if "" != "" {
    if "`markout'" != "" {
      markout `touse' `markout', strok
    }
  }
}

> _____ end _vce_pa
> rse _____
- if "`r(vce)'" != "" {
= if "" != "" {
    local vce1 `r(vce)' `r(vceargs)'
  }
- if ``vce'`vce1'"" != "" {
= if `"robust'" != "" {
- if ``vce1'"" != "" {

```

```

= if `"" != "" {
  if !inlist("oim",`""`vce'`,`""`vce1'`) & !inlist("eim",`""`vce'`,`""`vc
> e1'`) {
    opts_exclusive "vce(`vce') vce(`vce1')"
  }
  else if !inlist("oim",`""`vce'`,`""`vce1'`) {
    local eim eim
  }
  else if !inlist("eim",`""`vce'`,`""`vce1'`) {
    local oim oim
  }
  else {
    opts_exclusive "vce(`vce') vce(`vce1')"
  }
  local vce `vce' `vce1'
  local vce : list vce - oim
  local vce : list vce - eim
  local vce1
}
- if `""`vce' == "oim" {
= if `"robust" == "oim" {
  local oim oim
}
- if `""`vce' == "eim" {
= if `"robust" == "eim" {
  local eim eim
}
- if !("`vce"=="robust" & ""`oldvce'=="unbiased") & !("`vce"=="cluster
> " & ""`oldvce'=="unbiased") & !inlist("`oldvce'", "", "`vce'") & !inlist("`v
> ce'", "", "oim", "eim") {
  = if !("robust"=="robust" & ""=="unbiased") & !("robust"=="cluster" & ""
> == "unbiased") & !inlist("", "", "robust") & !inlist("robust", "", "oim", "ei
> m") {
    opts_exclusive ``vce(`vce') `oldvce'
  }
  - if !inlist("`vce'", "", "cluster") {
  = if !inlist("robust", "", "cluster") {
  - MapVCE "" irls `vce'
  = MapVCE "" robust
}

> ----- begin glm.Map
> VCE -----
- gettoken irls 0 : 0
- local 0 : list retok 0
- gettoken vce vcerest: 0, parse(", ")
- local vcerest : list retok vcerest

```

```

- local len : length local vce
- if inlist("`"vce"'', "jknife1", "jackknife1") {
= if inlist("`robust'", "jknife1", "jackknife1") {
  local norest norest
  local vce jknife1
  local vcespec jknife1
}
- else if `"vce"' == "hac" {
= else if "`robust'" == "hac" {
  if `"vcerest"' == "" {
    di as err "option vce(hac) misspecified"
    exit 198
  }
  local vcespec `"vcerest"'
  local vcerest
}
- else if `"vce"' == bsubstr("robust",1,max(1,`len')) {
= else if "`robust'" == bsubstr("robust",1,max(1,6)) {
- local norest norest
- local vce robust
- local vcespec robust
- }
- else if `"vce"' == bsubstr("unbiased",1,max(3,`len')) {
= else if "`robust'" == bsubstr("unbiased",1,max(3,6)) {
  local norest norest
  local vce unbiased
  local vcespec unbiased
}
- else if `"vce"' == "eim" {
= else if "`robust'" == "eim" {
  local norest norest
  local vce eim
  local vcespec eim
}
- else {
  if `"vce"' == "oim" {
    local norest norest
    local vcespec oim
  }
  else if `"vce"' == "opg" {
    local norest norest
    local vcespec opg
  }
  else if `"0"' != "" {
    if "`irls'" != "" {
      di as err "option vce(`0') is not allowed with irls"
    }
  }
}

```

```

        exit 198
    }
    else {
        local vceopt ``vce(`0')''
    }
}

- if ``norest'' != '' & ``vcerest'' != '' {
= if "norest" != "" & `"' != "" {
    di as err "option vce(`0') invalid"
    exit 198
}

- sreturn local vce ``vce''
= sreturn local vce ``robust''
- sreturn local vcespec ``vcespec''
= sreturn local vcespec ``robust''
- sreturn local vceopt ``vceopt''
= sreturn local vceopt `"'



---


> _____ end glm.Map
> VCE _____
- if ``s(vce)'' != '' {
= if ``robust'' != '' {
- local vce `s(vce)'
= local vce robust
- local `vce' `s(vcespec)'
= local robust robust
- }
- if ``s(vceopt)'' != '' {
= if `"' != "" {
    local options ``options' `s(vceopt)''
}
- }
- }
- else {
    if ``nwest'' != '' {
        local hac ``nwest''
    }
}

- c_local bstrap ``bstrap''
= c_local bstrap `"'
- c_local brep ``brep''
= c_local brep ``-1''
- c_local jknife ``jknife''
= c_local jknife `"'
- c_local jknife1 ``jknife1''

```

```

= c_local jknife1 `""'
- c_local hac `""hac""'
= c_local hac `""'
- c_local oim `""oim""'
= c_local oim `""'
- c_local opg `""opg""'
= c_local opg `""'
- c_local unbiased `""unbiased""'
= c_local unbiased `""'
- if "`robust'" != "" {
= if "robust" != "" {
- c_local robust `""robust""'
= c_local robust `robust'
- }
- c_local vce `""vce""'
= c_local vce `robust'
- c_local cluster `""cluster""'
= c_local cluster `""'
- c_local options `""options""'
= c_local options `""'

> _____ end glm.Get
> VCE _____
- if "`family'" != "" {
= if "nbinomial" != "" {
- local argfam `family(`family')'
= local argfam `family(nbinomial)'
- }
- if "`link'" != "" {
= if "log" != "" {
- local arglink `link(`link')'
= local arglink `link(log)'
- }
- if "`init'" != "" & "`mu'" != "" & "`init'" != "`mu'" {
= if "" != "" & "" != "" & "" != "" {
noi di as err "mu() and init() should use the same varname"
exit 198
}
- if "`init'" != "" {
= if "" != "" {
local init "`init'"
}
- else local init "`mu'"
= else local init ""
- if "`dots'" == "nodots" {
= if "" == "nodots" {

```



```

        local dots ""
    }
    - else local dots "dots"
    - if ""`unbiased'"" != "" {
= if "" != "" {
        local robust "robust"
    }
    - if `vfactor' <= 0.0 {
= if 1 <= 0.0 {
        di as err "vfactor() must be positive"
        exit 198
    }
    - if `iterate' < 0 {
= if 300 < 0 {
        di as err "iterate() must be positive"
        exit 198
    }
    - _get_diopts diopts options, `options'
    = _get_diopts diopts options,

```

```

> ----- begin _get_dio
> pts -----
    - version 11
    - syntax namelist(max=2) [, *]
    - gettoken name1 namelist : namelist
    - gettoken name2 namelist : namelist
    - GetDiopts diopts options, `options'
    = GetDiopts diopts options,

```

```

> ----- begin _get_diopts.GetDio
> pts -----
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels NO
> BASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NOOM
> ITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sforma
> t(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(st
> ring) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV AB
> BREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels ALL
> BASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EM
> PTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(s
> tring) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru
> ) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru
> ) NOABBREV ABBREV *]
    - opts_exclusive ""`allbaselevels' `noallbaselevels'""
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2

```

```

- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive
- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0

```

```

- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
  - opts_exclusive "`omitted' `noomitted'"
  = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ``opts'' != "" {
  = while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
  - exit

```

```

> ----- end opts_exclus
> ive -----
  - opts_exclusive "`lstretch' `nolstretch'"
  = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ``opts'' != "" {
  = while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind

```

```

    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

```

```

> _____ end opts_exclus
> ive _____
  - opts_exclusive "`nocl' `nopvalues'"
  = opts_exclusive " "
  _____
> _____ begin opts_exclus
> ive _____
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ```opts'`` != "" {
  = while `""' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
  - exit
  _____
> _____ end opts_exclus
> ive _____
  - local K : list sizeof namelist
  - gettoken c_diopts c_opts : namelist
  - opts_exclusive "`coding' `compare'"
  = opts_exclusive " "
  _____
> _____ begin opts_exclus
> ive _____
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ```opts'`` != "" {
  = while `""' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
  - exit
  _____
> _____ end opts_exclus

```

```

> ive
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "
-----
> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `"'`opts'" != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
-----
> ----- end opts_exclus
> ive
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "
-----
> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `"'`opts'" != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
-----
> ----- end opts_exclus
> ive
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'

```

```

= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
  }

```



```

        local cformat `"cformat(`cformat')"'
    }
    - if `:length local sformat' {
= if 0 {
        confirm numeric format `sformat'
        if fmtwidth(`"'sformat'"') > 8 {
            local sformat "%8.2f"
            di "{txt}note: invalid sformat(), using default"
        }
        sreturn local sformat `"'sformat'"'
        local sformat `"sformat(`sformat')"'
    }
    - if `:length local pformat' {
= if 0 {
        confirm numeric format `pformat'
        if fmtwidth(`"'pformat'"') > 5 {
            local pformat "%5.3f"
            di "{txt}note: invalid pformat(), using default"
        }
        sreturn local pformat `"'pformat'"'
        local pformat `"pformat(`pformat')"'
    }
    - if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
        syntax namelist(max=2) [, `DIOPTS']
    }
    - if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
        local levelopt level(`level')
    }
    - Wraupon , `fvwraupon'
= Wraupon ,

> _____ begin _get_diopts.Wra
> pon _____
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwraupon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive "'word' `width'" fvwraupon
= opts_exclusive " " fvwraupon

> _____ begin opts_exclus
> ive _____

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `""`opts'"" != "" {
= while `""' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }

> _____ end _get_diopts.Wra
> pon _____
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevels'
> `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend'
> `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwrap
> ' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts
    - if `K' == 2 {
    = if 2 == 2 {
    - c_local `c_opts' `""`options'""
    = c_local options `""'
    - }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'

```

```

= sreturn local level 95


---


> 

---

 end _get_diopts.GetDio
> pts 

---


- c_local `name1' ``diopts'
= c_local diopts `''
- local snames : s(macros)
- foreach sname of local snames {
- local S_`sname' = s(`sname')
= local S_level = s(level)
- }
- local S_`sname' = s(`sname')
= local S_vcespec = s(vcespec)
- }
- local S_`sname' = s(`sname')
= local S_vce = s(vce)
- }
- local S_`sname' = s(`sname')
= local S_namelist = s(namelist)
- }
- local S_`sname' = s(`sname')
= local S_after = s(after)
- }
- local S_`sname' = s(`sname')
= local S_before = s(before)
- }
- local S_`sname' = s(`sname')
= local S_optlist = s(optlist)
- }
- local S_`sname' = s(`sname')
= local S_fvops = s(fvops)
- }
- sreturn clear
- GetDiopts DEFAULT


---


> 

---

 begin _get_diopts.GetDio
> pts 

---


- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels NO
> BASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NOOM
> ITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sforma
> t(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(st
> ring) NOFVLABEL FVLABEL NOCI NOPvalues MARKDOWN CITYPE(passthru) NOABBREV AB
> BREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels ALL
> BASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EM

```

```

> PTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(s
> tring) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru
> ) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru
> ) NOABBREV ABBREV *)
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit


---


> 

---

 end opts_exclus
> ive 

---


    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit


---


> 

---

 end opts_exclus
> ive 

---



```

```

- opts_exclusive "`baselevels' `nobaselevels'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit


---


> 

---

 end opts_exclus
> ive 

---


- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit


---


> 

---

 end opts_exclus
> ive 

---


- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus

```

```

> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``cnsreport' `fullcnsreport'
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
        }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``coeflegend' `selegend'
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }

```



```

    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

```

```

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {

```

```

confirm numeric format `cformat'
if fmtwidth("`cformat'") > 9 {
local cformat "%9.0g"
di "{txt}note: invalid cformat(), using default"
}
sreturn local cformat "`cformat'"
local cformat `cformat(`cformat')"'
}
- if `:length local sformat' {
= if 0 {
confirm numeric format `sformat'
if fmtwidth("`sformat'") > 8 {
local sformat "%8.2f"
di "{txt}note: invalid sformat(), using default"
}
sreturn local sformat "`sformat'"
local sformat `sformat(`sformat')"'
}
- if `:length local pformat' {
= if 0 {
confirm numeric format `pformat'
if fmtwidth("`pformat'") > 5 {
local pformat "%5.3f"
di "{txt}note: invalid pformat(), using default"
}
sreturn local pformat "`pformat'"
local pformat `pformat(`pformat')"'
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
local levelopt level(`level')
}
- Wraon , `fvwraon'
= Wraon ,

> _____ begin _get_diopts.Wra
> pon _____
- capture syntax [, W0rd WIdth]
- if c(rc) {
di as err "invalid fvwrapon() option;"
syntax [, W0rd WIdth]
exit 198

```

```

    }
    - opts_exclusive "`word' `width'" fwrapon
    = opts_exclusive " " fwrapon
}
> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive
    - local fwrapon `word' `width'
    = local fwrapon
    - if `:length local fwrapon' {
    = if 0 {
        c_local fwrapon fwrapon(`fwrapon')
    }
}
> ----- end _get_diopts.Wra
> pon
    - if "`fwrap'" != "" {
    = if "" != "" {
        FVWrap, `fwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevels'
> `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend'
> `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fwrap
> ' `fwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DEFAULT
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options'
    }
    - sreturn local coding `coding'
    = sreturn local coding

```

```

- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> ----- end _get_diopts.GetDio
> pts -----
- if ``name2''' == "" {
= if `"options"' == "" {
  GetDiopts DUPS, `options'
}
- else {
- GetDiopts DUPS options, `options'
= GetDiopts DUPS options,

```

```

> ----- begin _get_diopts.GetDio
> pts -----
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels NO
> BASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NOOM
> ITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sforma
> t(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(st
> ring) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV AB
> BREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels ALL
> BASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EM
> PTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(s
> tring) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru
> ) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru
> ) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts''' != "" {
= while `''' != "" {
  local ++n
  gettoken item`n' opts : opts, bind

```

```

    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nobaselevels'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

```

```

> _____ end opts_exclus
> ive _____
  - opts_exclusive "`emptycells' `noemptycells'"
  = opts_exclusive " "
  _____

> _____ begin opts_exclus
> ive _____
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ``opts' != "" {
  = while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
  - exit
  _____

> _____ end opts_exclus
> ive _____
  - opts_exclusive "`fvlabel' `nofvlabel'"
  = opts_exclusive " "
  _____

> _____ begin opts_exclus
> ive _____
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ``opts' != "" {
  = while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
  - exit
  _____

> _____ end opts_exclus
> ive _____
  - opts_exclusive "`omitted' `noomitted'"

```

```

= opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----

```



```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev` `noabbrev`"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels` `noallbaselevels`
= local allbaselevels
    - local baselevels `baselevels` `nobaselevels`
= local baselevels
    - local emptycells `emptycells` `noemptycells`
= local emptycells
    - local omitted `omitted` `noomitted`
= local omitted
    - local lstretch `lstretch` `nolstretch`
= local lstretch
    - local fvlabel `fvlabel` `nofvlabel`
= local fvlabel
    - local abbrev `abbrev` `noabbrev`
= local abbrev
    - if "`allbaselevels` `baselevels`" == "" {

```

```

= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `"cformat(`cformat')"'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `"sformat(`sformat')"'
  }
}

```

```

- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth(`"pformat"') > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat `"pformat"'
    local pformat `pformat(pformat)'"'
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraupon , `fvwraupon'
= Wraupon ,

> ----- begin _get_diopts.Wra
> pon -----
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwrapon
    = opts_exclusive " " fvwrapon

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `"opts"' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {

```

```

        - exit
    }
}
> ----- end opts_exclus
> ive
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
}
> ----- end _get_diopts.Wra
> pon
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevels'
> `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend'
> `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwrap
> ' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DUPS
    - if `K' == 2 {
    = if 2 == 2 {
        - c_local `c_opts' ``options''
        = c_local options `''
        - }
        - sreturn local coding `coding'
        = sreturn local coding
        - sreturn local versus `versus'
        = sreturn local versus
        - sreturn local compare `compare'
        = sreturn local compare
        - sreturn local level `level'
        = sreturn local level 95
    }
}
> ----- end _get_diopts.GetDio
> pts
    - c_local `name2' ``options''
    = c_local options `''
    - }
    - local DUPS : list DUPS - DEFAULT
    - gettoken DUPS : DUPS
    - if ``DUPS'' != "" {
    = if `'' != "" {
        if (strpos(`DUPS', "(")) {

```

```

    gettoken DUPS : DUPS, parse("(")
    local DUPS ``DUPS'()'
  }
  di as err "duplicate {bf:`DUPS'} option not allowed"
  exit 198
}
- sreturn clear
- foreach sname of local snames {
- sreturn local `sname' = ``S_`sname''
= sreturn local level = ``95''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local vcespec = ``robust''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local vce = ``robust''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local namelist = ``cluster hac''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local after = `", vce()'
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local before = `", opt(EIM OIM OPG Robust jackknife1 jknife1 U
> NBIased NATIVE) argopt(CLuster HAC) old ''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local optlist = ``CLuster HAC''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local fvops = ``true''
- }

_____ end _get_dio
> pts _____
- mlopts mlopts, `options'
= mlopts mlopts,
_____

> _____ begin mlo
> pts _____
- version 8.0
- local maxoptlist MLOPTs(string) TRace TRACEDiffs GRADient HESSian SHOW
> STEP ITERate(int -1) TOLerance(real -1) LTOLerance(real -1) GTOLerance(real
> -1) NONRTOLerance SHOWNRTolerance SHOWTOLerance NRTOLerance(real -1) QTOLera
> nce(real -1) DIFFicult HALFSTEPonly CONSTraints(string) noCNSNOTes COLlinear

```

```

> TECHnique(passthru) VCE(passthru) DOTS depsilon(passthru) SHOWH NOTCONCAVE(
> passthru) USEVIEWS NOLOg LOg nrderiv
    - syntax namelist(id="macro names" min=1 max=2) [, `maxoptlist' * ]
    = syntax namelist(id="macro names" min=1 max=2) [, MLOPTs(string) TRace
> TRACEDIffs GRADient HESSian SHOWSTEP ITERate(int -1) TOLerance(real -1) LTOL
> erance(real -1) GTOLerance(real -1) NONRTOlerance SHOWNRtolerance SHOWTOLera
> nce NRTOLerance(real -1) QTOLerance(real -1) DIFFicult HALFSTEPonly CONSTrai
> nts(string) noCNSNOTEs COLLinear TECHnique(passthru) VCE(passthru) DOTS deps
> ilon(passthru) SHOWH NOTCONCAVE(passthru) USEVIEWS NOLOg LOg nrderiv * ]
    - gettoken dest rest : namelist
    - if "`rest'" == "" {
    = if "" == "" {
    - syntax namelist [, `maxoptlist' ]
    = syntax namelist [, MLOPTs(string) TRace TRACEDIffs GRADient HESSian SH
> OWSTEP ITERate(int -1) TOLerance(real -1) LTOLerance(real -1) GTOLerance(rea
> l -1) NONRTOlerance SHOWNRtolerance SHOWTOLerance NRTOLerance(real -1) QTOLe
> rance(real -1) DIFFicult HALFSTEPonly CONSTraints(string) noCNSNOTEs COLLine
> ar TECHnique(passthru) VCE(passthru) DOTS depsilon(passthru) SHOWH NOTCONCAV
> E(passthru) USEVIEWS NOLOg LOg nrderiv ]
    - }
    - local mlopts `mlopts' `trace' `tracediffs' `gradient' `hessian' `shows
> tep' `difficult' `halfsteponly' `nonrtolerance' `shownrtolerance' `showtoler
> ance' `dots' `depsilon' `showh' `notconcave' `useviews' `log' `nolog' `nrder
> iv'
    = local mlopts
    - if `tolerance' != -1 {
    = if -1 != -1 {
        local tol "tolerance(`tolerance')"
    }
    - if `ltolerance' != -1 {
    = if -1 != -1 {
        local ltol "ltolerance(`ltolerance')"
    }
    - if `gtolerance' != -1 {
    = if -1 != -1 {
        local gtol "gtolerance(`gtolerance')"
    }
    - if `nrtolerance' != -1 {
    = if -1 != -1 {
        local nrtol "nrtolerance(`nrtolerance')"
    }
    - if `qtolerance' != -1 {
    = if -1 != -1 {
        local qtol "qtolerance(`qtolerance')"
    }
    - if `iterate' != -1 {

```



```

= if -1 != -1 {
    local iter "iterate(`iterate')"
}
- sreturn clear
- if "`technique' `vce'" != "" {
= if "" != "" {
    ml_technique, `technique' `vce' nodefault
    if ``s(technique)'' != "" {
        local technique "technique(`s(technique)')"
    }
    if ``s(vce)'' != "" {
        local vce ``s(vceopt)''
    }
    sreturn local technique ``technique''
    sreturn local vce ``vce''
}
- if "`constraints'" != "" {
= if "" != "" {
    capture confirm name `constraints'
    if _rc {
        CheckClist constraints , constraints(`constraints')
    }
    else {
        CheckCmat `constraints'
    }
    local const "constraints(`constraints')"
    sreturn local constraints ``constraints''
}
- sreturn local collinear ``collinear''
= sreturn local collinear ""
- c_local `dest' `mlopts' `tol' `ltol' `gtol' `nrtol' `iter' `const' `cn
> snotes' `collinear' `technique' `vce' `qtol'
    = c_local mlopts
    - if ``rest'' != "" {
= if "" != "" {
    c_local `rest' ``options''
}



---


> _____ end mlo
> pts _____
    - local cns `s(constraints)'
    = local cns
    - local coll `s(collinear)'
    = local coll
    - if ``coll'' != "" & ``irls'' != "" {
    = if ``'' != "" & "" != "" {

```

```

    version 8: di as err "option {bf:collinear} not allowed"
    exit 198
  }
- if `ltolerance' == -1 {
= if -1 == -1 {
- local ltol 1e-6
- }
- else {
  local ltol `ltolerance'
  if ""irls"" == "" {
    local mlopts ``mlopts' ltol(`ltol)''
  }
}
- if `ltol'>=1 | `ltol'<0 {
= if 1e-6>=1 | 1e-6<0 {
  di as err "ltolerance() must be in [0,1)"
  exit 198
}
- if ""ml""!= "" & ""irls""!= "" {
= if ""!= "" & ""!= "" {
  di as err "ml and irls options may not be combined"
  exit 198
}
- if ""irls"" != "" {
= if "" != "" {
  if ``from"" != "" {
    di as err "option {bf:from()} not allowed " "in combination with option
> {bf:irls}"
    exit 198
  }
  if ""search"" != "" {
    di as err "option {bf:search} not allowed " "in combination with option
> {bf:irls}"
    exit 198
  }
}
- if ``score""!= "" {
= if ``""!= "" {
  if ""irls"" != "" {
    di as err "deviance scores are calculated " "postestimation via predict,
> score"
    exit 198
  }
  confirm new variable `score'
  local nword : word count `score'
  if `nword' > 1 {

```

```

di as err "score() must contain the name of only " "one new variable"
exit 198
}
tempvar scvar
local scopt score(`scvar')
}
- if ``cluster'""!="" {
= if `""'!="" {
  if "`jknife'`bstrap' == "" {
    local robust "robust"
  }
  local clopt cluster(`cluster')
}
- if "`bstrap' == "" {
= if "" == "" {
- if !inlist("`brep'", "", "-1") {
= if !inlist("-1", "", "-1") {
  di as err "brep() only valid with bstrap"
  exit 198
}
- else local brep -1
- }
- if "`bstrap' != "" {
= if "" != "" {
  if `brep' == -1 {
    local brep = 199
  }
  if `brep' <= 40 {
    di as err "bootstrap replications must be > 40"
    exit 198
  }
}
- local tsops = index("`varlist'", ".")
= local tsops = index("piq_timing_max rule_g rule_itr estimated i.wlth#i.o
> pen cb_authors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit vint_earl
> y vint_mid", ".")
- if `tsops' {
= if 43 {
- local PROB `bstrap' `jknife' `jknife1'
= local PROB
- if `:length local PROB' {
= if 0 {
  gettoken PROB : PROB
  if ``PROB'"" == "jknife1" {
    local PROB "vce(jackknife1)"
  }
}

```

```

di as error "{p}"
di as error "cannot use factor variables or "
di as error "time-series operators with option `PROB'"
di as error "{p_end}"
exit 101
}
- }
- if `fisher' != -1 {
= if -1 != -1 {
  if ""irls"" != "" {
    di as err "fisher() only valid with Newton-Raphson optimization"
    exit 198
  }
  if `fisher' <= 0 {
    di as err "fisher() must be positive"
    exit 198
  }
}
- if `disp' < 0 {
= if 1 < 0 {
  di as err "disp() must be positive"
  exit 198
}
- if ``scale'"" != "" {
= if `""' != "" {
  if ""robust""opg""hac""jknife1""jknife""bstrap"" != "" {
    di as err "cannot use scale() with alternate variances"
    exit 198
  }
  if ``scale'""=="x2" {
    local scale 0
  }
  else if ``scale'""=="dev" {
    local scale -1
  }
  else {
    capture confirm number `scale'
    if _rc {
      di as err "invalid scale()"
      exit 198
    }
    if `scale' <= 0 {
      di as err "scale(#) must be positive"
      exit 198
    }
  }
}
}

```

```

    }
    - if "`scale'" == "-1" & "`irls'" == "" {
= if "" == "-1" & "" == "" {
    di as err "option {bf:scale(dev)} allowed only with option {bf:irls}"
    exit 198
    }
    - if `disp' != 1 & "`irls'" == "" {
= if 1 != 1 & "" == "" {
    di as err "disp() allowed only with irls option"
    exit 198
    }
    - MapFL ``family'"" ``link'""
= MapFL ``nbinomial'"" ``log'""
}

> _____ begin glm.Ma
> pFL _____
    - args f ulink
    - MapFam `f'
    = MapFam nbinomial
}

> _____ begin glm.Map
> Fam _____
    - args f
    - local s1
    - local f = lower(trim(``f'""))
    = local f = lower(trim(`"nbinomial"'))
    - local l = length(``f'"" )
    = local l = length(`"nbinomial"')
    - if ``f'""=="" { local s1 "1" }
    = if `"nbinomial"'=="" { local s1 "1" }
    - else if ``f'""==bsubstr("gaussian",1,max(`l',3)) { local s1 "1" }
    = else if `"nbinomial"'==bsubstr("gaussian",1,max(9,3)) { local s1 "1" }
> }

    - else if ``f'""==bsubstr("normal",1,`l') { local s1 "1" }
    = else if `"nbinomial"'==bsubstr("normal",1,9) { local s1 "1" }
    - else if ``f'""==bsubstr("binomial",1,`l') { local s1 "2" }
    = else if `"nbinomial"'==bsubstr("binomial",1,9) { local s1 "2" }
    - else if ``f'""==bsubstr("bernoulli",1,`l') { local s1 "2" }
    = else if `"nbinomial"'==bsubstr("bernoulli",1,9) { local s1 "2" }
    - else if ``f'""==bsubstr("poisson",1,`l') { local s1 "3" }
    = else if `"nbinomial"'==bsubstr("poisson",1,9) { local s1 "3" }
    - else if ``f'""==bsubstr("gamma",1,max(`l',3)) { local s1 "4" }
    = else if `"nbinomial"'==bsubstr("gamma",1,max(9,3)) { local s1 "4" }
    - else if ``f'""==bsubstr("igaussian",1,max(`l',2)) { local s1 "5" }
    = else if `"nbinomial"'==bsubstr("igaussian",1,max(9,2)) { local s1 "5" }
> " }

```

```

- else if ``f''==bsubstr("inormal",1,max(`l',2)) { local s1 "5" }
= else if ``nbinomial''==bsubstr("inormal",1,max(9,2)) { local s1 "5"
> }

- else if ``f''=="ivg" { local s1 "5" }
= else if ``nbinomial''=="ivg" { local s1 "5" }
- else if ``f''==bsubstr("nbinomial",1,max(2,`l')) { local s1 "6" }
= else if ``nbinomial''==bsubstr("nbinomial",1,max(2,9)) { local s1 "6"
> " }

- if "`s1'" != "" {
= if "6" != "" {
- local s1 "glim_v`s1"
= local s1 "glim_v6"
- }
- else {
local s1 "f"
}
- ret local famcode `s1'
= ret local famcode glim_v6



---


> ----- end glm.Map
> Fam -----
- local fam ``r(famcode)''
= local fam "glim_v6"
- local mfixed 1
- local m 1
- local k 1
- if ``fam'=="glim_v2" {
= if "glim_v6'=="glim_v2" {
tokenize ``f''
if ``2'!=' {
capture confirm number `2'
if _rc {
unabbrev `2'
local m ``s(varlist)''
local mfixed 0
}
else {
if `2'>=. | `2'<1 {
di as err ``2' in family(binomial `2') invalid"
exit 198
}
local m `2'
}
}
if ``3'!=' {
di as err "family(`f') invalid"

```

```

        exit 198
    }
}
- if ``fam''=="glim_v6" {
= if "glim_v6"=="glim_v6" {
- tokenize ``f''
= tokenize "nbinomial"
- if ``2'' != "" {
= if "" != "" {
    if ``2'' != "ml" {
        confirm number `2'
        if `2'<=0 {
            di as err ``2' in family(nbinomial `2') invalid"
            exit 198
        }
    }
    local k `2'
}
- if ``3''!="" {
= if ""!="" {
    di as err "family(`f') invalid"
    exit 198
}
- }
- if ``fam''~="glim_v1" & ``fam''~="glim_v2" & ``fam''~="glim_v3"
> & ``fam''~="glim_v4" & ``fam''~="glim_v5" & ``fam''~="glim_v6" {
    = if "glim_v6"~="glim_v1" & "glim_v6"~="glim_v2" & "glim_v6"~="gli
> m_v3" & "glim_v6"~="glim_v4" & "glim_v6"~="glim_v5" & "glim_v6"~="glim
> _v6" {
        tokenize ``f''
        if ``2'' != "" {
            global SGLM_fa `2'
        }
        if ``3''!="" {
            di as err "family(`f') invalid"
            exit 198
        }
    }
- MapLink `ulink'
= MapLink log



---


> ----- begin glm.MapL
> ink -----
    - args ulink upow rest
    - if ``rest''!="" {
    = if ""!="" {

```

```

di as err "link(`ulink' `upow' `rest') invalid"
exit 198
}
- local ulink = lower("`ulink'")
= local ulink = lower("`log'")
- local l = length("`ulink'")
= local l = length("`log'")
- local s1 "11"
- local s2 .
- if "`ulink'"==" { local s1 "" }
= if "`log'"==" { local s1 "" }
- else if "`ulink'"==bsubstr("identity",1,`l') { local s1 "01"
= else if "`log'"==bsubstr("identity",1,3) { local s1 "01"
    local s2 1 }
- else if "`ulink'"==bsubstr("reciprocal",1,`l') { local s1 "09"
= else if "`log'"==bsubstr("reciprocal",1,3) { local s1 "09"
    local s2 -1 }
- else if "`ulink'"=="log" { local s1 "03"
= else if "`log'"=="log" { local s1 "03"
- local s2 0 }
- else if "`ulink'"==bsubstr("power",1,max(`l',3)) {
= else if "`log'"==bsubstr("power",1,max(3,3)) {
    capture confirm number `upow'
    if _rc {
di as err "invalid # in link(power #)"
exit 198
}
    if `upow' == 0 {
local s1 "03"
}
    if `upow' == -1 {
local s1 "09"
}
    else if `upow' == -2 {
local s1 "10"
}
    local s2 `upow'
    local upow
}
- else if "`ulink'"==bsubstr("opower",1,max(`l',3)) {
= else if "`log'"==bsubstr("opower",1,max(3,3)) {
    capture confirm number `upow'
    if _rc {
di as err "invalid # in link(opower #)"
exit 198
}
}

```



```

        if `upow'==0 {
            local s1 "02"
        }
        else {
            local s1 "12"
        }
        local s2 `upow'
        local upow
    }
- else if ```ulink'``==bsubstr("logit",1,`l') { local s1 "02"
= else if ``"log"``==bsubstr("logit",1,3) { local s1 "02"
    local s2 0 }
- else if ```ulink'``==bsubstr("nbinomial",1,max(`l',2)) { local s1 "0
> 4" }
= else if ``"log"``==bsubstr("nbinomial",1,max(3,2)) { local s1 "04" }
- else if ```ulink'``==bsubstr("logc",1,4) { local s1 "05" }
= else if ``"log"``==bsubstr("logc",1,4) { local s1 "05" }
- else if ```ulink'``==bsubstr("loglog",1,max(`l',4)) { local s1 "06"
> }
= else if ``"log"``==bsubstr("loglog",1,max(3,4)) { local s1 "06" }
- else if ```ulink'``==bsubstr("cloglog",1,`l') { local s1 "07" }
= else if ``"log"``==bsubstr("cloglog",1,3) { local s1 "07" }
- else if ```ulink'``==bsubstr("probit",1,`l') { local s1 "08" }
= else if ``"log"``==bsubstr("probit",1,3) { local s1 "08" }
- else {
    local s1
    local s2 ``upow'
    }
- if ```s1'`` != "" {
= if ``"03"`` != "" {
- local s1 "glm_l`s1'
= local s1 "glm_l03"
- }
- else {
    local s1 ``ulink'
    }
- ret local link `s1'
= ret local link glm_l03
- ret local power `s2'
= ret local power 0



---


> ----- end glm.MapL
> ink -----
- local link ```r(link)``
= local link ``glm_l03``
- local pow ```r(power)``

```

```

    = local pow ``0''
    - local scale1 = ``fam''=="glim_v2" | ``fam''=="glim_v3" | ``fam''
> == "glim_v6"
    = local scale1 = "glim_v6"=="glim_v2" | "glim_v6"=="glim_v3" | "gli
> m_v6"=="glim_v6"
    - if ``link''="" {
    = if "glim_l03"="" {
      local link "glim_l11"
      if ``fam''=="glim_v1" { local link "glim_l01"
      local pow 1 }
      else if ``fam''=="glim_v2" { local link "glim_l02"
      local pow 0 }
      else if ``fam''=="glim_v3" { local link "glim_l03"
      local pow 0 }
      else if ``fam''=="glim_v4" { local link "glim_l09"
      local pow -1 }
      else if ``fam''=="glim_v5" { local link "glim_l10"
      local pow -2 }
      else if ``fam''=="glim_v6" { local link "glim_l03"
      local pow 0 }
      else local link
    }
    - ret local family ``fam''
    = ret local family "glim_v6"
    - ret local link ``link''
    = ret local link "glim_l03"
    - ret local power `pow'
    = ret local power 0
    - ret local scale `scale1'
    = ret local scale 1
    - ret local m `m'
    = ret local m 1
    - ret local mfixed `mfixed'
    = ret local mfixed 1
    - ret local k `k'
    = ret local k 1

```

```

> ----- end glm.Ma
> pFL -----
    - local family ``r(family)''
    = local family "glim_v6"
    - local link ``r(link)''
    = local link "glim_l03"
    - local pow `r(power)'
    = local pow 0
    - local scale1 `r(scale)'

```

```

= local scale1 1
- local m `r(m)'
= local m 1
- local mfixed `r(mfixed)'
= local mfixed 1
- local k `r(k)'
= local k 1
- tempname canon
- scalar `canon' = 0
= scalar __000001 = 0
- if ("`family'" == "glim_v1" & "`link'" == "glim_l01") {
= if ("glim_v6" == "glim_v1" & "glim_l03" == "glim_l01") {
  scalar `canon' = 1
}
- else if ("`family'" == "glim_v5" & ("`link'" == "glim_l10")) {
= else if ("glim_v6" == "glim_v5" & ("glim_l03" == "glim_l10")) {
  scalar `canon' = 1
}
- else if ("`family'" == "glim_v3" & "`link'" == "glim_l03") {
= else if ("glim_v6" == "glim_v3" & "glim_l03" == "glim_l03") {
  scalar `canon' = 1
}
- else if ("`family'" == "glim_v6" & "`link'" == "glim_l04") {
= else if ("glim_v6" == "glim_v6" & "glim_l03" == "glim_l04") {
  scalar `canon' = 1
}
- else if ("`family'" == "glim_v2" & "`link'" == "glim_l02") {
= else if ("glim_v6" == "glim_v2" & "glim_l03" == "glim_l02") {
  scalar `canon' = 1
}
- else if ("`family'" == "glim_v4" & "`link'" == "glim_l09") {
= else if ("glim_v6" == "glim_v4" & "glim_l03" == "glim_l09") {
  scalar `canon' = 1
}
- if "`link'" == "" | "`family'" == "" {
= if "glim_l03" == "" | "glim_v6" == "" {
  di as err "incomplete specification of family() and link()"
  exit 198
}
- if "`offset'" != "" & "`lnoffset'" != "" {
= if "" != "" & "" != "" {
  di as err "only one of offset() or lnoffset() can be specified"
  exit 198
}
- if "`offset'" != "" & "`exposure'" != "" {
= if "" != "" & "" != "" {

```

```

di as err "only one of offset() or exposure() can be specified"
exit 198
}
- if "`lnoffset'" != "" & "`exposure'" != "" {
= if "" != "" & "" != "" {
di as err "only one of lnoffset() or exposure() can be specified"
exit 198
}
- if "`exposure'" != "" {
= if "" != "" {
local lnoffset `exposure'
local exposure
}
- if "`constant'" != "" {
= if "" != "" {
local nvar : word count `varlist'
if `nvar' == 1 {
di as err "independent variables required with noconstant option"
exit 100
}
}
- marksample touse
- if "`cluster'" != "" {
= if "" != "" {
markout `touse' `cluster', strok
}
- if "`m'" != "" {
= if "1" != "" {
- capture confirm number `m'
= capture confirm number 1
- if _rc { markout `touse' `m' }
= if _rc { markout __000002 1 }
- }
- if ``"hac""' == "" {
= if `""' == "" {
- if ``"t""' ~="" {
= if `""' ~="" {
di as err "t() only valid with vce(hac ...)"
exit 198
}
- }
- if "`hac'" != "" {
= if "" != "" {
capture xt_tis `t'
if _rc {
tsset `t', noquery

```

```

    }
    local tvar ``s(timevar)''
    markout `touse' `tvar'
    sort `touse' `tvar'
    cap assert `tvar'[_n-1] != `tvar' if `touse' & (`touse'[_n-1]==1)
    if _rc {
        di as err "repeated time values in sample"
        exit 451
    }
    sort `tvar'
}
- if "`lnoffset'" != "" {
= if "" != "" {
    capture assert `lnoffset' > 0 if `touse'
    if _rc {
        di as err "exposure() must be greater than zero"
        exit 459
    }
    tempvar offset
    qui gen double `offset' = ln(`lnoffset')
    local offvar "ln(`lnoffset')"
}
- if "`offset'" != "" {
= if "" != "" {
    markout `touse' `offset'
    local offopt "offset(`offset')"
    if "`offvar'" == "" {
        local offvar "`offset'"
    }
}
- gettoken y xvars : varlist
- _fv_check_depvar `y'
= _fv_check_depvar piq_timing_max
- tsunab y : `y'
= tsunab y : piq_timing_max
}

> _____ begin tsu
> nab _____
    - version 6
    - gettoken user 0: 0, parse(" :")
    - gettoken colon 0: 0, parse(" :")
    - if ``colon'' != ":" { error 198 }
    = if `":' != ":" { error 198 }
    - syntax [varlist(ts default=empty)] [, MIN(integer 1) MAX(integer 32000
> ) NAME(string)]
    - c_local `user' `varlist'

```

```

= c_local y piq_timing_max
- local n : word count `varlist'
= local n : word count piq_timing_max
- if `n'>=`min' & `n'<=`max' { exit }
= if 1>=1 & 1<=32000 { exit }

> _____ end tsu
> nab _____
- local yname : subinstr local y "." "_"
- summarize `y' if `touse', meanonly
= summarize piq_timing_max if __000002, meanonly
- if r(N) == 0 { error 2000 }
- if r(N) == 1 { error 2001 }
- local nobs = round(r(N),1)
- MapHAC `nobs' `hac'
= MapHAC 217

> _____ begin glm.Map
> HAC _____
- args nobs f lag
- local f = lower(trim("`f'"))
= local f = lower(trim(''))
- local l = length("`f'")
= local l = length('')
- local s1
- local s2
- local intlag 0
- if "`f'"==" { local s1 "" }
= if ""==" { local s1 "" }
- else if "`f'"==bsubstr("nwest",1,max(`l',2)) {
= else if ""==bsubstr("nwest",1,max(0,2)) {
    local s1 "glim_nw1"
    local intlag 1
  }
- else if "`f'"==bsubstr("bartlett",1,max(`l',2)) {
= else if ""==bsubstr("bartlett",1,max(0,2)) {
    local s1 "glim_nw1"
    local intlag 1
  }
- else if "`f'"==bsubstr("gallant",1,max(`l',2)) {
= else if ""==bsubstr("gallant",1,max(0,2)) {
    local s1 "glim_nw2"
    local intlag 1
  }
- else if "`f'"==bsubstr("parzen",1,max(`l',2)) {
= else if ""==bsubstr("parzen",1,max(0,2)) {

```

```

        local s1 "glim_nw2"
        local intlag 1
    }
- else if ``f''==bsubstr("anderson",1,max(`l',2)) { local s1 "glim_nw3
> " }
= else if ``''==bsubstr("anderson",1,max(0,2)) { local s1 "glim_nw3" }
- else { local s1 ``f'' }
= else { local s1 ``''' }
- if ``lag'' != ``''' {
= if ``''' != ``''' {
    if `intlag' {
        confirm integer number `lag'
    }
    if `lag' < 0 {
        noi di as err "Newey-West lag must be positive"
        exit 198
    }
}
- else local lag = `nobs'-2
= else local lag = 217-2
- if `lag' >= `nobs'-2 & `intlag' {
= if 215 >= 217-2 & 0 {
    local lag = `nobs'-2
}
- local s2 `lag'
= local s2 215
- ret local hacnam ``s1''
= ret local hacnam ``'''
- ret local haclag ``s2''
= ret local haclag "215"
- ret local intlag ``intlag''
= ret local intlag "0"

```

```

> _____ end glm.Map

```

```

> HAC _____
- local hacnam ``r(hacnam)''
= local hacnam ``''
- local haclag ``r(haclag)''
= local haclag ``215''
- local intlag ``r(intlag)''
= local intlag ``0''
- local fwt "1"
- local awt "1"
- tempvar wt
- if ``weight'' != ``''' {
= if ``''' != ``''' {

```

```

qui gen double `wt' `exp' if `touse'
local awt "`wt'"
if "`weight'"=="aweight" {
summ `wt' , meanonly
qui replace `wt' = `wt'/r(mean)
}
else if "`weight'"=="fweight" {
if "`bstrap'`hac'" != "" {
di as err "fweights not allowed"
exit 198
}
if "`jknife'" != "" & "`cluster'" != "" {
sort `cluster' `tvar' `touse'
cap by `cluster' : assert `wt' == `wt'[_n-1] if `touse' & (`touse'[_n-1]
> ==1)
if _rc {
noi di as err "weight must be " "constant within `cluster'"
exit 198
}
}
summ `touse' [fw=`wt'] if `touse' , meanonly
local nob = round(r(N),1)
local fwt "`wt'"
local awt "1"
}
if "`weight'" == "pweight" {
local robust "robust"
}
}
- else {
- qui gen byte `wt' = `touse' if `touse'
= qui gen byte __000003 = __000002 if __000002
- }
- if "`weight'"=="fweight" {
= if ""=="fweight" {
local wwt `[fw=`wt']'
}
- else if "`weight'" == "iweight" {
= else if "" == "iweight" {
local wwt `[iw=`wt']'
}
- else local wwt `[aw=`wt']'
= else local wwt `[aw=__000003]'
- if "`robust'" != "" {
= if "robust" != "" {
- if "`hacnam'`opg'`jknife'`jknife1'" != "" {

```



```

= if "" != "" {
  di as err "robust allowed only with hessian matrices"
  exit 198
}
- }
- if "`weight'" != "" {
= if "" != "" {
  if "`weight'" != "fweight" {
    if "`hacnam'`opg'`jknife'`jknife1'" != "" {
      di as err "only fweights (if any) are available for non-hessian, non-boo
> tstrap matrices"
      exit 198
    }
  }
}
- if `fvops' {
= if 1 {
- local rmcoll "version 11: _rmcoll"
- local fvexp expand
- }
- else local rmcoll _rmcoll
- local is_bin = "`family'" == "glim_v2"
= local is_bin = "glim_v6" == "glim_v2"
- if `is_bin' & "`m'" != "" {
= if 0 & "1" != "" {
  capture assert `m' == 1
  if c(rc) {
    local is_bin 0
  }
}
- if "`asis'" != "" {
= if "" != "" {
  if !`is_bin' {
    di as err "{p 0 0 2}" "option asis not allowed;{break}" "option asis is
> only allowed with family(binomial) " "with a denominator of 1" "{p_end}"
    exit 198
  }
  local is_bin 0
}
- if !`fvops' {
= if !1 {
  local is_bin 0
}
- if `is_bin' {
= if 0 {
  capture assert inlist(`y', 0, 1) if `touse'

```

```

        if c(rc) {
            local is_bin 0
        }
    }
    - if "`display'" != "" & "`weight'" == "aweight" {
= if "" != "" & "" == "aweight" {
        local rmwgt [iw`exp']
        local rmcoll qui `rmcoll'
    }
    - else {
    - local rmwgt [`weight'`exp']
= local rmwgt []
    - }
    - if `is_bin' {
= if 0 {
        if strmatch("`from'", "e(b)*") {
            tempname ehold
            _est hold `ehold', restore nullok
        }
        `rmcoll' `y' `xvars' `rmwgt' if `touse' , noskipline `coll' `constant' `
> fvexp' logit touse(`touse')
        if "`ehold'" != "" {
            _est unhold `ehold'
        }
        local VARLIST `r(varlist)'
        gettoken y xvars : VARLIST
        summ `touse' `wwt' if `touse' , meanonly
        local nobs = round(r(N),1)
    }
    - else {
    - `rmcoll' `xvars' `rmwgt' if `touse', `coll' `constant' `fvexp'
    = version 11: _rmcoll rule_g rule_itr estimated i.wlth#i.open cb_authors_
> ext ln_neq i.ntwrth#i.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid [] i
> f __000002, expand
note: 1.wlth#1.open omitted because of collinearity.
note: 1.ntwrth#0.bnkcrdit omitted because of collinearity.
note: 1.ntwrth#1.bnkcrdit omitted because of collinearity.
    - local xvars `r(varlist)'
    = local xvars rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.w
> lth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.n
> twrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnk
> crdit vint_early vint_mid
    - }
    - global SGLM_V ``family'
    = global SGLM_V `glim_v6'
    - global SGLM_L ``link'

```

```

= global SGLM_L `"glim_l03"'
- global SGLM_A `""ancilla""'
= global SGLM_A `""'
- global SGLM_y `""y""'
= global SGLM_y `"piq_timing_max"'
- global SGLM_m `""m""'
= global SGLM_m `"1"'
- if `""k""' == "ml" {
= if `"1"' == "ml" {
    qui nbreg `y' `xvars' if `touse' [`weight'`exp'], `constant' `offopt' `m
> lopts'
    local k = e(alpha)
    global SGLM_ml = "ml"
    }
- global SGLM_a `""k""'
= global SGLM_a `"1"'
- global SGLM_p `""pow""'
= global SGLM_p `""0""'
- global SGLM_f `""fisher""'
= global SGLM_f `"-1"'
- global SGLM_mu
- global SGLM_s1 `""scale1""'
= global SGLM_s1 `"1"'
- $SGLM_V -1 `touse'
= glim_v6 -1 __000002

```

```

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if -1 == -1 {
- local y "$SGLM_y"
= local y "piq_timing_max"
- local m "$SGLM_m"
= local m "1"
- local touse "`eta'"
= local touse "__000002"
- capture assert `y'>=0 if `touse'
= capture assert piq_timing_max>=0 if __000002
- if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
- capture assert `y'==int(`y') if `touse'
= capture assert piq_timing_max==int(piq_timing_max) if __000002

```

```

- if _rc {
  di as txt `"note: `y' has noninteger values"'
}
- global SGLM_vt "Neg. Binomial"
- local SGLM_a = round($SGLM_a,0.0001)
= local SGLM_a = round(1,0.0001)
- global SGLM_vf "u+(`SGLM_a')u^2"
= global SGLM_vf "u+(1)u^2"
- global SGLM_mu "glim_mu 0 ."
- exit

```

```

> _____ end glim
> _v6 _____
- $SGLM_L -1 `touse'
= glim_l03 -1 __000002

```

```

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if -1 == -1 {
- global SGLM_lt "Log"
- if "$SGLM_m" == "1" {
= if "1" == "1" {
- global SGLM_lf "ln(u)"
- }
- else global SGLM_lf "ln(u/$SGLM_m)"
= else global SGLM_lf "ln(u/1)"
- exit

```

```

> _____ end glim_
> l03 _____
- global SGLM_running 1
- if "$SGLM_V" == "glim_v1" | "$SGLM_V" == "glim_v3" | "$SGLM_V" == "glim_
> v4" | "$SGLM_V" == "glim_v5" {
= if "glim_v6" == "glim_v1" | "glim_v6" == "glim_v3" | "glim_v6" == "glim_
> v4" | "glim_v6" == "glim_v5" {
if "$SGLM_L" == "glim_l02" | "$SGLM_L" == "glim_l04" | "$SGLM_L" == "gli
> m_l05" | "$SGLM_L" == "glim_l06" | "$SGLM_L" == "glim_l07" | "$SGLM_L" == "g
> lim_l08" | "$SGLM_L" == "glim_l12" {
global SGLM_nonstan 1
}
}
- if "$SGLM_V" == "glim_v2" {
= if "glim_v6" == "glim_v2" {

```

```

        if "$SGLM_L" == "glim_l04" {
        global SGLM_nonstan 1
        }
    }
    - if "$SGLM_V" == "glim_v6" {
    = if "glim_v6" == "glim_v6" {
    - if "$SGLM_L" == "glim_l02" | "$SGLM_L" == "glim_l05" | "$SGLM_L" == "gli
> m_l06" | "$SGLM_L" == "glim_l07" | "$SGLM_L" == "glim_l08" | "$SGLM_L" == "g
> lim_l12" {
    = if "glim_l03" == "glim_l02" | "glim_l03" == "glim_l05" | "glim_l03" == "
> glim_l06" | "glim_l03" == "glim_l07" | "glim_l03" == "glim_l08" | "glim_l03"
> == "glim_l12" {
        global SGLM_nonstan 1
        }
    - }
    - local mllog `log' `nolog'
    = local mllog nolog
    - _parse_iterlog, `log' `nolog'
    = _parse_iterlog, nolog
}
> ----- begin _parse_iter
> log -----
    - version 16
    - syntax [, NOLOG LOG]
    - if "`log' `nolog'" != "" {
    = if "nolog" != "" {
    - opts_exclusive "`log' `nolog'"
    = opts_exclusive " nolog"
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts nolog
    - local n 0
    - while ```opts'"" != "" {
    = while ``nolog"" != "" {
    - local ++n
    - gettoken item`n' opts : opts, bind
    = gettoken item1 opts : opts, bind
    - }
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind

```

```

        }
        - if `n' < 2 {
        = if 1 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - }
    - else if "`c(iterlog)'" == "on" {
    = else if "on" == "on" {
        local log log
    }
    - else {
        local nolog nolog
    }
    - sreturn local log `log'
    = sreturn local log
    - sreturn local nolog `nolog'
    = sreturn local nolog nolog
}
> ----- end _parse_iter
> log -----
    - local log "`s(nolog)'"
    = local log "nolog"
    - if ``log'"" != "" {
    = if `"nolog'"" != "" {
    - local iflog `*'
    - }
    - else { local iflog "noi" }
    - if "`offset'" != "" {
    = if "" != "" {
        local moffset = "-`offset'"
    }
    - local reg = "$SGLM_V" == "glim_v1" & "$SGLM_L" == "glim_l01"
    = local reg = "glim_v6" == "glim_v1" & "glim_l03" == "glim_l01"
    - if "`irls'" == "" {
    = if "" == "" {
    - tempvar eta mu dm u v z W
    - tempname Wscale b0
    - if !inlist("`vce'", "eim", "opg") {
    = if !inlist("robust", "eim", "opg") {
    - local oim "oim"
    - }
    - if "`from'" == "" & "`search'" == "" {
    = if "" == "" & "" == "" {
    - quietly {

```

```

- if ``init''!=`` {
= if ``''!=`` {
  gen double `mu' = `init' if `touse'
}
- else {
- sum `y' `wgt' if `touse'
= sum piq_timing_max [aw=__000003] if __000002
- if "$SGLM_V" != "glim_v2" {
= if "glim_v6" != "glim_v2" {
- gen double `mu' = (`y'+r(mean))/(`m'+1)
= gen double __000005 = (piq_timing_max+r(mean))/(1+1)
- }
- else {
  gen double `mu' = `m'*(`y'+.5)/(`m'+1)
}
- }
- capture drop `eta'
= capture drop __000004
- $SGLM_L 0 `eta' `mu'
= glim_l03 0 __000004 __000005

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 0 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 0 == 0 {
- gen double `eta' = ln(`mu'/$SGLM_m)
= gen double __000004 = ln(__000005/1)
- exit

> _____ end glim_
> l03 _____
- $SGLM_V 1 `eta' `mu' `v'
= glim_v6 1 __000004 __000005 __000007

> _____ begin glim

```

```

> _v6
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"
  local SGLM_a = round($SGLM_a,0.0001)
  global SGLM_vf "u+(`SGLM_a')u^2"
  global SGLM_mu "glim_mu 0 ."
  exit
}
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = -ln1p($SGLM_a/`mu')
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
  = gen double __000007 = __000005+__000005*__000005*1
  - exit
}
}

> _____ end glim
> _v6
- $SGLM_L 2 `eta' `mu' `dmu'
= glim_l03 2 __000004 __000005 __000006

> _____ begin glim_
> l03
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"

```



```

    if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
    }
- if `todo' == 0 {
= if 2 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
    }
- if `todo' == 1 {
= if 2 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
    }
- if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = `mu'
    = gen double __000006 = __000005
    - exit
}

> _____ end glim_
> l03 _____
- gen double `z' = `eta' + (`y' - `mu')/`dmu' `moffset'
= gen double __000008 = __000004 + (piq_timing_max - __000005)/__000006
- gen double `W' = `dmu'*`dmu'/'v'
= gen double __000009 = __000006*__000006/__000007
- summ `W' `wgt', meanonly
= summ __000009 [aw=__000003], meanonly
- scalar `Wscale' = r(mean)
= scalar __00000A = r(mean)
- if "`cns'" != "" & _caller() >= 11 {
= if "" != "" & _caller() >= 11 {
    foreach c in `cns' {
        local c`c' : constraint `c'
        local c`c'new = subinstr( "`c`c'", "[`y']", "", .)
        constraint `c' `c`c'new'
    }
    `VV' cap noisily quietly cnsreg `z' `xvars' [iw=`W'*`wt'/'Wscale'], cons
> tr(`cns') mse1 `constant'
    foreach c in `cns' {
        constraint `c' `c`c''
    }
}
- else {

```

```

- `VV' cap noisily quietly _regress `z' `xvars' [iw=`W'*`wt'/'Wscale'], ms
> e1 `constant'
= version 18.5, missing: cap noisily quietly _regress __000008 rule_g rule
> _itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open
> cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#
> 0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid [
> iw=__000009*__000003/__00000A], mse1

-----
> ----- begin _ereturn_disp
> lay -----
- version 11
- if "`e(b)'" == "" & "`e(V)'" == "" {
= if "matrix" == "" & "matrix" == "" {
exit
}
- if "`e(cmd)'" == "regress" {
= if "regress" == "regress" {
- local ALTOPT Beta
- }
- syntax [, eform(passthru) First neq(integer -1) PPlus `ALTOPT' *]
= syntax [, eform(passthru) First neq(integer -1) PPlus Beta *]
- _get_mcompare, `options'
= _get_mcompare, level(95.00) fwrap(1)

-----
> ----- begin _get_mcomp
> are -----
- version 11
- local cmd ``e(cmd)''
= local cmd `regress'
- syntax [name(name=caller)] [, *]
- local is_pwc = inlist("pwcompare", "`cmd'", "`caller'") | inlist("pw
> mean", "`cmd'", "`caller'")
= local is_pwc = inlist("pwcompare", "regress", "") | inlist("pwmean",
> "regress", "")
- local is_marg = inlist("margins", "`cmd'", "`e(cmd2)'", "`caller'")
= local is_marg = inlist("margins", "regress", "", "")
- local OPTIONS NOADJust BONferroni DUNCAn DUNNett SCHeffe SIDak SNK T
> UKey ADJustall
- if `is_pwc' {
= if 0 {
syntax [name] [, `OPTIONS' MCOMPare(string) *]
}
- else {
- syntax [name] [, MCOMPare(string) *]
- local suboption mcompare()
- }

```

```

- sreturn clear
- sreturn local options ``options'''
= sreturn local options ``level(95.00) fvwrap(1)'''
- if `is_pwc' {
= if 0 {
    local METHOD `noadjust' `bonferroni' `duncan' `dunnett' `scheffe' `s
> idak' `snk' `tukey'
    local METHOD : list uniq METHOD
    opts_exclusive ``METHOD'''
    local ALL `adjustall'
    opts_exclusive ``noadjust' `ALL'''
    }
- local 0 `", `mcompare'''
= local 0 `", ""
- capture syntax [, `OPTIONS']
= capture syntax [, NOADJUST BONferroni DUNCAn DUNNett SCHeffe SIDak S
> NK TUKey ADJustall]
- if c(rc) {
    di as err "invalid mcompare() option;"
    di as err "method ``mcompare'' not allowed"
    error 198
    }
- local METHOD `METHOD' `noadjust' `bonferroni' `duncan' `dunnett' `sc
> heffe' `sidak' `snk' `tukey'
= local METHOD
- local METHOD : list uniq METHOD
- local ALL `ALL' `adjustall'
= local ALL
- local ALL : list uniq ALL
- opts_exclusive ``METHOD''' `suboption'
= opts_exclusive "" mcompare()



---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts''' != "" {
= while `""' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    }
}
> ----- end opts_exclus
> ive
    - if "`METHOD'" == "" {
    = if "" == "" {
    - if "`ALL'" != "" {
    = if "" != "" {
        di as err "multiple comparison method required with the adjustall op
> tion"
        exit 198
    }
    - local METHOD noadjust
    - local noadjust noadjust
    - }
    - opts_exclusive "`ALL' `noadjust'" `suboption'
    = opts_exclusive " noadjust" mcompare()
}
> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts  noadjust
    - local n 0
    - while ``opts' != "" {
    = while `"noadjust" != "" {
    - local ++n
    - gettoken item`n' opts : opts, bind
    = gettoken item1 opts : opts, bind
    - }
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 1 < 2 {
    - exit
    }
}
> ----- end opts_exclus
> ive
    - local EXCLUDE duncan dunnett snk tukey
    - if `:list METHOD in EXCLUDE' {
    = if 0 {
        if `is_pwc' {

```

```

    local opt : copy local METHOD
  }
  else {
    local opt "mcompare(`METHOD')"
  }
  local cmdlist anova manova regress mvreg pwmean
  if "`cmd'" == "pwcompare" {
    local cmd ``e(cmd_est)'''
  }
  if `is_marg' {
    local cmd margins
  }
  if !':list cmd in cmdlist' | !`is_pwc' {
    di as err `"method `opt' is not allowed with results from `cmd''
    exit 322
  }
  if ``e(prefix)''' == "svy" {
    di as err `"method `opt' is not allowed with results using the svy p
> refix"'
    exit 322
  }
  local vce ``e(vce)'''
  if !inlist("`vce'", "", "ols") {
    di as err `"method `opt' is not allowed with results using vce(`vce'
> )"''
    exit 322
  }
}
- local EXCLUDE duncan dunnett snk tukey scheffe
- if `:list METHOD in EXCLUDE' {
= if 0 {
  if "`ALL'" != "" {
    di as err `"option adjustall is not allowed with method `METHOD''
    exit 198
  }
}
- sreturn local method ``METHOD'''
= sreturn local method "noadjust"
- sreturn local adjustall ``ALL'''
= sreturn local adjustall ""''



---


> _____ end _get_mcomp
> are _____
- if "`s(method)'" != "" {
= if "noadjust" != "" {
- loca mcopt mcompare(`s(method)' `s(adjustall)')

```

```

= loca mcopt mcompare(noadjust )
- }
- _get_diopts diopts, `s(options)'
= _get_diopts diopts, level(95.00) fvwrap(1)

```

```

> ----- begin _get_dio
> pts -----
- version 11
- syntax namelist(max=2) [, *]
- gettoken name1 namelist : namelist
- gettoken name2 namelist : namelist
- GetDiopts diopts options, `options'
= GetDiopts diopts options, level(95.00) fvwrap(1)

```

```

> ----- begin _get_diopts.GetDio
> pts -----
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels
> NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NO
> OMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sfor
> mat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(
> string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels A
> LLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells
> EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat
> (string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passth
> ru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passth
> ru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
local ++n
gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____

```

```

- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit


---


> 

---

 end opts_exclus
> ive 

---


- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit


---


> 

---

 end opts_exclus
> ive 

---


- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus

```



```

> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``coeflegend' `selegend'
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``noci' `nopvalues'
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind

```

```

    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
= local baselevels
    - local emptycells `emptycells' `noemptycells'
= local emptycells
    - local omitted `omitted' `noomitted'
= local omitted
    - local lstretch `lstretch' `nolstretch'
= local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
    - local abbrev `abbrev' `noabbrev'
= local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
    - if c(showbaselevels) == "all" {
        local allbaselevels allbaselevels

```

```

    }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')'
  }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'

```

```

        if fmtwidth("`pformat'") > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
        }
        sreturn local pformat "`pformat'"
        local pformat `pformat(`pformat')'
    }
    - if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
        syntax namelist(max=2) [, `DIOPTS'
    }
    - if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
        local levelopt level(`level')
    }
    - Wrapon , `fvwrapon'
= Wrapon ,

> ----- begin _get_diopts.Wra
> pon -----
    - capture syntax [, WOrd WIdth]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, WOrd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while "`opts'" != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit

> ----- end opts_exclus

```

```

> ive
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }

>
> pon
    - if "`fvwrap'" != "" {
    = if "fvwrap(1)" != "" {
    - FVWrap, `fvwrap'
    = FVWrap, fvwrap(1)

>
> rap
    - syntax, fvwrap(integer)
    - if `fvwrap' < 0 {
    = if 1 < 0 {
        di as err "{p}invalid option {bf:fvwrap(`fvwrap')}"; " "nonnegati
> ve integer required{p_end}"
        exit 198
    }

>
> rap
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevel
> s' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend
> ' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwr
> ap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts fvwrap(1)
    - if `K' == 2 {
    = if 2 == 2 {
    - c_local `c_opts' ``options'``
    = c_local options ````
    - }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95

```

```

> _____ end _get_diopts.GetDio
> pts _____
    - c_local `name1' ```diopts''
    = c_local diopts ``fvwrap(1)''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)
    - }
    - local S_`sname' = s(`sname')
    = local S_method = s(method)
    - }
    - local S_`sname' = s(`sname')
    = local S_options = s(options)
    - }
    - sreturn clear
    - GetDiopts DEFAULT

```

```

> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels
> NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NO
> OMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sfor
> mat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(
> string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels A
> LLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells
> EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat
> (string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passth
> ru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passth
> ru) NOABBREV ABBREV *]
    - opts_exclusive ``allbaselevels' `noallbaselevels''
    = opts_exclusive " "

```

```

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'' != "" {
    = while `""' != "" {

```



```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nbaselevels'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nbaselevels'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus

```

```

> ive
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""`opts'"" != "" {
    = while `""' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""`opts'"" != "" {
    = while `""' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'

```

```

= local abbrev
- if "`allbaselevels'"`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"
}

```

```

    local sformat ``sformat(`sformat')''
  }
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth(`"pformat"') > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat ``pformat''
  local pformat `pformat(`pformat')''
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
  syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
  local levelopt level(`level')
}
- Wrapon , `fvwrapon'
= Wrapon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
  di as err "invalid fvwrapon() option;"
  syntax [, W0rd WIdth]
  exit 198
}
- opts_exclusive ``word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

> _____ begin opts_exclus

> ive _____

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}

```



```

        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}

> ----- end opts_exclus
> ive -----
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
}

> ----- end _get_diopts.Wra
> pon -----
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevel
> s' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend
> ' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwr
> ap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DEFAULT
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options''
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95
}

> ----- end _get_diopts.GetDio
> pts -----
    - if ``name2'' == "" {
    = if ``' == "" {
        GetDiopts DUPS, `options'
    = GetDiopts DUPS,
    }

> ----- begin _get_diopts.GetDio
> pts -----

```

```

- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels
> NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NO
> OMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sfor
> mat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(
> string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels A
> LLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells
> EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat
> (string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passth
> ru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passth
> ru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `""opts"" != "" {
= while `"" != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

> _____ end opts_exclus
> ive _____
- opts_exclusive "`allbaselevels' `nobaselevels'"
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `""opts"" != "" {
= while `"" != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus

```

```

> ive
- opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
-----
> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `"'opts'"' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
-----
> ----- end opts_exclus
> ive
- opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "
-----
> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `"'opts'"' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
-----
> ----- end opts_exclus
> ive
- opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`nocl' `nopvalues'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if ``allbaselevels'`baselevels' == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if ``emptycells' == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if ``omitted' == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
}

```



```

- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"
  local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth("`pformat'") > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat "`pformat'"
  local pformat `pformat(`pformat')'
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
  syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
  local levelopt level(`level')
}
- Wraapon , `fvwraapon'

```

```

= Wraapon ,
> _____ begin _get_diopts.Wra
> pon _____
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fwraapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fwraapon
= opts_exclusive " " fwraapon
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
        - exit
> _____ end opts_exclus
> ive _____
    - local fwraapon `word' `width'
= local fwraapon
    - if `:length local fwraapon' {
= if 0 {
        c_local fwraapon fwraapon(`fwraapon')
    }
> _____ end _get_diopts.Wra
> pon _____
    - if "`fwwrap'" != "" {
= if "" != "" {
        FWWrap, `fwwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevel
> s' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend

```

```

> ' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwr
> ap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DUPS
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options''
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95



---


> _____ end _get_diopts.GetDio
> pts _____
    - }
    - else {
        GetDiopts DUPS options, `options'
        c_local `name2' ``options''
    }
    - local DUPS : list DUPS - DEFAULT
    - gettoken DUPS : DUPS
    - if ``DUPS'' != "" {
    = if ``' != "" {
        if (strpos(`DUPS', "(")) {
            gettoken DUPS : DUPS, parse("(")
            local DUPS `DUPS'()'
        }
        di as err "duplicate {bf:`DUPS'} option not allowed"
        exit 198
    }
    - sreturn clear
    - foreach sname of local snames {
    - sreturn local `sname' = ``S_`sname''
    = sreturn local level = `95'
    - }
    - sreturn local `sname' = ``S_`sname''
    = sreturn local method = `noadjust'
    - }
    - sreturn local `sname' = ``S_`sname''
    = sreturn local options = `level(95.00) fvwrap(1)'
    - }

```

```

> _____ end _get_dio
> pts _____
  - if !`:length local first' {
  = if !0 {
  - local first showeqns
  - }
  - if `neq' < 1 {
  = if -1 < 1 {
  - _ms_eq_info
  - local neqopt neq(`r(k_eq)')
  = local neqopt neq(1)
  - }
  - else {
    local neqopt neq(`neq')
  }
  - _coef_table, nodiparm nocnsreport noeqcheck eformall `eform' `first' `
> neqopt' `plus' `beta' `mcopt' `diopts'
  = _coef_table, nodiparm nocnsreport noeqcheck eformall showeqns neq(1)
> mcompare(noadjust ) fvwrap(1)
  _____
> _____ begin _coef_ta
> ble _____
  - version 11
  - local vv : di "version " string(max(11,_caller())) ", missing:"
  - if (!c(noisily) & c(coeftabresults) == "off") {
    exit
  }
  - _check_eclass
  - syntax [, BMATrix(passthru) VMATrix(passthru) * ]
  - if "`e(b)'" == "" & "`e(V)'" == "" & "`bmatrix'`vmatrix'"==" {
  = if "matrix" == "" & "matrix" == "" & ""==" {
    exit
  }
  - local mc_cmds contrast margins pwcompare pwmean
  - local cmd "`e(cmd)'"
  = local cmd "regress"
  - local keepmc : list cmd in mc_cmds
  - if inlist("`cmd'", "pwcompare", "pwmean") {
  = if inlist("regress", "pwcompare", "pwmean") {
    local groups GROUPS
  }
  - if "`cmd'" == "regress" {
  = if "regress" == "regress" {
    - local beta Beta
    - }
    - if "`cmd'" == "gsem" {

```

```

= if "regress" == "gsem" {
    local fvignore fvignore(int 0)
    local flignore flignore
}
- if "`cmd'" == "sem" {
= if "regress" == "sem" {
    local standardized STANDARDIZED
    local showginvariant SHOWGINvariant
    local nolabel NOLABel LABel NOFVLABel FVLABel
    local nofootnote NOFOOTnote
    local wrap wrap(numlist max=1) fvwrap(passthru)
}
- if "`e(mi)'"=="mi" {
= if ""=="mi" {
    local dftable DFTable
    local dfonly DFONLY
    local noclustreport NOCLUSTReport
    local pisematrix PISEMATrix(string)
}
- syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(string)
> DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) PMATrix
> (string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat(s
> tring) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATrix(
> string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name) c
> nszero(string) suffix(name) OFFSETlist(string asis) `beta' `standardized' NO
> CI NOPValues `dftable' `dfonly' dfci DFPValues dfmissing `groups' COEFLegend
> selegend `fvignore' `flignore' `showginvariant' `nolabel' `nofootnote' `wra
> p' sort `pisematrix' depname(string) COEFTitle(string) coeftitle2(string) pt
> itle(string) cititle(string) NOMCLEGend noCNSReport FULLCNSReport `noclustre
> port' cformat(passthru) sformat(passthru) pformat(passthru) NOFirst First SH
> OWEQns neq(integer -1) NODIPARM NOTEST SEParator(integer 0) NOSKIP OFFSETONL
> Y1 PPlus NOEQCHECK EFORMALL CITYPE(string) eqselect(string) cidisjoint noEFOR
> MNOTE * ]
    = syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(string)
> DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) PMATrix
> (string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat(s
> tring) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATrix(
> string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name) c
> nszero(string) suffix(name) OFFSETlist(string asis) Beta NOCI NOPValues d
> fci DFPValues dfmissing COEFLegend selegend sort depname(string) COE
> FTitle(string) coeftitle2(string) ptitle(string) cititle(string) NOMCLEGend
> noCNSReport FULLCNSReport cformat(passthru) sformat(passthru) pformat(passt
> hru) NOFirst First SHOWEQns neq(integer -1) NODIPARM NOTEST SEParator(intege
> r 0) NOSKIP OFFSETONLY1 PPlus NOEQCHECK EFORMALL CITYPE(string) eqselect(stri
> ng) cidisjoint noEFORMNOTE * ]
    - if "`wrap'" != "" {

```

```

= if "" != "" {
    opts_exclusive `wrap(`wrap') `fvwrap'''
    local fvwrap fvwrap(`wrap')
}
- if "`nolabel'`label'" != "" {
= if "" != "" {
    opts_exclusive "`nolabel' `label'"
    opts_exclusive "`nofvlabel' `fvlabel'"
    opts_exclusive "`nolabel' `fvlabel'"
    opts_exclusive "`nofvlabel' `label'"
    if "`nolabel'" != "" {
        local fvlabel nofvlabel
    }
    else {
        local fvlabel fvlabel
    }
}
- _novce_other_mats, cimatrix("`cimatrix'") pmatrix("`pmatrix'") tsmat
> rix("`tsmatrix'") `noc' `nopvalues' `dfci' `dfpvalues' `cidisjoint'
= _novce_other_mats, cimatrix("") pmatrix("") tsmatrix("")

> _____ begin _coef_table._novce_other_m
> ats _____
- syntax [anything], [ CIMATrix(string) PMATrix(string) TSMATrix(str
> ing) NOCI NOPValues dfci DFPValues cidisjoint ]
- if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"!=" & "`noc'"!=") {
    = if ("'"=="" & "'"=="" & "matrix"=="" & '"!=" & '"!=") {
        sreturn local cimatrix "empty"
    }
    - if ("`pmatrix'"=="" & "`tsmatrix'"!=" & "`e(V)'"=="" & "`cimatrix
> '"=="" & ("`noc'"!="|"`nopvalues'"!=")) {
        = if ("'"=="" & '"!=" & "matrix"=="" & '"=="" & ("'"!="|'"!=")) {
            sreturn local noc "empty"
            sreturn local nopvalues "empty"
        }
        - if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"!=" & "`noc'"!=") {
            = if ("'"=="" & "'"=="" & "matrix"=="" & '"!=" & '"!=") {
                sreturn local cimatrix "empty"
            }
            - if ("`pmatrix'"!=" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"=="" & "`nopvalues'"!=") {
                = if ("'"!=" & "'"=="" & "matrix"=="" & '"=="" & '"!=") {
                    sreturn local pmatrix "empty"
                }
            }
        }
    }
}

```

```

- if ("`pmatrx'"!=" & "`tsmatrx'"==" & "`e(V)'"==" & "`cimatrix
> '"==" & "`noci'"!=") {
= if ("'"!=" & '"==" & "matrix"==" & '"==" & '"!=") {
    sreturn local noci "empt"
  }
- if ("`pmatrx'"!=") {
= if ("'"!=") {
    tempname pmat invalid
    capture matrix `pmat' = `pmatrx'
    local rc = _rc
    if (`rc') {
      display as error "invalid {bf:pmatrx()}"
      exit 198
    }
    else {
      local k: rowsof `pmat'
      if (`k'>1 | `k'==0) {
        display as error "{bf:pmatrx()} should be a row vector"
        exit 198
      }
    }
    scalar `invalid' = 1
    mata: _is_in_0_1void("`pmat'", "`invalid'")
    if (`invalid'==0) {
      display as error "invalid {bf:pmatrx()}"
      di as txt "{p 4 4 2}"
      di as smcl as err "{bf:pmatrx()}"
      di as smcl as err "should contain values inside"
      di as smcl as err "the interval [0,1]."
      di as smcl as err "{p_end}"
      exit 198
    }
  }
- if ("`tsmatrx'"!=") {
= if ("'"!=") {
    tempname tmat
    capture matrix `tmat' = `tsmatrx'
    local rc = _rc
    if (`rc') {
      display as error "invalid {bf:tsmatrx()}"
      exit 198
    }
    else {
      local k: rowsof `tmat'
      if (`k'>1 | `k'==0) {
        display as error "{bf:tsmatrx()} should be a row vector"

```

```

exit 198
}
}
}
- if ("`cimatrix'"!="") {
= if ("'"!="") {
  tempname cimat isvalid
  capture matrix `cimat' = `cimatrix'
  local rc = _rc
  if (`rc') {
    display as error "invalid {bf:cimatrix()}"
    exit 198
  }
  if ("`cidisjoint'"==" & "`e(disjoint)'"=="") {
    scalar `isvalid' = 1
    mata: _is_in_ci("`cimat'", "`isvalid'")
    if (`isvalid'==0) {
      display as error "invalid {bf:cimatrix()}"
      di as txt "{p 4 4 2}"
      di as smcl as err "{bf:cimatrix()}"
      di as smcl as err "lower limit should be"
      di as smcl as err "smaller than the upper"
      di as smcl as err "limit."
      di as smcl as err "{p_end}"
      exit 198
    }
  }
}
}
- if ("`e(V)'"==" & "`dfci'"!=" & "`pmatrx'"!="") {
= if ("matrix"==" & "'"!=" & "'"!="") {
  opts_exclusive "pmatrx() `dfci'"
}
- if ("`e(V)'"==" & "`dfci'"!=" & "`tsmatrix'"!="") {
= if ("matrix"==" & "'"!=" & "'"!="") {
  opts_exclusive "tsmatrix() `dfci'"
}
- if ("`e(V)'"==" & "`dfci'"!=" & "`cimatrix'"!="") {
= if ("matrix"==" & "'"!=" & "'"!="") {
  opts_exclusive "cimatrix() `dfci'"
}
- if ("`e(V)'"==" & "`dfpvalues'"!=" & "`pmatrx'"!="") {
= if ("matrix"==" & "'"!=" & "'"!="") {
  opts_exclusive "pmatrx() `dfci'"
}
- if ("`e(V)'"==" & "`dfpvalues'"!=" & "`tsmatrix'"!="") {
= if ("matrix"==" & "'"!=" & "'"!="") {

```



```

    opts_exclusive "tsmatrix() `dfpvalues'"
  }
- if ("`e(V)'"==" & "`dfpvalues'"!=" & "`cimatrix'"!=") {
= if ("matrix"==" & ""!=" & ""!=") {
    opts_exclusive "cimatrix() `dfpvalues'"
  }
}

> _____ end _coef_table._novce_other_m
> ats _____
- if ("`s(cimatrix)'"=="empt") {
= if ("=="empt") {
    local cimatrix ""
  }
- if ("`s(pmatrix)'"=="empt") {
= if ("=="empt") {
    local pmatrix ""
  }
- if ("`s(noci)'"=="empt") {
= if ("=="empt") {
    local noci ""
  }
- if ("`s(nopvalues)'"=="empt") {
= if ("=="empt") {
    local nopvalues ""
  }
- if ("`e(V)'"!=" & "`cidisjoint'"!=") {
= if ("matrix"!=" & ""!=") {
    display in red "option {bf:cidisjoint} not allowed when e(V) is post
> ed"
    exit 198
  }
- else local fvlabel `nofvlabel' `fvlabel'
= else local fvlabel
- local type `coeflegend' `selegend'
= local type
- opts_exclusive "`type' `standardized'"
= opts_exclusive " "
}

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {

```

```

= while `"" != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- if (``rowmatrix'=='' & ``norowci'!='') {
= if (`''=='' & ''!='') {
    di as err "{bf:norowci} requires {bf:rowmatrix()}"
    exit 198
}
- if (``dftable'!='' & ``norowci'!='') {
= if (''!='' & ''!='') {
    di as err "{bf:norowci} is not allowed with {bf:`dftable'}"
    exit 198
}
- if (``dfonly'!='' & ``norowci'!='') {
= if (''!='' & ''!='') {
    di as err "{bf:norowci} is not allowed with {bf:`dfonly'}"
    exit 198
}
- if (``noci'!='' & ``norowci'!='') {
= if (''!='' & ''!='') {
    di as err "{bf:norowci} is not allowed with {bf:`noci'}"
    exit 198
}
- if ``standardized' != '' & ``bstdmatrix' != '' {
= if '' != '' & '' != '' {
    local beta beta
    local standardized
}
- if `:length local type' == 0 {
= if 0 == 0 {
- local type `beta' `noci' `nopvalues' `dftable' `dfonly' `groups' `df
> ci' `dfpvalues'
= local type
- }
- opts_exclusive ``type'
= opts_exclusive ""
}

> ----- begin opts_exclus
> ive -----

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`first' `nofirst'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`first' `showeqns'"
    = opts_exclusive " showeqns"

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts showeqns

```

```

- local n 0
- while ``opts'' != "" {
= while "showeqns" != "" {
- local ++n
- gettoken item`n' opts : opts, bind
= gettoken item1 opts : opts, bind
- }
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 1 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local cnsreport `cnsreport' `fullcnsreport'
= local cnsreport nocnsreport
- opts_exclusive "`cnsreport'"
= opts_exclusive "nocnsreport"
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts nocnsreport
- local n 0
- while ``opts'' != "" {
= while "nocnsreport" != "" {
- local ++n
- gettoken item`n' opts : opts, bind
= gettoken item1 opts : opts, bind
- }
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 1 < 2 {
- exit
}

> ----- end opts_exclus

```

```

> ive
- if ``showeqns`` != "" {
= if ``showeqns`` != "" {
- local nofirst nofirst
- }
- if ``ematrix``noeqcheck`` == "" {
= if ``noeqcheck`` == "" {
  if ``e(error)`` == "matrix" {
    local ematrix e(error)
  }
}
- _get_mcompare, `options'
= _get_mcompare, mcompare(noadjust ) fvwrap(1)

> _____ begin _get_mcomp
> are
- version 11
- local cmd ``e(cmd)``
= local cmd ``regress``
- syntax [name(name=caller)] [, *]
- local is_pwc = inlist("pwcompare", ``cmd``, ``caller``) | inlist("
> pwmean", ``cmd``, ``caller``)
= local is_pwc = inlist("pwcompare", "regress", "") | inlist("pwmean
> ", "regress", "")
- local is_marg = inlist("margins", ``cmd``, ``e(cmd2)``, ``caller``
> )
= local is_marg = inlist("margins", "regress", "", "")
- local OPTIONS NOADJUST BONferroni DUNCAn DUNNett SCHeffe SIDak SNK
> TUKey ADJustall
- if `is_pwc` {
= if 0 {
  syntax [name] [, `OPTIONS' MCOMPare(string) *]
}
- else {
- syntax [name] [, MCOMPare(string) *]
- local suboption mcompare()
- }
- sreturn clear
- sreturn local options ``options``
= sreturn local options ``fvwrap(1)``
- if `is_pwc` {
= if 0 {
  local METHOD `noadjust' `bonferroni' `duncan' `dunnett' `scheffe'
> `sidak' `snk' `tukey'
  local METHOD : list uniq METHOD
  opts_exclusive ``METHOD``

```

```

        local ALL `adjustall'
        opts_exclusive "`noadjust' `ALL'"
    }
    - local 0 `", `mcompare'"
    = local 0 `", noadjust'"
    - capture syntax [, `OPTIONS']
    = capture syntax [, NOADJUST BONferroni DUNCAn DUNNett SCHeffe SIDak
> SNK TUKey ADJustall]
    - if c(rc) {
        di as err "invalid mcompare() option;"
        di as err "method ``mcompare'' not allowed"
        error 198
    }
    - local METHOD `METHOD' `noadjust' `bonferroni' `duncan' `dunnett' `
> scheffe' `sidak' `snk' `tukey'
    = local METHOD noadjust
    - local METHOD : list uniq METHOD
    - local ALL `ALL' `adjustall'
    = local ALL
    - local ALL : list uniq ALL
    - opts_exclusive "`METHOD'" `suboption'
    = opts_exclusive "noadjust" mcompare()



---


> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts noadjust
    - local n 0
    - while ```opts' != "" {
    = while `"noadjust" != "" {
    - local ++n
    - gettoken item`n' opts : opts, bind
    = gettoken item1 opts : opts, bind
    - }
    - while ```opts' != "" {
    = while `"" != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 1 < 2 {
    - exit



---


> ----- end opts_exclus

```

```

> ive
- if ``METHOD'' == "" {
= if "noadjust" == "" {
  if ``ALL'' != "" {
    di as err "multiple comparison method required with the adjustall
> option"
    exit 198
  }
  local METHOD noadjust
  local noadjust noadjust
}
- opts_exclusive ``ALL' `noadjust' `suboption'
= opts_exclusive " noadjust" mcompare()

> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts noadjust
- local n 0
- while ``opts'' != "" {
= while `noadjust' != "" {
- local ++n
- gettoken item`n' opts : opts, bind
= gettoken item1 opts : opts, bind
- }
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 1 < 2 {
- exit

> ----- end opts_exclus
> ive
- local EXCLUDE duncan dunnett snk tukey
- if `:list METHOD in EXCLUDE' {
= if 0 {
  if `is_pwc' {
    local opt : copy local METHOD
  }
  else {
    local opt "mcompare(`METHOD')"

```

```

    }
    local cmdlist anova manova regress mvreg pwmean
    if "`cmd'" == "pwcompare" {
        local cmd ``e(cmd_est)'''
    }
    if `is_marg' {
        local cmd margins
    }
    if !`:list cmd in cmdlist' | !`is_pwc' {
        di as err `"method `opt' is not allowed with results from `cmd'"
        exit 322
    }
    if ``e(prefix)''' == "svy" {
        di as err `"method `opt' is not allowed with results using the svy
> prefix"
        exit 322
    }
    local vce ``e(vce)'''
    if !inlist("`vce'", "", "ols") {
        di as err `"method `opt' is not allowed with results using vce(`vc
> e')"
        exit 322
    }
    }
    - local EXCLUDE duncan dunnett snk tukey scheffe
    - if `:list METHOD in EXCLUDE' {
    = if 0 {
        if "`ALL'" != "" {
            di as err `"option adjustall is not allowed with method `METHOD'"
            exit 198
        }
    }
    - sreturn local method ``METHOD'''
    = sreturn local method `noadjust'
    - sreturn local adjustall ``ALL'''
    = sreturn local adjustall `'''

> _____ end _get_mcomp
> are _____
    - local method ``s(method)'''
    = local method `noadjust'
    - local all ``s(adjustall)'''
    = local all `'''
    - local options ``s(options)'''
    = local options `fvwrap(1)'
    - if "`method'" != "noadjust" {

```



```

= if "noadjust" != "noadjust" {
    local keepmc 1
    }
- opts_exclusive "`all' `groups'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - if "`method'" == "dunnett" {
    = if "noadjust" == "dunnett" {
        opts_exclusive "`method' `groups'"
    }
    - _get_diopts diopts options, `options' `fvwrap' `fvlabel'
    = _get_diopts diopts options, fvwrap(1)

```

```

> ----- begin _get_dio
> pts -----
    - version 11
    - syntax namelist(max=2) [, *]
    - gettoken name1 namelist : namelist
    - gettoken name2 namelist : namelist
    - GetDiopts diopts options, `options'
    = GetDiopts diopts options, fvwrap(1)

```

```

> ----- begin _get_diopts.GetDio
> pts -----
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0

```

```

> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit


---


> 

---

 end opts_exclus
> ive 

---


    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus

```

```

> ive -----
      - opts_exclusive "`fvlabel' `nofvlabel'"
      = opts_exclusive " "
-----

> ----- begin opts_exclus
> ive -----
      - version 8.2
      - args opts optname errcode optslab
      - local opts `opts'
      = local opts
      - local n 0
      - while ```opts'"" != "" {
      = while `""' != "" {
          local ++n
          gettoken item`n' opts : opts, bind
          }
      - if `n' < 2 {
      = if 0 < 2 {
      - exit
-----

> ----- end opts_exclus
> ive -----
      - opts_exclusive "`omitted' `noomitted'"
      = opts_exclusive " "
-----

> ----- begin opts_exclus
> ive -----
      - version 8.2
      - args opts optname errcode optslab
      - local opts `opts'
      = local opts
      - local n 0
      - while ```opts'"" != "" {
      = while `""' != "" {
          local ++n
          gettoken item`n' opts : opts, bind
          }
      - if `n' < 2 {
      = if 0 < 2 {
      - exit
-----

> ----- end opts_exclus
> ive -----
      - opts_exclusive "`lstretch' `nolstretch'"
      = opts_exclusive " "
-----

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`nocl' `nopvalues'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n

```

```

        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
= local baselevels
    - local emptycells `emptycells' `noemptycells'
= local emptycells
    - local omitted `omitted' `noomitted'
= local omitted
    - local lstretch `lstretch' `nolstretch'
= local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
    - local abbrev `abbrev' `noabbrev'
= local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
    - if c(showbaselevels) == "all" {
        local allbaselevels allbaselevels
    }
    - else if c(showbaselevels) == "on" {
        local baselevels baselevels
    }
    - }
    - if "`emptycells'" == "" {
= if "" == "" {
    - if c(showemptycells) == "off" {
        local emptycells noemptycells
    }
    - }
    - if "`omitted'" == "" {
= if "" == "" {
    - if c(showomitted) == "off" {
        local omitted noomitted
    }
    - }
    - if "`lstretch'" == "" {
= if "" == "" {
    - if c(lstretch) == "off" {

```



```

    local lstretch nolstretch
    }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth(`""cformat'""') > 9 {
        local cformat "%9.0g"
        di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat `""cformat'""'
    local cformat `cformat(`cformat')"'
    }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth(`""sformat'""') > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat `""sformat'""'
    local sformat `sformat(`sformat')"'
    }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth(`""pformat'""') > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat `""pformat'""'
    local pformat `pformat(`pformat')"'
    }
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
    }
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
    }
- Wraon , `fvwraon'
= Wraon ,

```

```

> _____ begin _get_diopts.Wra
> pon _____

```

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
  di as err "invalid fwwrapon() option;"
  syntax [, W0rd WIdth]
  exit 198
}
- opts_exclusive "`word' `width'" fwwrapon
= opts_exclusive " " fwwrapon

```

```

> ----- begin opts_exclus
> ive -----
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while ```opts'"" != "" {
  = while `""' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
    - exit
  }

```

```

> ----- end opts_exclus
> ive -----
  - local fwwrapon `word' `width'
  = local fwwrapon
  - if `:length local fwwrapon' {
  = if 0 {
    c_local fwwrapon fwwrapon(`fwwrapon')
  }

```

```

> ----- end _get_diopts.Wra
> pon -----
  - if "`fwwrap'" != "" {
  = if "fwwrap(1)" != "" {
  - FWWrap, `fwwrap'
  = FWWrap, fwwrap(1)

```

```

> ----- begin _get_diopts.FW
> rap -----
  - syntax, fwwrap(integer)
  - if `fwwrap' < 0 {
  = if 1 < 0 {

```

```

        di as err "{p}invalid option {bf:fvwrap(`fvwrap')}); " "nonnega
> tive integer required{p_end}"
        exit 198
    }
}



---


> ----- end _get_diopts.FVW
> rap -----
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts          fvwrap(1)
    - if `K' == 2 {
    = if 2 == 2 {
    - c_local `c_opts' ``options'''
    = c_local options `'''
    - }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95



---


> ----- end _get_diopts.GetDio
> pts -----
    - c_local `name1' ``diopts'''
    = c_local diopts `fvwrap(1)'''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)
    - }
    - local S_`sname' = s(`sname')
    = local S_method = s(method)
    - }
    - local S_`sname' = s(`sname')
    = local S_options = s(options)
    - }
    - sreturn clear
    - GetDiopts DEFAULT



---


> ----- begin _get_diopts.GetDio

```

```

> pts -----
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "
    -----
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""opts"" != "" {
    = while `"" != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    -----
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""opts"" != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels` `nobaselevels`"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells` `noemptycells`"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}

```

```

- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""opts'" != "" {
    = while `"" != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""opts'" != "" {
    = while `"" != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "

> ----- begin opts_exclus

```



```

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus

> ive
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus

> ive
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "

> ----- begin opts_exclus

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if ``allbaselevels'`baselevels' == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if ``emptycells' == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if ``omitted' == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }

```

```

    }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')'
  }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth("`pformat'") > 5 {
      local pformat "%5.3f"
      di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat "`pformat'"
    local pformat `pformat(`pformat')'
  }
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
  }
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
  }
}

```

```

- Wraapon , `fvwraapon'
= Wraapon ,

```

```

> ----- begin _get_diopts.Wra
> pon -----
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
        - exit

```

```

> ----- end opts_exclus
> ive -----
    - local fvwrapon `word' `width'
= local fvwrapon
    - if `:length local fvwrapon' {
= if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }

```

```

> ----- end _get_diopts.Wra
> pon -----
    - if "`fvwrap'" != "" {
= if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev

```

```

> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DEFAULT
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options''
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95


---


> _____ end _get_diopts.GetDio
> pts _____
    - if ``name2'' == "" {
    = if ``options'' == "" {
        GetDiopts DUPS, `options'
    }
    - else {
    - GetDiopts DUPS options, `options'
    = GetDiopts DUPS options,


---


> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive ``allbaselevels' `noallbaselevels''
    = opts_exclusive " "


---


> _____ begin opts_exclus

```

```

> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`allbaselevels' `nbaselevels'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`baselevels' `nbaselevels'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``emptycells' `noemptycells'
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``fvlabel' `nofvlabel'
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```



```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus

```

```

> ive
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

>
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> ive
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "

>
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> ive
    - opts_exclusive "`versus' `compare'"

```

```

= opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'

```

```

= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
}

```

```

- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"
  local sformat "`sformat(`sformat')'"
}
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth("`pformat'") > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat "`pformat'"
  local pformat "`pformat(`pformat')'"
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
  syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
  local levelopt level(`level')
}
- Wraapon , `fvwraapon'
= Wraapon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
  di as err "invalid fvwraapon() option;"
  syntax [, W0rd WIdth]
  exit 198
}
- opts_exclusive "`word' `width'" fvwraapon
= opts_exclusive " " fvwraapon

```

> _____ begin opts_exclus

> ive _____

```

- version 8.2
- args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> _____ end opts_exclus
> ive _____
- local fwwrapon `word' `width'
= local fwwrapon
- if `:length local fwwrapon' {
= if 0 {
    c_local fwwrapon fwwrapon(`fwwrapon')
}

> _____ end _get_diopts.Wra
> pon _____
- if ``fwwrap' != "" {
= if "" != "" {
    FVWrap, `fwwrap'
}
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fwwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
= c_local DUPS
- if `K' == 2 {
= if 2 == 2 {
- c_local `c_opts' ``options'
= c_local options `"'
- }
- sreturn local coding `coding'
= sreturn local coding
- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> _____ end _get_diopts.GetDio
> pts _____
    - c_local `name2' ``options''
    = c_local options ``''
    - }
    - local DUPS : list DUPS - DEFAULT
    - gettoken DUPS : DUPS
    - if ``DUPS'' != "" {
    = if ``'' != "" {
        if (strpos(`DUPS'', "(")) {
        gettoken DUPS : DUPS, parse("(")
        local DUPS `DUPS'()'
        }
        di as err "duplicate {bf:`DUPS'} option not allowed"
        exit 198
    }
    - sreturn clear
    - foreach sname of local snames {
    - sreturn local `sname' = ``S_`sname''
    = sreturn local level = ``95''
    - }
    - sreturn local `sname' = ``S_`sname''
    = sreturn local method = ``noadjust''
    - }
    - sreturn local `sname' = ``S_`sname''
    = sreturn local options = ``fvwrap(1)''
    - }

> _____ end _get_dio
> pts _____
    - local legend coeflegend selegend
    - local legend : list legend & diopts
    - local diopts : list diopts - legend
    - local lstretch lstretch nolstretch
    - local lstretch : list lstretch & diopts
    - local diopts : list diopts - lstretch
    - local markdown markdown
    - local markdown : list markdown & diopts
    - local diopts : list diopts - markdown
    - _get_diopts ignore, `diopts' `cformat' `sformat' `pformat'
    = _get_diopts ignore, fvwrap(1)

> _____ begin _get_dio
> pts _____
    - version 11
    - syntax namelist(max=2) [, *]

```

```

- gettoken name1 namelist : namelist
- gettoken name2 namelist : namelist
- GetDiopts diopts options, `options'
= GetDiopts diopts options, fwrap(1)

```

```

> ----- begin _get_diopts.GetDio
> pts -----
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
local ++n
gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`allbaselevels' `nobaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----

```



```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
        }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n

```

```

        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    end opts_exclus
>
> ive
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive

```

```

- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive ``coding' `compare'
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'`` != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive ``versus' `compare'
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'`` != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive ``abbrev' `noabbrev'
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----
> ----- end opts_exclus
> ive
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells
    - local omitted `omitted' `noomitted'
    = local omitted
    - local lstretch `lstretch' `nolstretch'
    = local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
    = local fvlabel
    - local abbrev `abbrev' `noabbrev'
    = local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
    = if "" == "" {
    - if c(showbaselevels) == "all" {
        local allbaselevels allbaselevels
    }
    - else if c(showbaselevels) == "on" {
        local baselevels baselevels
    }
    - }
    - if "`emptycells'" == "" {
    = if "" == "" {
    - if c(showemptycells) == "off" {
        local emptycells noemptycells

```

```

    }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')"'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')"'
  }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth("`pformat'") > 5 {
      local pformat "%5.3f"
      di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat "`pformat'"
    local pformat `pformat(`pformat')"'
  }
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {

```

```

        syntax namelist(max=2) [, `DIOPTS']
    }
    - if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
        local levelopt level(`level')
    }
    - Wraupon , `fvwraupon'
= Wraupon ,

> _____ begin _get_diopts.Wra
> pon _____
        - capture syntax [, W0rd WIdth]
        - if c(rc) {
            di as err "invalid fvwraupon() option;"
            syntax [, W0rd WIdth]
            exit 198
        }
        - opts_exclusive "`word' `width'" fvwraupon
= opts_exclusive " " fvwraupon

> _____ begin opts_exclus
> ive _____
        - version 8.2
        - args opts optname errcode optslab
        - local opts `opts'
= local opts
        - local n 0
        - while ``opts' != "" {
= while `"' != "" {
            local ++n
            gettoken item`n' opts : opts, bind
        }
        - if `n' < 2 {
= if 0 < 2 {
            - exit

> _____ end opts_exclus
> ive _____
        - local fvwraupon `word' `width'
= local fvwraupon
        - if `:length local fvwraupon' {
= if 0 {
            c_local fvwraupon fvwraupon(`fvwraupon')
        }

> _____ end _get_diopts.Wra

```



```

> pon
    - if "`fvwrap'" != "" {
    = if "fvwrap(1)" != "" {
    - FVWrap, `fvwrap'
    = FVWrap, fvwrap(1)

> _____ begin _get_diopts.FVW
> rap
    - syntax, fvwrap(integer)
    - if `fvwrap' < 0 {
    = if 1 < 0 {
        di as err "{p}invalid option {bf:fvwrap(`fvwrap')}"; " "nonnega
> tive integer required{p_end}"
        exit 198
    }

> _____ end _get_diopts.FVW
> rap
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts          fvwrap(1)
    - if `K' == 2 {
    = if 2 == 2 {
    - c_local `c_opts' ``options''
    = c_local options `''
    - }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95

> _____ end _get_diopts.GetDio
> pts
    - c_local `name1' ``diopts''
    = c_local ignore ``fvwrap(1)''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)

```

```

- }
- local S_`sname' = s(`sname')
= local S_options = s(options)
- }
- local S_`sname' = s(`sname')
= local S_method = s(method)
- }
- sreturn clear
- GetDiopts DEFAULT

```

```

> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "

```

```

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> _____ end opts_exclus
> ive _____

```

```

- opts_exclusive "`allbaselevels' `nobaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`baselevels' `nobaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus

```

```

> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `''` != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `''` != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
}

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells
    - local omitted `omitted' `noomitted'
    = local omitted
    - local lstretch `lstretch' `nolstretch'
    = local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
    = local fvlabel
    - local abbrev `abbrev' `noabbrev'
    = local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
    = if "" == "" {
    - if c(showbaselevels) == "all" {
        local allbaselevels allbaselevels
    }
    - else if c(showbaselevels) == "on" {
        local baselevels baselevels
    }
}

```



```

- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"
  local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth("`pformat'") > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
}

```

```

        sreturn local pformat ``pformat'''
        local pformat `pformat(pformat)'''
    }
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wrapon , `fvwrapon'
= Wrapon ,

> _____ begin _get_diopts.Wra
> pon _____
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive ``word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
        - exit

> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
= local fvwrapon
    - if `:length local fvwrapon' {

```

```

        = if 0 {
            c_local fwrapon fwrapon(`fwrapon`)
        }
    }
}

> _____ end _get_diopts.Wra
> pon _____
    - if "`fwrap'" != "" {
    = if "" != "" {
        FVWrap, `fwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DEFAULT
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options'
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95
}

> _____ end _get_diopts.GetDio
> pts _____
    - if ``name2' == "" {
    = if `'' == "" {
    - GetDiopts DUPS, `options'
    = GetDiopts DUPS,
}

> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell

```

```

> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *)
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____

```

```

- opts_exclusive "`baselevels' `nobaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus

```

```

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus

> ive
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus

> ive
- opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "

> ----- begin opts_exclus

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "
}
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
= opts_exclusive " "
}
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----

```



```

        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}

```

```

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {

```

```

confirm numeric format `cformat'
if fmtwidth("`cformat'") > 9 {
local cformat "%9.0g"
di "{txt}note: invalid cformat(), using default"
}
sreturn local cformat "`cformat'"
local cformat `cformat(`cformat')"'
}
- if `:length local sformat' {
= if 0 {
confirm numeric format `sformat'
if fmtwidth("`sformat'") > 8 {
local sformat "%8.2f"
di "{txt}note: invalid sformat(), using default"
}
sreturn local sformat "`sformat'"
local sformat `sformat(`sformat')"'
}
- if `:length local pformat' {
= if 0 {
confirm numeric format `pformat'
if fmtwidth("`pformat'") > 5 {
local pformat "%5.3f"
di "{txt}note: invalid pformat(), using default"
}
sreturn local pformat "`pformat'"
local pformat `pformat(`pformat')"'
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
local levelopt level(`level')
}
- Wraon , `fvwraon'
= Wraon ,

```

```

> _____ begin _get_diopts.Wra
> pon _____

```

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
di as err "invalid fvwraon() option;"
syntax [, W0rd WIdth]
exit 198

```

```

    }
    - opts_exclusive "`word' `width'" fvwrapon
    = opts_exclusive " " fvwrapon
}
> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
}
> ----- end _get_diopts.Wra
> pon
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DUPS
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options'
    }
    - sreturn local coding `coding'
    = sreturn local coding

```

```

- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> ----- end _get_diopts.GetDio
> pts -----
- }
- else {
  GetDiopts DUPS options, `options'
  c_local `name2' ``options'''
}
- local DUPS : list DUPS - DEFAULT
- gettoken DUPS : DUPS
- if ``DUPS''' != "" {
= if `''' != "" {
  if (strpos(`DUPS'', "(")) {
    gettoken DUPS : DUPS, parse("(")
    local DUPS `DUPS'()'
  }
  di as err "duplicate {bf:`DUPS'} option not allowed"
  exit 198
}
- sreturn clear
- foreach sname of local snames {
- sreturn local `sname' = ``S_`sname''''
= sreturn local level = `95''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local options = `fvwrap(1)''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local method = `noadjust''
- }

```

```

> ----- end _get_dio
> pts -----
- local cformat ``s(cformat)'''
= local cformat `'''
- local sformat ``s(sformat)'''
= local sformat `'''
- local pformat ``s(pformat)'''
= local pformat `'''
- _get_eformopts , eformopts(`options') allowed(__all__) soptions

```

```

= _get_eformopts , eformopts() allowed(__all__) soptions
> _____ begin _get_eformo
> pts _____
    - version 8.0
    - syntax [, eformopts(string asis) soptions ALLOWed(string) ]
    - local 0 , `eformopts'
    = local 0 ,
    - local EFALL NOHR hr NOSHR shr or IRr RRr tr TRatio
    - if "`soptions'" != "" {
    = if "soptions" != "" {
    - syntax [, EForm1(passthru) EForm `EFALL' * ]
    = syntax [, EForm1(passthru) EForm NOHR hr NOSHR shr or IRr RRr tr T
> Ratio * ]
    - local opts ``options'""
    = local opts `""'
    - local 0 , `eform1' `eform' `nohr' `hr' `noshr' `shr' `or' `irr' `r
> rr' `tr' `tratio'
    = local 0 ,
    - }
    - if "`allowed'" == "__all__" {
    = if "__all__" == "__all__" {
    - local allowed `EFALL'
    = local allowed NOHR hr NOSHR shr or IRr RRr tr TRatio
    - }
    - foreach ef of local allowed {
    - capture confirm name `ef'
    = capture confirm name NOHR
    - if _rc {
        di as err "`ef' is not a valid name"
        exit 198
    }
    - local efopts `efopts' `=lower("`ef'")'
    = local efopts nohr
    - }
    - capture confirm name `ef'
    = capture confirm name hr
    - if _rc {
        di as err "`ef' is not a valid name"
        exit 198
    }
    - local efopts `efopts' `=lower("`ef'")'
    = local efopts nohr hr
    - }
    - capture confirm name `ef'
    = capture confirm name NOSHR

```

```

- if _rc {
    di as err "`ef' is not a valid name"
    exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr
- }
- capture confirm name `ef'
= capture confirm name shr
- if _rc {
    di as err "`ef' is not a valid name"
    exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr
- }
- capture confirm name `ef'
= capture confirm name or
- if _rc {
    di as err "`ef' is not a valid name"
    exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or
- }
- capture confirm name `ef'
= capture confirm name IRr
- if _rc {
    di as err "`ef' is not a valid name"
    exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or irr
- }
- capture confirm name `ef'
= capture confirm name RRr
- if _rc {
    di as err "`ef' is not a valid name"
    exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or irr rrr
- }
- capture confirm name `ef'
= capture confirm name tr
- if _rc {

```



```

- }
- if ``eform1``=="Odds ratio" {
= if ``=="Odds ratio" {
  local eform_cons_ti = "Odds"
  local eform_cons_note = "odds ratios"
}
- else if ``eform1``=="Haz. ratio" {
= else if ``=="Haz. ratio" {
  local eform_cons_ti = "Hazard"
  local eform_cons_note = "hazard ratios"
}
- else if ``eform1``=="Time ratio" {
= else if ``=="Time ratio" {
  local eform_cons_ti = "Time"
  local eform_cons_note = "time ratios"
}
- else if ``eform1``=="IRR" {
= else if ``=="IRR" {
  local eform_cons_ti = "Inc. rate"
  local eform_cons_note = "incidence-rate ratios"
  local eform_cons_ti2 = "incidence rate"
}
- else if ``eform1``=="Risk ratio" {
= else if ``=="Risk ratio" {
  local eform_cons_ti = "Risk"
  local eform_cons_note = "risk ratios"
}
- else if ``eform1``=="Hlth ratio" {
= else if ``=="Hlth ratio" {
  local eform_cons_ti = "Health"
  local eform_cons_note = "health ratios"
}
- local k : list sizeof eform
- if `k' {
= if 0 {
  opts_exclusive "`eform'"
  if `:length local eform1' {
    opts_exclusive "eform() `eform'"
  }
}
- if `k' {
= if 0 {
  if ("`eform'"=="eform") local eform1 = "exp(b)"
  else if ("`eform'"=="hr") {
    local eform1 = "Haz. ratio"
    local eform_cons_ti = "Hazard"

```

```

local eform_cons_note = "hazard ratios"
}
else if ("`eform'"=="shr") {
local eform1 = "SHR"
local eform_cons_ti = "Subhazard"
local eform_cons_note = "subhazard ratios"
}
else if ("`eform'"=="tr") {
local eform1 = "Time ratio"
local eform_cons_ti = "Time"
local eform_cons_note = "time ratios"
}
else if ("`eform'"=="tratio") {
local eform1 = "Time ratio"
local eform_cons_ti = "Time"
local eform_cons_note = "time ratios"
}
else if ("`eform'"=="or") {
local eform1 = "Odds ratio"
if ("`e(cmd)'"=="asclogit" | "`e(cmd)'"=="cmclogit" | "`e(cmd)'"==
> "cmmixlogit" | "`e(cmd)'"=="cmxtmixlogit" ) {
local eform_cons_ti = "Rel. risk"
local eform_cons_note = "relative-risk ratios"
local eform_cons_ti2 = "relative risk"
}
else {
local eform_cons_ti = "Odds"
local eform_cons_note = "odds ratios"
}
}
else if ("`eform'"=="irr") {
local eform1 = "IRR"
local eform_cons_ti = "Inc. rate"
local eform_cons_note = "incidence-rate ratios"
local eform_cons_ti2 = "incidence rate"
}
else if ("`eform'"=="rrr") {
local eform1 = "RRR"
local eform_cons_ti = "Rel. risk"
local eform_cons_note = "relative-risk ratios"
local eform_cons_ti2 = "relative risk"
}
else if ("`eform'"!="nohr") & ("`eform'"!="noshr") {
local eform1 = upper("`eform'")
}
}

```

```

- sreturn clear
- sreturn local options ``opts'''
= sreturn local options `'''
- sreturn local opt `eform'
= sreturn local opt
- sreturn local str `eform1'
= sreturn local str
- if ``eform1''' != "" {
= if `''' != "" {
    sreturn local eform eform(`eform1''')
  }
- else sreturn local eform ""
- sreturn local eform_cons_ti `eform_cons_ti'
= sreturn local eform_cons_ti
- sreturn local eform_cons_note `eform_cons_note'
= sreturn local eform_cons_note
- sreturn local eform_cons_ti2 `eform_cons_ti2'
= sreturn local eform_cons_ti2

```

```

> _____ end _get_eformo
> pts _____
- local tr_item = ``s(eform_cons_note)'''
= local tr_item = `'''
- if inlist(`bmatrix', "", "e(b)") & "e(k_eform)'" == "0" {
= if inlist("", "", "e(b)") & "" == "0" {
    local eform
  }
- else local eform ``s(str)'''
= else local eform `'''
- local eform_cons_ti ``s(eform_cons_ti)'''
= local eform_cons_ti `'''
- local eform_cons_ti2 ``s(eform_cons_ti2)'''
= local eform_cons_ti2 `'''
- if ("e(consonly)'"!="1" | ``eform_cons_ti'"==") {
= if ("!"="1" | `'"==") {
- local eformdi ``s(str)'''
= local eformdi `'''
- }
- else {
    local eformdi ``eform_cons_ti'''
    local tr_item = cond(missing(`eform_cons_ti2'), strlower(`eform_
> cons_ti'), ``eform_cons_ti2')
  }
- local coefttl = cond(`eform'"==`'', ``coeftitle'', ``eformdi'
> '')
= local coefttl = cond(`'"==`'', `'', `')

```

```

- local options ``s(options)'''
= local options `'''
- _get_diparmopts, diparmopts(`options') level(`level')
= _get_diparmopts, diparmopts() level(95)

```

```

> _____ begin _get_diparmo
> pts _____
- if _caller() < 8.2 {
  _get_diparmopts_8 `0'
  exit
}
- version 9
- local version : di "version " string(_caller()) ":"
- syntax [, diparmopts(string asis) EXECute soptions bottom plus SYN
> TAXonly * ]
- local goptions `options'
= local goptions level(95)
- SpliceGlobals junk , `goptions'
= SpliceGlobals junk , level(95)

```

```

> _____ begin _get_diparmopts.SpliceGlob
> als _____
- syntax name(id="macro name" name=c_diparm) [, diparm(string asis
> ) Level(cilevel) dof(passthru) syntaxonly ]
- if ``dof''' == "" & ``e(df_r)''' != "" {
= if `''' == "" & `217''' != "" {
- capture confirm integer number `e(df_r)''
= capture confirm integer number 217
- if !c(rc) {
- local gdof dof(`e(df_r)')
= local gdof dof(217)
- }
- capture
- }
- else local gdof ``dof'''
= else local gdof `'''
- local glevel `level'
= local glevel 95
- local lopt level(`glevel')
= local lopt level(95)
- if (``diparm''' == "") exit
= if (``''' == "") exit

```

```

> _____ end _get_diparmopts.SpliceGlob
> als _____
- local 0 `", `diparmopts'''

```

```

= local 0 `", ""
- local diparmopts
- local k 0
- syntax [, diparm(string asis) * ]
- while ``diparm''' != "" {
= while `''' != "" {
    local ++k
    local single 1
    local after
    capture _on_colon_parse `diparm'
    if !c(rc) {
        local single 0
        local diparm ``s(before)'''
        local after ``s(after)'''
    }
    capture
    SpliceGlobals diparm , diparm(`diparm') `syntaxonly' `goptions'
    local diparm`k' ``diparm'''
    local diparmopts ``diparmopts' : `diparm'''
    if `single' | ``:list retok after''' == "" {
        local 0 `", `options'''
        syntax [, diparm(string asis) * ]
    }
    else {
        local diparm ``after'''
    }
}
- if ``soptions''' == "" {
= if `''' == "" {
- syntax [, diparm(string asis) ]
- }
- if ("`execute" == "") {
= if ("" == "") {
- sreturn clear
- sreturn local options ``options'''
= sreturn local options `'''
- sreturn local diparm ``diparmopts'''
= sreturn local diparm `'''
- forval i = 1/`k' {
= forval i = 1/0 {
    sreturn local diparm`i' ``diparm`i'''
}
- sreturn local k `k'
= sreturn local k 0
- exit

```

```

> _____ end _get_diparmo
> pts _____
- local NODIPARM : length local nodiparm
- if `NODIPARM' {
= if 8 {
- local options
- }
- local GTOPTS parse(":") bind quotes
- local k 0
- if !`NODIPARM' {
= if !8 {
  if ``e(diparm)'' != "" {
    local i 0
    local diparm ``e(diparm)''
  }
  else if ``e(diparm1)'' != "" {
    local i 1
    local diparm ``e(diparm1)''
  }
  while `:length local diparm' {
    gettoken diparm rest : diparm, `GTOPTS'
    while `:length local rest' {
      local ++k
      local diparm`k' : copy local diparm
      gettoken COLON rest : rest, `GTOPTS'
      gettoken diparm rest : rest, `GTOPTS'
    }
    local ++k
    local diparm`k' : copy local diparm
    local ++i
    local diparm ``e(diparm`i)''
  }
}
- if `:length local options' {
= if 0 {
  local 0 `", `options''
  syntax [, diparm(string asis) *]
  if `k' & `:length local diparm' {
    local ++k
    local diparm`k' __sep__
  }
  while `:length local diparm' {
    gettoken diparm rest : diparm, `GTOPTS'
    while `:length local rest' {
      local ++k
      local diparm`k' : copy local diparm

```

```

    gettoken COLON rest : rest, `GTPTS'
    gettoken diparm rest : rest, `GTPTS'
  }
  local ++k
  local diparm`k' : copy local diparm
  local 0 `", `options'""
  syntax [, diparm(string asis) *]
}
}
- if "`markdown'" != "" {
= if "" != "" {
  local lsize = c(linesize)
  set linesize 255
}
- `vv' mata: _coef_table()
= version 11, missing: mata: _coef_table()
}

> ----- begin _get_dio
> pts -----
  - version 11
  - syntax namelist(max=2) [, *]
  - gettoken name1 namelist : namelist
  - gettoken name2 namelist : namelist
  - GetDiopts diopts options, `options'
  = GetDiopts diopts options, fwrap(1)
}

> ----- begin _get_diopts.GetDio
> pts -----
  - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
  - syntax namelist(max=2) [, `DIOPTS' *]
  = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
  - opts_exclusive "`allbaselevels' `noallbaselevels'"
  = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nbaselevels'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nbaselevels'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```



```

- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n

```

```

        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    end opts_exclus
>
> ive
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive

```

```

- opts_exclusive "`nocl' `nopvalues'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``abbrev' `noabbrev'
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells

```

```

- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {

```

```

= if 0 {
    confirm numeric format `sformat'
    if fmtwidth(`""sformat'") > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat `""sformat'
    local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth(`""pformat'") > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat `""pformat'
    local pformat `pformat(`pformat')'
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraupon , `fvwraupon'
= Wraupon ,

```

```

> ----- begin _get_diopts.Wra
> pon -----

```

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
    di as err "invalid fvwrapon() option;"
    syntax [, W0rd WIdth]
    exit 198
}
- opts_exclusive ""`word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

```

> ----- begin opts_exclus
> ive -----

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'

```

```

        = local opts
        - local n 0
        - while ``opts'' != "" {
        = while `'' != "" {
            local ++n
            gettoken item`n' opts : opts, bind
        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }

> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }

> _____ end _get_diopts.Wra
> pon _____
    - if ``fvwrap'' != "" {
    = if "fvwrap(1)" != "" {
    - FVWrap, `fvwrap'
    = FVWrap, fvwrap(1)

> _____ begin _get_diopts.FVW
> rap _____
    - syntax, fvwrap(integer)
    - if `fvwrap' < 0 {
    = if 1 < 0 {
        di as err "{p}invalid option {bf:fvwrap(`fvwrap')}"; " "nonnega
> tive integer required{p_end}"
        exit 198
    }

> _____ end _get_diopts.FVW
> rap _____
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts          fvwrap(1)
    - if `K' == 2 {

```



```

= if 2 == 2 {
- c_local `c_opts' ``options'''
= c_local options `'''
- }
- sreturn local coding `coding'
= sreturn local coding
- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95
}

> ----- end _get_diopts.GetDio
> pts -----
- c_local `name1' ``diopts'''
= c_local __00000C ``fvwrap(1)'''
- local snames : s(macros)
- foreach sname of local snames {
- local S_`sname' = s(`sname')
= local S_level = s(level)
- }
- local S_`sname' = s(`sname')
= local S_k = s(k)
- }
- sreturn clear
- GetDiopts DEFAULT
}

> ----- begin _get_diopts.GetDio
> pts -----
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
- opts_exclusive ``allbaselevels' `noallbaselevels'''
= opts_exclusive " "
}

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`omitted` `noomitted`"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch` `nolstretch`"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}

```

```

- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____

```

```

- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'"" != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'"" != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels

```

```

- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')'

```



```

    }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth("`pformat'") > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat "`pformat'"
    local pformat `pformat(`pformat')'
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wrapon , `fvwrapon'
= Wrapon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
    di as err "invalid fvwrapon() option;"
    syntax [, W0rd WIdth]
    exit 198
}
- opts_exclusive "`word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

> _____ begin opts_exclus

> ive _____

- version 8.2

```

- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> _____ end opts_exclus
> ive _____
- local fvwrapon `word' `width'
= local fvwrapon
- if `:length local fvwrapon' {
= if 0 {
    c_local fvwrapon fvwrapon(`fvwrapon')
}

> _____ end _get_diopts.Wra
> pon _____
- if "`fvwrap'" != "" {
= if "" != "" {
    FVWrap, `fvwrap'
}
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
= c_local DEFAULT
- if `K' == 2 {
= if 1 == 2 {
    c_local `c_opts' ``options'
}
- sreturn local coding `coding'
= sreturn local coding
- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> _____ end _get_diopts.GetDio
> pts _____
    - if ``name2'' == '' {
    = if ``'' == '' {
    - GetDiopts DUPS, `options'
    = GetDiopts DUPS,
    _____

> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITttd OMITttd NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITttd OMITttd NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive ``allbaselevels' `noallbaselevels''
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != '' {
    = while ``'' != '' {
        local ++n
        gettoken item`n' opts : opts, bind
        }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive ``allbaselevels' `nobaselevels''
    = opts_exclusive " "
    _____

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch` `nolstretch`"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport` `fullcnsreport`"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}

```

```

- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

```

```

> _____ end opts_exclus
> ive _____
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____

```



```

- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {

```

```

- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')"'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')"'
  }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth("`pformat'") > 5 {
      local pformat "%5.3f"
      di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat "`pformat'"
    local pformat `pformat(`pformat')"'
  }
}

```

```

- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraapon , `fvwraapon'
= Wraapon ,

> ----- begin _get_diopts.Wra
> pon -----
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwraapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwraapon
    = opts_exclusive " " fvwraapon

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit

> ----- end opts_exclus
> ive -----
    - local fvwraapon `word' `width'
    = local fvwraapon
    - if `:length local fvwraapon' {
    = if 0 {
        c_local fvwraapon fvwraapon(`fvwraapon')
    }

```

```

> _____ end _get_diopts.Wra
> pon _____
    - if "`fvwrap'" != "" {
    = if "" != "" {
      FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DUPS
    - if `K' == 2 {
    = if 1 == 2 {
      c_local `c_opts' ``options''
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95
  _____
> _____ end _get_diopts.GetDio
> pts _____
    - }
    - else {
      GetDiopts DUPS options, `options'
      c_local `name2' ``options''
    }
    - local DUPS : list DUPS - DEFAULT
    - gettoken DUPS : DUPS
    - if ``DUPS'' != "" {
    = if ``' != "" {
      if (strpos(`DUPS', "(")) {
        gettoken DUPS : DUPS, parse("(")
        local DUPS `DUPS'()'
      }
      di as err "duplicate {bf:`DUPS'} option not allowed"
      exit 198
    }
    - sreturn clear
    - foreach sname of local snames {
    - sreturn local `sname' = ``S_`sname''

```

```

= sreturn local level = `95'
- }
- sreturn local `sname' = `S_`sname'
= sreturn local k = `0'
- }

> _____ end _get_dio
> pts _____
- if "`markdown'" != "" {
= if "" != "" {
  set linesize `lsize'
}
- if "`eqselect'" == "" & "`eformnote'" != "noeformnote" {
= if "" == "" & "" != "noeformnote" {
- if "`eformdi'" != "" & ("`e(cmd)'" != "gnbreg" | "`e(prefix)'" == "svy")
> {
= if `""' != "" & ("regress" != "gnbreg" | "" == "svy") {
  local k_eform `r(k_eform)'
  if ("`k_eform'" == "") {
    local k_eform = 1
  }
  local k_eq `e(k_eq)'
  if ("`k_eq'" == "") {
    local k_eq = 1
  }
  if (`k_eform' < `k_eq' & `k_eform') {
    if ("`e(cmd2)'" == "stintreg" | "`e(cmd2)'" == "streg") & ("`e(cmd)'"
> " == "weibull" | "`e(cmd)'" == "gompertz") {
    }
    else {
      _eform_multeq_note efnote : `k_eform' ``tr_item'
      di as txt "`efnote'"
    }
  }
}
- if ("`eformdi'" != "" & "`e(noconstant)'" == "0" & "`eform_cons_ti'"
> != "" & "`e(consonly)'" != "1") {
= if (`""' != "" & "" == "0" & `""' != "" & "" != "1") {
  if ("`e(cmd)'" == "asclogit" | "`e(cmd)'" == "cmclogit" | "`e(cmd)'" == "c
> mmixlogit" | "`e(cmd)'" == "cmxtmixlogit" ) {
    local asvar = "`e(ranvars)'"`e(fixvars)'"`e(indvars)'" != ""
    if "`e(casevars)'" != "" & `asvar' {
      local note1 "Exponentiated coefficients represent odds ratios for "
      local note2 "alternative-specific variables (first equation) and "
      local note3 "{help j_cmrisk##_new:relative-risk ratios} "
      local note4 "for case-specific variables."

```

```

di as txt "{p 0 6 2}Note: `note1``note2``note3``note4'{p_end}"
}
}
if "`eform_cons_ti'" == "Inc. rate" {
local eform_cons_note "incidence rate"
}
else if "`eform_cons_ti'" == "Rel. risk" {
local eform_cons_note "relative risk"
}
else if "`eform_cons_ti'" == "Health" {
local eform_cons_note "health (probability of no disease)"
}
else {
local eform_cons_note = strlower("`eform_cons_ti'")
}
local eform_cons_note "baseline `eform_cons_note'"
if ("`e(cmd2)'"=="") {
local cmd `e(cmd)'
}
else {
local cmd `e(cmd2)'
}
local f2 = udsustr("`cmd'",1,2)
local is_re = ("`f2'"=="xt" | "`f2'"=="me")
if `is_re' {
local extranote " (conditional on zero random effects)"
}
di as txt "{p 0 6 2}Note: {bf:_cons} estimates " "`eform_cons_note'`
> extranote'" _c
if ("`e(cmd)'"=="mlogit" | "`e(cmd)'"=="asclogit" | "`e(cmd)'"=="cmc
> logit" | "`e(cmd)'"=="cmmixlogit" | "`e(cmd)'"=="cmxtmixlogit" ) {
di as txt " for each outcome" _c
}
di as txt ".{p_end}"
}
- }
- return add
- if !`keepmc' {
= if !0 {
- return local mcmethod
- return local mctitle
- }
}

> _____ end _coef_ta
> ble _____

```

```

> _____ end _ereturn_disp
> lay _____
  - }
  - if e(N)>=. {exit 459}
  - if _rc {exit _rc}
  - capture drop `eta'
  = capture drop __000004
  - capture drop `mu'
  = capture drop __000005
  - predict double `eta', xb
  = predict double __000004, xb
  _____

> _____ begin pred
> ict _____
  - version 8.2, missing
  - if "`e(cmd)'" == "rocreg" & "`e(predict)'" == "" {
  = if "regress" == "rocreg" & "regres_p" == "" {
    di as err "predict not allowed after nonparametric ROC"
    exit 198
  }
  - if "`e(mi)'"!="> & "`e(b)'"!="matrix" {
  = if ""!="> & "matrix"!="matrix" {
    error 321
  }
  - if _caller()<=5 | "`e(predict)'"=="> {
  = if _caller()<=5 | "regres_p"=="> {
    _predict `0'
  }
  - else {
  - local v : display string(_caller())
  - version `v', missing
  = version 7, missing
  - `e(predict)' `0'
  = regres_p double __000004, xb
  _____

> _____ begin regre
> s_p _____
  - local vv : display "version " string(_caller()) ", missing:"
  - version 6, missing
  - syntax [anything] [if] [in] [, SCores * ]
  - if ``scores'" != "" {
  = if `""' != "" {
    GenScores `0'
    exit
  }
  - local xopts COVratio DFBeta(string) DFIIts E(string) Hat Leverage Pr(

```

```

> string) Welsch YStar(string)
  - if "`e(prefix)'" != "" {
= if "" != "" {
  local svyopts Residuals Cooksd RSTandard RSTudent STDF STDR
}
  - else if "`e(vcetype)'" == "Robust" {
= else if "" == "Robust" {
  local svyopts Cooksd RSTandard RSTudent STDF STDR
}
  - local myopts `xopts' `svyopts'
= local myopts COVratio DFBeta(string) DFIts E(string) Hat Leverage Pr
> (string) Welsch YStar(string)
  - if "`e(prefix)'" != "" | "`e(vcetype)'" == "Robust" {
= if "" != "" | "" == "Robust" {
  if "`e(vcetype)'" == "Robust" {
    local msg after robust estimation
  }
  else local msg after `e(prefix)' estimation
  syntax [anything] [if] [in] [, `myopts' *]
  if "`dfbeta'" != "" {
    local dfbopt dfbeta(`dfbeta')
  }
  if "`e(prefix)'" == "svy" | "`e(vcetype)'" == "Robust" {
    if "`e'" != "" {
      local eopt e(`e')
    }
    if "`pr'" != "" {
      local propt pr(`pr')
    }
    if "`ystar'" != "" {
      local ysopt ystar(`ystar')
    }
  }
  _prefix_noption `msg', `cooksd' `covrati' `dfbopt' `dfits' `eopt'
> `hat' `leverage' `propt' `rstandard' `rstudent' `stdf' `stdr' `welsch' `ysopt
> '
  }
  - _pred_se "`myopts'" `0'
= _pred_se "COVratio DFBeta(string) DFIts E(string) Hat Leverage Pr(st
> ring) Welsch YStar(string)" double __000004, xb

_____  

> _____ begin _pred
> _se _____
  - version 6.0, missing
  - sret clear
  - gettoken ouser 0 : 0

```



```

- local orig ``0''
= local orig `` double __000004, xb''
- gettoken varn 0 : 0, parse(" ,[")
- gettoken nxt : 0, parse(" ,[")
- if !(```nxt'''=="" | ``nxt'''=="if" | ``nxt'''=="in" | ``nxt'''
> ==",") {
= if !(``__000004'''=="" | ``__000004'''=="if" | ``__000004'''=="in" |
> ``__000004'''==",") {
- local typ `varn'
= local typ double
- gettoken varn 0 : 0, parse(" ,[")
- }
- syntax [if] [in] [, `ouser' CONSTANT(varname numeric) noOFFSET *]
= syntax [if] [in] [, COVratio DFBeta(string) DFIIts E(string) Hat Le
> verage Pr(string) Welsch YStar(string) CONSTANT(varname numeric) noOFFSET *]
- if ``options''' != "" {
= if ``xb''' != "" {
- _predict `orig'
= _predict double __000004, xb
- sret local done 1
- exit
}

> _____ end _pred
> _se _____
- if `s(done)' { exit }
= if 1 { exit }

> _____ end regre
> s_p _____
- }

> _____ end pred
> ict _____
- $SGLM_L 1 `eta' `mu'
= glim_l03 1 __000004 __000005

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
global SGLM_lt "Log"
if "$SGLM_m" == "1" {
global SGLM_lf "ln(u)"
}

```

```

        else global SGLM_lf "ln(u/$SGLM_m)"
        exit
    }
- if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
}
- if `todo' == 1 {
= if 1 == 1 {
    - gen double `mu' = $SGLM_m*exp(`eta')
= gen double __000005 = 1*exp(__000004)
- exit
}

> _____ end glim_
> l03 _____
    - $SGLM_mu `mu'
= glim_mu 0 . __000005
}

> _____ begin glim
> _mu _____
    - version 7
    - args min max mu
    - tempname eps
    - scalar `eps' = 1e-8
= scalar __00000C = 1e-8
    - replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __000005 = 0+__00000C if __000005<=0+__00000C
    - replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __000005 = .-__00000C if __000005>=.-__00000C
}

> _____ end glim
> _mu _____
    - mat `b0' = e(b)
= mat __00000B = e(b)
    - _post_vce_rank, checksize
}

> _____ begin _post_vce_r
> ank _____
    - syntax, [Checksize]
    - if "`checksize'" != "" {
= if "checksize" != "" {
    - tempname V
    - capture matrix `V' = e(V)
= capture matrix __00000C = e(V)
    - if _rc {

```

```

        exit
    }
    - local cols = colsof(`V')
    = local cols = colsof(__00000C)
    - if `cols' == 0 {
    = if 19 == 0 {
        exit
    }
    - }
    - tempname V Vi rank
    - mat `V' = e(V)
    = mat __00000D = e(V)
    - mat `Vi' = invsym(`V')
    = mat __00000E = invsym(__00000D)
    - sca `rank' = rowsof(`V') - diag0cnt(`Vi')
    = sca __00000F = rowsof(__00000D) - diag0cnt(__00000E)
    - mata:st_numscalar("e(rank)", st_numscalar("`rank'"))
    = mata:st_numscalar("e(rank)", st_numscalar("__00000F"))

```

```

> _____ end _post_vce_r
> ank _____
    - local k = e(rank)
    - local from "`b0'"
    = local from "__00000B"
    - replace `z' = sum(`wt'*(`y'-'mu')^2/`v')
    = replace __000008 = sum(__000003*(piq_timing_max-__000005)^2/__000007)
    - global SGLM_ph = `z'[_N]/(`nobs'-'k')
    = global SGLM_ph = __000008[_N]/(217-14)
    - }
    - }
    - if "`from'" != "" & "`search'" == "" {
    = if "__00000B" != "" & "" == "" {
        - local initopt `"init(`from') search(off)'"
        = local initopt `"init(__00000B) search(off)'"
        - }
        - else local initopt `"search(on) maxfeas(100)'"
        - if `reg' & "`offopt'FROM_ORIG'" == "" {
        = if 0 & `"' == "" {
            local iterate 0
            local nowarn "nowarning"
        }
        - local initopt "`initopt' iter(`iterate') `nowarn' "
        = local initopt "init(__00000B) search(off) iter(300) "
        - if "`estscale'" != "" & "$SGLM_V" == "glim_v1" {
        = if "" != "" & "glim_v6" == "glim_v1" {
            local MM d0

```

```

mat `b0' = e(b), $SGLM_ph
local from "`b0'"
local initopt `"init(`from', copy) search(off)'"
}
- if ``scale'""=="" {
= if `""'=="" {
- local scale `scale1'
= local scale 1
- local cd
- }
- else if `scale'==0 {
= else if 1==0 {
if `scale1' {
local cd "square root of Pearson"
local cd "`cd' X2-based dispersion"
}
}
- else {
if !`scale1' | (`scale1' & `scale'!=1) {
local cd "dispersion equal to square"
local cd "`cd' root of `scale'"
}
}
- if `scale1' == 1 & `scale' != `scale1' {
= if 1 == 1 & 1 != 1 {
global SGLM_s1 = `scale1'
local fixme 1
}
- else {
- global SGLM_s1 = `scale'
= global SGLM_s1 = 1
- local fixme 0
- }
- tempname newdev
- if "`opg'`unbiased'" != "" {
= if "" != "" {
if "`scopt'" == "" {
tempvar scvar
local scopt score(`scvar')
}
if "`unbiased'" != "" {
local cll "`clopt'"
local robust
local clopt
}
}
}

```

```

- if (!`iterate`) { `iflog' di }
= if (!300) { * di }
- global GLIST : all globals "SGLM_*"
- if "`estscale'" != "" & "$SGLM_V" == "glim_v1" {
= if "" != "" & "glim_v6" == "glim_v1" {
    local sigma2 /sigma2
    local glim_lf glim_lf_scale
    global SGLM_estscale 1
  }
- else {
- local glim_lf glim_lf
- global SGLM_estscale 0
- }
- `VV' ml model `MM' `glim_lf' (`yname': `y' = `xvars', `constant' `offopt
> ') `sigma2' [`weight'`exp'] if `touse', collinear missing max nooutput nopre
> serve `mlopts' title(Generalized linear models) `scopt' `robust' `clopt' `ml
> log' `initopt' `trace' `negh' `moptobj'
    = version 18.5, missing: ml model e2 glim_lf (piq_timing_max: piq_timing_m
> ax = rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open
> 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnk
> crdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_
> early vint_mid, ) [] if __000002, collinear missing max nooutput nopreserv
> e title(Generalized linear models) robust nolog init(__00000B) search(off
> ) iter(300) negh

-----
> ----- begin
> ml -----
    - if _caller()>=11 {
    - mopt `0'
    = mopt model e2 glim_lf (piq_timing_max: piq_timing_max = rule_g rule_it
> r estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb
> _authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.
> bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid, )
> [] if __000002, collinear missing max nooutput nopreserve title(Generalize
> d linear models) robust nolog init(__00000B) search(off) iter(300) negh
>

-----
> ----- begin m
> opt -----
    - local vv : di "version " string(_caller()) ":"
    - version 12
    - gettoken subcmd 0 : 0, parse(" ,")
    - local lcmd : length local subcmd
    - if "`subcmd'" == "clear" {
    = if "model" == "clear" {
      Clear

```

```

    exit
}
- if ``subcmd`` == "hold" {
= if "model" == "hold" {
    ml_hold `0'
    exit
}
- if ``subcmd`` == "unhold" {
= if "model" == "unhold" {
    ml_unhold `0'
    exit
}
- if "subcmd" == "score" {
= if "model" == "score" {
    if _caller() > 12 {
        `vv' Score `0'
    }
    else {
        Score `0'
    }
    exit
}
- if "subcmd" == "count" {
= if "model" == "count" {
    Count `0'
    exit
}
- if "subcmd" == "trace" {
= if "model" == "trace" {
    Trace `0'
    exit
}
- if "subcmd" == bsubstr("graph",1,max(2,`lcmd')) {
= if "model" == bsubstr("graph",1,max(2,5)) {
    Graph `0'
    exit
}
- if "subcmd" == "init" {
= if "model" == "init" {
    Interactive 0
    `vv' Init `0'
    exit
}
- if "subcmd" == bsubstr("maximize",1,max(3,`lcmd')) {
= if "model" == bsubstr("maximize",1,max(3,5)) {
    Interactive 1

```

```

`vv' MaxMin maximize `0'
exit
}
- if "`subcmd'" == bsubstr("minimize",1,max(3,`lcmd')) {
= if "model" == bsubstr("minimize",1,max(3,5)) {
Interactive 1
`vv' MaxMin minimize `0'
exit
}
- if "`subcmd'" == bsubstr("display",1,max(2,`lcmd')) {
= if "model" == bsubstr("display",1,max(2,5)) {
Display `0'
exit
}
- if "`subcmd'" == bsubstr("footnote",1,max(4,`lcmd')) {
= if "model" == bsubstr("footnote",1,max(4,5)) {
ml_footnote `0'
exit
}
- if "`subcmd'" == "," {
= if "model" == "," {
Display, `0'
exit
}
- if "`subcmd'" == bsubstr("model",1,max(3,`lcmd')) {
= if "model" == bsubstr("model",1,max(3,5)) {
- `vv' capture noi Model `0'
= version 18.5: capture noi Model e2 glim_lf (piq_timing_max: piq_tim
> ing_max = rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b
> .open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#
> 1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit
> vint_early vint_mid, ) [] if __000002, collinear missing max nooutput nopr
> eserve title(Generalized linear models) robust nolog init(__00000B) searc
> h(off) iter(300) negh

> _____ begin mopt.Mo
> del _____
- local version = string(_caller())
- local vv : di "version `version':"
= local vv : di "version 18.5:"
- version 12
- syntax anything(id="model" name=spec) [if] [in] [fw aw pw iw] [, B
> RACKET COLLinear MISSing CONTinue lf0(string) INIT(string) CONSTraints(strin
> g) noCNSNOTEs VCE(passthru) NEGH NOPREserve TECHnique(string) ITERID(string)
> CRITtype(string) OBS(numlist max=1 >0) TITle(string) WALDtest(integer -1) S
> VY SUBpop(string) SRSSubpop noSVYadjust SCore(passthru) noOUTput noSCVARS GR

```

```

> OUP(varname) ITERPROLOG(string) DERIVPROLOG(string) DERIVH(string) DERIVSCAL
> E(string) DERIV(string) nrderiv GNWMATrix(string) KAUXiliary(numlist int >=0
> ) MAXimize MINimize USERinfo(namelist) PUPDATED Robust CLuster(passthru) mop
> tobj(string) USEVIEWS * ]
    - if ``moptobj''' == "" {
    = if `''' == "" {
    - Clear
    _____
> _____ begin mopt.Cl
> ear _____
    - syntax [, NOTANOPTION]
    - capture drop $ML_w
    = capture drop
    - capture drop $ML_samp
    = capture drop
    - capture drop $ML_sample
    = capture drop
    - capture drop $ML_subv
    = capture drop
    - capture drop $ML_grp
    = capture drop
    - mata: Mopt_drop_external()
    - macro drop ML_*
    _____
> _____ end mopt.Cl
> ear _____
    - }
    - local maxmin `minimize' `maximize'
    = local maxmin maximize
    - opts_exclusive ``maxmin''
    = opts_exclusive "maximize"
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts maximize
    - local n 0
    - while ``opts''' != "" {
    = while `"maximize"' != "" {
    - local ++n
    - gettoken item`n' opts : opts, bind
    = gettoken item1 opts : opts, bind
    - }
    - while ``opts''' != "" {

```



```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 1 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - if `:length local maxmin` {
= if 8 {
    - local interactive 0
    - }
    - else local interactive 1
    - _get_diparmopts, diparmopts(`options`) soptions syntaxonly
= _get_diparmopts, diparmopts(nolog search(off) iter(300)) soptions
> syntaxonly
    -----
> ----- begin _get_diparmo
> pts -----
    - if _caller() < 8.2 {
        _get_diparmopts_8 `0`
        exit
    }
    - version 9
    - local version : di "version " string(_caller()) ":"
    - syntax [, diparmopts(string asis) EXECute soptions bottom plus S
> YNTAXonly * ]
    - local goptions `options`
    = local goptions
    - SpliceGlobals junk , `goptions`
    = SpliceGlobals junk ,
    -----
> ----- begin _get_diparmopts.SpliceGlob
> als -----
    - syntax name(id="macro name" name=c_diparm) [, diparm(string as
> is) Level(cilevel) dof(passthru) syntaxonly ]
    - if ``dof`` == "" & ``e(df_r)`` != "" {
= if `""` == "" & ``217`` != "" {
    - capture confirm integer number `e(df_r)`
= capture confirm integer number 217
    - if !c(rc) {
    - local gdof dof(`e(df_r)`)
= local gdof dof(217)
    - }

```

```

- capture
- }
- else local gdof ``dof'''
= else local gdof `'''
- local glevel `level'
= local glevel 95
- local lopt level(`glevel')
= local lopt level(95)
- if (``diparm''' == "") exit
= if (`''' == "") exit

```

```

> _____ end _get_diparmopts.SpliceGlob
> als _____
- local 0 `", `diparmopts'''
= local 0 `", nolog search(off) iter(300)'''
- local diparmopts
- local k 0
- syntax [, diparm(string asis) * ]
- while ``diparm''' != "" {
= while `''' != "" {
    local ++k
    local single 1
    local after
    capture _on_colon_parse `diparm'
    if !c(rc) {
        local single 0
        local diparm ``s(before)'''
        local after ``s(after)'''
    }
    capture
    SpliceGlobals diparm , diparm(`diparm') `syntaxononly' `goptions'
    local diparm`k' ``diparm'''
    local diparmopts ``diparmopts' : `diparm'''
    if `single' | ``:list retok after''' == "" {
        local 0 `", `options'''
        syntax [, diparm(string asis) * ]
    }
    else {
        local diparm ``after'''
    }
}
- if ``soptions''' == "" {
= if `soptions''' == "" {
    syntax [, diparm(string asis) ]
}
- if (``execute''' == "") {

```

```

= if ("" == "") {
- sreturn clear
- sreturn local options ``options'''
= sreturn local options ``nolog search(off) iter(300)'''
- sreturn local diparm ``diparmopts'''
= sreturn local diparm ``'''
- forval i = 1/`k' {
= forval i = 1/0 {
    sreturn local diparm`i' ``diparm`i''''
  }
- sreturn local k `k'
= sreturn local k 0
- exit

```

```

> _____ end _get_diparmo
> pts _____
- forval k = 1/`s(k)' {
= forval k = 1/0 {
    local diparm`k' ``s(diparm`k')'''
  }
- local ndiparm `s(k)'
= local ndiparm 0
- local options ``s(options)'''
= local options ``nolog search(off) iter(300)'''
- if `interactive' & `:length local options' {
= if 0 & 27 {
    local 0 , `options'
    syntax [, NOTANOPTION]
  }
- if `:length local nopreserve' | `:length local score' {
= if 10 | 0 {
- global ML_preserve no
- }
- else global ML_preserve yes
- gettoken evaltype spec : spec
- confirm name `evaltype'
= confirm name e2
- gettoken evaluator spec : spec
- global ML_evaltype : copy local evaltype
- if `:length local srssubpop' {
= if 0 {
    local 0 , srssubpop
    syntax [, NONOPTION]
    exit 198
  }
- if `:length local subpop' {

```

```

= if 0 {
  local svy svy
}
- if `:length local svy' {
= if 0 {
  opts_exclusive "`vce' `svy'" vce
  if `:length local weight' {
    di as err "svy option not allowed with weights"
    exit 198
  }
  global ML_noadj : copy local svyadjust
}
- global ML_svy : copy local svy
- if _caller() >= 15 & c(userversion) >= 15 {
- local fparm "", freeparm"
- }
- local eqspec : copy local spec
- local ieq 0
- while `:length local eqspec' {
= while 280 {
- local ++ieq
- gettoken eq eqspec : eqspec, bind
- if bsubstr(trim("`eq'"),1,1) == "/" {
  = if bsubstr(trim("(piq_timing_max: piqu_timing_max = rule_g rule_itr
> estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_
> authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.b
> nkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid,  )")
> ,1,1) == "/" {
  local eq = bsubstr(trim("`eq'"),2,..)
  local spec : subinstr local spec "/"`eq'" "("("`eq'":`fparm')"'', wor
> d
  }
- else {
- gettoken p1 p2 : eq, parse("(")")
- local p2 = trim("`p2'")
  = local p2 = trim("`piq_timing_max: piqu_timing_max = rule_g rule_itr
> estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_
> authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.b
> nkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid,  )"'
> )
  - if bsubstr("`p2'",1,1) == ",", {
    = if bsubstr("piq_timing_max: piqu_timing_max = rule_g rule_itr estim
> ated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_author
> s_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdi
> t 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid,  )",1,1) =
> = ",", {

```

```

        gettoken comma p2 : p2, parse(",")
        local p2 = trim("`"p2"'")
    }
    - if `"p1"p2"' == "()" {
    = if `(piq_timing_max: piq_timing_max = rule_g rule_itr estimated 0
> b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext
> ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.n
> twrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid, )" == "()" {
        local spec : subinstr local spec "`eq'" "(eq`ieq':)", word
    }
    - }
    - }
    - while `:length local eqspec' {
    = while 4 {
    - local ++ieq
    - gettoken eq eqspec : eqspec, bind
    - if bsubstr(trim("`eq'"),1,1) == "/" {
    = if bsubstr(trim("[]"),1,1) == "/" {
        local eq = bsubstr(trim("`eq'"),2,.)
        local spec : subinstr local spec "/"`eq'" `("`eq'":`fparm')"' , wor
> d
    }
    - else {
    - gettoken p1 p2 : eq, parse("()")
    - local p2 = trim("`"p2"'")
    = local p2 = trim(`"")
    - if bsubstr("`p2'",1,1) == "," {
    = if bsubstr("",1,1) == "," {
        gettoken comma p2 : p2, parse(",")
        local p2 = trim("`"p2"'")
    }
    - if `"p1"p2"' == "()" {
    = if `"[]"' == "()" {
        local spec : subinstr local spec "`eq'" "(eq`ieq':)", word
    }
    - }
    - }
    - while `:length local eqspec' {
    = while 0 {
        local ++ieq
        gettoken eq eqspec : eqspec, bind
        if bsubstr(trim("`eq'"),1,1) == "/" {
        local eq = bsubstr(trim("`eq'"),2,.)
        local spec : subinstr local spec "/"`eq'" `("`eq'":`fparm')"' , wor
> d
    }

```

```

else {
  gettoken p1 p2 : eq, parse("(")
  local p2 = trim("`p2'")
  if bsubstr("`p2'",1,1) == "," {
    gettoken comma p2 : p2, parse(",")
    local p2 = trim("`p2'")
  }
  if "`p1'`p2'" == "(" {
    local spec : subinstr local spec "`eq'" "(eq`ieq':)", word
  }
}
- if `interactive' {
= if 0 {
  SetGlobalVar ML_sample : _MLtua
}
- else {
- global ML_sample : tempvar
- if "`moptobj'" != "" {
= if `""' != "" {
  global ML_sample `moptobj'$ML_sample
}
- }
- if "`moptobj'" != "" {
= if `""' != "" {
  cap drop $ML_sample
}
- `vv' mark $ML_sample `if' `in' [`weight'`exp'], `zeroweight'
= version 18.5: mark __00000D if __000002 [],
- tempname eqlist
- `eqlist' = _eqlist.new, eqopts(NOCONSTANT FREEParam) eqargopts(OF
> Fset EXPosure) noneedvarlist numdepvars(0) needequal
= __00000E = _eqlist.new, eqopts(NOCONSTANT FREEParam) eqargopts(OF
> Fset EXPosure) noneedvarlist numdepvars(0) needequal
-class {
-instance:
-      array          n_depvars
-      string          needequal
-      string          noneedvarlist
-      string          wtypes
-      string          collinear
-      string          nocons
-      string          nocons_ignore
-      double          n_eq
-      string          gl_if
-      string          gl_in

```

```

-      string      gl_wgt
-      string      gl_options
-      _optlist     eq_opts_allowed
-class {
-instance:
-      array options = {}
-      array names   = {}
-}
-      array      eq_list
-      _optlist    mark_opts
-      array      mark_list
-      array      mark_name
-      _optlist    smark_opts
-      array      smark_list
-      array      smark_name
-      _optlist    rmcoll_opts
-      array      rmcoll_list
-      array      rmcoll_name
-      _optlist    rmdcoll_opts
-      array      rmdcoll_list
-      array      rmdcoll_name
-}

> _____ begin _eqlist.
> new _____
- version 9.2
- ._init
- _____

> _____ begin _eqlist._i
> nit _____
- version 9.2
- .n_eq = 0
- .gl_if = ""
- .gl_in = ""
- .gl_wgt = ""
- .gl_options = ""
- .eq_list = {}
- _____

> _____ end _eqlist._i
> nit _____
- ._setup `0'
- = ._setup , eqopts(NOCONSTant FREEParam) eqargopts(OFFset EXPosure)
> noneedvarlist numdepvars(0) needequal
- _____

> _____ begin _eqlist._se
> tup _____

```

```

- version 9.2
- .mark_list = {}
- .mark_name = {}
- .mark_opts.reset

```

```

> ----- begin _optlist.re
> set -----
- version 9
- .options = {}
- .names = {}

```

```

> ----- end _optlist.re
> set -----
- .smark_list = {}
- .smark_name = {}
- .smark_opts.reset

```

```

> ----- begin _optlist.re
> set -----
- version 9
- .options = {}
- .names = {}

```

```

> ----- end _optlist.re
> set -----
- .rmcoll_list = {}
- .rmcoll_name = {}
- .rmcoll_opts.reset

```

```

> ----- begin _optlist.re
> set -----
- version 9
- .options = {}
- .names = {}

```

```

> ----- end _optlist.re
> set -----
- .rmdcoll_list = {}
- .rmdcoll_name = {}
- .rmdcoll_opts.reset

```

```

> ----- begin _optlist.re
> set -----
- version 9
- .options = {}
- .names = {}

```



```

> _____ end _optlist.re
> set _____
      - syntax [, EQopts(namelist) EQARGopts(namelist) noNEEDVARLIST C
> OMMONopts(string asis) MARKopts(namelist) SMARKopts(namelist) RMCOLLOpts(nam
> elist) RMDCOLLOpts(namelist) NUMDEPvars(numlist integer >=0) NEEDEQUAL WTYPE
> S(string) IGNORENOCONS ]
      - .eq_opts_allowed.reset
> _____ begin _optlist.re
> set _____
      - version 9
      - .options = {}
      - .names = {}
> _____ end _optlist.re
> set _____
      - foreach opt of local eqopts {
      - .eq_opts_allowed.addopt `opt'
      = .eq_opts_allowed.addopt NOCONSTant
> _____ begin _optlist.add
> opt _____
      - version 9
      - syntax name(name=optspec) [, PASSthru]
      - local mname = lower("`optspec'")
      = local mname = lower("NOCONSTant")
      - if "`passthru'" != "" {
      = if "" != "" {
        local optspec `optspec'(passthru)
      }
      - .options.Arrpush "`optspec'"
      = .options.Arrpush "NOCONSTant"
      - .names.Arrpush "`mname'"
      = .names.Arrpush "noconstant"
> _____ end _optlist.add
> opt _____
      - }
      - .eq_opts_allowed.addopt `opt'
      = .eq_opts_allowed.addopt FREEParm
> _____ begin _optlist.add
> opt _____
      - version 9
      - syntax name(name=optspec) [, PASSthru]

```

```

- local mname = lower("`optspec'")
= local mname = lower("FREEParm")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "FREEParm"
- .names.Arrpush "`mname'"
= .names.Arrpush "freeparm"

```

```

> _____ end _optlist.add
> opt _____
- }
- foreach opt of local eqargopts {
- .eq_opts_allowed.addopt `opt', passthru
= .eq_opts_allowed.addopt OFFset, passthru

```

```

> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("OFFset")
- if "`passthru'" != "" {
= if "passthru" != "" {
- local optspec `optspec'(passthru)
= local optspec OFFset(passthru)
- }
- .options.Arrpush "`optspec'"
= .options.Arrpush "OFFset(passthru)"
- .names.Arrpush "`mname'"
= .names.Arrpush "offset"

```

```

> _____ end _optlist.add
> opt _____
- }
- .eq_opts_allowed.addopt `opt', passthru
= .eq_opts_allowed.addopt EXPosure, passthru

```

```

> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("EXPosure")

```

```

- if "`passthru'" != "" {
= if "passthru" != "" {
- local optspec `optspec'(passthru)
= local optspec EXPosure(passthru)
- }
- .options.Arrpush "`optspec'"
= .options.Arrpush "EXPosure(passthru)"
- .names.Arrpush "`mname'"
= .names.Arrpush "exposure"
}

> _____ end _optlist.add
> opt _____
- }
- local i 0
- foreach opt of local markopts {
  .mark_opts.addopt `opt', passthru
}
- local i 0
- foreach opt of local smarkopts {
  .smark_opts.addopt `opt', passthru
}
- local i 0
- foreach opt of local rmcollopts {
  .rmcoll_opts.addopt `opt', passthru
}
- local i 0
- foreach opt of local rmdcollopts {
  .rmdcoll_opts.addopt `opt', passthru
}
- .n_depvars.Arrdropall
- if "`numdepvars'" != "" {
= if "0" != "" {
- foreach ndep of local numdepvars {
  .n_depvars.Arrpush `ndep'
= .n_depvars.Arrpush 0
- }
- }
- else .n_depvars.Arrpush 1
- .needequal = "`needequal'"
= .needequal = "needequal"
- .wtypes = "`wtypes'"
= .wtypes = ""
- .noneedvarlist = "`needvarlist'"
= .noneedvarlist = "noneedvarlist"
- .nocons_ignore = "`ignorenocons'"
= .nocons_ignore = ""

```

```

> _____ end _eqlist._se
> tup _____

> _____ end _eqlist.
> new _____
    - .`eqlist'.parse `spec'
    = .__00000E.parse (piq_timing_max: piq_timing_max = rule_g rule_itr
> estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_
> authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.b
> nkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid, )
> []

> _____ begin _eqlist.pa
> rse _____
    - version 9.2
    - .__init

> _____ begin _eqlist._i
> nit _____
    - version 9.2
    - .n_eq = 0
    - .gl_if = ""
    - .gl_in = ""
    - .gl_wgt = ""
    - .gl_options = ""
    - .eq_list = {}

> _____ end _eqlist._i
> nit _____
    - local MARKOPTLIST ```.mark_opts.dumpoptions'''

> _____ begin _optlist.dumpopti
> ons _____
    - version 9
    - forval i = 1/`.options.arnels' {
    = forval i = 1/0 {
        local optlist `optlist' `.options['i']'
        }
    - local optlist : list uniq optlist
    - sreturn local optlist ```.optlist'''
    = sreturn local optlist `'''
    - class exit ```.optlist'''
    = class exit `'''

> _____ end _optlist.dumpopti

```

```

> ons -----
    = local MARKOPTLIST `""'
    - local SMARKOPTLIST `""`.smark_opts.dumpoptions'""'
-----

> ----- begin _optlist.dumpopti
> ons -----
    - version 9
    - forval i = 1/`.options.arnels' {
    = forval i = 1/0 {
        local optlist `optlist' `.options['i']'
    }
    - local optlist : list uniq optlist
    - sreturn local optlist `""`optlist'""'
    = sreturn local optlist `""'
    - class exit `""`optlist'""'
    = class exit `""'
-----

> ----- end _optlist.dumpopti
> ons -----
    = local RMCOLLOPTLIST `""'
    - local RMCOLLOPTLIST `""`.rmcoll_opts.dumpoptions'""'
-----

> ----- begin _optlist.dumpopti
> ons -----
    - version 9
    - forval i = 1/`.options.arnels' {
    = forval i = 1/0 {
        local optlist `optlist' `.options['i']'
    }
    - local optlist : list uniq optlist
    - sreturn local optlist `""`optlist'""'
    = sreturn local optlist `""'
    - class exit `""`optlist'""'
    = class exit `""'
-----

> ----- end _optlist.dumpopti
> ons -----
    = local RMDCOLLOPTLIST `""'
    - local RMDCOLLOPTLIST `""`.rmdcoll_opts.dumpoptions'""'
-----

> ----- begin _optlist.dumpopti
> ons -----
    - version 9
    - forval i = 1/`.options.arnels' {
    = forval i = 1/0 {
        local optlist `optlist' `.options['i']'

```

```

    }
    - local optlist : list uniq optlist
    - sreturn local optlist ``"optlist"``
    = sreturn local optlist ``"``"``
    - class exit ``"optlist"``
    = class exit ``"``"``

```

```

> _____ end _optlist.dumpopti
> ons _____
    = local RMDCOLLOPTLIST ``"``"``
    - _parse expand EQ GL : 0, gweight common( `MARKOPTLIST' `SMARKOPT
> LIST' `RMCOLLOPTLIST' `RMDCOLLOPTLIST' COLLlinear )
    = _parse expand EQ GL : 0, gweight common( COLLlinear )
    - if `EQ_n' == 0 & ``.noneedvarlist' == "" {
    = if 1 == 0 & "noneedvarlist" == "" {
        syntax varlist
    }
    - local rebind = `EQ_n' == 1
    = local rebind = 1 == 1
    - if `rebind' {
    = if 1 {
        - tokenize ``"0"`, parse(",()[]")
        = tokenize ``(piq_timing_max: piq_timing_max = rule_g rule_itr est
> imated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_auth
> ors_ext ln_neq 0b.ntwrth#0b.bnkcrcdit 0b.ntwrth#1.bnkcrcdit 1o.ntwrth#0b.bnkcrc
> dit 1o.ntwrth#1o.bnkcrcdit ntwrth open bnkcrcdit vint_early vint_mid, ) []"
> , parse(",()[]")
        - local rebind = ``"1" != "(" & ``"2" == "("
        = local rebind = ``( "(" != "(" & "piq_timing_max: piq_timing_max =
> rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.
> wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrcdit 0b.ntwrth#1.bnkcrcdi
> t 1o.ntwrth#0b.bnkcrcdit 1o.ntwrth#1o.bnkcrcdit ntwrth open bnkcrcdit vint_earl
> y vint_mid" == "("
        - }
        - if `rebind' {
    = if 0 {
        while `:length local 0' {
            gettoken tok 0 : 0, parse(" ,") bind match(par)
            if ``par' == "(" {
                gettoken nxt : 0
                local nxt = substr(`"nxt"`,1,1)
                if inlist(`"nxt"`, "#", "|") {
                    local ZERO ``"ZERO' (`tok')"
                }
            }
            else {
                local ZERO ``"(`ZERO') (`tok') `0"
            }
        }
    }

```

```

        local 0
    }
}
else if inlist("`tok'", "if", "in", ",") {
    local ZERO `("`ZERO') `tok' `0'`"
    local 0
}
else if inlist(substr("`tok'",1,1), "#", "|") {
    local ZERO ``"ZERO" `tok'`"
}
else {
    local ZERO ``"ZERO" `tok'`"
}
}

_parse expand EQ GL : ZERO, gweight common( `MARKOPTLIST' `SMARK
> OPTLIST' `RMCOLLOPTLIST' `RMDCOLLOPTLIST' COLLlinear )
}
- forval i = 1/`EQ_n' {
= forval i = 1/1 {
- if `i' <= `.n_depvars.arrnels' {
= if 1 <= 1 {
- local ndep = `.n_depvars[`i']'
= local ndep = 0
- }
- .eq_list[`i'] = .eqspec.new, numdepvars(`ndep') `.needequal' `.
> noneedvarlist' wtypes(`.wtypes') `.nocons_ignore'
= .eq_list[1] = .eqspec.new, numdepvars(0) needequal noneedvarlis
> t wtypes()
-class {
-instance:
-         double  n_depvars
-         double  needequal
-         double  needvarlist
-         string  wtypes
-         string  name
-         string  depvars
-         double  hasequal
-         string  indepvars
-         string  other_markvars
-         double  nocons
-         double  nocons_ignore
-         string  offset
-         double  exposure
-         double  freeparm
-         string  if
-         string  in

```

```

-         string wgt
-         string options
-}

```

```

> _____ begin _eqspec.
> new _____
- version 9.2
- syntax [, NUMDEPvars(numlist integer >=0 max=1) NEEDEQUAL noNE
> EDVARLIST WTYPES(string) IGNORENOCONS ]
- .n_depvars = 1
- if `:list sizeof numdepvars' {
= if 1 {
- .n_depvars = `numdepvars'
= .n_depvars = 0
- }
- .needequal = ("`needequal'" != "")
= .needequal = ("needequal" != "")
- .needvarlist = ("`needvarlist'" == "")
= .needvarlist = ("noneedvarlist" == "")
- .nocons_ignore = ("`ignorenocons'" != "")
= .nocons_ignore = ("" != "")
- .wtypes = "`wtypes'"
= .wtypes = ""

```

```

> _____ end _eqspec.
> new _____
- .eq_list['i'].parse `.eq_opts_allowed.objkey':`EQ`i'
= .eq_list[1].parse __key(600000279740):piq_timing_max: piq_timing
> _max = rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.op
> en 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.b
> nkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vin
> t_early vint_mid

```

```

> _____ begin _eqspec.pa
> rse _____
- version 9.2
- _on_colon_parse `0'
= _on_colon_parse __key(600000279740):piq_timing_max: piq_timing
> _max = rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.op
> en 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.b
> nkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vin
> t_early vint_mid

```

```

> _____ begin _on_colon_pa
> rse _____
- version 8.2

```



```

- sreturn local before ""
- sreturn local after ""
- gettoken before after : 0, parse(":") bind match(par) quotes
- if "`par'" != "" {
= if "" != "" {
  local before `("`before')'
}
- if ``"before"' == ":" {
= if ``__key(600000279740)' == ":" {
  sreturn local after ``"after"'
  exit
}
- while ``"COLON"' != ":" & ``"after"' != "" {
= while ``"' != ":" & ``:piq_timing_max: piqu_timing_max = rule
> _g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#
> 1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.
> ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vin
> t_mid"' != "" {
  - gettoken COLON after : after, parse(":") bind match(par) quo
> tes
  - if "`par'" != "" {
= if "" != "" {
  local before `before' (`COLON')
  local COLON
}
- else if ``"COLON"' != ":" {
= else if ``:"' != ":" {
  local before ``"before' `COLON"'
  local COLON
}
- }
- while ``"COLON"' != ":" & ``"after"' != "" {
= while ``:"' != ":" & ``piq_timing_max: piqu_timing_max = rule
> _g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#
> 1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.
> ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vin
> t_mid"' != "" {
  gettoken COLON after : after, parse(":") bind match(par) quo
> tes
  if "`par'" != "" {
  local before `before' (`COLON')
  local COLON
}
else if ``"COLON"' != ":" {
  local before ``"before' `COLON"'
  local COLON

```

```

    }
    }
    - if ``COLON`` != ":" {
= if `":` != ":" {
    di as err "" found where ':' expected"
    exit 198
    }
    - sreturn local before ``before``
    = sreturn local before `__key(600000279740)`
    - sreturn local after ``after``
    = sreturn local after `piq_timing_max: piq_timing_max = rule_
> g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1
> o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.n
> twrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint
> _mid``

> _____ end _on_colon_pa
> rse _____
    - local ol `s(before)`
    = local ol __key(600000279740)
    - if `:list sizeof ol` {
    = if 1 {
    - local OPTLIST ``.ol'.dumpoptions``

> _____ begin _optlist.dumpopti
> ons _____
    - version 9
    - forval i = 1/`.options.arnels` {
    = forval i = 1/4 {
    - local optlist `optlist' `.options['i']`
    = local optlist NOCONSTant
    - }
    - local optlist `optlist' `.options['i']`
    = local optlist NOCONSTant FREEParam
    - }
    - local optlist `optlist' `.options['i']`
    = local optlist NOCONSTant FREEParam OFFset(passthru)
    - }
    - local optlist `optlist' `.options['i']`
    = local optlist NOCONSTant FREEParam OFFset(passthru) EXPosure(
> passthru)
    - }
    - local optlist : list uniq optlist
    - sreturn local optlist ``optlist``
    = sreturn local optlist `NOCONSTant FREEParam OFFset(passthru)
> EXPosure(passthru)``

```

```

- class exit ```optlist''
= class exit ``NOCONSTant FREEParm OFFset(passthru) EXPosure(p
> assthru)''
-
-
- end _optlist.dumpopti
> ons
= local OPTLIST ``NOCONSTant FREEParm OFFset(passthru) EXPosure(
> passthru)''
- local OPTNAMES ```.ol'.dumpnames''
-
-
- begin _optlist.dumpna
> mes
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/4 {
- local namelist `namelist' `.names['i']'
= local namelist noconstant
- }
- local namelist `namelist' `.names['i']'
= local namelist noconstant freeparm
- }
- local namelist `namelist' `.names['i']'
= local namelist noconstant freeparm offset
- }
- local namelist `namelist' `.names['i']'
= local namelist noconstant freeparm offset exposure
- }
- local namelist : list uniq namelist
- sreturn local namelist ```namelist''
= sreturn local namelist ``noconstant freeparm offset exposure
> ``
- class exit ```namelist''
= class exit ``noconstant freeparm offset exposure''
-
-
- end _optlist.dumpna
> mes
= local OPTNAMES ``noconstant freeparm offset exposure''
- local NOCONS constant noconstant
- if ```:list NOCONS & OPTNAMES'' == '' {
= if ``noconstant'' == '' {
local NOCONS noCONSTant
}
- else local NOCONS
- }
- local 0 ``s(after)''
= local 0 ``piq_timing_max: piq_timing_max = rule_g rule_itr est

```

```

> imated 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_auth
> ors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcr
> dit 1o.ntwrth#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid''
    - if ``.wtypes'' != '' {
    = if `'' != '' {
      local wgt "[`.wtypes']"
    }
    - syntax [anything(equalok)] [if] [in] `wgt' [, `OPTLIST' `NOCON
> S' *]
    = syntax [anything(equalok)] [if] [in]  [, NOCONSTant FREEParam 0
> FFset(passthru) EXPosure(passthru) *]
    - local HOLDIF ``if''
    = local HOLDIF `''
    - local HOLDIN ``in''
    = local HOLDIN `''
    - if ``wgt'' != '' & ``weight'' != '' {
    = if '' != '' & '' != '' {
      local HOLDWGT ``[`weight'`exp]''
    }
    - gettoken NAME REST : anything, parse(":")
    - if trim("`NAME'") == ":" {
    = if trim("piq_timing_max") == ":" {
      di as err "nothing found where equation name expected"
      exit 198
    }
    - if `:list sizeof REST' {
    = if 21 {
    - if strpos("`NAME'", ":") {
    = if strpos("`piq_timing_max'", ":") {
      capture _msparse `NAME'
      if c(rc) {
        di as err "{bf:`NAME'} invalid equation specification"
        exit 198
      }
    }
    - else {
    - ._confirm_eqname `NAME'
    = ._confirm_eqname piq_timing_max

> _____ begin _eqspec._confirm_eqn
> ame _____
    - version 9.2
    - capture confirm number `0'
    = capture confirm number piq_timing_max
    - if (! c(rc)) exit
    - local hasdot = strpos("`0'", ".")

```

```

= local hasdot = strpos(`"piq_timing_max"',".")
- if `hasdot' {
= if 0 {
    syntax varname(ts)
    exit
  }
- capture syntax name(id="equation name")
- if (! c(rc)) exit

```

```

> _____ end _eqspec._confirm_eqn
> ame _____
    - }
    - local NAME : list retok NAME
    - gettoken COLON anything : REST, parse(":")
    - }
    - else {
      local NAME
    }
    - gettoken DEPVARs REST : anything, parse("=") bind
    - if "`DEPVARs'" == "=" {
= if "piq_timing_max " == "=" {
      local REST : copy local anything
      local DEPVARs
    }
    - if (`:length local DEPVARs' == 0 & `.n_depvars') {
= if (15 == 0 & 0) {
      di as err "nothing found where depvar expected"
      exit 198
    }
    - if `"'REST'"' == "" {
= if `"'= rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.ope
> n 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit
> 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth ope
> n bnkcrdit vint_early vint_mid"' == "" {
      if 0`.needequal' {
      if `.needvarlist' | (`:length local DEPVARs' != 0 & `.n_depvar
> s') {
      di as err "equal sign required to separate depvars from indepv
> ars"
      exit 198
    }
  }
  local DEPVARs
  local n_deps : list sizeof anything
  if `n_deps' != 0 | `.needvarlist' {
  if `n_deps' < `.n_depvars' {

```

```

        error 102
    }
    forval j = 1/`.n_depvars' {
        gettoken YY anything : anything
        local DEPVARS `DEPVARS' `YY'
    }
    local INDEPVARS : list retok anything
}
local hasequal 0
}
- else {
- gettoken EQ INDEPVARS : REST, parse("=") bind
- local hasequal 1
- }
- if `:list sizeof DEPVARS' {
= if 1 {
- _fv_check_depvar `DEPVARS'
= _fv_check_depvar piq_timing_max
- tsunab DEPVARS : `DEPVARS'
= tsunab DEPVARS : piq_timing_max
}

> _____ begin tsu
> nab _____
        - version 6
        - gettoken user 0: 0, parse(" :")
        - gettoken colon 0: 0, parse(" :")
        - if `"'colon'" != ":" { error 198 }
        = if `":'" != ":" { error 198 }
        - syntax [varlist(ts default=empty)] [, MIN(integer 1) MAX(int
> eger 32000) NAME(string)]
        - c_local `user' `varlist'
        = c_local DEPVARS piq_timing_max
        - local n : word count `varlist'
        = local n : word count piq_timing_max
        - if `n'>=`min' & `n'<=`max' { exit }
        = if 1>=1 & 1<=32000 { exit }

> _____ end tsu
> nab _____
        - }
        - if `:list sizeof INDEPVARS' {
        = if 18 {
        - fvunab INDEPVARS : `INDEPVARS'
        = fvunab INDEPVARS : rule_g rule_itr estimated 0b.wlth#0b.open
> 0b.wlth#1.open 1.wlth#0b.open 10.wlth#10.open cb_authors_ext ln_neq 0b.ntwrth
> h#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 10.ntwrth#0b.bnkcrdit 10.ntwrth#10.bnkcrd

```

```

> it ntwrth open bnkcrdit vint_early vint_mid
> _____ begin fvu
> nab _____
      - version 11
      - _on_colon_parse `0'
      = _on_colon_parse INDEPVARs : rule_g rule_itr estimated 0b.wl
> th#0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_n
> eq 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#
> h#1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid
> _____ begin _on_colon_pa
> rse _____
      - version 8.2
      - sreturn local before ""
      - sreturn local after ""
      - gettoken before after : 0, parse(":") bind match(par) quot
> es
      - if "`par'" != "" {
      = if "" != "" {
        local before "`('before')'"
      }
      - if ``"before"' == ":" {
      = if `"INDEPVARs "' == ":" {
        sreturn local after ``"after"'
        exit
      }
      - while ``"COLON"' != ":" & ``"after"' != "" {
      = while ``"' != ":" & `": rule_g rule_itr estimated 0b.wlth
> #0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq
> 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#
> 1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid"' != "" {
      - gettoken COLON after : after, parse(":") bind match(par) q
> uotes
      - if "`par'" != "" {
      = if "" != "" {
        local before `before' (`COLON')
        local COLON
      }
      - else if ``"COLON"' != ":" {
      = else if `":' != ":" {
        local before ``"before' `COLON"'
        local COLON
      }
      - }
      - while ``"COLON"' != ":" & ``"after"' != "" {

```

```

= while `":'" != ":" & `"` rule_g rule_itr estimated 0b.wlth
> #0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq
> 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#
> 1o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid'" != "" {
    gettoken COLON after : after, parse(":") bind match(par) q
> uotes
    if "`par'" != "" {
        local before `before' (`COLON')
        local COLON
    }
    else if `""COLON'" != ":" {
        local before `""before' `COLON'"
        local COLON
    }
}
- if `""COLON'" != ":" {
= if `":'" != ":" {
    di as err "" found where ':' expected"
    exit 198
}
- sreturn local before `""before'"
= sreturn local before `INDEPVARS "'
- sreturn local after `""after'"
= sreturn local after `"` rule_g rule_itr estimated 0b.wlth#
> 0b.open 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq
> 0b.ntwrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1
> o.bnkcrdit ntwrth open bnkcrdit vint_early vint_mid'"

_____ end _on_colon_pa
> rse _____
    - local 0 `""s(before)'"
    = local 0 `INDEPVARS "'
    - syntax name(id="macro name" name=user)
    - local 0 `""s(after)'"
    = local 0 `"` rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlt
> h#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.b
> nkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntw
> rth open bnkcrdit vint_early vint_mid'"
    - syntax [varlist(ts fv default=none)] [, MIN(integer 1) MAX(i
> nteger 32000) NAME(string) ]
    - c_local `user' `varlist'
    = c_local INDEPVARS rule_g rule_itr estimated 0b.wlth#0b.open
> 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwr
> h#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrd
> it ntwrth open bnkcrdit vint_early vint_mid
    - local n : list sizeof varlist

```



```

- if (`min' <= `n' & `n' <= `max') exit
= if (1 <= 18 & 18 <= 32000) exit

```

```

> ----- end fvu
> nab -----
- }
- if inlist("noconstant", "`noconstant'", "`constant'") {
= if inlist("noconstant", "", "") {
  if "`INDEPVARs'" == "" & !`.nocons_ignore' {
    di as err "independent variables are required when option noco
> nstant is specified"
    exit 102
  }
  if "`NOCONS'" != "" {
    local options `options' noconstant
  }
  local NOCONS 1
}
- else local NOCONS 0
- if ``offset'`exposure'"" != "" {
= if `""' != "" {
  . _check_offset, `offset' `exposure'
  local OFFSET "`s(offset)'"
  local EXPOSURE = ("`exposure'" != "")
  local OTHER `s(offset)'
}
- else local EXPOSURE 0
- local FREEARM = ``freeparm'"" != ""
= local FREEARM = `""' != ""
- local OPTIONS
- local SPACE
- foreach OPTNAME of local OPTNAMES {
- if ``OPTNAME'"" != "" {
= if `""' != "" {
  local OPTIONS ``OPTIONS'`SPACE'`OPTNAME'""
  local SPACE " "
}
- }
- if ``OPTNAME'"" != "" {
= if `""' != "" {
  local OPTIONS ``OPTIONS'`SPACE'`OPTNAME'""
  local SPACE " "
}
- }
- if ``OPTNAME'"" != "" {
= if `""' != "" {

```

```

        local OPTIONS ```OPTIONS``SPACE```OPTNAME````
        local SPACE " "
    }
- }
- if `````OPTNAME```` != "" {
= if ```` != "" {
    local OPTIONS ```OPTIONS``SPACE```OPTNAME````
    local SPACE " "
    }
- }
- if `````options`` != "" {
= if ```` != "" {
    local OPTIONS ```OPTIONS``SPACE```options``
    }
- .name = ````NAME````
= .name = `"piq_timing_max"`
- .depvars = ````DEPVAR````
= .depvars = `"piq_timing_max"`
- .hasequal = `hasequal`
= .hasequal = 1
- .indepvars = ````:list retok INDEPVAR````
= .indepvars = `"rule_g rule_itr estimated 0b.wlth#0b.open 0b.wl
> th#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.
> bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit nt
> wrth open bnkcrdit vint_early vint_mid``
    - .other_markvars = ````:list retok OTHER````
    = .other_markvars = ````
    - .nocons = `NOCONS`
    = .nocons = 0
    - .if = ````HOLDIF````
    = .if = ````
    - .in = ````HOLDIN````
    = .in = ````
    - .wgt = ````HOLDWGT````
    = .wgt = ````
    - .offset = ````OFFSET````
    = .offset = ````
    - .exposure = `EXPOSURE`
    = .exposure = 0
    - .freeparm = `FREEPARM`
    = .freeparm = 0
    - .options = ````OPTIONS````
    = .options = ````

```

```

> _____ end _eqspec.pa
> rse _____

```

```

- }
- local MARKOPTLIST ```.mark_opts.dumpoptions'''
_____
> _____ begin _optlist.dumpopti
> ons _____
- version 9
- forval i = 1/`.options.arnels' {
= forval i = 1/0 {
  local optlist `optlist' `options['i']'
}
- local optlist : list uniq optlist
- sreturn local optlist ```optlist'''
= sreturn local optlist `'''
- class exit ```optlist'''
= class exit `'''
_____
> _____ end _optlist.dumpopti
> ons _____
= local MARKOPTLIST `'''
- local MARKOPTNAMES ```.mark_opts.dumpnames'''
_____
> _____ begin _optlist.dumpna
> mes _____
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/0 {
  local namelist `namelist' `names['i']'
}
- local namelist : list uniq namelist
- sreturn local namelist ```namelist'''
= sreturn local namelist `'''
- class exit ```namelist'''
= class exit `'''
_____
> _____ end _optlist.dumpna
> mes _____
= local MARKOPTNAMES `'''
- local OPTLIST1 : substr local MARKOPTLIST "(passthru)" "(strin
> g asis)", all
- local SMARKOPTLIST ```.smark_opts.dumpoptions'''
_____
> _____ begin _optlist.dumpopti
> ons _____
- version 9
- forval i = 1/`.options.arnels' {
= forval i = 1/0 {

```

```

        local optlist `optlist' `.options['i']'
    }
    - local optlist : list uniq optlist
    - sreturn local optlist ```optlist'''
= sreturn local optlist `'''
    - class exit ```optlist'''
= class exit `'''

```

```

> ----- end _optlist.dumpopti
> ons -----
    = local SMARKOPTLIST `'''
    - local SMARKOPTNAMES ```.smark_opts.dumpnames'''

```

```

> ----- begin _optlist.dumpna
> mes -----
    - version 9
    - forval i = 1/`.names.arnels' {
= forval i = 1/0 {
    local namelist `namelist' `.names['i']'
    }
    - local namelist : list uniq namelist
    - sreturn local namelist ```namelist'''
= sreturn local namelist `'''
    - class exit ```namelist'''
= class exit `'''

```

```

> ----- end _optlist.dumpna
> mes -----
    = local SMARKOPTNAMES `'''
    - local OPTLIST1s : substr local SMARKOPTLIST "(passthru)" "(str
> ing asis)", all
    - local RMCOLLOPTLIST ```.rmcoll_opts.dumpoptions'''

```

```

> ----- begin _optlist.dumpopti
> ons -----
    - version 9
    - forval i = 1/`.options.arnels' {
= forval i = 1/0 {
    local optlist `optlist' `.options['i']'
    }
    - local optlist : list uniq optlist
    - sreturn local optlist ```optlist'''
= sreturn local optlist `'''
    - class exit ```optlist'''
= class exit `'''

```

```

> _____ end _optlist.dumpopti
> ons _____
    = local RMCOLLOPTLIST `''`
    - local RMCOLLOPTNAMES ```.rmcoll_opts.dumpnames'''
    _____
> _____ begin _optlist.dumpna
> mes _____
    - version 9
    - forval i = 1/`.names.arnels' {
    = forval i = 1/0 {
        local namelist `namelist' `.names['i']'
    }
    - local namelist : list uniq namelist
    - sreturn local namelist ```namelist'''
    = sreturn local namelist `''`
    - class exit ```namelist'''
    = class exit `''`
    _____
> _____ end _optlist.dumpna
> mes _____
    = local RMCOLLOPTNAMES `''`
    - local OPTLIST2 : substr local RMCOLLOPTLIST "(passthru)" "(str
> ing asis)", all
    - local RMDCOLLOPTLIST ```.rmdcoll_opts.dumpoptions'''
    _____
> _____ begin _optlist.dumpopti
> ons _____
    - version 9
    - forval i = 1/`.options.arnels' {
    = forval i = 1/0 {
        local optlist `optlist' `.options['i']'
    }
    - local optlist : list uniq optlist
    - sreturn local optlist ```optlist'''
    = sreturn local optlist `''`
    - class exit ```optlist'''
    = class exit `''`
    _____
> _____ end _optlist.dumpopti
> ons _____
    = local RMDCOLLOPTLIST `''`
    - local RMDCOLLOPTNAMES ```.rmdcoll_opts.dumpnames'''
    _____
> _____ begin _optlist.dumpna
> mes _____
    - version 9

```

```

- forval i = 1/`.names.arnels' {
= forval i = 1/0 {
    local namelist `namelist' `.names[`i']'
}
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist `""'
- class exit ``namelist'''
= class exit `""'

```

```

> ----- end _optlist.dumpna
> mes
    = local RMDCOLLOPTNAMES `""'
    - local OPTLIST3 : substr local RMDCOLLOPTLIST "(passthru)" "(st
> ring asis)", all
    - local 0 `", `GL_op'''
    = local 0 `", ""
    - syntax [, `OPTLIST1' `OPTLIST1s' `OPTLIST2' `OPTLIST3' COLLlinear
> noCONSTant * ]
    = syntax [, COLLlinear noCONSTant * ]
    - local GL_op ``options'''
    = local GL_op `""'
    - if ``collinear''' != "" {
    = if `""' != "" {
        local GL_op ``GL_op' `collinear'''
        .collinear = ""collinear""
    }
    - if ``constant''' != "" {
    = if `""' != "" {
        local GL_op ``GL_op' `constant'''
        .nocons = ""constant""
    }
    - local I 0
    - foreach OPT of local MARKOPTNAMES {
        if `:list sizeof `OPT'' {
            .mark_list[`++I'] = .eqspec.new, numdep(0) wtypes(`.wtypes')
            .mark_list[`I'].parse "" : ``OPT''
            .mark_name.Arrpush ""OPT""
        }
    }
    - local I 0
    - foreach OPT of local SMARKOPTNAMES {
        if `:list sizeof `OPT'' {
            .smark_list[`++I'] = .eqspec.new, numdep(0) wtypes(`.wtypes')
            .smark_list[`I'].parse "" : ``OPT''
            .smark_name.Arrpush ""OPT""

```

```

    }
  }
  - local I 0
  - foreach OPT of local RMCOLLOPTNAMES {
    if `:list sizeof `OPT'' {
      .rmcoll_list[`++I'] = .eqspec.new, numdep(0) wtypes(`.wtypes')
      .rmcoll_list[`I'].parse "" : ``OPT''
      .rmcoll_name.Arrpush ``OPT''
    }
  }
  - local I 0
  - foreach OPT of local RMDCOLLOPTNAMES {
    if `:list sizeof `OPT'' {
      .rmdcoll_list[`++I'] = .eqspec.new, wtypes(`.wtypes')
      .rmdcoll_list[`I'].parse "" : ``OPT''
      .rmdcoll_name.Arrpush ``OPT''
    }
  }
  - .n_eq = `EQ_n'
  = .n_eq = 1
  - .gl_if = ``GL_if'''
  = .gl_if = `'''
  - .gl_in = ``GL_in'''
  = .gl_in = `'''
  - .gl_wgt = ``GL_wt'''
  = .gl_wgt = ``[]'''
  - .gl_options = ``:list retok GL_op'''
  = .gl_options = `'''

```

```
> _____ end _eqlist.pa
```

```
> rse _____
- if ``subpop'' != "" & ``missing'' == "" {
= if "" != "" & "missing" == "" {
  if `interactive' {
    SetGlobalVar ML_samp : _MLtu
  }
  else {
    global ML_samp : tempvar
    if ``moptobj''' != "" {
      global ML_samp `moptobj'$ML_samp
    }
  }
  tempvar markuse
  _svy_setup $ML_sample $ML_samp, svy subpop(`subpop')
  quietly gen byte `markuse' = $ML_samp
}

```

```

- else global ML_samp $ML_sample
= else global ML_samp __00000D
- if "`collinear'" == "" {
= if "collinear" == "" {
    .`eqlist'.rmcoll $ML_samp, replace `missing'
    }
- else if "`missing'" == "" {
= else if "missing" == "" {
    .`eqlist'.markout $ML_samp, replace
    }
- if "$ML_samp" != "$ML_sample" {
= if "__00000D" != "__00000D" {
    `vv' quietly replace $ML_sample = 0 if `markuse' & !$ML_samp
    drop `markuse'
    }
- local eq_n = `.`eqlist'.eq count'

> ----- begin _eqlist
> .eq -----
    - version 9.2
    - args item idx
    - if "`item'" == "count" {
    = if "count" == "count" {
    - sreturn local value ```.n_eq'''
    = sreturn local value ``1'''
    - class exit ``s(value)'''
    = class exit ``1'''

> ----- end _eqlist
> .eq -----
    = local eq_n = 1
    - global ML_n : copy local eq_n
    - global ML_dim = `.`eqlist'.dim'

> ----- begin _eqlist.
> dim -----
    - version 9.2
    - syntax
    - local neq = `.`eq count'

> ----- begin _eqlist
> .eq -----
    - version 9.2
    - args item idx
    - if "`item'" == "count" {
    = if "count" == "count" {

```



```

- sreturn local value ```.n_eq'''
= sreturn local value `1'
- class exit ```s(value)'''
= class exit `1'

> _____ end _eqlist
> .eq _____
    = local neq = 1
    - forval i = 1/`neq' {
    = forval i = 1/1 {
    - local xvarsi ```.eq indepvars `i'''

> _____ begin _eqlist
> .eq _____
    - version 9.2
    - args item idx
    - if ``item' == "count" {
    = if "indepvars" == "count" {
      sreturn local value ```.n_eq'''
      class exit ```s(value)'''
    }
    - else if 0 < `idx' & `idx' <= `.n_eq' {
    = else if 0 < 1 & 1 <= 1 {
    - .eq_list[`idx'].item `item'
    = .eq_list[1].item indepvars

> _____ begin _eqspec.i
> tem _____
    - version 9.2
    - args item
    - .check_eq_item_name item, `item'
    = .check_eq_item_name item, indepvars

> _____ begin _eqspec._check_eq_item_n
> ame _____
    - version 9.2
    - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars
> IF IN WGT OTHER NOCONS OFFset EXPosure FREEparm OPTions ]
    - if c(rc) error 198
    - if `:list sizeof other' {
    = if 0 {
      local other other_markvars
    }
    - local item `name' `depvars' `indepvars' `if' `in' `wgt' `o
> ther' `nocons' `offset' `exposure' `freeparm' `options'
    = local item indepvars

```

```

- if `:list sizeof item' != 1 {
= if 1 != 1 {
    error 198
  }
- c_local `c_item' `item'
= c_local item indepvars

```

```

> _____ end _eqspec._check_eq_item_n
> ame _____
    - local value ```.item'''
    = local value `rule_g rule_itr estimated 0b.wlth#0b.open 0b.w
> lth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b
> .bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit n
> twrth open bnkcrdit vint_early vint_mid''
    - sreturn local value ```.value'''
    = sreturn local value `rule_g rule_itr estimated 0b.wlth#0b.o
> pen 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.n
> twrth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bn
> kcrdit ntwrth open bnkcrdit vint_early vint_mid''
    - class exit ```.value'''
    = class exit `rule_g rule_itr estimated 0b.wlth#0b.open 0b.wl
> th#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.
> bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit nt
> wrth open bnkcrdit vint_early vint_mid''

```

```

> _____ end _eqspec.i
> tem _____
    - class exit ```.s(value)'''
    = class exit `rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth
> #1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bn
> kcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwr
> th open bnkcrdit vint_early vint_mid''

```

```

> _____ end _eqlist
> .eq _____
    = local xvarsi `rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth
> #1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bn
> kcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwr
> th open bnkcrdit vint_early vint_mid''
    - local noconsi ```.eq nocons `i'''

```

```

> _____ begin _eqlist
> .eq _____
    - version 9.2
    - args item idx
    - if ``.item' == "count" {

```

```

= if "nocons" == "count" {
  sreturn local value ```.n_eq'''
  class exit ```s(value)'''
}
- else if 0 < `idx' & `idx' <= `.n_eq' {
= else if 0 < 1 & 1 <= 1 {
- .eq_list[`idx'].item `item'
= .eq_list[1].item nocons
}

> _____ begin _eqspec.i
> tem _____
- version 9.2
- args item
- .check_eq_item_name item, `item'
= .check_eq_item_name item, nocons
}

> _____ begin _eqspec._check_eq_item_n
> ame _____
- version 9.2
- capture syntax name(name=c_item) [, NAME DEPvars INDEPvars
> IF IN WGT OTHER NOCONS OFFset EXPosure FREEParam OPTions ]
- if c(rc) error 198
- if `:list sizeof other' {
= if 0 {
  local other other_markvars
}
- local item `name' `depvars' `indepvars' `if' `in' `wgt' `o
> ther' `nocons' `offset' `exposure' `freeparm' `options'
= local item nocons
- if `:list sizeof item' != 1 {
= if 1 != 1 {
  error 198
}
- c_local `c_item' `item'
= c_local item nocons
}

> _____ end _eqspec._check_eq_item_n
> ame _____
- local value ```.`item''''
= local value ``0'''
- sreturn local value ``value'''
= sreturn local value ``0'''
- class exit ``value'''
= class exit ``0'''
}

> _____ end _eqspec.i

```

```

> tem -----
      - class exit ``s(value)''
      = class exit ``0''
-----

> ----- end _eqlist
> .eq -----
      = local noconsi ``0''
      - local dim = `dim' + `:list sizeof xvarsi'
      = local dim = + 18
      - if ``noconsi'' == "0" {
      = if "0" == "0" {
      - local ++dim
      - }
      - }
      - sreturn local value ``dim''
      = sreturn local value ``19''
      - class exit ``s(value)''
      = class exit ``19''
-----

> ----- end _eqlist.
> dim -----
      = global ML_dim = 19
      - forval i = 1/`eq_n' {
      = forval i = 1/1 {
      - local depvars `.`eqlist'.eq depvars `i''
-----

> ----- begin _eqlist
> .eq -----
      - version 9.2
      - args item idx
      - if ``item'' == "count" {
      = if "depvars" == "count" {
      sreturn local value ``.n_eq''
      class exit ``s(value)''
      }
      - else if 0 < `idx' & `idx' <= `.n_eq' {
      = else if 0 < 1 & 1 <= 1 {
      - .eq_list[`idx'].item `item'
      = .eq_list[1].item depvars
-----

> ----- begin _eqspec.i
> tem -----
      - version 9.2
      - args item
      - .check_eq_item_name item, `item'
      = .check_eq_item_name item, depvars

```

```

> _____ begin _eqspec._check_eq_item_n
> ame _____
      - version 9.2
      - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPosure FREEParam OPTions ]
      - if c(rc) error 198
      - if `:list sizeof other' {
      = if 0 {
        local other other_markvars
      }
      - local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
      = local item depvars
      - if `:list sizeof item' != 1 {
      = if 1 != 1 {
        error 198
      }
      - c_local `c_item' `item'
      = c_local item depvars
      _____
> _____ end _eqspec._check_eq_item_n
> ame _____
      - local value ```.`item'```
      = local value ``piq_timing_max```
      - sreturn local value ```.`value'```
      = sreturn local value ``piq_timing_max```
      - class exit ```.`value'```
      = class exit ``piq_timing_max```
      _____
> _____ end _eqspec.i
> tem _____
      - class exit ``s(value)```
      = class exit ``piq_timing_max```
      _____
> _____ end _eqlist
> .eq _____
      = local depvars piq_timing_max
      - local eq `.`eqlist'.eq name `i'
      _____
> _____ begin _eqlist
> .eq _____
      - version 9.2
      - args item idx
      - if ``item' == "count" {
      = if "name" == "count" {

```

```

        sreturn local value ```.n_eq'''
        class exit ```s(value)'''
    }
    - else if 0 < `idx' & `idx' <= `.n_eq' {
    = else if 0 < 1 & 1 <= 1 {
    - .eq_list[`idx'].item `item'
    = .eq_list[1].item name
    }
}

> _____ begin _eqspec.i
> tem _____
    - version 9.2
    - args item
    - .check_eq_item_name item, `item'
    = .check_eq_item_name item, name
}

> _____ begin _eqspec._check_eq_item_n
> ame _____
    - version 9.2
    - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPOsure FREEparm OPTions ]
    - if c(rc) error 198
    - if `:list sizeof other' {
    = if 0 {
        local other other_markvars
    }
    - local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
    = local item name
    - if `:list sizeof item' != 1 {
    = if 1 != 1 {
        error 198
    }
    - c_local `c_item' `item'
    = c_local item name
}

> _____ end _eqspec._check_eq_item_n
> ame _____
    - local value ```.`item''''
    = local value `piq_timing_max''
    - sreturn local value ```value'''
    = sreturn local value `piq_timing_max''
    - class exit ```value'''
    = class exit `piq_timing_max''
}

> _____ end _eqspec.i
> tem _____

```

```

- class exit ``s(value)''
= class exit "piq_timing_max"

> _____ end _eqlist
> .eq _____
= local eq piq_timing_max
- if !`:length local eq' {
= if !14 {
  local eq eq`i'
}
- if `:list eq in eqnames' {
= if 0 {
  di as err "equation ``eq'' multiply defined"
  exit 110
}
- local eqnames `eqnames' `eq'
= local eqnames piq_timing_max
- fvexpand `.`_eqlist'.eq indepvars `i'' if $ML_samp

> _____ begin _eqlist
> .eq _____
- version 9.2
- args item idx
- if ""item"" == "count" {
= if "indepvars" == "count" {
  sreturn local value ``.n_eq''
  class exit ``s(value)''
}
- else if 0 < `idx' & `idx' <= `.n_eq' {
= else if 0 < 1 & 1 <= 1 {
- .eq_list[`idx'].item `item'
= .eq_list[1].item indepvars

> _____ begin _eqspec.i
> tem _____
- version 9.2
- args item
- ._check_eq_item_name item, `item'
= ._check_eq_item_name item, indepvars

> _____ begin _eqspec._check_eq_item_n
> ame _____
- version 9.2
- capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPosure FREEParam OPTions ]
- if c(rc) error 198

```

```

- if `:list sizeof other' {
= if 0 {
  local other other_markvars
}
- local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
= local item indepvars
- if `:list sizeof item' != 1 {
= if 1 != 1 {
  error 198
}
- c_local `c_item' `item'
= c_local item indepvars

```

```

> _____ end _eqspec._check_eq_item_n
> ame _____
- local value ```.`item'```
= local value ``rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlt
> h#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.b
> nkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntw
> rth open bnkcrdit vint_early vint_mid```
- sreturn local value ```.value```
= sreturn local value ``rule_g rule_itr estimated 0b.wlth#0b.ope
> n 0b.wlth#1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntw
> rth#0b.bnkcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnk
> rdit ntwrth open bnkcrdit vint_early vint_mid```
- class exit ```.value```
= class exit ``rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth
> #1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bn
> kcrdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwr
> th open bnkcrdit vint_early vint_mid```

```

```

> _____ end _eqspec.i
> tem _____
- class exit ```.s(value)```
= class exit ``rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1
> .open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnk
> rdit 0b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth
> open bnkcrdit vint_early vint_mid```

```

```

> _____ end _eqlist
> .eq _____
= fvexpand rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#1.open
> 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnkcrdit 0
> b.ntwrth#1.bnkcrdit 1o.ntwrth#0b.bnkcrdit 1o.ntwrth#1o.bnkcrdit ntwrth open
> bnkcrdit vint_early vint_mid if __00000D

```



```

- if "`r(tsops)'" == "true" {
= if "" == "true" {
    global ML_preserve no
}
- local eq_rhs`i' ``r(varlist)''
= local eq_rhs1 `"rule_g rule_itr estimated 0b.wlth#0b.open 0b.wlth#
> 1.open 1.wlth#0b.open 1o.wlth#1o.open cb_authors_ext ln_neq 0b.ntwrth#0b.bnk
> crdit 0b.ntwrth#1.bnkcredit 1o.ntwrth#0b.bnkcredit 1o.ntwrth#1o.bnkcredit ntwrt
> h open bnkcredit vint_early vint_mid"'
- local eq_nocons`i' ``.eqlist'.eq nocons `i''

> _____ begin _eqlist
> .eq _____
    - version 9.2
    - args item idx
    - if "`item'" == "count" {
= if "nocons" == "count" {
    sreturn local value ``.n_eq"'
    class exit ``s(value)''
}
- else if 0 < `idx' & `idx' <= `.n_eq' {
= else if 0 < 1 & 1 <= 1 {
- .eq_list[`idx'].item `item'
= .eq_list[1].item nocons

> _____ begin _eqspec.i
> tem _____
    - version 9.2
    - args item
    - .check_eq_item_name item, `item'
= .check_eq_item_name item, nocons

> _____ begin _eqspec._check_eq_item_n
> ame _____
    - version 9.2
    - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPosure FREEParm OPTions ]
    - if c(rc) error 198
    - if `:list sizeof other' {
= if 0 {
    local other other_markvars
}
- local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
= local item nocons
- if `:list sizeof item' != 1 {

```

```

        = if 1 != 1 {
            error 198
        }
        - c_local `c_item' `item'
        = c_local item nocons
    _____
> _____ end _eqspec._check_eq_item_n
> ame _____
    - local value ```.`item'```
    = local value ``0```
    - sreturn local value ```.value```
    = sreturn local value ``0```
    - class exit ```.value```
    = class exit ``0```
    _____
> _____ end _eqspec.i
> tem _____
    - class exit ```.s(value)```
    = class exit ``0```
    _____
> _____ end _eqlist
> .eq _____
    = local eq_nocons1 0
    - local eq_offset`i' ``.eqlist'.eq offset `i'
    _____
> _____ begin _eqlist
> .eq _____
    - version 9.2
    - args item idx
    - if ``item' == "count" {
    = if "offset" == "count" {
        sreturn local value ```.n_eq```
        class exit ```.s(value)```
    }
    - else if 0 < `idx' & `idx' <= `.n_eq' {
    = else if 0 < 1 & 1 <= 1 {
        - .eq_list[`idx'].item `item'
        = .eq_list[1].item offset
    _____
> _____ begin _eqspec.i
> tem _____
    - version 9.2
    - args item
    - .check_eq_item_name item, `item'
    = .check_eq_item_name item, offset
    _____

```

```

> _____ begin _eqspec._check_eq_item_n
> ame _____
      - version 9.2
      - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPosure FREEParam OPTions ]
      - if c(rc) error 198
      - if `:list sizeof other' {
      = if 0 {
        local other other_markvars
      }
      - local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
      = local item      offset
      - if `:list sizeof item' != 1 {
      = if 1 != 1 {
        error 198
      }
      - c_local `c_item' `item'
      = c_local item offset
      _____

> _____ end _eqspec._check_eq_item_n
> ame _____
      - local value ```.`item'```
      = local value ````
      - sreturn local value ```.value```
      = sreturn local value ````
      - class exit ```.value```
      = class exit ````
      _____

> _____ end _eqspec.i
> tem _____
      - class exit ```.s(value)```
      = class exit ````
      _____

> _____ end _eqlist
> .eq _____
      = local eq_offset1
      - local eq_exposure`i' `.`eqlist'.eq exposure `i'
      _____

> _____ begin _eqlist
> .eq _____
      - version 9.2
      - args item idx
      - if ``item' == "count" {
      = if "exposure" == "count" {
        sreturn local value ```.n_eq```

```

```

        class exit ``s(value)'''
    }
    - else if 0 < `idx' & `idx' <= `.n_eq' {
    = else if 0 < 1 & 1 <= 1 {
    - .eq_list[`idx'].item `item'
    = .eq_list[1].item exposure
    }
}

> _____ begin _eqspec.i
> tem _____
    - version 9.2
    - args item
    - .check_eq_item_name item, `item'
    = .check_eq_item_name item, exposure
    }
}

> _____ begin _eqspec._check_eq_item_n
> ame _____
    - version 9.2
    - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPOsure FREEparm OPTions ]
    - if c(rc) error 198
    - if `:list sizeof other' {
    = if 0 {
        local other other_markvars
    }
    - local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
    = local item exposure
    - if `:list sizeof item' != 1 {
    = if 1 != 1 {
        error 198
    }
    - c_local `c_item' `item'
    = c_local item exposure
    }
}

> _____ end _eqspec._check_eq_item_n
> ame _____
    - local value ``.`item''''
    = local value ``0'''
    - sreturn local value ``value'''
    = sreturn local value ``0'''
    - class exit ``value'''
    = class exit ``0'''
    }
}

> _____ end _eqspec.i
> tem _____
    - class exit ``s(value)'''

```

```

        = class exit ``0''
    _____
> _____ end _eqlist
> .eq _____
    = local eq_exposure1 0
    - local eq_freeparm`i' ``.eqlist'.eq freeparm `i''
    _____
> _____ begin _eqlist
> .eq _____
    - version 9.2
    - args item idx
    - if ``item'' == "count" {
    = if "freeparm" == "count" {
        sreturn local value ``.n_eq''
        class exit ``s(value)''
    }
    - else if 0 < `idx' & `idx' <= `.n_eq' {
    = else if 0 < 1 & 1 <= 1 {
        - .eq_list[`idx'].item `item'
    = .eq_list[1].item freeparm
    _____
> _____ begin _eqspec.i
> tem _____
    - version 9.2
    - args item
    - .check_eq_item_name item, `item'
    = .check_eq_item_name item, freeparm
    _____
> _____ begin _eqspec._check_eq_item_n
> ame _____
    - version 9.2
    - capture syntax name(name=c_item) [, NAME DEPvars INDEPvars I
> F IN WGT OTHER NOCONS OFFset EXPosure FREEParM OPTions ]
    - if c(rc) error 198
    - if `:list sizeof other' {
    = if 0 {
        local other other_markvars
    }
    - local item `name' `depvars' `indepvars' `if' `in' `wgt' `oth
> er' `nocons' `offset' `exposure' `freeparm' `options'
    = local item freeparm
    - if `:list sizeof item' != 1 {
    = if 1 != 1 {
        error 198
    }
    - c_local `c_item' `item'

```

```

        = c_local item freeparm
    _____
> _____ end _eqspec._check_eq_item_n
> ame _____
    - local value ```.`item'```
    = local value ``0```
    - sreturn local value ```value'```
    = sreturn local value ``0```
    - class exit ```value'```
    = class exit ``0```
    _____
> _____ end _eqspec.i
> tem _____
    - class exit ```s(value)'```
    = class exit ``0```
    _____
> _____ end _eqlist
> .eq _____
    = local eq_freeparm1 0
    - local lhs `lhs' `depvars'
    = local lhs piq_timing_max
    - local eq_names `eq_names' `eq'
    = local eq_names piq_timing_max
    - local nxvars :list sizeof eq_rhs`i'
    = local nxvars :list sizeof eq_rhs1
    - local TMP : copy local eq_rhs`i'
    = local TMP : copy local eq_rhs1
    - forval j = 1/`nxvars' {
    = forval j = 1/18 {
    - gettoken xvar TMP : TMP
    - local colna ```colna' `eq':`xvar'```
    = local colna `` piq_timing_max:rule_g```
    - }
    - gettoken xvar TMP : TMP
    - local colna ```colna' `eq':`xvar'```
    = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr```
    - }
    - gettoken xvar TMP : TMP
    - local colna ```colna' `eq':`xvar'```
    = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated```
    - }
    - gettoken xvar TMP : TMP
    - local colna ```colna' `eq':`xvar'```
    = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open```

```

```

- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrcd
> it''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar''
= local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing

```

```

> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit''
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit''
> '
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit
> piq_timing_max:1o.ntwrth#1o.bnkcrdit''
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit
> piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth''
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit
> piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_timing_max:op
> en''
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd

```



```

> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit
> piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_timing_max:op
> en piq_timing_max:bnkcrdit''
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit
> piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_timing_max:op
> en piq_timing_max:bnkcrdit piq_timing_max:vint_early''
    - }
    - gettoken xvar TMP : TMP
    - local colna ``colna' `eq':`xvar'''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit
> piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_timing_max:op
> en piq_timing_max:bnkcrdit piq_timing_max:vint_early piq_timing_max:vint_mid
> ''
    - }
    - if `eq_freeparm`i'' {
= if 0 {
    if `nxvars' {
      di as err "{it:xvars} not allowed with option {bf:freeparm}"
      exit 198
    }
    if strpos(`"eq"', ":") {
      local colna ``colna' /`eq'''
    }
    else {
      local colna ``colna' /:`eq'''
    }
  }
  - else if !`eq_nocons`i'' {
= else if !0 {
    - local colna ``colna' `eq':_cons''
      = local colna `` piq_timing_max:rule_g piq_timing_max:rule_itr piq_t
> iming_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.
> open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing
> _max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrd
> it piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit

```

```

> piq_timing_max:lo.ntwrth#lo.bnkrdit piq_timing_max:ntwrth piq_timing_max:op
> en piq_timing_max:bnkrdit piq_timing_max:vint_early piq_timing_max:vint_mid
> piq_timing_max:_cons''
- }
- if ``moptobj'' != '' {
= if '' != '' {
  if ``eq_offset`i'' != '' {
    if usubstr(`eq_offset`i'',1,2)=="__" {
      capture clonevar `moptobj`eq_offset`i' = `eq_offset`i'
      local dropmetoo `dropmetoo' `moptobj`eq_offset`i'
    }
  }
}
- }
- if ``moptobj'' != '' {
= if '' != '' {
  foreach y of local lhs {
    if usubstr(`y'',1,2)=="__" {
      capture clonevar `moptobj`y' = `y'
      local tmp `tmp' `moptobj`y'
      local dropmetoo `dropmetoo' `moptobj`y'
    }
    else {
      local tmp `tmp' `y'
    }
  }
  local lhs `tmp'
}
- if `waldtest' > 0 & `:length local lf0' {
= if -1 > 0 & 0 {
  opts_exclusive "waldtest() lf0()"
}
- if `:length local continue' {
= if 0 {
  if !`:length local init' {
    tempname init
    local ub0 `init'
  }
  else tempname ub0
  mat `ub0' = e(b)
  tempname v0 iv0
  mat `v0' = e(V)
  mat `iv0' = syminv(`v0')
  global ML_rank0 = colsof(`v0') - diag0cnt(`iv0')
  if !`:length local lf0' & !missing(e(ll)) {
    local lf0 `=colsof(`ub0')' `e(ll)'
  }
}
}

```

```

    }
    local ub0
  }
  - if `:length local init' {
= if 8 {
  - tempname b0
  - local dim : list sizeof colna
  - matrix `b0' = J(1,`dim',0)
  = matrix __00000F = J(1,19,0)
  - matrix colna `b0' = `colna'
  = matrix colna __00000F = piq_timing_max:rule_g piq_timing_max:rule
> _itr piq_timing_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:
> 0b.wlth#1.open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open
> piq_timing_max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth
> #0b.bnkcrdit piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b
> .bnkcrdit piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_tim
> ing_max:open piq_timing_max:bnkcrdit piq_timing_max:vint_early piq_timing_ma
> x:vint_mid piq_timing_max:_cons
    - `vv' _mkvec `b0', from(`init') update first error("initial vector"
> )
    = version 18.5: _mkvec __00000F, from(__00000B) update first error("
> initial vector")
  }
}

> ----- begin _mk
> vec -----
  - local vv : display "version " string(_caller()) ":"
  - version 6
  - gettoken matout 0 : 0, parse(", ")
  - if "`matout'"==" " | "`matout'"=="", " {
= if "__00000F"==" " | "__00000F"=="", " {
  error 198
}
  - syntax [, FROM(str) UPdate COLnames(str) ERRor(str) FIRST ]
  - if "`update'"!=" " {
= if "update"!=" " {
  - if "`colname'"!=" " {
= if ""!=" " {
    di in red "update and colnames() cannot both be " "specified"
    exit 198
  }
  - capture local colname : colfullnames `matout'
= capture local colname : colfullnames __00000F
  - if _rc {
    di in red "matrix `matout' not found"
    exit 111
  }
}
}

```

```

- if rowsof(matrix(`matout`)) != 1 {
= if rowsof(matrix(__00000F)) != 1 {
  di in red ``matout' matrix to be updated must be " "a row vector
> "

  exit 503
  }
- local matin `matout'
= local matin __00000F
- }
- if ``error'!='' {
= if ``initial vector'!='' {
- local error ``error': ''
= local error ``initial vector: ''
- }
- tempname mat
- if ``from'!='' {
= if ``__00000B'!='' {
- From `from'
= From __00000B
}

> _____ begin _mkvec.F
> rom _____
  - gettoken from 0 : 0, parse(",") bind
  - sret clear
  - sret local from `from'
= sret local from __00000B
  - if ``0'!='' {
= if ``'!='' {
    syntax [, COPY SKIP ]
  }
  - sret local copy `copy'
= sret local copy
  - sret local skip `skip'
= sret local skip
}

> _____ end _mkvec.F
> rom _____
  - local copy `s(copy)'
= local copy
  - local skip `s(skip)'
= local skip
  - if ``copy'!='' {
= if ``'!='' {
    CopyMat `mat' ``error' `s(from)'
    if ``colname'!='' {
      local dim : list sizeof colname

```

```

        if `dim' != colsof(matrix(`mat')) {
        di in red `"'error'matrix must "' `be dimension "' `dim'
        exit 503
        }
        `vv' mat colnames `mat' = `colname'
        }
    }
- else {
- `vv' BuildMat `mat' `"'error'"' "'skip'" `s(from)'
= version 18.5: BuildMat __00000G `initial vector: "' "' __00000B
_____
> _____ begin _mkvec.Build
> Mat _____
    - local vv : display "version " string(_caller()) ":"
    - version 6
    - args matall error skip
    - macro shift 3
    - tokenize `"'*'"', parse(" =")
    = tokenize `__00000B"', parse(" =")
    - tempname matadd
    - local i 1
    - while `"'i'"'!="' {
    = while "__00000B"!="' {
    - local k = `i' + 1
    = local k = 1 + 1
    - if `"'k'"'=="' {
    = if "'=="' {
        local j = `i' + 2
        `vv' MkMat `matadd' `i' `j' `skip'
        local i = `i' + 3
    }
    - else {
    - ChkMat `matadd' `i' `"'error'"'
    = ChkMat __00000H __00000B `initial vector: "'
    _____
> _____ begin _mkvec.Chk
> Mat _____
    - args matout matin error
    - capture di matrix(`matin'[1,1])
    = capture di matrix(__00000B[1,1])
    - if _rc {
        capture confirm number `matin'
        if _rc {
            di in red `"'error'matrix `matin' not found"'
            exit 111
        }
    }

```

```

        di in red ``error'list of numbers requires copy option''
        exit 198
    }
    - if matrix(rowsof(`matin')!=1 & colsof(`matin')!=1) {
    = if matrix(rowsof(__00000B)!=1 & colsof(__00000B)!=1) {
        di in red ``matin' " matrix(rowsof(`matin')) " x " matrix(co
> lsof(`matin')) " is not a vector"
        exit 503
    }
    - if matrix(rowsof(`matin')) == 1 {
    = if matrix(rowsof(__00000B)) == 1 {
        - mat `matout' = `matin'
        = mat __00000H = __00000B
        - }
        - else mat `matout' = `matin''
        = else mat __00000H = __00000B'
    }
}


---


> _____ end _mkvec.Chk
> Mat _____
    - local i = `i' + 1
    = local i = 1 + 1
    - }
    - mat `matall' = nullmat(`matall') , `matadd'
    = mat __00000G = nullmat(__00000G) , __00000H
    - }
    - while ``i'!='' {
    = while ''!='' {
        local k = `i' + 1
        if ``k'=='' {
            local j = `i' + 2
            `vv' MkMat `matadd' ``i' ``j' `skip'
            local i = `i' + 3
        }
        else {
            ChkMat `matadd' ``i' ``error''
            local i = `i' + 1
        }
        mat `matall' = nullmat(`matall') , `matadd'
    }
}


---


> _____ end _mkvec.Build
> Mat _____
    - if ``colname'!='' {
    = if `piq_timing_max:rule_g piq_timing_max:rule_itr piq_timing_ma
> x:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.open piq
> _timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing_max:cb_

```

```

> authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrdit piq_t
> iming_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit piq_timi
> ng_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_timing_max:open piq_t
> iming_max:bnkcrdit piq_timing_max:vint_early piq_timing_max:vint_mid piq_tim
> ing_max:_cons''!=''' {
    - `vv' ChkNames `mat' ``matin'`` ``colname'`` ``error'`` ``skip''
> ``first''
    = version 18.5: ChkNames __00000G "__00000F" ``piq_timing_max:rule
> _g piq_timing_max:rule_itr piq_timing_max:estimated piq_timing_max:0b.wlth#0
> b.open piq_timing_max:0b.wlth#1.open piq_timing_max:1.wlth#0b.open piq_timin
> g_max:1o.wlth#1o.open piq_timing_max:cb_authors_ext piq_timing_max:ln_neq pi
> q_timing_max:0b.ntwrth#0b.bnkcrdit piq_timing_max:0b.ntwrth#1.bnkcrdit piq_t
> iming_max:1o.ntwrth#0b.bnkcrdit piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_tim
> ing_max:ntwrth piq_timing_max:open piq_timing_max:bnkcrdit piq_timing_max:vi
> nt_early piq_timing_max:vint_mid piq_timing_max:_cons'' ``initial vector: ''
> "" "first"

> _____ begin _mkvec.ChkNa
> mes _____
    - local vv : display "version " string(_caller()) ":"
    - version 6
    - args mat matin names error skip first
    - local dim = colsof(matrix(`mat'))
    = local dim = colsof(matrix(__00000G))
    - if ``first''!=''' {
    = if "first"!=''' {
    - gettoken name : names
    - local eq1 = bsubstr(```name'``,1,index(```name'``,":"))
    = local eq1 = bsubstr(`"piq_timing_max:rule_g"',1,index(`"piq_ti
> ming_max:rule_g"',":"))
    - }
    - tempname new fill x
    - local k : list sizeof names
    - if ``matin'``!=''' {
    = if "__00000F"!=''' {
    - mat `new' = `matin'
    = mat __00000H = __00000F
    - }
    - else {
    mat `new' = J(1,`k',0)
    `vv' mat colnames `new' = `names'
    }
    - mat `fill' = J(1,`k',0)
    = mat __00000I = J(1,19,0)
    - local i 1
    - while `i' <= `dim' {

```

```

= while 1 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,1..1]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
  }
- else {
- local spec "`eq':`name'"
= local spec "_:rule_g"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "_:rule_g"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse rule_g, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)'"
    local ospec "`eq':`oname'"
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,"_:rule_g'')
- if `j' == . & ``oname'" != "" {
= if 1 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & ``sspec'" != "" {
= if 1 == . & "_:rule_g" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 1==. & ``_'""=="_" & ``piq_timing_max:'""!="" {
    `vv' local j = colnumb(`new',``eq1``name'')
    if `j' == . & ``oname'" != "" {
      `vv' local j = colnumb(`new',``eq1``oname'')
    }
  }
}

```



```

    }
- if `j'!=. {
= if 1!=. {
- if `fill'[1,`j'] {
= if __00000I[1,1] {
    _ms_parse_parts `name'
    if !r(omit) {
    if "`eq'"=="_" {
    di in red `"'error'duplicate "' `"'entries for `name' found"'
    }
    else di in red `"'error'"' `"'duplicate entries for `eq':`name'
> found"'

    exit 507
    }
    else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
    }
    }
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,1] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,1] = __00000G[1,1]
- }
- else if "`skip'"==" " {
= else if ""==" " {
    if "`eq'"=="_" {
    local spec : copy local name
    }
    di in red `"'error'extra parameter "' `"'`spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
    }
- local i = `i' + 1
= local i = 1 + 1
- }
- while `i' <= `dim' {
= while 2 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,2..2]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"

```

```

    }
- else {
- local spec "`eq':`name'"
= local spec "_:rule_itr"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:rule_itr"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse rule_itr, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)''
    local ospec "`eq':`oname'"
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_:rule_itr')
- if `j' == . & ``oname' != "" {
= if 2 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec' != "" {
= if 2 == . & "/_:rule_itr" != "" {
  `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 2==. & ``_"=="_" & ``piq_timing_max:''!="" {
  `vv' local j = colnumb(`new',``eq1``name'')
  if `j' == . & ``oname' != "" {
    `vv' local j = colnumb(`new',``eq1``oname'')
  }
}
- if `j'!=. {
= if 2!=. {
- if `fill'[1,`j'] {
= if __00000I[1,2] {
  _ms_parse_parts `name'
  if !r(omit) {
    if "`eq'=="_" {
      di in red ``error'duplicate "" ``entries for `name' found''
    }
  }
}

```

```

else di in red ``error'' ``duplicate entries for `eq':`name'
> found''
    exit 507
}
else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,2] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,2] = __00000G[1,2]
- }
- else if ``skip'')=="" {
= else if ""==" {
    if ``eq'')==_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 2 + 1
- }
- while `i' <= `dim' {
= while 3 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,3..3]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if ``eq'')== "/" {
= if "_" == "/" {
    local spec "/`name'"
}
- else {
- local spec ``eq':`name'"
= local spec "_:estimated"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:estimated"
- }
- }
}

```

```

- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse estimated, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)'''
    local ospec ``eq':`name'''
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:estimated'')
- if `j' == . & ``oname''' != "" {
= if 3 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec''' != "" {
= if 3 == . & "/_:estimated" != "" {
  `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'''=="_" & ``eq1'''!="" {
= if 3==. & ``_'=="_" & ``piq_timing_max:''!="" {
  `vv' local j = colnumb(`new',``eq1``name'')
  if `j' == . & ``oname''' != "" {
    `vv' local j = colnumb(`new',``eq1``oname'')
  }
}
- if `j'!=. {
= if 3!=. {
- if `fill'[1,`j'] {
= if __00000I[1,3] {
  _ms_parse_parts `name'
  if !r(omit) {
    if ``eq'''=="_" {
      di in red ``error'duplicate "'`entries for `name' found"'
    }
    else di in red ``error''''`duplicate entries for `eq':`name'
> found'''
  exit 507
}
else if !missing(`new'[1,`j']) {
  matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,3] = 1

```

```

- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,3] = __00000G[1,3]
- }
- else if "`skip'"==" {
= else if ""==" {
    if "`eq'"=="_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 3 + 1
- }
- while `i' <= `dim' {
= while 4 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,4..4]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'
}
- else {
- local spec "`eq':`name'"
= local spec "_:0b.wlth#0b.open"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "/_:0b.wlth#0b.open"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 0b.wlth#0b.open, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)'"
= local oname ``0.wlth#0.open'"
- local ospec "`eq':`oname'"
= local ospec "_:0.wlth#0.open"

```

```

- }
- }
- `vv' local j = colnumb(`new',`""`spec'')
= version 18.5: local j = colnumb(__00000H,`":0b.wlth#0b.open"')
> )

- if `j' == . & ""`oname'" != "" {
= if 4 == . & "0.wlth#0.open" != "" {
  `vv' local j = colnumb(`new',`""`ospec'')
  }
- if `j' == . & ""`sspec'" != "" {
= if 4 == . & "/_0b.wlth#0b.open" != "" {
  `vv' local j = colnumb(`new',`""`sspec'')
  }
- if `j'==. & ""`eq'""=="_" & ""`eq1'""!="" {
= if 4==. & ""_'"=="_" & ""`piq_timing_max:'"!="" {
  `vv' local j = colnumb(`new',`""`eq1'`name'')
  if `j' == . & ""`oname'" != "" {
    `vv' local j = colnumb(`new',`""`eq1'`oname'')
  }
}
- if `j'!=. {
= if 4!=. {
- if `fill'[1,`j'] {
= if __00000I[1,4] {
  _ms_parse_parts `name'
  if !r(omit) {
    if ""`eq'""=="_" {
      di in red ""`error'duplicate "'`"entries for `name' found""
    }
    else di in red ""`error'"" "`duplicate entries for `eq':`name'
> found""

  exit 507
  }
  else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
  }
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,4] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,4] = __00000G[1,4]
- }
- else if ""`skip'""=="" {
= else if ""=="" {
  if ""`eq'""=="_" {
    local spec : copy local name

```

```

    }
    di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
  }
  - local i = `i' + 1
= local i = 4 + 1
  - }
  - while `i' <= `dim' {
= while 5 <= 19 {
  - mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,5..5]
  - local eq : coleq `x'
= local eq : coleq __00000J
  - local name : colname `x'
= local name : colname __00000J
  - if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
  }
  - else {
  - local spec "`eq':`name'"
= local spec "_:0b.wlth#1.open"
  - if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
  - local sspec "/"`eq':`name'"
= local sspec "/_:0b.wlth#1.open"
  - }
  - }
  - local oname
  - local ospec
  - capture _msparse `name', noomit
= capture _msparse 0b.wlth#1.open, noomit
  - if c(rc) == 0 {
  - if r(omit) {
    local oname ``r(stripe)'"
    local ospec "`eq':`oname'"
  }
  - }
  - `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_:0b.wlth#1.open'')
  - if `j' == . & ``oname'" != "" {
= if 5 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
  - if `j' == . & ``sspec'" != "" {

```

```

= if 5 == . & "/_:#0b.wlth#1.open" != "" {
  `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=='_' & ``eq1'!='' {
= if 5==. & ``_'=='_' & ``piq_timing_max:''!='' {
  `vv' local j = colnumb(`new',``eq1'`name'')
  if `j' == . & ``oname' != "" {
    `vv' local j = colnumb(`new',``eq1'`oname'')
  }
}
- if `j'!=. {
= if 5!=. {
- if `fill'[1,`j'] {
= if __00000I[1,5] {
  _ms_parse_parts `name'
  if !r(omit) {
    if ``eq'=='_' {
      di in red ``error'duplicate "'`entries for `name' found"'
    }
    else di in red ``error'""`duplicate entries for `eq':`name'
  }
}
exit 507
}
else if !missing(`new'[1,`j']) {
  matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,5] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,5] = __00000G[1,5]
- }
- else if ``skip'=='' {
= else if ''=='' {
  if ``eq'=='_' {
    local spec : copy local name
  }
  di in red ``error'extra parameter "'`spec' found"' _n "spe
> cify skip option if necessary"
  exit 111
}
- local i = `i' + 1
= local i = 5 + 1
- }
- while `i' <= `dim' {
= while 6 <= 19 {

```



```

- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,6..6]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_.1.wlth#0b.open"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "_.1.wlth#0b.open"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1.wlth#0b.open, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)'"
    local ospec "`eq':`oname'"
}
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_.1.wlth#0b.open'')
- if `j' == . & ``oname' != "" {
= if 6 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec' != "" {
= if 6 == . & "_.1.wlth#0b.open" != "" {
    `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 6==. & ``_"=="_" & ``piq_timing_max:''!="" {
    `vv' local j = colnumb(`new',``eq1``name'')
    if `j' == . & ``oname' != "" {
        `vv' local j = colnumb(`new',``eq1``oname'')
    }
}
}

```

```

- if `j'!=. {
= if 6!=. {
- if `fill'[1,`j'] {
= if __00000I[1,6] {
    _ms_parse_parts `name'
    if !r(omit) {
    if "`eq'"=="_" {
    di in red ``error'duplicate "' `"'entries for `name' found"'
    }
    else di in red ``error'''' `duplicate entries for `eq':`name'

> found"'

    exit 507
    }
    else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
    }
    }
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,6] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,6] = __00000G[1,6]
- }
- else if "`skip'"==" " {
= else if ""==" " {
    if "`eq'"=="_" {
    local spec : copy local name
    }
    di in red ``error'extra parameter "' `"'spec' found"' _n "spe

> cify skip option if necessary"
    exit 111
    }
- local i = `i' + 1
= local i = 6 + 1
- }
- while `i' <= `dim' {
= while 7 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,7..7]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
    }
}

```

```

- else {
- local spec "`eq':`name'"
= local spec "_:1o.wlth#1o.open"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:1o.wlth#1o.open"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1o.wlth#1o.open, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)''
= local oname ``1.wlth#1.open''
- local ospec "`eq':`oname'"
= local ospec "_:1.wlth#1.open"
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:1o.wlth#1o.open''

> )

- if `j' == . & "`oname'" != "" {
= if 7 == . & "1.wlth#1.open" != "" {
`vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & "`sspec'" != "" {
= if 7 == . & "/_:1o.wlth#1o.open" != "" {
`vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'""=="_ & ``eq1'""!=" {
= if 7==. & ``_""=="_ & ``piq_timing_max:'""!=" {
`vv' local j = colnumb(`new',``eq1``name'')
if `j' == . & "`oname'" != "" {
`vv' local j = colnumb(`new',``eq1``oname'')
}
}
- if `j'!=. {
= if 7!=. {
- if `fill'[1,`j'] {
= if __00000I[1,7] {
_ms_parse_parts `name'
if !r(omit) {
if "`eq'""=="_ {

```

```

        di in red ``error'duplicate "' ``entries for `name' found''
    }
    else di in red ``error'' ``duplicate entries for `eq':`name'
> found''

    exit 507
    }
    else if !missing(`new'[1,`j']) {
        matrix `mat'[1,`i'] = `new'[1,`j']
    }
    }
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,7] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,7] = __00000G[1,7]
- }
- else if ``skip'""=="" {
= else if ""=="" {
    if ``eq'""=="_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "' ``spec' found'' _n "spe
> cify skip option if necessary"
    exit 111
    }
- local i = `i' + 1
= local i = 7 + 1
- }
- while `i' <= `dim' {
= while 8 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,8..8]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if ``eq'"" == "/" {
= if "_" == "/" {
    local spec "/`name'"
    }
- else {
- local spec ``eq':`name'"
= local spec "_:cb_authors_ext"
- if (usubstr("eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:cb_authors_ext"

```

```

- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse cb_authors_ext, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)'''
    local ospec ``eq':`name'''
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:cb_authors_ext'')
- if `j' == . & ``oname'' != "" {
= if 8 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & ``sspec'' != "" {
= if 8 == . & "/_:cb_authors_ext" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq''=="_" & ``eq1''!="" {
= if 8==. & ``_'=="_" & ``piq_timing_max:''!="" {
    `vv' local j = colnumb(`new',``eq1``name'')
    if `j' == . & ``oname'' != "" {
      `vv' local j = colnumb(`new',``eq1``oname'')
    }
  }
- if `j'!=. {
= if 8!=. {
- if `fill'[1,`j'] {
= if __00000I[1,8] {
    _ms_parse_parts `name'
    if !r(omit) {
      if ``eq''=="_" {
        di in red ``error'duplicate '' ``entries for `name' found''
      }
      else di in red ``error'' ``duplicate entries for `eq':`name'
> found''

    exit 507
  }
  else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
  }
}
}

```

```

- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,8] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,8] = __00000G[1,8]
- }
- else if "`skip'"==" {
= else if ""==" {
  if "`eq'"=="_" {
    local spec : copy local name
  }
  di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
  exit 111
}
- local i = `i' + 1
= local i = 8 + 1
- }
- while `i' <= `dim' {
= while 9 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,9..9]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
  local spec "/`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_:ln_neq"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:ln_neq"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse ln_neq, noomit
- if c(rc) == 0 {
- if r(omit) {
  local oname ``r(stripe)'"
  local ospec "`eq':`oname'"

```

```

    }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_:ln_neq'')
- if `j' == . & ``oname' != "" {
= if 9 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
    }
- if `j' == . & ``sspec' != "" {
= if 9 == . & "/_:ln_neq" != "" {
    `vv' local j = colnumb(`new',``sspec'')
    }
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 9==. & ``_'=="_" & ``piq_timing_max:'!="" {
    `vv' local j = colnumb(`new',``eq1`name'')
    if `j' == . & ``oname' != "" {
        `vv' local j = colnumb(`new',``eq1`oname'')
    }
    }
- if `j'!=. {
= if 9!=. {
- if `fill'[1,`j'] {
= if __00000I[1,9] {
    _ms_parse_parts `name'
    if !r(omit) {
        if ``eq'=="_" {
            di in red ``error'duplicate "" ``entries for `name' found''
        }
        else di in red ``error'' ``duplicate entries for `eq':`name'
> found''

    exit 507
    }
    else if !missing(`new'[1,`j']) {
        matrix `mat'[1,`i'] = `new'[1,`j']
    }
    }
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,9] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,9] = __00000G[1,9]
- }
- else if ``skip'=="" {
= else if ""=="" {
    if ``eq'=="_" {
        local spec : copy local name
    }

```

```

        di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 9 + 1
- }
- while `i' <= `dim' {
= while 10 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,10..10]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_:0b.ntwrth#0b.bnkcrcdit"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "/_:0b.ntwrth#0b.bnkcrcdit"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 0b.ntwrth#0b.bnkcrcdit, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)'"
= local oname ``0.ntwrth#0.bnkcrcdit'"
- local ospec "`eq':`oname'"
= local ospec "_:0.ntwrth#0.bnkcrcdit"
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:0b.ntwrth#0b.bnkc
> rdit'')
- if `j' == . & "`oname'" != "" {
= if 10 == . & "0.ntwrth#0.bnkcrcdit" != "" {
    `vv' local j = colnumb(`new',``ospec'')

```



```

    }
- if `j' == . & "`sspec'" != "" {
= if 10 == . & "/*_0b.ntwrth#0b.bnkcrcdit" != "" {
  `vv' local j = colnumb(`new', "`sspec'")
}
- if `j'==. & "`eq'"=="_" & "`eq1'"!="" {
= if 10==. & "`_"=="_" & "`piq_timing_max:'"!="" {
  `vv' local j = colnumb(`new', "`eq1`name'")
  if `j' == . & "`oname'" != "" {
    `vv' local j = colnumb(`new', "`eq1`oname'")
  }
}
- if `j'!=. {
= if 10!=. {
- if `fill'[1,`j'] {
= if __00000I[1,10] {
  _ms_parse_parts `name'
  if !r(omit) {
    if "`eq'"=="_" {
      di in red "`error'duplicate "' `"'entries for `name' found"'
    }
    else di in red "`error'" "`duplicate entries for `eq':`name'
> found"'

    exit 507
  }
  else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
  }
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,10] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,10] = __00000G[1,10]
- }
- else if "`skip'"=="" {
= else if ""=="" {
  if "`eq'"=="_" {
    local spec : copy local name
  }
  di in red "`error'extra parameter "' `"'spec' found"' _n "spe
> cify skip option if necessary"
  exit 111
}
- local i = `i' + 1
= local i = 10 + 1
- }

```

```

- while `i' <= `dim' {
= while 11 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,11..11]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
  }
- else {
- local spec "`eq':`name'"
= local spec "_:0b.ntwrth#1.bnkcrcdit"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:0b.ntwrth#1.bnkcrcdit"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 0b.ntwrth#1.bnkcrcdit, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)''
    local ospec "`eq':`oname'"
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_:0b.ntwrth#1.bnkcrc
> dit'')

- if `j' == . & "`oname'" != "" {
= if 11 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & "`sspec'" != "" {
= if 11 == . & "/_:0b.ntwrth#1.bnkcrcdit" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 11==. & ``_""=="_" & ``piq_timing_max:'""!="" {
    `vv' local j = colnumb(`new',``eq1``name'')
    if `j' == . & "`oname'" != "" {

```

```

        `vv' local j = colnumb(`new', ``eq1'`oname'``)
    }
}
- if `j'!=. {
= if 11!=. {
- if `fill'[1,`j'] {
= if __00000I[1,11] {
    _ms_parse_parts `name'
    if !r(omit) {
        if ``eq'=="_" {
            di in red ``error'duplicate "'`entries for `name' found"'
        }
        else di in red ``error'""' ``duplicate entries for `eq':`name'
> found"'
    }
    exit 507
}
else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,11] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,11] = __00000G[1,11]
- }
- else if ``skip'==" " {
= else if ""==" " {
    if ``eq'=="_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "'`spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 11 + 1
- }
- while `i' <= `dim' {
= while 12 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,12..12]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if ``eq' == "/" {

```

```

= if "_" == "/" {
    local spec "/`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_:1o.ntwrth#0b.bnkcrcdit"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:1o.ntwrth#0b.bnkcrcdit"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1o.ntwrth#0b.bnkcrcdit, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)''
= local oname ``1.ntwrth#0.bnkcrcdit''
- local ospec "`eq':`oname'"
= local ospec "_:1.ntwrth#0.bnkcrcdit"
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:1o.ntwrth#0b.bnkc
> rdit'')
- if `j' == . & ``oname' != "" {
= if 12 == . & "1.ntwrth#0.bnkcrcdit" != "" {
    `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec' != "" {
= if 12 == . & "/_:1o.ntwrth#0b.bnkcrcdit" != "" {
    `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 12==. & ``_"=="_" & ``piq_timing_max:' != "" {
    `vv' local j = colnumb(`new',``eq1`name'')
    if `j' == . & ``oname' != "" {
        `vv' local j = colnumb(`new',``eq1`oname'')
    }
}
- if `j'!=. {
= if 12!=. {
- if `fill'[1,`j'] {
= if __00000I[1,12] {

```

```

        _ms_parse_parts `name'
        if !r(omit) {
            if "`eq'"=="_" {
                di in red `""error'duplicate "' `"'entries for `name' found"'
            }
            else di in red `""error'"' `"'duplicate entries for `eq':`name'
> found"'

        exit 507
        }
        else if !missing(`new'[1,`j']) {
            matrix `mat'[1,`i'] = `new'[1,`j']
        }
        }
        - else mat `fill'[1,`j'] = 1
        = else mat __00000I[1,12] = 1
        - mat `new'[1,`j'] = `mat'[1,`i']
        = mat __00000H[1,12] = __00000G[1,12]
        - }
        - else if "`skip'"==" " {
        = else if ""==" " {
            if "`eq'"=="_" {
                local spec : copy local name
            }
            di in red `""error'extra parameter "' `""spec' found"' _n "spe
> cify skip option if necessary"
            exit 111
        }
        - local i = `i' + 1
        = local i = 12 + 1
        - }
        - while `i' <= `dim' {
        = while 13 <= 19 {
            - mat `x' = `mat'[1,`i'..`i']
            = mat __00000J = __00000G[1,13..13]
            - local eq : coleq `x'
            = local eq : coleq __00000J
            - local name : colname `x'
            = local name : colname __00000J
            - if "`eq'" == "/" {
            = if "_" == "/" {
                local spec "/"`name'"
            }
            - else {
            - local spec "`eq':`name'"
            = local spec "_.ntwrth#1o.bnkcrdit"
            - if (usubstr("`eq'",1,1) != "/") {

```

```

= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:1o.ntwrth#1o.bnkcrcdit"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1o.ntwrth#1o.bnkcrcdit, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)''
= local oname ``1.ntwrth#1.bnkcrcdit''
- local ospec ``eq':`oname''
= local ospec "_:1.ntwrth#1.bnkcrcdit"
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:1o.ntwrth#1o.bnkc
> rdit'')
- if `j' == . & ``oname'' != "" {
= if 13 == . & "1.ntwrth#1.bnkcrcdit" != "" {
`vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec'' != "" {
= if 13 == . & "/_:1o.ntwrth#1o.bnkcrcdit" != "" {
`vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq''=="_" & ``eq1''!="" {
= if 13==. & ``_''=="_" & ``piq_timing_max:''!="" {
`vv' local j = colnumb(`new',``eq1``name'')
if `j' == . & ``oname'' != "" {
`vv' local j = colnumb(`new',``eq1``oname'')
}
}
- if `j'!=. {
= if 13!=. {
- if `fill'[1,`j'] {
= if __00000I[1,13] {
_ms_parse_parts `name'
if !r(omit) {
if ``eq''=="_" {
di in red ``error'duplicate '' ``entries for `name' found''
}
else di in red ``error'' ``duplicate entries for `eq':`name'
> found''

```

```

        exit 507
    }
    else if !missing(`new'[1,`j']) {
        matrix `mat'[1,`i'] = `new'[1,`j']
    }
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,13] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,13] = __00000G[1,13]
- }
- else if "`skip'"==" {
= else if ""==" {
    if "`eq'"=="_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 13 + 1
- }
- while `i' <= `dim' {
= while 14 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,14..14]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_:ntwrth"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:ntwrth"
- }
- }
- local oname
- local ospec

```

```

- capture _msparse `name', noomit
= capture _msparse ntwrth, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)'''
    local ospec ``eq':`oname'''
  }
- }
- `vv' local j = colnumb(`new',``spec''')
= version 18.5: local j = colnumb(__00000H,``_:ntwrth'')
- if `j' == . & ``oname''' != "" {
= if 14 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec''')
}
- if `j' == . & ``sspec''' != "" {
= if 14 == . & "/_:ntwrth" != "" {
  `vv' local j = colnumb(`new',``sspec''')
}
- if `j'==. & ``eq'''=="_" & ``eq1'''!="" {
= if 14==. & ``_'=="_" & ``piq_timing_max:'!="" {
  `vv' local j = colnumb(`new',``eq1``name''')
  if `j' == . & ``oname''' != "" {
    `vv' local j = colnumb(`new',``eq1``oname''')
  }
}
- if `j'!=. {
= if 14!=. {
- if `filll'[1,`j'] {
= if __00000I[1,14] {
  _ms_parse_parts `name'
  if !r(omit) {
    if ``eq'''=="_" {
      di in red ``error'duplicate "' ``entries for `name' found"'
    }
    else di in red ``error'''' ``duplicate entries for `eq':`name'
> found"'
  }
  exit 507
}
else if !missing(`new'[1,`j']) {
  matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `filll'[1,`j'] = 1
= else mat __00000I[1,14] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,14] = __00000G[1,14]

```



```

- }
- else if "`skip'"==" {
= else if ""==" {
  if "`eq'"=="_" {
    local spec : copy local name
  }
  di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
  exit 111
}
- local i = `i' + 1
= local i = 14 + 1
- }
- while `i' <= `dim' {
= while 15 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,15..15]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
  local spec "/`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_:open"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "_:open"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse open, noomit
- if c(rc) == 0 {
- if r(omit) {
  local oname ``r(stripe)'"
  local ospec "`eq':`oname'"
}
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:open'')

```

```

- if `j' == . & ""`oname'" != "" {
= if 15 == . & "" != "" {
  `vv' local j = colnumb(`new',`""`ospec'')
}
- if `j' == . & ""`sspec'" != "" {
= if 15 == . & "/_:"open" != "" {
  `vv' local j = colnumb(`new',`""`sspec'')
}
- if `j'==. & ""`eq'""=="_" & ""`eq1'""!="" {
= if 15==. & ""_"'=="_" & ""`piq_timing_max:'""!="" {
  `vv' local j = colnumb(`new',`""`eq1'`name'')
  if `j' == . & ""`oname'" != "" {
    `vv' local j = colnumb(`new',`""`eq1'`oname'')
  }
}
- if `j'!=. {
= if 15!=. {
- if `fill'[1,`j'] {
= if __00000I[1,15] {
  _ms_parse_parts `name'
  if !r(omit) {
    if ""`eq'""=="_" {
      di in red ""`error'duplicate "'`entries for `name' found""
    }
    else di in red ""`error'"" "`duplicate entries for `eq':`name'
> found""
  }
  exit 507
}
else if !missing(`new'[1,`j']) {
  matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,15] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,15] = __00000G[1,15]
- }
- else if ""`skip'""=="" {
= else if ""=="" {
  if ""`eq'""=="_" {
    local spec : copy local name
  }
  di in red ""`error'extra parameter "'`""`spec' found"" _n "spe
> cify skip option if necessary"
  exit 111
}

```

```

- local i = `i' + 1
= local i = 15 + 1
- }
- while `i' <= `dim' {
= while 16 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,16..16]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
    }
- else {
- local spec "`eq':`name'"
= local spec "_:bnkcrdit"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "_:bnkcrdit"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse bnkcrdit, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)'"
    local ospec "`eq':`oname'"
    }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_:bnkcrdit'')
- if `j' == . & ``oname'" != "" {
= if 16 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
    }
- if `j' == . & ``sspec'" != "" {
= if 16 == . & "_:bnkcrdit" != "" {
    `vv' local j = colnumb(`new',``sspec'')
    }
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 16==. & ``_'""=="_" & ``piq_timing_max:'""!="" {

```

```

        `vv' local j = colnumb(`new', ``eq1'`name'`)
        if `j' == . & ``oname' != "" {
            `vv' local j = colnumb(`new', ``eq1'`oname'`)
        }
    }
- if `j' != . {
= if 16 != . {
- if `fill'[1,`j'] {
= if __00000I[1,16] {
    _ms_parse_parts `name'
    if !r(omit) {
        if ``eq'=="_" {
            di in red ``error'duplicate "'`entries for `name' found"'
        }
        else di in red ``error'""' "duplicate entries for `eq':`name'
> found"'

        exit 507
    }
    else if !missing(`new'[1,`j']) {
        matrix `mat'[1,`i'] = `new'[1,`j']
    }
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,16] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,16] = __00000G[1,16]
- }
- else if ``skip'==" " {
= else if ""==" " {
    if ``eq'=="_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "'`spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 16 + 1
- }
- while `i' <= `dim' {
= while 17 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,17..17]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'

```

```

= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
  }
- else {
- local spec "`eq':`name'"
= local spec "_:vint_early"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "_:vint_early"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse vint_early, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)''
    local ospec "`eq':`oname'"
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,`_:vint_early')
- if `j' == . & ``oname' != "" {
= if 17 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & ``sspec' != "" {
= if 17 == . & "_:vint_early" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'=="_" & ``eq1'!='' {
= if 17==. & ``_'=="_" & ``piq_timing_max:'!=' {
    `vv' local j = colnumb(`new',``eq1`name'')
    if `j' == . & ``oname' != "" {
      `vv' local j = colnumb(`new',``eq1`oname'')
    }
  }
- if `j'!=. {
= if 17!=. {
- if `filll'[1,`j'] {
= if __00000I[1,17] {
    _ms_parse_parts `name'

```

```

        if !r(omit) {
        if "`eq'"=="_" {
        di in red `"'error'duplicate "' `"'entries for `name' found"'
        }
        else di in red `"'error'"'"' `"'duplicate entries for `eq':`name'
> found"'

        exit 507
        }
        else if !missing(`new'[1,`j']) {
        matrix `mat'[1,`i'] = `new'[1,`j']
        }
        }
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,17] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,17] = __00000G[1,17]
- }
- else if "`skip'"==" " {
= else if ""==" " {
    if "`eq'"=="_" {
        local spec : copy local name
    }
    di in red `"'error'extra parameter "' `"'`spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
    }
- local i = `i' + 1
= local i = 17 + 1
- }
- while `i' <= `dim' {
= while 18 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,18..18]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
    }
- else {
- local spec "`eq':`name'"
= local spec "_:vint_mid"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {

```

```

- local sspec "/`eq':`name'"
= local sspec "/_:vint_mid"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse vint_mid, noomit
- if c(rc) == 0 {
- if r(omit) {
    local oname ``r(stripe)''
    local ospec ``eq':`oname'
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:vint_mid'')
- if `j' == . & ``oname' != "" {
= if 18 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec' != "" {
= if 18 == . & "/_:vint_mid" != "" {
  `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 18==. & ``_'=="_" & ``piq_timing_max:'!="" {
  `vv' local j = colnumb(`new',``eq1``name'')
  if `j' == . & ``oname' != "" {
    `vv' local j = colnumb(`new',``eq1``oname'')
  }
}
- if `j'!=. {
= if 18!=. {
- if `fill'[1,`j'] {
= if __00000I[1,18] {
  _ms_parse_parts `name'
  if !r(omit) {
    if ``eq'=="_" {
      di in red ``error'duplicate "" ``entries for `name' found''
    }
    else di in red ``error'' ``duplicate entries for `eq':`name'
  }
}
> found''

exit 507
}
else if !missing(`new'[1,`j']) {
matrix `mat'[1,`i'] = `new'[1,`j']

```

```

    }
    }
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,18] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,18] = __00000G[1,18]
- }
- else if "`skip'"==" {
= else if ""==" {
    if "`eq'"=="_" {
        local spec : copy local name
    }
    di in red ``error'extra parameter "' `"'spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 18 + 1
- }
- while `i' <= `dim' {
= while 19 <= 19 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000J = __00000G[1,19..19]
- local eq : coleq `x'
= local eq : coleq __00000J
- local name : colname `x'
= local name : colname __00000J
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
}
- else {
- local spec "`eq':`name'"
= local spec ":_cons"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:_cons"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse _cons, noomit
- if c(rc) == 0 {
- if r(omit) {

```



```

        local oname ``r(stripe)''
        local ospec ``eq':`oname''
    }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__00000H,``_:cons'')
- if `j' == . & ``oname'' != "" {
= if 19 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
    }
- if `j' == . & ``sspec'' != "" {
= if 19 == . & "/_:cons" != "" {
    `vv' local j = colnumb(`new',``sspec'')
    }
- if `j'==. & ``eq''=="_" & ``eq1''!="" {
= if 19==. & ``_'=="_" & ``piq_timing_max:''!="" {
    `vv' local j = colnumb(`new',``eq1`name'')
    if `j' == . & ``oname'' != "" {
        `vv' local j = colnumb(`new',``eq1`oname'')
    }
}
- if `j'!=. {
= if 19!=. {
- if `fill'[1,`j'] {
= if __00000I[1,19] {
    _ms_parse_parts `name'
    if !r(omit) {
        if ``eq''=="_" {
            di in red ``error'duplicate ''`entries for `name' found''
        }
        else di in red ``error'' ``duplicate entries for `eq':`name'
> found''
    }
    exit 507
}
else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
}
}
- else mat `fill'[1,`j'] = 1
= else mat __00000I[1,19] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __00000H[1,19] = __00000G[1,19]
- }
- else if ``skip''=="" {
= else if ""=="" {
    if ``eq''=="_" {

```

```

        local spec : copy local name
    }
    di in red ``error'extra parameter "' ``spec' found"' _n "spe
> cify skip option if necessary"
    exit 111
}
- local i = `i' + 1
= local i = 19 + 1
- }
- while `i' <= `dim' {
= while 20 <= 19 {
    mat `x' = `mat'[1,`i'..`i']
    local eq : coleq `x'
    local name : colname `x'
    if ``eq' == "/" {
        local spec "/`name'"
    }
    else {
        local spec ``eq':`name'"
        if (usubstr("eq",1,1) != "/") {
            local sspec "/`eq':`name'"
        }
    }
    local oname
    local ospec
    capture _msparse `name', noomit
    if c(rc) == 0 {
        if r(omit) {
            local oname ``r(stripe)'"
            local ospec ``eq':`oname'"
        }
    }
    `vv' local j = colnumb(`new',``spec'')
    if `j' == . & ``oname' != "" {
        `vv' local j = colnumb(`new',``ospec'')
    }
    if `j' == . & ``sspec' != "" {
        `vv' local j = colnumb(`new',``sspec'')
    }
    if `j'==. & ``eq'=="_" & ``eq1'!='' {
        `vv' local j = colnumb(`new',``eq1`name'')
    }
    if `j' == . & ``oname' != "" {
        `vv' local j = colnumb(`new',``eq1`oname'')
    }
}
if `j'!=. {

```

```

        if `fill'[1,`j'] {
        _ms_parse_parts `name'
        if !r(omit) {
        if "`eq'"=="_" {
        di in red `"'`error'duplicate "' `"'entries for `name' found"'
        }
        else di in red `"'`error'"'"' `"'duplicate entries for `eq':`name'
> found"'
        exit 507
        }
        else if !missing(`new'[1,`j']) {
        matrix `mat'[1,`i'] = `new'[1,`j']
        }
        }
        else mat `fill'[1,`j'] = 1
        mat `new'[1,`j'] = `mat'[1,`i']
        }
        else if "`skip'"==" " {
        if "`eq'"=="_" {
        local spec : copy local name
        }
        di in red `"'`error'extra parameter "' `"'`spec' found"' _n "spe
> cify skip option if necessary"
        exit 111
        }
        local i = `i' + 1
        }
        - mat `fill' = `fill'*`fill'
        = mat __00000I = __00000I*__00000I'
        - sret clear
        - sret local k_fill = `fill'[1,1]
        = sret local k_fill = __00000I[1,1]
        - mat `mat' = `new'
        = mat __00000G = __00000H
        _____
> _____ end _mkvec.ChkNa
> mes _____
        - local fill `s(k_fill)'
        = local fill 19
        - }
        - }
        - }
        - }
        - else {
        local fill 0
        if "`update'"!=" " {
        mat `mat' = `matin'

```

```

    }
    else if ``colname''!='' {
        local dim : list sizeof colname
        mat `mat' = J(1,`dim',0)
        `vv' mat colnames `mat' = `colname'
    }
    else {
        di in red "from() must be specified"
        exit 198
    }
}

- sret clear
- sret local copy `copy'
= sret local copy
- sret local skip `skip'
= sret local skip
- sret local k = colsof(matrix(`mat'))
= sret local k = colsof(matrix(__00000G))
- if ``fill''!='' {
= if "19"!='' {
- sret local k_fill `fill'
= sret local k_fill 19
- }
- else sret local k_fill = colsof(matrix(`mat'))
= else sret local k_fill = colsof(matrix(__00000G))
- matrix rename `mat' `matout', replace
= matrix rename __00000G __00000F, replace

```

```

> _____ end _mk
> vec _____
    - }
    - tempname bomit
    - matrix `bomit' = J(1,`:list sizeof colna',0)
    = matrix __00000G = J(1,19,0)
    - matrix colna `bomit' = `colna'
    = matrix colna __00000G = piq_timing_max:rule_g piq_timing_max:rule
> _itr piq_timing_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:
> 0b.wlth#1.open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open
> piq_timing_max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth
> #0b.bnkcrdit piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b
> .bnkcrdit piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_tim
> ing_max:open piq_timing_max:bnkcrdit piq_timing_max:vint_early piq_timing_ma
> x:vint_mid piq_timing_max:_cons
    - _ms_omit_info `bomit'
    = _ms_omit_info __00000G
    - if r(k_omit) | `:length local constraints' {

```

```

= if r(k_omit) | 0 {
- nobreak {
- capture noisily break {
- _b_post0 `colna'
= _b_post0 piq_timing_max:rule_g piq_timing_max:rule_itr piq_timing
> _max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_max:0b.wlth#1.open
> piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.open piq_timing_max:
> cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.ntwrth#0b.bnkcrdit pi
> q_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrth#0b.bnkcrdit piq_t
> iming_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq_timing_max:open pi
> q_timing_max:bnkcrdit piq_timing_max:vint_early piq_timing_max:vint_mid piq_
> timing_max:_cons

> _____ begin _b_po
> st0 _____
- version 11
- tempname b V
- local dim : list sizeof 0
- matrix `b' = J( 1,`dim',0)
= matrix __00000H = J( 1,19,0)
- matrix `V' = J(`dim',`dim',0)
= matrix __00000I = J(19,19,0)
- matrix colnames `b' = `0'
= matrix colnames __00000H = piq_timing_max:rule_g piq_timing_max:
> rule_itr piq_timing_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_
> max:0b.wlth#1.open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.o
> pen piq_timing_max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.nt
> wrth#0b.bnkcrdit piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrt
> h#0b.bnkcrdit piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq
> _timing_max:open piq_timing_max:bnkcrdit piq_timing_max:vint_early piq_timin
> g_max:vint_mid piq_timing_max:_cons
- matrix colnames `V' = `0'
= matrix colnames __00000I = piq_timing_max:rule_g piq_timing_max:
> rule_itr piq_timing_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_
> max:0b.wlth#1.open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.o
> pen piq_timing_max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.nt
> wrth#0b.bnkcrdit piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrt
> h#0b.bnkcrdit piq_timing_max:1o.ntwrth#1o.bnkcrdit piq_timing_max:ntwrth piq
> _timing_max:open piq_timing_max:bnkcrdit piq_timing_max:vint_early piq_timin
> g_max:vint_mid piq_timing_max:_cons
- matrix rownames `V' = `0'
= matrix rownames __00000I = piq_timing_max:rule_g piq_timing_max:
> rule_itr piq_timing_max:estimated piq_timing_max:0b.wlth#0b.open piq_timing_
> max:0b.wlth#1.open piq_timing_max:1.wlth#0b.open piq_timing_max:1o.wlth#1o.o
> pen piq_timing_max:cb_authors_ext piq_timing_max:ln_neq piq_timing_max:0b.nt
> wrth#0b.bnkcrdit piq_timing_max:0b.ntwrth#1.bnkcrdit piq_timing_max:1o.ntwrt

```

```

> h#0b.bnkcrcdit piq_timing_max:10.ntwrth#10.bnkcrcdit piq_timing_max:ntwrth piq
> _timing_max:open piq_timing_max:bnkcrcdit piq_timing_max:vint_early piq_timin
> g_max:vint_mid piq_timing_max:_cons
    - ereturn post `b' `V'
    = ereturn post __00000H __00000I
    _____
> _____ end _b_po
> st0 _____
    - local clist : substr local constraints "," " ", all
    - `vv' makecns `clist', `cnsnotes'
    = version 18.5: makecns ,
    _____
> _____ begin make
> cns _____
    - if _caller() < 11 {
      MakeCns `0'
      exit
    }
    - _makecns `0'
    = _makecns ,
    _____
> _____ end make
> cns _____
    - local constraints ``r(clist)''
    = local constraints `''''
    - local k_autoCns = e(k_autoCns)
    - tempname C
    - capture matrix `C' = e(Cns)
    = capture matrix __00000H = e(Cns)
    - local rcc = c(rc)
    - }
    - local rc = c(rc)
    - ereturn clear
    - }
    - if (`rc') exit `rc'
    = if (0) exit 0
    - if (`rcc') {
    = if (0) {
      local C
      local k_autoCns
    }
    - }
    - if `:length local constraints' {
    = if 0 {
      local lf0
    }

```

```

- local vce `vce' `robust' `cluster'
= local vce robust
- if `:length local vce' {
= if 6 {
- _vce_parse, argoptlist(Cluster) optlist(OIM OPG Robust) old : `wt'
> , `vce'
= _vce_parse, argoptlist(Cluster) optlist(OIM OPG Robust) old : , ro
> bust
_____
> _____ begin _vce_pa
> rse _____
- version 10
- _on_colon_parse `0'
= _on_colon_parse , argoptlist(Cluster) optlist(OIM OPG Robust) ol
> d : , robust
_____
> _____ begin _on_colon_pa
> rse _____
- version 8.2
- sreturn local before ""
- sreturn local after ""
- gettoken before after : 0, parse(":") bind match(par) quotes
- if "`par'" != "" {
= if "" != "" {
local before `("`before')'
}
- if ``"before"' == ":" {
= if `", argoptlist(Cluster) optlist(OIM OPG Robust) old "" == "
> ":" {
sreturn local after ``"after"'
exit
}
- while ``COLON"' != ":" & ``"after"' != "" {
= while `"' != ":" & `": , robust"' != "" {
- gettoken COLON after : after, parse(":") bind match(par) quote
> s
- if "`par'" != "" {
= if "" != "" {
local before `before' (`COLON')
local COLON
}
- else if ``COLON"' != ":" {
= else if `":' != ":" {
local before ``"before' `COLON"'
local COLON
}

```

```

- }
- while `""COLON"" != ":" & `""after"" != "" {
= while `":"" != ":" & `", robust"" != "" {
  gettoken COLON after : after, parse(":") bind match(par) quote
> s
    if "`par'" != "" {
      local before `before' (`COLON')
      local COLON
    }
    else if `""COLON"" != ":" {
      local before `""before' `COLON""
      local COLON
    }
  }
- if `""COLON"" != ":" {
= if `":"" != ":" {
  di as err "" found where ':' expected"
  exit 198
}
- sreturn local before `""before""
= sreturn local before `", argoptlist(CList) optlist(OIM OPG R
> obust) old ""
  - sreturn local after `""after""
  = sreturn local after `", robust""
}

> _____ end _on_colon_pa
> rse _____
  - local 0 `""s(before)""
  = local 0 `", argoptlist(CList) optlist(OIM OPG Robust) old ""
  - local user `""s(after)""
  = local user `", robust""
  - syntax [varname(default=none)], [ OPTlist(namelist) ARGOPTlist(n
> amelist) Pwallowed(namelist) OLD_is_ok NOARGOPT_is_ok ]
  - local touse `varlist'
  = local touse
  - tempname o a
  - `o' = `._optlist.new
  = `._00000I = `._optlist.new
  - local optlist : list uniq optlist
  - foreach name of local optlist {
  - `o'.addopt `name'
  = `._00000I.addopt OIM
}

> _____ begin _optlist.add
> opt _____
  - version 9

```



```

- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("OIM")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "OIM"
- .names.Arrpush "`mname'"
= .names.Arrpush "oim"

```

```

> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__00000I.addopt OPG

```

```

> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("OPG")
- if "`passthru'" != "" {
= if "" != "" {
    local optspec `optspec'(passthru)
  }
- .options.Arrpush "`optspec'"
= .options.Arrpush "OPG"
- .names.Arrpush "`mname'"
= .names.Arrpush "opg"

```

```

> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__00000I.addopt Robust

```

```

> _____ begin _optlist.add
> opt _____
- version 9
- syntax name(name=optspec) [, PASSthru]
- local mname = lower("`optspec'")
= local mname = lower("Robust")
- if "`passthru'" != "" {

```

```

= if "" != "" {
    local optspec `optspec'(passthru)
}
- .options.Arrpush "`optspec'"
= .options.Arrpush "Robust"
- .names.Arrpush "`mname'"
= .names.Arrpush "robust"

```

```

> _____ end _optlist.add
> opt _____
    - }
    - .`a' = ._optlist.new
    = __00000J = ._optlist.new
    - local argoptlist : list uniq argoptlist
    - foreach name of local argoptlist {
    - .`a'.addopt `name'
    = __00000J.addopt CLuster

```

```

> _____ begin _optlist.add
> opt _____
    - version 9
    - syntax name(name=optspec) [, PASSthru]
    - local mname = lower("`optspec'")
    = local mname = lower("CLuster")
    - if "`passthru'" != "" {
    = if "" != "" {
        local optspec `optspec'(passthru)
    }
    - .options.Arrpush "`optspec'"
    = .options.Arrpush "CLuster"
    - .names.Arrpush "`mname'"
    = .names.Arrpush "cluster"

```

```

> _____ end _optlist.add
> opt _____
    - }
    - local OPTNAMES ```.`o'.dumpnames'''

```

```

> _____ begin _optlist.dumpna
> mes _____
    - version 9
    - forval i = 1/`.names.arnels' {
    = forval i = 1/3 {
    - local namelist `namelist' `.names['i']'
    = local namelist oim
    - }

```

```

- local namelist `namelist' `.names['i']'
= local namelist oim opg
- }
- local namelist `namelist' `.names['i']'
= local namelist oim opg robust
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist `oim opg robust'
- class exit ``namelist'''
= class exit `oim opg robust'

> _____ end _optlist.dumpna
> mes _____
= local OPTNAMES `oim opg robust'
- local ARGOPTNAMES ```.a'.dumpnames'''

> _____ begin _optlist.dumpna
> mes _____
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/1 {
- local namelist `namelist' `.names['i']'
= local namelist cluster
- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist `cluster'
- class exit ``namelist'''
= class exit `cluster'

> _____ end _optlist.dumpna
> mes _____
= local ARGOPTNAMES `cluster'
- local AND : list OPTNAMES & ARGOPTNAMES
- if `:list sizeof AND' {
= if 0 {
di as err "vcetype `:word 1 of `AND'' specified twice"
exit 198
}
- local pwallowed : list uniq pwallowed
- local pwlist robust cluster
- foreach n of local pwallowed {
local lc = lower("`n'")
local pwlist `pwlist' `lc'
}

```

```

- if "`old_is_ok'" != "" {
= if "old_is_ok" != "" {
- local ROBUST robust
- if `:list ROBUST in OPTNAMES' {
= if 1 {
- local robust Robust
- }
- local CLUSTER cluster
- if `:list CLUSTER in ARGOPTNAMES' {
= if 1 {
- local cluster CLuster(varname)
- }
- }
- local 0 ``user'''
= local 0 `` , robust''
- syntax [fw aw pw iw] [, VCE(string asis) `robust' `cluster' ]
= syntax [fw aw pw iw] [, VCE(string asis) Robust CLuster(varname)

> ]

- if ``vce''' != "" {
= if `''' != "" {
gettoken name args : vce, parse(" ,")
local args : list retok args
capture ParseVCE `o' `a' , `name'
if c(rc) {
di as err "vcetype {bf:`name'} not allowed"
exit 198
}
local name `r(vce)'
if "`weight'" == "pweight" & !`:list name in pwlist' {
di as err "pweights are not allowed with vce(`name')"
exit 198
}
return add
if ``robust''' != "" & !inlist("`name'", "", "robust", "eim", "oim"
> , "cluster") {
opts_exclusive "`robust' vce(`name')"
}
if "`name'" == "cluster" {
if ``args''' == "" {
di as err "invalid vce(cluster) option"
exit 198
}
confirm var `args'
unab args : `args'
local markout `args'
if "`cluster'" != "" {

```

```

if !`:list args == cluster' {
di as err "options cluster() and vce(cluster) are in conflict"
exit 100
}
}
}
else if "`cluster'" != "" & !inlist("`name'", "eim", "oim") {
opts_exclusive "cluster() vce(`name')"
}
if ``args' != "" {
if !`:list name in ARGOPTNAMES' {
di as err "invalid vce() option"
exit 198
}
if "`name'" == "hc2" {
confirm var `args'
unab args : `args'
local markout `args'
}
return local vceargs ```:list retok args'''
}
else if `:list name in ARGOPTNAMES' {
if "`noargopt_is_ok'" == "" {
di as err "invalid vce() option"
exit 198
}
}
}
- else {
- if "`weight'" == "pweight" | "`robust'" != "" {
= if "" == "pweight" | "robust" != "" {
- local vce robust
- }
- if "`cluster'" != "" {
= if "" != "" {
local vce cluster
local args `cluster'
local markout `cluster'
}
- return local vce ``vce'''
= return local vce `robust'
- return local vceargs ``args'''
= return local vceargs `'''
- }
- if "`return(vce)'" != "" {
= if "robust" != "" {

```

```

- if ``return(vceargs)'' != "" {
= if `""' != "" {
    local ss " "
  }
- return local vceopt `vce(`return(vce)''ss'`return(vceargs))'
= return local vceopt `vce(robust)'
- }
- if "`return(vce)'" == "cluster" {
= if "robust" == "cluster" {
    return local cluster `return(vceargs)'
  }
- if inlist("`return(vce)'", "robust", "cluster") {
= if inlist("robust", "robust", "cluster") {
- return local robust robust
- }
- if ""touse'' != "" {
= if "" != "" {
    if ""markout'' != "" {
      markout `touse' `markout', strok
    }
  }
}

```

```

> _____ end _vce_pa
> rse _____

```

```

- local vce `r(vce)'
= local vce robust
- }
- else if ""weight'' == "pweight" {
= else if "" == "pweight" {
    local vce robust
  }
- if `:length local svy' {
= if 0 {
    local vce svy
  }
- if ``vce'' == "cluster" {
= if `robust' == "cluster" {
    local cluster `r(cluster)'
    markout $ML_sample `cluster', strok
  }
- if `:length local group' {
= if 0 {
    markout $ML_sample `group', strok
    global ML_group : copy local group
    if bsubstr("`type `group'",1,3) == "str" {
      if `interactive' {

```

```

SetGlobalVar ML_grp : _MLgr
}
else {
global ML_grp : tempvar
if ```moptobj`'' != "" {
global ML_grp `moptobj'$ML_grp
}
}
quietly egen $ML_grp = group(`group')
if ! `:length local cluster' {
local cluster : copy global ML_grp
}
}
else {
if ! `:length local cluster' {
local cluster : copy global ML_group
}
}
}
- if "$ML_samp" != "$ML_sample" {
= if "__00000D" != "__00000D" {
quietly replace $ML_samp = 0 if $ML_samp & $ML_sample == 0
}
- if `:length local vce' == 0 {
= if 6 == 0 {
if ```technique`'' == "bhhh" {
local vce opg
}
else local vce oim
}
- if inlist("`vce'", "robust", "cluster") & bsubstr("`evaltype'",1,1
> ) == "d" {
= if inlist("robust", "robust", "cluster") & bsubstr("e2",1,1) == "d
> " {
local optname = cond("`svy'"=="", "vce(`vce')", "svy")
di as err "option `optname' is not allowed with evaltype `evaltype
> ""
exit 198
}
- if `:length local crittype' == 0 {
= if 0 == 0 {
- if inlist("`vce'", "cluster", "robust", "svy") {
= if inlist("robust", "cluster", "robust", "svy") {
- local crittype "log pseudolikelihood"
- }
- else local crittype "log likelihood"

```

```

- }
- global ML_waldtest `waldtest'
= global ML_waldtest -1
- if `:length local svy' | `:length local weight' {
= if 0 | 0 {
    if `interactive' {
        SetGlobalVar ML_w : _MLw
    }
    else {
        global ML_w : tempvar
        if ``moptobj' != "" {
            global ML_w `moptobj'$ML_w
        }
    }
}
- if `:length local weight' {
= if 0 {
    if ``moptobj' != "" {
        cap drop $ML_w
    }
    quietly gen double $ML_w `exp' if $ML_samp
    local wtype ``weight'
    local wexp ``exp'
    global ML_wtyp : copy local wtype
    global ML_wexp : copy local wexp
    if "wtype" == "iweight" | "svy" != "" {
        local zeroweight zeroweight
    }
}
- if inlist("`vce'", "svy", "cluster") {
= if inlist("robust", "svy", "cluster") {
    if `interactive' {
        SetGlobalVar ML_subv : _MLsu
    }
    else {
        global ML_subv : tempvar
        if ``moptobj' != "" {
            global ML_subv `moptobj'$ML_subv
        }
    }
}
- quietly count if $ML_sample
= quietly count if __00000D
- if r(N) == 0 {
    error 2000
}

```



```

- if ``weight'' != "" {
= if ``'' != "" {
  quietly count if $ML_sample & $ML_w != 0
  if r(N) == 0 {
    di in gr "all observation have zero weights"
    exit 2000
  }
}
- if `:length local lf0' {
= if 0 {
  if `:list sizeof lf0' != 2 {
    di as err "lf0(`lf0') invalid"
    exit 198
  }
  global ML_k0 : word 1 of `lf0'
  global ML_ll_0 : word 2 of `lf0'
  confirm integer number $ML_k0
  confirm number $ML_ll_0
  if $ML_k0 < 0 {
    di as err "lf0(`lf0') invalid"
    exit 198
  }
}
- if ``obs'' == "" {
= if "" == "" {
- if ``wtype'' != "fweight" {
= if "" != "fweight" {
- quietly count if $ML_sample
= quietly count if __00000D
- local obs = r(N)
- }
- else {
  sum $ML_w if $ML_sample, mean
  local obs = r(sum)
}
- }
- if `:length local iterid' {
= if 0 {
  local iterid = trim(bsubstr(```iterid'',1,32))
}
- if `:length local crittype' {
= if 20 {
- local crittype = trim(bsubstr(```crittype'',1,32))
= local crittype = trim(bsubstr(`"log pseudolikelihood"',1,32))
- }
- if `:length local derivscale' {

```

```

= if 0 {
  confirm matrix `derivscale'
}
- if `:length local derivh' {
= if 0 {
  confirm matrix `derivh'
}
- local l : strlen local deriv
- ParseUsemin, `deriv' len(`l')
= ParseUsemin, len(0)
}

> ----- begin mopt.ParseUse
> min -----
  - capture syntax [, LEN(integer 0), MIN1(numlist min=2 max=2 missi
> ngokay) NOMIN MIN]
  - if (_rc) {
    syntax [, LEN(integer 0) MIN1(string) NOMIN MIN]
  }
  - if ``min1'' != "" & ``nomin'' != "" {
= if `'' != "" & `'' != "" {
  di as err "options {bf:min()} and {bf:nomin} are mutually exclus
> ive"

  exit 198
}
- if ``min1'' != "" & ``nomin'' != "" {
= if `'' != "" & `'' != "" {
  di as err "options {bf:min} and {bf:nomin} are mutually exclusiv
> e"

  exit 198
}
- if (``min1'' != "") {
= if (``'' != "") {
  if (_rc) {
    capture confirmmatrix `min1'
    if (_rc) {
      di as err "option {bf:min()} requires that either two numbers or
> a 1X2 matrix be specified"
      exit 198
    }
  }
  local deriv `min1'
}
- else if (``min1'' != "") {
= else if (``'' != "") {
  if (`len' > length(`min1')) {
    di as err "option {bf:min()} requires that either two numbers or

```

```

> a 1X2 matrix be specified"
    exit 198
  }
  local deriv 1
}
- else {
- if (`len' > 0) {
= if (0 > 0) {
  if ("`nomin'" == "") {
    di as err "option {bf:min()} requires that either two numbers or
> a 1X2 matrix be specified"
    exit 198
  }
  else if (`len' > length("`nomin'")) {
    di as err "options {bf:min()} and {bf:nomin} are mutually exclus
> ive"
    exit 198
  }
  else {
    local deriv 0
  }
}
- else {
- local deriv 0
- }
- }
- sreturn local deriv `deriv'
= sreturn local deriv 0
}

> _____ end mopt.ParseUse
> min _____
  - local deriv = s(deriv)
  - if `:length local gnwmatrix' {
= if 0 {
  confirm matrix `gnwmatrix'
}
  - `vv' mata: Mopt_model()
= version 18.5: mata: Mopt_model()
}

> _____ begin mopt_check_prog
> ram _____
  - version 10
  - args name
  - capture confirm name `name'
= capture confirm name glim_lf
  - if c(rc) {

```

```

        di as err "unrecognized command:  `name' invalid command name"
        exit 199
    }
    - capture which `name'
    = capture which glim_lf
    - if !c(rc) {
    - exit
}
> _____ end mopt_check_prog
> ram _____
    - global ML_stat model
    - if !`interactive' {
    = if !0 {
    - `vv' MaxMin `maxmin', nooutput `score' `options' moptobj(`moptobj'
> )
    = version 18.5: MaxMin maximize, nooutput  nolog search(off) iter(30
> 0) moptobj()
    _____
> _____ begin mopt.Max
> Min _____
    - local vv : di "version " string(_caller()) ":"
    - version 12
    - ChkMacros
    _____
> _____ begin mopt.ChkMac
> ros _____
    - if "$ML_stat" != "model" {
    = if "model" != "model" {
        di as err "you must issue -ml model- first"
        exit 198
    }
    - capture confirm var $ML_samp $ML_w $ML_subv
    = capture confirm var __00000D __00000D
    - if c(rc) {
        di as err "Since issuing the -ml model- statement, you have do
> ne something to drop" _n "temporary variables ml added to your data.  You mu
> st start again."
        Clear
        exit 111
    }
    _____
> _____ end mopt.ChkMac
> ros _____
    - syntax name(name=maxmin) [, NEGH TOLerance(numlist max=1 >=0) LT
> OLerance(numlist max=1 >=0) NONRTOLerance NRTOLerance(numlist max=1 >=0) QT0
> Lerance(numlist max=1 >=0) GTOLerance(numlist max=1 >=0) ITERate(numlist max

```

```

> =1 >=0) NOTCONCAVE(numlist max=1 >=0) ndami DIFFicult HALFSTEPonly D0OPT SEA
> rch(name) Repeat(passthru) Bounds(string) SCore(string) noWARNING noCLEAR NO
> L0g L0g DOTs SHOWTOLerance SHOWNRtolerance SHOWSTEP TRace COEFDiffs GRADient
> HESSian noSKIPline noOUTput Level(cilevel) moptobj(string) * ]
    - if `:length local nrtolerance' {
    = if 0 {
      opts_exclusive "nrtolerance() `nonrtolerance'"
    }
    - if `:length local qtolerance' {
    = if 0 {
      opts_exclusive "qtolerance() `nonrtolerance'"
    }
    - opts_exclusive "`nonrtolerance' `shownrtolerance'"
    = opts_exclusive " "
  }

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
      - exit
    }

> ----- end opts_exclus
> ive -----
    - if `:length local shownrtolerance' {
    = if 0 {
      local showtolerance showtolerance
    }
    - if "$ML_preserve" != "no" {
    = if "no" != "no" {
      if `:length local score' {
        di as smcl as err "May not specify {bf:score()} option unless" _
> n "      a) estimation subsample is the entire data in memory, or" _n "      b
> ) you specify option {bf:nopreserve} on {bf:ml model} statement (meaning" _n
> "      your evaluation program explicitly restricts itself to obs" _n "
>      for which {bf:$}" "{bf:ML_samp==1}."
      exit 198
    }
  }

```

```

    }
    preserve
    local N = c(N)
    quietly keep if $ML_sample
    if (`N' != c(N)) {
    mata: Mopt_init_regetviews()
    }
    }
- if `:length local score' {
= if 0 {
    SetupScore ``$ML_evaltype"' ``score'''' $ML_n $ML_dim
    local scores ``s(scores)''
    }
- _parse_iterlog, `nolog' `log'
= _parse_iterlog, nolog

```

```

> ----- begin _parse_iter
> log -----
    - version 16
    - syntax [, NOLOG LOG]
    - if ``log'`nolog'' != "" {
    = if "nolog" != "" {
    - opts_exclusive ``log' `nolog''
    = opts_exclusive " nolog"

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts nolog
    - local n 0
    - while ``opts'' != "" {
    = while ``nolog'' != "" {
    - local ++n
    - gettoken item`n' opts : opts, bind
    = gettoken item1 opts : opts, bind
    - }
    - while ``opts'' != "" {
    = while ``'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 1 < 2 {
    - exit

```

```

> _____ end opts_exclus
> ive _____
    - }
    - else if "`c(iterlog)'" == "on" {
= else if "on" == "on" {
    local log log
    }
    - else {
    local nolog nolog
    }
    - sreturn local log `log'
= sreturn local log
    - sreturn local nolog `nolog'
= sreturn local nolog nolog
_____

> _____ end _parse_iter
> log _____
    - local log "`s(nolog)'"
= local log "nolog"
    - local tropts `log' `dots' `showtolerance' `showstep' `trace' `co
> efdiffs'
= local tropts nolog
    - if "`tropts'" != "nolog" {
= if "nolog" != "nolog" {
    local tropts `tropts' `gradient' `hessian'
    if "`skipline'" == "" {
    if "`iterate'" != "0" | `:length local tropts' {
    di
    }
    }
    }
    - else {
    - local gradient
    - local hessian
    - local nolog nolog
    - }
    - if "`search'" != "off" & "`iterate'" != "0" {
= if "off" != "off" & "300" != "0" {
    if "`search'" == "quietly" {
    local sopts nolog
    }
    else {
    if `:length local tropts' {
    if "`tropts'" != "nolog" {
    local sopts trace

```

```

    }
    else local sopts nolog
    }
    }
    if "`search'" == "norescale" {
        local sopts `sopts' norescale
    }
    Search `bounds', nopreserve `sopts' `repeat' `maxmin'
    }
- tempname value b V iV
- `vv' mata: Mopt_maxmin()
= version 18.5: mata: Mopt_maxmin()

```

```

> ----- begin glim
> _lf -----
    - version 6
    - args todo b lnf g H score
    - local VV : di "version " string(_caller()) ":"
    - tempvar eta mu dm u v
    - mlevel `eta' = `b'
    = mlevel __00000X = __00000R
    - qui $SGLM_L 1 `eta' `mu'
    = qui glim_l03 1 __00000X __00000Y

```

```

> ----- begin glim_
> l03 -----
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
    = if 1 == -1 {
        global SGLM_lt "Log"
        if "$SGLM_m" == "1" {
            global SGLM_lf "ln(u)"
        }
        else global SGLM_lf "ln(u/$SGLM_m)"
        exit
    }
    - if `todo' == 0 {
    = if 1 == 0 {
        gen double `eta' = ln(`mu'/$SGLM_m)
        exit
    }
    - if `todo' == 1 {
    = if 1 == 1 {
        - gen double `mu' = $SGLM_m*exp(`eta')
        = gen double __00000Y = 1*exp(__00000X)

```



```

- exit
-
> ----- end glim_
> l03 -----
- qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y
-
> ----- begin glim
> _mu -----
- version 7
- args min max mu
- tempname eps
- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011
-
> ----- end glim
> _mu -----
- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z
-
> ----- begin glim_
> l03 -----
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
global SGLM_lt "Log"
if "$SGLM_m" == "1" {
global SGLM_lf "ln(u)"
}
else global SGLM_lf "ln(u/$SGLM_m)"
exit
}
- if `todo' == 0 {
= if 2 == 0 {
gen double `eta' = ln(`mu'/$SGLM_m)
exit
}
- if `todo' == 1 {
= if 2 == 1 {
gen double `mu' = $SGLM_m*exp(`eta')
exit
}
}
}

```

```

    }
    - if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = `mu'
= gen double __00000Z = __00000Y
    - exit
}
> _____ end glim_
> l03 _____
    - qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010
}
> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
    - if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
    - if `todo' == 1 {
= if 1 == 1 {
    - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
= gen double __000010 = __00000Y+__00000Y*__00000Y*1
    - exit

```

```

> _____ end glim
> _v6 _____
- if "$ML_off" == "" {
= if "" == "" {
- global ML_off 0
- }
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".2639785217643881" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .2639785217643881
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
local y "$SGLM_y"
local m "$SGLM_m"
local touse "`eta'"
capture assert `y'>=0 if `touse'
if _rc {
di as err `"dependent variable `y' has negative values"'
exit 499
}
capture assert `y'==int(`y') if `touse'
if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 5 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}

```

```

    }
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
    }
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
    }
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu*$SGLM_a))))
    exit
    }
- if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
    exit
    }
- if `todo' == 5 {
= if 5 == 5 {
    - local y "$SGLM_y"
    = local y "piq_timing_max"
    - if "`y'" == "" {
    = if "piq_timing_max" == "" {
        local y "`e(depvar)'"
    }
    - tempname sig2
    - if $SGLM_s1 {
    = if 1 {
    - scalar `sig2' = $SGLM_s1
    = scalar __000013 = 1

```

```

- }
- else {
  if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
  }
  else {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
  }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')
= local lnalpha = ln(1)
- gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/'m') + `y'*ln(`mu'/(`mu'+`m'))))
= gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
- exit

_____
> _____ end glim
> _v6 _____
- mlsun `lnf' = `lli'
= mlsun __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 2 == 0 | __00000S==. { exit }
- qui replace `score' = ($ML_y1 - `mu')/(`phi'*`v')*`dmu' if $ML
> _samp
= qui replace __00000V = (piq_timing_max - __00000Y)/(__000011*_
> _000010)*__00000Z if __00000D
- $ML_ec mlvecsum `lnf' `g' = `score'
= * mlvecsum __00000S __00000T = __00000V
- if `todo' == 1 | `lnf'==. { exit }
= if 2 == 1 | __00000S==. { exit }
- tempvar tt
- qui gen double `tt' = sum($ML_w*($ML_y1-`mu')*($ML_y1-`mu')/`v
> ')
= qui gen double __000013 = sum(__00000D*(piq_timing_max-__00000
> Y)*(piq_timing_max-__00000Y)/__000010)
- if _caller() >= 11 {
- global SGLM_ph = `tt'[_N]/($ML_N-$ML_kCns)
= global SGLM_ph = __000013[_N]/(217-14)
- }
- else {
global SGLM_ph = `tt'[_N]/($ML_N-$ML_k)
}

```

```

- qui $SGLM_A
= qui
- tempvar t1
- qui gen double `t1' = `dmu'*`dmu'/`v' if $ML_samp
= qui gen double __000014 = __00000Z*__00000Z/__000010 if __0000
> 0D

- if "$SGLM_f" == "" {
= if "-1" == "" {
    global SGLM_f = -1
}
- if $SGLM_f >= $ML_ic {
= if -1 >= 0 {
    mlfmatsum `lnf' `H' = (`t1')/`phi'
    exit
}
- tempvar dv d2mu
- qui $SGLM_V 2 `eta' `mu' `dv'
= qui glim_v6 2 __00000X __00000Y __000015

> _____ begin glim
> _v6 _____

- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
        di as err `"dependent variable `y' has negative values"'
        exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
        di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
}
- if `todo' == 0 {
= if 2 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')

```

```

        exit
    }
- if `todo' == 1 {
= if 2 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
}
- if `todo' == 2 {
= if 2 == 2 {
- gen double `return' = 1 + 2*`mu'*$SGLM_a
= gen double __000015 = 1 + 2*__00000Y*1
- exit
}
}

> _____ end glim
> _v6 _____
- qui $SGLM_L 3 `eta' `mu' `d2mu'
= qui glim_l03 3 __00000X __00000Y __000016
}

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 3 == -1 {
    global SGLM_lt "Log"
    if "$SGLM_m" == "1" {
        global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
}
- if `todo' == 0 {
= if 3 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
}
- if `todo' == 1 {
= if 3 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
}
- if `todo' == 2 {
= if 3 == 2 {
    gen double `return' = `mu'
    exit
}
}

```

```

- if `todo' == 3 {
= if 3 == 3 {
- gen double `return' = `mu'
= gen double __000016 = __00000Y
- exit
}
}

> _____ end glim_
> l03 _____
- tempvar t2 t3
- qui gen double `t2' = `t1'/'v'*`dv' if $ML_samp
= qui gen double __000017 = __000014/__000010*__000015 if __0000
> 0D
- qui gen double `t3' = `d2mu'/'v' if $ML_samp
= qui gen double __000018 = __000016/__000010 if __00000D
- mmlatsum `lnf' `H' = (`t1'-(`mu'-$ML_y1)*(`t2'-`t3'))/'phi'
= mmlatsum __00000S __00000U = (__000014-(__00000Y-piq_timing_ma
> x)*(__000017-__000018))/__000011
}
}

> _____ end glim
> _lf _____
}
}

> _____ begin glim
> _lf _____
- version 6
- args todo b lnf g H score
- local VV : di "version " string(_caller()) ":"
- tempvar eta mu dm u v
- mlevel `eta' = `b'
= mlevel __00000X = __00000R
- qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __00000X __00000Y
}
}

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
global SGLM_lt "Log"
if "$SGLM_m" == "1" {
global SGLM_lf "ln(u)"
}
else global SGLM_lf "ln(u/$SGLM_m)"
exit
}
- if `todo' == 0 {

```



```

= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
- gen double `mu' = $SGLM_m*exp(`eta')
= gen double __00000Y = 1*exp(__00000X)
- exit
}
}

> _____ end glim_
> l03 _____
- qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y
}

> _____ begin glim
> _mu _____
- version 7
- args min max mu
- tempname eps
- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011
}

> _____ end glim
> _mu _____
- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z
}

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {

```

```

= if 2 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
}
- if `todo' == 1 {
= if 2 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
}
- if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = `mu'
= gen double __00000Z = __00000Y
    - exit
}
}
> _____ end glim_
> l03 _____
    - qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010
}
> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
    - if `todo' == 0 {
= if 1 == 0 {

```

```

        gen double `eta' = -ln1p($SGLM_a/`mu')
        exit
    }
    - if `todo' == 1 {
= if 1 == 1 {
    - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
= gen double __000010 = __00000Y+__00000Y*__00000Y*1
    - exit
}

> _____ end glim
> _v6 _____
    - if "$ML_off" == "" {
= if "0" == "" {
    global ML_off 0
    }
    - tempname phi
    - if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".4003032150048503" == "" { global SGLM_ph = 1 }
    - scalar `phi' = $SGLM_ph
= scalar __000011 = .4003032150048503
    - if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
    - tempvar lli
    - qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012

> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 5 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)

```

```

global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 5 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
}
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
}
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
}
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
    exit
}
- if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
    exit
}
- if `todo' == 5 {
= if 5 == 5 {
    - local y "$SGLM_y"
    = local y "piq_timing_max"
    - if "`y'" == "" {

```

```

= if "piq_timing_max" == "" {
  local y ``e(depvar)''
}
- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
  if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
  }
  else {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
  }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')
= local lnalpha = ln(1)
- gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/`m') + `y'*ln(`mu'/(`mu'+`m'))))
= gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
- exit
}

> _____ end glim
> _v6 _____
- msum `lnf' = `lli'
= msum __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 0 == 0 | __00000S==. { exit }
}

> _____ end glim
> _lf _____
}

> _____ begin glim
> _lf _____
- version 6
- args todo b lnf g H score
- local VV : di "version " string(_caller()) ":"
- tempvar eta mu dm u v
- mlevel `eta' = `b'
= mlevel __00000X = __00000R

```

```

- qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __00000X __00000Y

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `mu' = $SGLM_m*exp(`eta')
  = gen double __00000Y = 1*exp(__00000X)
  - exit
}

> _____ end glim_
> l03 _____
- qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y

> _____ begin glim
> _mu _____
- version 7
- args min max mu
- tempname eps
- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011

> _____ end glim
> _mu _____

```

```

- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  - gen double `return' = `mu'
  = gen double __00000Z = __00000Y
  - exit
}

> _____ end glim_
> l03 _____
- qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"

```

```

capture assert `y'>=0 if `touse'
if _rc {
di as err `"dependent variable `y' has negative values"'
exit 499
}
capture assert `y'==int(`y') if `touse'
if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 1 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}
- if `todo' == 1 {
= if 1 == 1 {
- gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
= gen double __000010 = __00000Y+__00000Y*__00000Y*1
- exit
}
}

> _____ end glim
> _v6 _____
- if "$ML_off" == "" {
= if "0" == "" {
global ML_off 0
}
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".4003032150048503" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .4003032150048503
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012

> _____ begin glim
> _v6 _____
- version 7

```



```

- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
- if `todo' == 0 {
= if 5 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
    }
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
    }
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
    local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
    exit

```

```

    }
    - if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
    local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) - `mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
    exit
    }
    - if `todo' == 5 {
= if 5 == 5 {
    - local y "$SGLM_y"
= local y "piq_timing_max"
    - if "`y'" == "" {
= if "piq_timing_max" == "" {
    local y "`e(depvar)'"
    }
    - tempname sig2
    - if $SGLM_s1 {
= if 1 {
    - scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
    - }
    - else {
    if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph*($ML_N-$ML_kCns)/$ML_N)
    }
    else {
    scalar `sig2' = $SGLM_ph*($ML_N-$ML_k)/$ML_N)
    }
    }
    - local m = 1/$SGLM_a
= local m = 1/1
    - local lnalpha = ln(`m')
= local lnalpha = ln(1)
    - gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/`m') + `y'*ln(`mu'/(`mu'+`m'))))
    = gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
    - exit
}
> ----- end glim

```

```

> _v6
- mlsun `lnf' = `lli'
= mlsun __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 0 == 0 | __00000S==. { exit }

>
> _lf
>
>
> _lf
- version 6
- args todo b lnf g H score
- local VV : di "version " string(_caller()) ":"
- tempvar eta mu dm u v
- mlevel `eta' = `b'
= mlevel __00000X = __00000R
- qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __00000X __00000Y

>
> l03
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `mu' = $SGLM_m*exp(`eta')
= gen double __00000Y = 1*exp(__00000X)
- exit

>
> l03

```

```

- qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y

```

```

> begin glim
> _mu
- version 7
- args min max mu
- tempname eps
- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011

```

```

> end glim
> _mu
- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z

```

```

> begin glim_
> l03
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  gen double `return' = `mu'

```

```

= gen double __00000Z = __00000Y
- exit


---


> 

---

 end glim_
> l03 

---


- qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010


---


> 

---

 begin glim
> _v6 

---


- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
local y "$SGLM_y"
local m "$SGLM_m"
local touse "`eta'"
capture assert `y'>=0 if `touse'
if _rc {
di as err `"dependent variable `y' has negative values"'
exit 499
}
capture assert `y'==int(`y') if `touse'
if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 1 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}
- if `todo' == 1 {
= if 1 == 1 {
- gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
= gen double __000010 = __00000Y+__00000Y*__00000Y*1
- exit


---


> 

---

 end glim
> _v6 

---


- if "$ML_off" == "" {

```

```

= if "0" == "" {
  global ML_off 0
}
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".4003032150048503" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .4003032150048503
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012

```

```

> _____ begin glim
> _v6 _____
  - version 7
  - args todo eta mu return
  - if `todo' == -1 {
= if 5 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"
  local SGLM_a = round($SGLM_a,0.0001)
  global SGLM_vf "u+(`SGLM_a')u^2"
  global SGLM_mu "glim_mu 0 ."
  exit
}
- if `todo' == 0 {
= if 5 == 0 {
  gen double `eta' = -ln1p($SGLM_a/`mu')
  exit
}
- if `todo' == 1 {
= if 5 == 1 {
  gen double `return' = `mu'+`mu'*`mu'*$SGLM_a

```

```

        exit
    }
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
}
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/'mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
        exit
    }
- if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
        exit
    }
- if `todo' == 5 {
= if 5 == 5 {
- local y "$SGLM_y"
= local y "piq_timing_max"
- if "`y'" == "" {
= if "piq_timing_max" == "" {
    local y "`e(depvar)'"
}
- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
    if _caller() >= 11 {
        scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
    }
}
}

```

```

    }
    else {
        scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
    }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')
= local lnalpha = ln(1)
- gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/'m') + `y'*ln(`mu'/(`mu'+`m'))))
= gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
- exit

_____
> _____ end glim
> _v6 _____
- msum `lnf' = `lli'
= msum __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 0 == 0 | __00000S==. { exit }

_____
> _____ end glim
> _lf _____
_____
> _____ begin glim
> _lf _____
- version 6
- args todo b lnf g H score
- local VV : di "version " string(_caller()) ":"
- tempvar eta mu dmu v
- mlevel `eta' = `b'
= mlevel __00000X = __00000R
- qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __00000X __00000Y

_____
> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
global SGLM_lt "Log"
if "$SGLM_m" == "1" {
global SGLM_lf "ln(u)"

```



```

    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
  }
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `mu' = $SGLM_m*exp(`eta')
= gen double __00000Y = 1*exp(__00000X)
  - exit
}
}

> _____ end glim_
> l03 _____
    - qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y
}

> _____ begin glim
> _mu _____
    - version 7
    - args min max mu
    - tempname eps
    - scalar `eps' = 1e-8
= scalar __000011 = 1e-8
    - replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
    - replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011
}

> _____ end glim
> _mu _____
    - qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z
}

> _____ begin glim_
> l03 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"

```

```

    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
  }
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  - gen double `return' = `mu'
= gen double __00000Z = __00000Y
- exit
}
}
> _____ end glim_
> l03 _____
    - qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010
}
> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"
  local SGLM_a = round($SGLM_a,0.0001)
  global SGLM_vf "u+(`SGLM_a')u^2"

```

```

        global SGLM_mu "glim_mu 0 ."
        exit
    }
- if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
}
- if `todo' == 1 {
= if 1 == 1 {
    - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    = gen double __000010 = __00000Y+__00000Y*__00000Y*1
    - exit
}
}
> _____ end glim
> _v6 _____
    - if "$ML_off" == "" {
= if "0" == "" {
    global ML_off 0
}
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".4003032150048503" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .4003032150048503
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012
}
> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 5 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'

```

```

if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 5 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}
- if `todo' == 1 {
= if 5 == 1 {
gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
exit
}
- if `todo' == 2 {
= if 5 == 2 {
gen double `return' = 1 + 2*`mu'*$SGLM_a
exit
}
- if `todo' == 3 {
= if 5 == 3 {
local y "$SGLM_y"
if "`y'" == "" {
local y "`e(depvar)'"
}
gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
exit
}
- if `todo' == 4 {
= if 5 == 4 {
local y "$SGLM_y"
local k "$SGLM_a"
if "`y'" == "" {
local y "`e(depvar)'"
}
gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
exit
}

```

```

- if `todo' == 5 {
= if 5 == 5 {
- local y "$SGLM_y"
= local y "piq_timing_max"
- if "`y'" == "" {
= if "piq_timing_max" == "" {
  local y "`e(depvar)'"
}
- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
  if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph*($ML_N-$ML_kCns)/$ML_N
  }
  else {
    scalar `sig2' = $SGLM_ph*($ML_N-$ML_k)/$ML_N
  }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')
= local lnalpha = ln(1)
- gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/'m') + `y'*ln(`mu'/(`mu'+`m'))))
= gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*ln
> n(__00000Y/(__00000Y+1)))
- exit
}

> _____ end glim
> _v6 _____
- mlsun `lnf' = `lli'
= mlsun __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 2 == 0 | __00000S==. { exit }
- qui replace `score' = ($ML_y1 - `mu')/(`phi'*`v')*`dmu' if $ML
> _samp
= qui replace __00000V = (piq_timing_max - __00000Y)/(__000011*_
> __000010)*__00000Z if __00000D
- $ML_ec mlvecsum `lnf' `g' = `score'
= * mlvecsum __00000S __00000T = __00000V
- if `todo' == 1 | `lnf'==. { exit }

```

```

= if 2 == 1 | __00000S==. { exit }
- tempvar tt
- qui gen double `tt' = sum($ML_w*($ML_y1-`mu')*($ML_y1-`mu')/`v
> ')
= qui gen double __000013 = sum(__00000D*(piq_timing_max-__00000
> Y)*(piq_timing_max-__00000Y)/__000010)
- if _caller() >= 11 {
- global SGLM_ph = `tt'[_N]/($ML_N-$ML_kCns)
= global SGLM_ph = __000013[_N]/(217-14)
- }
- else {
global SGLM_ph = `tt'[_N]/($ML_N-$ML_k)
}
- qui $SGLM_A
= qui
- tempvar t1
- qui gen double `t1' = `dmu'*`dmu'/`v' if $ML_samp
= qui gen double __000014 = __00000Z*__00000Z/__000010 if __0000
> 0D
- if "$SGLM_f" == "" {
= if "-1" == "" {
global SGLM_f = -1
}
- if $SGLM_f >= $ML_ic {
= if -1 >= 1 {
mlmatsum `lnf' `H' = (`t1')/`phi'
exit
}
- tempvar dv d2mu
- qui $SGLM_V 2 `eta' `mu' `dv'
= qui glim_v6 2 __00000X __00000Y __000015

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
local y "$SGLM_y"
local m "$SGLM_m"
local touse "`eta'"
capture assert `y'>=0 if `touse'
if _rc {
di as err `"dependent variable `y' has negative values"'
exit 499
}
}

```

```

capture assert `y'==int(`y') if `touse'
if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 2 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}
- if `todo' == 1 {
= if 2 == 1 {
gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
exit
}
- if `todo' == 2 {
= if 2 == 2 {
- gen double `return' = 1 + 2*`mu'*$SGLM_a
= gen double __000015 = 1 + 2*__00000Y*1
- exit
}
}
}
> _____ end glim
> _v6 _____
- qui $SGLM_L 3 `eta' `mu' `d2mu'
= qui glim_l03 3 __00000X __00000Y __000016
}
> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 3 == -1 {
global SGLM_lt "Log"
if "$SGLM_m" == "1" {
global SGLM_lf "ln(u)"
}
else global SGLM_lf "ln(u/$SGLM_m)"
exit
}
- if `todo' == 0 {
= if 3 == 0 {

```



```

> l03
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `mu' = $SGLM_m*exp(`eta')
  = gen double __00000Y = 1*exp(__00000X)
  - exit
}
}

> end glim_
> l03
- qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y

> begin glim_
> _mu
- version 7
- args min max mu
- tempname eps
- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011

> end glim_
> _mu
- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z

> begin glim_

```

```

> l03 -----
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  - gen double `return' = `mu'
= gen double __00000Z = __00000Y
- exit
}

> ----- end glim_
> l03 -----
- qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010

> ----- begin glim
> _v6 -----
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
}

```

```

    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
- if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
- if `todo' == 1 {
= if 1 == 1 {
    - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    = gen double __000010 = __00000Y+__00000Y*__00000Y*1
    - exit
}

> _____ end glim
> _v6 _____
    - if "$ML_off" == "" {
    = if "0" == "" {
        global ML_off 0
    }
    - tempname phi
    - if "$SGLM_ph" == "" { global SGLM_ph = 1 }
    = if ".303924650780859" == "" { global SGLM_ph = 1 }
    - scalar `phi' = $SGLM_ph
    = scalar __000011 = .303924650780859
    - if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
    = if 1 { scalar __000011 = 1 }
    - tempvar lli
    - qui `VV' $SGLM_V 5 `eta' `mu' `lli'
    = qui version 18.5: glim_v6 5 __00000X __00000Y __000012

> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
    = if 5 == -1 {
        local y "$SGLM_y"

```

```

local m "$SGLM_m"
local touse "`eta'"
capture assert `y'>=0 if `touse'
if _rc {
di as err `"dependent variable `y' has negative values"'
exit 499
}
capture assert `y'==int(`y') if `touse'
if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 5 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}
- if `todo' == 1 {
= if 5 == 1 {
gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
exit
}
- if `todo' == 2 {
= if 5 == 2 {
gen double `return' = 1 + 2*`mu'*$SGLM_a
exit
}
- if `todo' == 3 {
= if 5 == 3 {
local y "$SGLM_y"
if "`y'" == "" {
local y "`e(depvar)'"
}
gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
exit
}
- if `todo' == 4 {
= if 5 == 4 {
local y "$SGLM_y"

```

```

        local k "$SGLM_a"
        if "`y'" == "" {
            local y ``e(depvar)''
        }
        gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) - `mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
        exit
    }
    - if `todo' == 5 {
    = if 5 == 5 {
    - local y "$SGLM_y"
    = local y "piq_timing_max"
    - if "`y'" == "" {
    = if "piq_timing_max" == "" {
        local y ``e(depvar)''
    }
    - tempname sig2
    - if $SGLM_s1 {
    = if 1 {
    - scalar `sig2' = $SGLM_s1
    = scalar __000013 = 1
    - }
    - else {
        if _caller() >= 11 {
            scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
        }
        else {
            scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
        }
    }
    - local m = 1/$SGLM_a
    = local m = 1/1
    - local lnalpha = ln(`m')
    = local lnalpha = ln(1)
    - gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/`m') + `y'*ln(`mu'/(`mu'+`m'))))
    = gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
    - exit
}

> _____ end glim
> _v6 _____
    - mlsun `lnf' = `lli'
    = mlsun __000005 = __000012
    - if `todo' == 0 | `lnf'==. { exit }

```

```

= if 0 == 0 | __00000S==. { exit }
> _____ end glim
> _lf _____
> _____ begin glim
> _lf _____
- version 6
- args todo b lnf g H score
- local VV : di "version " string(_caller()) ":"
- tempvar eta mu dm u v
- mlevel `eta' = `b'
= mlevel __00000X = __00000R
- qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __00000X __00000Y
> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
global SGLM_lt "Log"
if "$SGLM_m" == "1" {
global SGLM_lf "ln(u)"
}
else global SGLM_lf "ln(u/$SGLM_m)"
exit
}
- if `todo' == 0 {
= if 1 == 0 {
gen double `eta' = ln(`mu'/$SGLM_m)
exit
}
- if `todo' == 1 {
= if 1 == 1 {
- gen double `mu' = $SGLM_m*exp(`eta')
= gen double __00000Y = 1*exp(__00000X)
- exit
> _____ end glim_
> l03 _____
- qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y
> _____ begin glim

```

```

> _mu
- version 7
- args min max mu
- tempname eps
- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011

> _____ end glim_
> _mu
- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z

> _____ begin glim_
> l03
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  - gen double `return' = `mu'
= gen double __00000Z = __00000Y
- exit

> _____ end glim_

```

```

> l03 -----
      - qui $SGLM_V 1 `eta' `mu' `v'
      = qui glim_v6 1 __00000X __00000Y __000010
-----

> ----- begin glim
> _v6 -----
      - version 7
      - args todo eta mu return
      - if `todo' == -1 {
      = if 1 == -1 {
          local y "$SGLM_y"
          local m "$SGLM_m"
          local touse "`eta'"
          capture assert `y'>=0 if `touse'
          if _rc {
              di as err `"dependent variable `y' has negative values"'
              exit 499
          }
          capture assert `y'==int(`y') if `touse'
          if _rc {
              di as txt `"note: `y' has noninteger values"'
          }
          global SGLM_vt "Neg. Binomial"
          local SGLM_a = round($SGLM_a,0.0001)
          global SGLM_vf "u+(`SGLM_a')u^2"
          global SGLM_mu "glim_mu 0 ."
          exit
      }
      - if `todo' == 0 {
      = if 1 == 0 {
          gen double `eta' = -ln1p($SGLM_a/`mu')
          exit
      }
      - if `todo' == 1 {
      = if 1 == 1 {
          - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
          = gen double __000010 = __00000Y+__00000Y*__00000Y*1
          - exit
      }
-----

> ----- end glim
> _v6 -----
      - if "$ML_off" == "" {
      = if "0" == "" {
          global ML_off 0
      }
      - tempname phi

```



```

- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".303924650780859" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .303924650780859
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012

```

```

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
        di as err `"dependent variable `y' has negative values"'
        exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
        di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
}
- if `todo' == 0 {
= if 5 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
}
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
}
- if `todo' == 2 {
= if 5 == 2 {

```

```

        gen double `return' = 1 + 2*`mu'*$SGLM_a
        exit
    }
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/'mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
        exit
    }
- if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
        exit
    }
- if `todo' == 5 {
= if 5 == 5 {
- local y "$SGLM_y"
= local y "piq_timing_max"
- if "`y'" == "" {
= if "piq_timing_max" == "" {
    local y "`e(depvar)'"
}
- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
    if _caller() >= 11 {
        scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
    }
    else {
        scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
    }
}

```

```

    }
    - local m = 1/$SGLM_a
    = local m = 1/1
    - local lnalpha = ln(`m')
    = local lnalpha = ln(1)
    - gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/`m') + `y'*ln(`mu'/(`mu'+`m'))))
    = gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
    - exit
    _____
> _____ end glim
> _v6 _____
    - mlsun `lnf' = `lli'
    = mlsun __00000S = __000012
    - if `todo' == 0 | `lnf'==. { exit }
    = if 0 == 0 | __00000S==. { exit }
    _____
> _____ end glim
> _lf _____
    _____
> _____ begin glim
> _lf _____
    - version 6
    - args todo b lnf g H score
    - local VV : di "version " string(_caller()) ":"
    - tempvar eta mu dm u v
    - mlevel `eta' = `b'
    = mlevel __00000X = __00000R
    - qui $SGLM_L 1 `eta' `mu'
    = qui glim_l03 1 __00000X __00000Y
    _____
> _____ begin glim_
> l03 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
    = if 1 == -1 {
        global SGLM_lt "Log"
        if "$SGLM_m" == "1" {
            global SGLM_lf "ln(u)"
        }
        else global SGLM_lf "ln(u/$SGLM_m)"
        exit
    }

```

```

- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `mu' = $SGLM_m*exp(`eta')
  = gen double __00000Y = 1*exp(__00000X)
  - exit
}
}

> _____ end glim_
> l03 _____
  - qui $SGLM_mu `mu'
  = qui glim_mu 0 . __00000Y
}

> _____ begin glim
> _mu _____
  - version 7
  - args min max mu
  - tempname eps
  - scalar `eps' = 1e-8
  = scalar __000011 = 1e-8
  - replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
  = replace __00000Y = 0+__000011 if __00000Y<=0+__000011
  - replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
  = replace __00000Y = .-__000011 if __00000Y>=.-__000011
}

> _____ end glim
> _mu _____
  - qui $SGLM_L 2 `eta' `mu' `dmu'
  = qui glim_l03 2 __00000X __00000Y __00000Z
}

> _____ begin glim_
> l03 _____
  - version 7
  - args todo eta mu return
  - if `todo' == -1 {
  = if 2 == -1 {
    global SGLM_lt "Log"
    if "$SGLM_m" == "1" {
      global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
  }
}

```

```

- if `todo' == 0 {
= if 2 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
}
- if `todo' == 1 {
= if 2 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
}
- if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = `mu'
    = gen double __00000Z = __00000Y
    - exit
}
}

> _____ end glim_
> l03 _____
    - qui $SGLM_V 1 `eta' `mu' `v'
    = qui glim_v6 1 __00000X __00000Y __000010
}

> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
- if `todo' == 0 {

```

```

= if 1 == 0 {
  gen double `eta' = -ln1p($SGLM_a/`mu')
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
- gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
= gen double __000010 = __00000Y+__00000Y*__00000Y*1
- exit
}

> _____ end glim
> _v6 _____
- if "$ML_off" == "" {
= if "0" == "" {
  global ML_off 0
}
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".303924650780859" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .303924650780859
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"

```

```

    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
}
- if `todo' == 0 {
= if 5 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
}
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
}
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
}
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu*$SGLM_a))))
    exit
}
- if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
    exit
}
- if `todo' == 5 {
= if 5 == 5 {
    - local y "$SGLM_y"
    = local y "piq_timing_max"

```

```

- if "`y'" == "" {
= if "piq_timing_max" == "" {
  local y "`e(depvar)'"
}
- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
  if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
  }
  else {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
  }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')
= local lnalpha = ln(1)
- gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/`m') + `y'*ln(`mu'/(`mu'+`m'))))
= gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*ln
> n(__00000Y/(__00000Y+1)))
- exit
}

> _____ end glim
> _v6 _____
- mlsun `lnf' = `lli'
= mlsun __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 2 == 0 | __00000S==. { exit }
- qui replace `score' = ($ML_y1 - `mu')/(`phi'*`v')*`dmu' if $ML
> _samp
= qui replace __00000V = (piq_timing_max - __00000Y)/(__000011*_
> __000010)*__00000Z if __00000D
- $ML_ec mlvecsum `lnf' `g' = `score'
= * mlvecsum __00000S __00000T = __00000V
- if `todo' == 1 | `lnf'==. { exit }
= if 2 == 1 | __00000S==. { exit }
- tempvar tt
- qui gen double `tt' = sum($ML_w*($ML_y1-`mu')*($ML_y1-`mu')/`v
> ')

```



```

= qui gen double __000013 = sum(__00000D*(piq_timing_max-__00000
> Y)*(piq_timing_max-__00000Y)/__000010)
- if _caller() >= 11 {
- global SGLM_ph = `tt'[_N]/($ML_N-$ML_kCns)
= global SGLM_ph = __000013[_N]/(217-14)
- }
- else {
  global SGLM_ph = `tt'[_N]/($ML_N-$ML_k)
  }
- qui $SGLM_A
= qui
- tempvar t1
- qui gen double `t1' = `dmu'*`dmu'/'`v' if $ML_samp
= qui gen double __000014 = __00000Z*__00000Z/__000010 if __0000
> 0D

- if "$SGLM_f" == "" {
= if "-1" == "" {
  global SGLM_f = -1
  }
- if $SGLM_f >= $ML_ic {
= if -1 >= 2 {
  mlmatsum `lnf' `H' = (`t1')/`phi'
  exit
  }
- tempvar dv d2mu
- qui $SGLM_V 2 `eta' `mu' `dv'
= qui glim_v6 2 __00000X __00000Y __000015

> _____ begin glim
> _v6 _____

- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
}

```

```

    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
  }
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = -ln1p($SGLM_a/`mu')
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  - gen double `return' = 1 + 2*`mu'*$SGLM_a
= gen double __000015 = 1 + 2*__00000Y*1
  - exit
}
}

> _____ end glim
> _v6 _____
  - qui $SGLM_L 3 `eta' `mu' `d2mu'
= qui glim_l03 3 __00000X __00000Y __000016
}

> _____ begin glim_
> l03 _____
  - version 7
  - args todo eta mu return
  - if `todo' == -1 {
= if 3 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 3 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {

```

```

= if 3 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 3 == 2 {
  gen double `return' = `mu'
  exit
}
- if `todo' == 3 {
= if 3 == 3 {
  - gen double `return' = `mu'
= gen double __000016 = __00000Y
  - exit
}

> _____ end glim_
> l03 _____
  - tempvar t2 t3
  - qui gen double `t2' = `t1'/'v'*`dv' if $ML_samp
= qui gen double __000017 = __000014/__000010*__000015 if __0000
> 0D
  - qui gen double `t3' = `d2mu'/'v' if $ML_samp
= qui gen double __000018 = __000016/__000010 if __00000D
  - mlatsum `lnf' `H' = (`t1'-(`mu'-$ML_y1)*(`t2'-`t3'))/'phi'
= mlatsum __00000S __00000U = (__000014-(__00000Y-piq_timing_ma
> x)*(__000017-__000018))/__000011
  _____
> _____ end glim
> _lf _____
  _____
> _____ begin glim
> _lf _____
  - version 6
  - args todo b lnf g H score
  - local VV : di "version " string(_caller()) ":"
  - tempvar eta mu dm u v
  - mlevel `eta' = `b'
= mlevel __00000X = __00000R
  - qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __00000X __00000Y
  _____
> _____ begin glim_
> l03 _____
  - version 7
  - args todo eta mu return
  - if `todo' == -1 {

```

```

= if 1 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `mu' = $SGLM_m*exp(`eta')
= gen double __00000Y = 1*exp(__00000X)
- exit
}

> _____ end glim_
> l03 _____
  - qui $SGLM_mu `mu'
= qui glim_mu 0 . __00000Y
}

> _____ begin glim
> _mu _____
  - version 7
  - args min max mu
  - tempname eps
  - scalar `eps' = 1e-8
= scalar __000011 = 1e-8
  - replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
  - replace `mu' = `max'-'`eps' if `mu'>=`max'-'`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011
}

> _____ end glim
> _mu _____
  - qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z
}

> _____ begin glim_
> l03 _____
  - version 7
  - args todo eta mu return
  - if `todo' == -1 {

```

```

= if 2 == -1 {
    global SGLM_lt "Log"
    if "$SGLM_m" == "1" {
        global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
}
- if `todo' == 0 {
= if 2 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
}
- if `todo' == 1 {
= if 2 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
}
- if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = `mu'
= gen double __00000Z = __00000Y
    - exit
}
}
> _____ end glim_
> l03 _____
    - qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010
}
> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
        di as err `"dependent variable `y' has negative values"
        exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
        di as txt `"note: `y' has noninteger values"

```

```

    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
  }
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = -ln1p($SGLM_a/`mu')
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
  = gen double __000010 = __00000Y+__00000Y*__00000Y*1
  - exit
}
}
> _____ end glim
> _v6 _____
- if "$ML_off" == "" {
= if "0" == "" {
  global ML_off 0
}
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".3071764291250987" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .3071764291250987
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012
}
> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {

```

```

di as err `"dependent variable `y' has negative values"'
exit 499
}
capture assert `y'==int(`y') if `touse'
if _rc {
di as txt `"note: `y' has noninteger values"'
}
global SGLM_vt "Neg. Binomial"
local SGLM_a = round($SGLM_a,0.0001)
global SGLM_vf "u+(`SGLM_a')u^2"
global SGLM_mu "glim_mu 0 ."
exit
}
- if `todo' == 0 {
= if 5 == 0 {
gen double `eta' = -ln1p($SGLM_a/`mu')
exit
}
- if `todo' == 1 {
= if 5 == 1 {
gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
exit
}
- if `todo' == 2 {
= if 5 == 2 {
gen double `return' = 1 + 2*`mu'*$SGLM_a
exit
}
- if `todo' == 3 {
= if 5 == 3 {
local y "$SGLM_y"
if "`y'" == "" {
local y "`e(depvar)'"
}
gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
exit
}
- if `todo' == 4 {
= if 5 == 4 {
local y "$SGLM_y"
local k "$SGLM_a"
if "`y'" == "" {
local y "`e(depvar)'"
}
}

```

```

        gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) - `mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
        exit
    }
    - if `todo' == 5 {
    = if 5 == 5 {
    - local y "$GLM_y"
    = local y "piq_timing_max"
    - if "`y'" == "" {
    = if "piq_timing_max" == "" {
        local y "`e(depvar)'"
    }
    - tempname sig2
    - if $GLM_s1 {
    = if 1 {
    - scalar `sig2' = $GLM_s1
    = scalar __000013 = 1
    - }
    - else {
        if _caller() >= 11 {
        scalar `sig2' = $GLM_ph*($ML_N-$ML_kCns)/$ML_N
        }
        else {
        scalar `sig2' = $GLM_ph*($ML_N-$ML_k)/$ML_N
        }
    }
    - local m = 1/$GLM_a
    = local m = 1/1
    - local lnalpha = ln(`m')
    = local lnalpha = ln(1)
    - gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/`m') + `y'*ln(`mu'/(`mu'+`m'))))
    = gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
        - exit
    }

> _____ end glim
> _v6 _____
    - mlsun `lnf' = `lli'
    = mlsun __00000S = __000012
    - if `todo' == 0 | `lnf'==. { exit }
    = if 0 == 0 | __00000S==. { exit }

> _____ end glim
> _lf _____

```



```

> _____ begin glim
> _lf _____
    - version 6
    - args todo b lnf g H score
    - local VW : di "version " string(_caller()) ":"
    - tempvar eta mu dmu v
    - mlevel `eta' = `b'
    = mlevel __00000X = __00000R
    - qui $SGLM_L 1 `eta' `mu'
    = qui glim_l03 1 __00000X __00000Y
    _____

> _____ begin glim_
> l03 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
    = if 1 == -1 {
        global SGLM_lt "Log"
        if "$SGLM_m" == "1" {
            global SGLM_lf "ln(u)"
        }
        else global SGLM_lf "ln(u/$SGLM_m)"
        exit
    }
    - if `todo' == 0 {
    = if 1 == 0 {
        gen double `eta' = ln(`mu'/$SGLM_m)
        exit
    }
    - if `todo' == 1 {
    = if 1 == 1 {
        - gen double `mu' = $SGLM_m*exp(`eta')
        = gen double __00000Y = 1*exp(__00000X)
        - exit
    }
    _____

> _____ end glim_
> l03 _____
    - qui $SGLM_mu `mu'
    = qui glim_mu 0 . __00000Y
    _____

> _____ begin glim
> _mu _____
    - version 7
    - args min max mu
    - tempname eps

```

```

- scalar `eps' = 1e-8
= scalar __000011 = 1e-8
- replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
= replace __00000Y = 0+__000011 if __00000Y<=0+__000011
- replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
= replace __00000Y = .-__000011 if __00000Y>=.-__000011

```

```

> _____ end glim
> _mu _____
- qui $SGLM_L 2 `eta' `mu' `dmu'
= qui glim_l03 2 __00000X __00000Y __00000Z

```

```

> _____ begin glim_
> l03 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  global SGLM_lt "Log"
  if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
  }
  else global SGLM_lf "ln(u/$SGLM_m)"
  exit
}
- if `todo' == 0 {
= if 2 == 0 {
  gen double `eta' = ln(`mu'/$SGLM_m)
  exit
}
- if `todo' == 1 {
= if 2 == 1 {
  gen double `mu' = $SGLM_m*exp(`eta')
  exit
}
- if `todo' == 2 {
= if 2 == 2 {
  - gen double `return' = `mu'
  = gen double __00000Z = __00000Y
  - exit
}

```

```

> _____ end glim_
> l03 _____
- qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010

```

```

> ----- begin glim
> _v6
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
    - if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
    - if `todo' == 1 {
= if 1 == 1 {
    - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    = gen double __000010 = __00000Y+__00000Y*__00000Y*1
    - exit
    }
}

> ----- end glim
> _v6
    - if "$ML_off" == "" {
= if "0" == "" {
    global ML_off 0
    }
    - tempname phi
    - if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".3071764291250987" == "" { global SGLM_ph = 1 }
    - scalar `phi' = $SGLM_ph
= scalar __000011 = .3071764291250987

```

```

- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012
_____
> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
        di as err `"dependent variable `y' has negative values"'
        exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
        di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
}
- if `todo' == 0 {
= if 5 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
}
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
}
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
}
- if `todo' == 3 {

```

```

= if 5 == 3 {
  local y "$SGLM_y"
  if "`y'" == "" {
    local y "`e(depvar)'"
  }
  gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/'mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
  exit
}
- if `todo' == 4 {
= if 5 == 4 {
  local y "$SGLM_y"
  local k "$SGLM_a"
  if "`y'" == "" {
    local y "`e(depvar)'"
  }
  gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) -`mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
  exit
}
- if `todo' == 5 {
= if 5 == 5 {
- local y "$SGLM_y"
= local y "piq_timing_max"
- if "`y'" == "" {
= if "piq_timing_max" == "" {
  local y "`e(depvar)'"
}
- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
  if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_kCns)/$ML_N)
  }
  else {
    scalar `sig2' = $SGLM_ph* (($ML_N-$ML_k)/$ML_N)
  }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')

```

```

        = local lnalpha = ln(1)
        - gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/'m') + `y'*ln(`mu'/(`mu'+`m'))))
        = gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*ln
> n(__00000Y/(__00000Y+1)))
        - exit
    _____
> _____ end glim
> _v6 _____
        - mlsun `lnf' = `lli'
        = mlsun __00000S = __000012
        - if `todo' == 0 | `lnf'==. { exit }
        = if 0 == 0 | __00000S==. { exit }
    _____
> _____ end glim
> _lf _____
    _____
> _____ begin glim
> _lf _____
        - version 6
        - args todo b lnf g H score
        - local VV : di "version " string(_caller()) ":"
        - tempvar eta mu dm u v
        - mlevel `eta' = `b'
        = mlevel __00000X = __00000R
        - qui $SGLM_L 1 `eta' `mu'
        = qui glim_l03 1 __00000X __00000Y
    _____
> _____ begin glim_
> l03 _____
        - version 7
        - args todo eta mu return
        - if `todo' == -1 {
        = if 1 == -1 {
            global SGLM_lt "Log"
            if "$SGLM_m" == "1" {
                global SGLM_lf "ln(u)"
            }
            else global SGLM_lf "ln(u/$SGLM_m)"
            exit
        }
        - if `todo' == 0 {
        = if 1 == 0 {
            gen double `eta' = ln(`mu'/$SGLM_m)
            exit

```

```

    }
    - if `todo' == 1 {
    = if 1 == 1 {
    - gen double `mu' = $SGLM_m*exp(`eta')
    = gen double __00000Y = 1*exp(__00000X)
    - exit
    }
    }
> _____ end glim_
> l03 _____
    - qui $SGLM_mu `mu'
    = qui glim_mu 0 . __00000Y
    }
> _____ begin glim
> _mu _____
    - version 7
    - args min max mu
    - tempname eps
    - scalar `eps' = 1e-8
    = scalar __000011 = 1e-8
    - replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
    = replace __00000Y = 0+__000011 if __00000Y<=0+__000011
    - replace `mu' = `max'-`eps' if `mu'>=`max'-`eps'
    = replace __00000Y = .-__000011 if __00000Y>=.-__000011
    }
> _____ end glim
> _mu _____
    - qui $SGLM_L 2 `eta' `mu' `dmu'
    = qui glim_l03 2 __00000X __00000Y __00000Z
    }
> _____ begin glim_
> l03 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
    = if 2 == -1 {
    global SGLM_lt "Log"
    if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
    }
    - if `todo' == 0 {
    = if 2 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
    }
    }

```

```

    }
    - if `todo' == 1 {
= if 2 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
    }
    - if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = `mu'
= gen double __00000Z = __00000Y
    - exit
}
}
> _____ end glim_
> l03 _____
    - qui $SGLM_V 1 `eta' `mu' `v'
= qui glim_v6 1 __00000X __00000Y __000010
}
> _____ begin glim
> _v6 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
    - if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }

```



```

- if `todo' == 1 {
= if 1 == 1 {
- gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
= gen double __000010 = __00000Y+__00000Y*__00000Y*1
- exit
}

> _____ end glim
> _v6 _____
- if "$ML_off" == "" {
= if "0" == "" {
  global ML_off 0
}
- tempname phi
- if "$SGLM_ph" == "" { global SGLM_ph = 1 }
= if ".3071764291250987" == "" { global SGLM_ph = 1 }
- scalar `phi' = $SGLM_ph
= scalar __000011 = .3071764291250987
- if $SGLM_s1 { scalar `phi' = $SGLM_s1 }
= if 1 { scalar __000011 = 1 }
- tempvar lli
- qui `VV' $SGLM_V 5 `eta' `mu' `lli'
= qui version 18.5: glim_v6 5 __00000X __00000Y __000012
}

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 5 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"
  local SGLM_a = round($SGLM_a,0.0001)
  global SGLM_vf "u+(`SGLM_a')u^2"
  global SGLM_mu "glim_mu 0 ."
  exit
}

```

```

    }
- if `todo' == 0 {
= if 5 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
- if `todo' == 1 {
= if 5 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
    }
- if `todo' == 2 {
= if 5 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
    }
- if `todo' == 3 {
= if 5 == 3 {
    local y "$SGLM_y"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGL
> M_a, 2*(`y'*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`
> mu'*$SGLM_a))))
    exit
    }
- if `todo' == 4 {
= if 5 == 4 {
    local y "$SGLM_y"
    local k "$SGLM_a"
    if "`y'" == "" {
        local y "`e(depvar)'"
    }
    gen double `return' = ((_hyp2f1(-`k'*`y') - _hyp2f1(-`k'*`mu
> ')) + 3/2*(`y'^(2/3) - `mu'^(2/3)))/((`mu'+`k'*`mu'^2)^(1/6))
    exit
    }
- if `todo' == 5 {
= if 5 == 5 {
    - local y "$SGLM_y"
    = local y "piq_timing_max"
    - if "`y'" == "" {
    = if "piq_timing_max" == "" {
        local y "`e(depvar)'"
    }
}

```

```

- tempname sig2
- if $SGLM_s1 {
= if 1 {
- scalar `sig2' = $SGLM_s1
= scalar __000013 = 1
- }
- else {
  if _caller() >= 11 {
    scalar `sig2' = $SGLM_ph*($ML_N-$ML_kCns)/$ML_N
  }
  else {
    scalar `sig2' = $SGLM_ph*($ML_N-$ML_k)/$ML_N
  }
}
- local m = 1/$SGLM_a
= local m = 1/1
- local lnalpha = ln(`m')
= local lnalpha = ln(1)
- gen double `return' = `sig2'*(lngamma(`m'+`y')-lngamma(`y'+1
> ) - lngamma(`m') - `m'*ln1p(`mu'/'m') + `y'*ln(`mu'/(`mu'+`m'))))
= gen double __000012 = __000013*(lngamma(1+piq_timing_max)-ln
> gamma(piq_timing_max+1) - lngamma(1) - 1*ln1p(__00000Y/1) + piq_timing_max*l
> n(__00000Y/(__00000Y+1)))
- exit

> _____ end glim
> _v6 _____
- mlsun `lnf' = `lli'
= mlsun __00000S = __000012
- if `todo' == 0 | `lnf'==. { exit }
= if 2 == 0 | __00000S==. { exit }
- qui replace `score' = ($ML_y1 - `mu')/(`phi'*`v')*`dmu' if $ML
> _samp
= qui replace __00000V = (piq_timing_max - __00000Y)/(__000011*_
> _000010)*__00000Z if __00000D
- $ML_ec mlvecsum `lnf' `g' = `score'
= * mlvecsum __00000S __00000T = __00000V
- if `todo' == 1 | `lnf'==. { exit }
= if 2 == 1 | __00000S==. { exit }
- tempvar tt
- qui gen double `tt' = sum($ML_w*($ML_y1-`mu')*($ML_y1-`mu')/`v
> ')
= qui gen double __000013 = sum(__00000D*(piq_timing_max-__00000
> Y)*(piq_timing_max-__00000Y)/__000010)
- if _caller() >= 11 {
- global SGLM_ph = `tt'[_N]/($ML_N-$ML_kCns)

```

```

= global SGLM_ph = __000013[_N]/(217-14)
- }
- else {
  global SGLM_ph = `tt'[_N]/($ML_N-$ML_k)
}
- qui $SGLM_A
= qui
- tempvar t1
- qui gen double `t1' = `dmu'*`dmu'/'`v' if $ML_samp
= qui gen double __000014 = __00000Z*__00000Z/__000010 if __0000

> 0D

- if "$SGLM_f" == "" {
= if "-1" == "" {
  global SGLM_f = -1
}
- if $SGLM_f >= $ML_ic {
= if -1 >= 3 {
  mmlatsum `lnf' `H' = (`t1')/`phi'
  exit
}
- tempvar dv d2mu
- qui $SGLM_V 2 `eta' `mu' `dv'
= qui glim_v6 2 __00000X __00000Y __000015

> _____ begin glim
> _v6 _____
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 2 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"
  local SGLM_a = round($SGLM_a,0.0001)
  global SGLM_vf "u+(`SGLM_a')u^2"
  global SGLM_mu "glim_mu 0 ."

```

```

        exit
    }
    - if `todo' == 0 {
= if 2 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
    - if `todo' == 1 {
= if 2 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
    }
    - if `todo' == 2 {
= if 2 == 2 {
    - gen double `return' = 1 + 2*`mu'*$SGLM_a
= gen double __000015 = 1 + 2*__00000Y*1
    - exit
}

> _____ end glim
> _v6 _____
    - qui $SGLM_L 3 `eta' `mu' `d2mu'
= qui glim_l03 3 __00000X __00000Y __000016

> _____ begin glim_
> l03 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 3 == -1 {
    global SGLM_lt "Log"
    if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
    }
    - if `todo' == 0 {
= if 3 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
    }
    - if `todo' == 1 {
= if 3 == 1 {
    gen double `mu' = $SGLM_m*exp(`eta')
    exit
    }

```

```

- if `todo' == 2 {
= if 3 == 2 {
  gen double `return' = `mu'
  exit
}
- if `todo' == 3 {
= if 3 == 3 {
  - gen double `return' = `mu'
  = gen double __000016 = __00000Y
  - exit
}
}

> _____ end glim_
> l03 _____
- tempvar t2 t3
- qui gen double `t2' = `t1'/'v'*`dv' if $ML_samp
= qui gen double __000017 = __000014/__000010*__000015 if __0000
> 0D
- qui gen double `t3' = `d2mu'/'v' if $ML_samp
= qui gen double __000018 = __000016/__000010 if __00000D
- mllmatsum `lnf' `H' = (`t1'-(`mu'-$ML_y1)*(`t2'-`t3'))/'phi'
= mllmatsum __00000S __00000U = (__000014-(__00000Y-piq_timing_ma
> x)*(__000017-__000018))/__000011
_____
> _____ end glim
> _lf _____
_____
> _____ begin mopt_p
> ost _____
- args b V C wgt touse depvar
- if `:length local touse' {
= if 8 {
- tempname esamp
- quietly gen byte `esamp' = `touse'
= quietly gen byte __000010 = __00000D
- }
- ereturn post `b' `V' `C' `wtg', esample(`esamp') depname(`depv
> ar') buildfvinfo
= ereturn post __00000X __00000Y __00000Z [], esample(__000010)
> depname(piq_timing_max) buildfvinfo
_____
> _____ end mopt_p
> ost _____
- ereturn scalar k_eq_model = abs($ML_waldtest)
= ereturn scalar k_eq_model = abs(-1)
- if `:length global ML_ll_0' {
= if 0 {

```

```

        ereturn scalar ll_0 = $ML_ll_0
        if `:length global ML_rank0' {
            ereturn scalar rank0 = $ML_rank0
            ereturn scalar df_m = e(rank) - $ML_rank0
        }
        else ereturn scalar df_m = e(rank) - $ML_k0
    }
    - if !`:length global ML_ll_0' | inlist(e(vce), "robust", "cluster
> ", "linearized") {
        = if !0 | inlist(e(vce), "robust", "cluster", "linearized") {
            - if "$ML_svy" == "" {
                = if "" == "" {
                    - _prefix_model_test, minimum
                }
            }
        }
    }

> ----- begin _prefix_model_t
> est -----
    - version 9
    - syntax [name(name=cmd)] [, svy noADJust noSVYadjust MINimum ]
    - if (inlist("`cmd'", "asmixlogit", "cmmixlogit", "cmxtmixlogit"))
> {
        = if (inlist("", "asmixlogit", "cmmixlogit", "cmxtmixlogit")) {
            _asmixtest, `svy' `adjust' `svyadjust'
            exit
        }
        - local K = 1
        - if "`e(k_eq_model)'" != "" {
            = if "1" != "" {
                - capture confirm integer number `e(k_eq_model)'
                = capture confirm integer number 1
                - if ! c(rc) {
                    - local K = e(k_eq_model)
                }
            }
            - if "`e(k_eq_model_skip)'" != "" {
                = if "" != "" {
                    capture confirm integer number `e(k_eq_model_skip)'
                    if ! c(rc) {
                        local skip = e(k_eq_model_skip)
                    }
                }
            }
        - ereturn local df_m
        - ereturn local F
        - ereturn local chi2
        - ereturn local chi2type
        - ereturn local p
        - if (`K' == 0) {

```

```

= if (1 == 0) {
  exit
}
- forval i = 1/`K' {
= forval i = 1/1 {
- if !`:list i in skip' {
= if !0 {
- local spec "`spec' ([#`i'])"
= local spec " ([#1])"
- }
- }
- tempname cns omit
- capture matrix `cns' = get(Cns)
= capture matrix __000010 = get(Cns)
- if (c(rc)) local cns
- if "`cns'" != "" {
= if "__000010" != "" {
- local k_autoCns = e(k_autoCns)
- if missing(`k_autoCns') {
= if missing(5) {
  local k_autoCns 0
}
- if rowsof(`cns') == `k_autoCns' {
= if rowsof(__000010) == 5 {
- local cns
- }
- }
- if "`svy'" != "" & "`adjust'`svyadjust'" != "" {
= if "" != "" & "" != "" {
  local adjust nosvyadjust
}
- else local adjust
- _ms_omit_info e(b), nocons
- matrix `omit' = r(omit)
= matrix __000011 = r(omit)
- capture test `spec', `adjust'
= capture test ([#1]),

```

```
> _____ begin t
```

```
> est _____
- if "`e(cmd)'" == "anova" | "`e(cmd)'" == "manova" {
= if "mopt" == "anova" | "mopt" == "manova" {
  AnovaTestChooser `0'
  return add
  exit
}

```



```

- if "`e(cmd)'" == "mixed" {
= if "mopt" == "mixed" {
    MixedTestChooser `0'
    return add
    exit
  }
- if _caller() < 8 {
  _test `0'
  return add
  exit
  }
- is_svy

```

```

> ----- begin is_
> svy -----
    - version 8
    - syntax [, Regression]
    - if "`e(prefix)'" == "svy" {
= if `""' == "svy" {
    return scalar is_svy = 1
  }
    - else if "`regression'" == "" {
= else if "" == "" {
    - return scalar is_svy = ("`e(N_psu)'" != "")
= return scalar is_svy = (" != ")
    - }
    - else {
      capt mat list e(b)
      local rc = _rc
      return scalar is_svy = ("`e(N_psu)'" != "") & (`rc' == 0)
    }

```

```

> ----- end is_
> svy -----
    - local is_svy = r(is_svy)
    - if replay() & `is_svy' == 0 {
= if replay() & 0 == 0 {
    _test `0'
    return add
    exit
  }
    - version 8
    - WaldTest `is_svy' `0'
= WaldTest 0 ([#1]),

```

```

> ----- begin test.WaldT

```

```

> est
- gettoken is_svy 0 : 0
- if ``is_svy'' == "1" {
= if "0" == "1" {
    if ``e(complete)'' == "available" {
        di as err ``must run svy command with "complete" "' "optio
> n before using this command"
        exit 301
    }
}
- gettoken tok : 0, parse(" (,") bind
- local single single
- local zero ``0''
= local zero ``([#1]), ''
- if ``tok'' == "(" {
= if ``('' == "(" {
- local single
- while ``tok'' == "(" {
= while ``('' == "(" {
- gettoken tok 0 : 0, parse(" (,") match(paren) bind
- local eqns ``eqns' ``tok''
= local eqns ``[#1]''
- gettoken tok : 0, parse(" (,") bind
- }
- while ``tok'' == "(" {
= while ``(",'' == "(" {
    gettoken tok 0 : 0, parse(" (,") match(paren) bind
    local eqns ``eqns' ``tok''
    gettoken tok : 0, parse(" (,") bind
    }
- if ``tok'' != "," & ``tok'' != "" {
= if ``(",'' != "," & ``(",'' != "" {
    local single single
    }
- }
- if ``single'=="single" {
= if ""=="single" {
    local eqns
    local 0 ``zero''
    while ``tok'' != "" & ``tok'' != "," {
        gettoken tok 0 : 0, parse(" ,") bind
        local eqns ``eqns' `tok''
        gettoken tok : 0, parse(" ,") bind
    }
    local eqns ``eqns''
}

```

```

- syntax [, Accumulate noSVYadjust common CONStant MINimum M
> test Mtest2(passthru) noTest COEF matvlc(passthru) DF(numlist max=1 >0) ]
- if "`df'" != "" {
= if "" != "" {
    local dof = `df'
}
- else if "`e(df_r)'" != "" {
= else if "" != "" {
    local dof = e(df_r)
}
- else {
- local dof = .
- }
- if ``mtest'`mtest2'"" != "" {
= if `""' != "" {
    if inlist("`e(cmd)'", "svymean", "svytotal", "svyratio") {
    di as err "`mtest option not allowed after `e(cmd)'"
    exit 301
    }
    _mtest syntax, `mtest' `mtest2'
    local mtmethod `r(method)'
}
- if "mtmethod" != "" {
= if "" != "" {
    if "test" != "" {
    di as err "options mtest and notest may not be combined"
    exit 198
    }
    if "accumulate" != "" {
    di as err "options mtest and accumulate may not be combine
> d"

    exit 198
    }
}
- if `is_svy' {
= if 0 {
    if ``e(cmd)'" == "margins" {
    local vce ``e(model_vce)'"
    }
    else local vce ``e(vce)'"
    if inlist(`"vce"', "bootstrap", "sdr") {
    if missing(`dof') {
    local svyadjust nosvyadjust
    }
    }
}
}
}

```

```

- if !`is_svy' & "`svyadjust'" != "" {
= if !0 & "" != "" {
    di as err "option nosvyadjust is allowed only with svy dat
> a"

    exit 198
  }
- local svy_adjust = `is_svy' & "`svyadjust'" == ""
= local svy_adjust = 0 & "" == ""
- if `svy_adjust' & missing(`dof') & (missing(e(N_psu)) | mi
> ssing(e(N_strata))) {
= if 0 & missing(.) & (missing(e(N_psu)) | missing(e(N_strat
> a))) {

    di as err "impossible to adjust p-values"
    exit 198
  }
- if `is_svy' & ``test'" == "" {
= if 0 & ``'" == "" {
    if "`svyadjust'" == "" {
      di _n as txt "Adjusted Wald test"
    }
    else di _n as txt "Unadjusted Wald test"
  }
- local tmp : subinstr local eqns "==" "", all count(local
> nch)

- if `nch' > 0 {
= if 0 > 0 {
    di as err "==" invalid operator"
    exit 198
  }
- local eqns : subinstr local eqns "==" "=", all
- ExpandExpns ``eqns'" "common" "constant"
= ExpandExpns ``"#[1]"' "" ""



---


> ----- begin test.ExpandEx
> pns -----
    - args expns common constant
    - gettoken exp expns : expns
    - while ``exp'" != "" {
= while ``#[1]" != "" {
    - Expand ``exp'" "common" "constant"
= Expand ``#[1]" "" ""



---


> ----- begin test.Exp
> and -----
    - args exp common constant
    - local junk : subinstr local exp "=" "=", count(local n

```

> ch) all

```
- if `nch' <= 1 {  
= if 0 <= 1 {  
- return local explist ````exp''''  
= return local explist ````[#1]''''  
- return local nexp 1  
- exit
```

> _____ end test.Exp

> and _____

```
- local explist ```explist' `r(explist)'''  
= local explist `` ``[#1]''''  
- local nexp = `nexp' + `r(nexp)'  
= local nexp = + 1  
- gettoken exp expns : expns  
- }  
- while ```exp'''' != "" {  
= while ``'''' != "" {  
  Expand ```exp'''' ``common'' ``constant''  
  local explist ```explist' `r(explist)'''  
  local nexp = `nexp' + `r(nexp)'  
  gettoken exp expns : expns  
}  
- return local explist ```explist''''  
= return local explist `` ``[#1]''''  
- return local nexp `nexp'  
= return local nexp 1
```

> _____ end test.ExpandEx

> pns _____

```
- local explist ```r(explist)'''  
= local explist `` ``[#1]''''  
- local nexp `r(nexp)'  
= local nexp 1  
- local acc `accumulate'  
= local acc  
- foreach exp of local explist {  
- quietly _test `exp' , `acc' `common' `constant' notest  
= quietly _test [#1] , notest  
- local acc acc  
- }  
- if ``test'' == "notest" {  
= if "" == "notest" {  
  _test , notest  
  exit  
}
```

```

- quietly _test , `matvlc'
= quietly _test ,
- if `dof' < . {
= if . < . {
    local testtype F
    tempname F df1 df2 p drop
    if "`r(F)'" != "" {
        scalar `F' = r(F)
    }
    else {
        scalar `F' = r(chi2)/r(df)
    }
    scalar `df1' = r(df)
    scalar `df2' = `dof'
    scalar `p' = Ftail(`df1', `df2', `F')
    scalar `drop' = r(drop)
}
- else {
- local testtype chi2
- tempname chi2 df1 p drop
- scalar `chi2' = r(chi2)
= scalar __000012 = r(chi2)
- scalar `df1' = r(df)
= scalar __000013 = r(df)
- scalar `p' = r(p)
= scalar __000014 = r(p)
- scalar `drop' = r(drop)
= scalar __000015 = r(drop)
- }
- local i 1
- while "`r(dropped_`i')'" != "" {
= while "4" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_1 4
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "7" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_2 7
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "10" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_3 10

```

```

- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "12" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_4 12
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "13" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_5 13
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "" != "" {
    local dropped_`i' `r(dropped_`i')'
    local ++i
  }
- if `svy_adjust' {
= if 0 {
    if "`df'" != "" {
      local svy_d = `df'
    }
    else if !missing(e(df_r)) {
      local svy_d = e(df_r)
    }
    else local svy_d = e(N_psu)-e(N_strata)
    if missing(`svy_d') {
      di as err "internal error in adjustment for svy"
      exit 10001
    }
    if "`testtype'" == "F" {
      scalar `F' = ((`svy_d'-`df1'+1)/`svy_d') * `F'
    }
    else {
      tempname F df2
      scalar `F' = ((`svy_d'-`df1'+1)/(`svy_d'*`df1')) * `chi2'
    }
    scalar `df2' = `svy_d' - `df1' + 1
    if `df2' < 0 {
      scalar `F' = .
    }
    scalar `p' = Ftail(`df1', `df2', `F')
    local testtype F
  }
}

```

```

- if "`mtmethod'" != "" {
= if "" != "" {
    tempname R r Rr V b VR tm Wald
    mat `V' = e(V)
    mat `b' = e(b)
    mat `Rr' = get(Rr)
    local nb = rowsof(`V')
    local nr = rowsof(`Rr')
    mat `R' = `Rr'[1...,1..`nb']
    mat `VR' = `R'*`V'*`R''
    mat `r' = `R'*`b'' - `Rr'[1...,`=nb'+1']
    mat `Wald' = `r'*syminv(`VR')*`r'
    mat `tm' = J(`nr', 3, .)
    mat colnames `tm' = `testtype' df p
    forvalues it = 1 / `nr' {
        mat `Wald' = `r'[`it',1]^2 / `VR'[`it',`it']
        mat `tm'[`it',1] = `Wald'[1,1]
        mat `tm'[`it',2] = 1
        if `svy_adjust' {
            mat `tm'[`it',3] = Ftail(1,`svy_d',`tm'[`it',1])
        }
        else if "`testtype'" == "chi2" {
            mat `tm'[`it',3] = chi2tail(1,`tm'[`it',1])
        }
        else {
            mat `tm'[`it',3] = Ftail(1,`df2',`tm'[`it',1])
        }
    }
    if "`mtmethod'" != "noadjust" {
        _mtest adjust `tm', mtest(`mtmethod') pindex(3) append
        mat `tm' = r(result)
        local pindex 4
    }
    else local pindex 3
    }
- _test, notest
- local i 1
- while "`dropped_`i'" != "" {
= while "4" != "" {
- di as txt "          Constraint `dropped_`i' dropped"
= di as txt "          Constraint 4 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "7" != "" {
- di as txt "          Constraint `dropped_`i' dropped"

```



```

= di as txt "          Constraint 7 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "10" != "" {
- di as txt "          Constraint `dropped_`i'" dropped"
= di as txt "          Constraint 10 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "12" != "" {
- di as txt "          Constraint `dropped_`i'" dropped"
= di as txt "          Constraint 12 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "13" != "" {
- di as txt "          Constraint `dropped_`i'" dropped"
= di as txt "          Constraint 13 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "" != "" {
  di as txt "          Constraint `dropped_`i'" dropped"
  local ++i
}
- if "`mtmethod'" == "" {
= if "" == "" {
- di
- if "`testtype'" == "chi2" {
= if "chi2" == "chi2" {
- di as txt _col(12) "chi2(" %3.0f `df1' ") =" as res %8.2f
> `chi2'
= di as txt _col(12) "chi2(" %3.0f __000013 ") =" as res %8.
> 2f __000012
- di as txt _col(10) "Prob > chi2 = " as res %8.4f `p'
= di as txt _col(10) "Prob > chi2 = " as res %8.4f __000014
- }
- else {
  di as txt "          F(" %3.0f `df1' ", " %6.0f `df2' ") =" as
> res %8.2f `F'
  di as txt _col(13) "Prob > F =" as res %10.4f `p'
  }
- }
- else {
  local nexp = rowsof(`tm')

```

```

local nc = colsof(`tm')
if "`testtype'" == "F" {
local d "F(df,`=df2')"
local dip "p > F"
}
else {
local d chi2
local dip "p > chi2"
}
di
di as txt "{hline 7}{c TT}{hline 30}"
di as txt "          {c |}{ralign 12:`d'}      df{ralign 11:`d
> ip'}"

di as txt "{hline 7}{c +}{hline 30}"
forvalues it = 1 / `nexp' {
di as txt "    (`it') {c |}" as res _col(12) %9.2f `tm'[`it
> ',1] _col(22) %6.0f `tm'[`it',2] _col(33) %6.4f `tm'[`it',pindex'] as txt "
> *"

}
di as txt "{hline 7}{c +}{hline 30}"
di as txt "    All {c |}" as res _col(12) %9.2f ``testtype'
> ' _col(22) %6.0f `df1' _col(33) %6.4f `p'
di as txt "{hline 7}{c BT}{hline 30}"
_mttest footer 1 ``mtmethod'" "*" "
}
- if "`coef'" != "" {
= if "" != "" {
Table
}
- return scalar p = `p'
= return scalar p = __000014
- return scalar df = `df1'
= return scalar df = __000013
- if "`testtype'" == "F" {
= if "chi2" == "F" {
return scalar F = `F'
return scalar df_r = `df2'
}
- else return scalar chi2 = `chi2'
= else return scalar chi2 = __000012
- return scalar drop = `drop'
= return scalar drop = __000015
- local i 1
- while "`dropped_`i'" != "" {
= while "4" != "" {
- return scalar dropped_`i' = `dropped_`i'

```

```

= return scalar dropped_1 = 4
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "7" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_2 = 7
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "10" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_3 = 10
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "12" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_4 = 12
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "13" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_5 = 13
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "" != "" {
return scalar dropped_`i' = `dropped_`i''
local ++i
}
- if "`mtmethod'" != "" {
= if "" != "" {
return matrix mtest `tm'
return local mtmethod `mtmethod'
}

```

```
> ----- end test.WaldT
```

```
> est -----
- return add
-----
```

```
> ----- end t
```

```
> est -----
- if !c(rc) {
- if `:length local minimum' {
```

```

= if 7 {
- quietly test, min
}

> ----- begin t
> est -----
- if "`e(cmd)'" == "anova" | "`e(cmd)'" == "manova" {
= if "mopt" == "anova" | "mopt" == "manova" {
  AnovaTestChooser `0'
  return add
  exit
}
- if "`e(cmd)'" == "mixed" {
= if "mopt" == "mixed" {
  MixedTestChooser `0'
  return add
  exit
}
- if _caller() < 8 {
  _test `0'
  return add
  exit
}
- is_svy
}

> ----- begin is_
> svy -----
- version 8
- syntax [, Regression]
- if "`e(prefix)'" == "svy" {
= if `""' == "svy" {
  return scalar is_svy = 1
}
- else if "`regression'" == "" {
= else if "" == "" {
- return scalar is_svy = ("`e(N_psu)'" != "")
= return scalar is_svy = (" " != "")
- }
- else {
  capt mat list e(b)
  local rc = _rc
  return scalar is_svy = ("`e(N_psu)'" != "") & (`rc' == 0)
}
}

> ----- end is_
> svy -----
- local is_svy = r(is_svy)

```

```

- if replay() & `is_svy' == 0 {
= if replay() & 0 == 0 {
- _test `0'
= _test , min
- return add
- exit

```

```

> ----- end t
> est -----
- }
- ereturn scalar df_m = r(df)
- local drop = r(drop)
- if `drop' {
= if 1 {
- local drop 0
- local i 1
- local idrop ``r(dropped_`i')'''
= local idrop `4"'
- while ``idrop'" != "" {
= while "4" != "" {
- if !`omit'[1,`idrop'] {
= if !__000011[1,4] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop `7"'
- }
- while ``idrop'" != "" {
= while "7" != "" {
- if !`omit'[1,`idrop'] {
= if !__000011[1,7] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop `10"'
- }
- while ``idrop'" != "" {
= while "10" != "" {
- if !`omit'[1,`idrop'] {
= if !__000011[1,10] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''

```

```

= local idrop ``12''
- }
- while ``idrop'' != "" {
= while "12" != "" {
- if !`omit'[1,`idrop'] {
= if !__000011[1,12] {
  local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')''
= local idrop ``13''
- }
- while ``idrop'' != "" {
= while "13" != "" {
- if !`omit'[1,`idrop'] {
= if !__000011[1,13] {
  local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')''
= local idrop ``''
- }
- while ``idrop'' != "" {
= while "" != "" {
  if !`omit'[1,`idrop'] {
    local ++drop
  }
  local ++i
  local idrop ``r(dropped_`i')''
  }
- }
- if `drop' == 0 | ``cns'' != "" {
= if 0 == 0 | "" != "" {
- if !missing(e(df_r)) {
  ereturn scalar F = r(F)
  }
- else {
- ereturn scalar chi2 = r(chi2)
- ereturn local chi2type Wald
- }
- ereturn scalar p = r(p)
- }
- else {
  if !missing(e(df_r)) {
    ereturn scalar F = .
  }
}

```

```

        else {
            ereturn scalar chi2 = .
            ereturn local chi2type Wald
        }
        ereturn scalar p = .
    }
- }
- else {
    if missing(e(df_r)) {
        ereturn local chi2type Wald
    }
    ereturn scalar df_m = 0
}

> _____ end _prefix_model_t
> est _____
- }
- else {
    _prefix_model_test, $ML_svy $ML_noadj
}
- }
- else if `:length global ML_ll_0' {
= else if 0 {
    if ``"maxmin" == "minimize" {
        ereturn scalar chi2 = -2*(e(ll)-e(ll_0))
    }
    else {
        ereturn scalar chi2 = 2*(e(ll)-e(ll_0))
    }
    ereturn scalar p = chi2tail(e(df_m), e(chi2))
    ereturn local chi2type "LR"
}
- if `:length global ML_wexp' {
= if 0 {
    ereturn local wtype ``$ML_wtyp"
    ereturn local wexp ``$ML_wexp"
}
- if `:length global ML_group' {
= if 0 {
    ereturn local group ``$ML_group"
    if ``"e(clustvar)" == "$ML_grp" & "$ML_grp" != "" {
        ereturn local clustvar ``$ML_group"
    }
}
- ereturn local cmd "ml"
- if `:length local score' {

```

```

= if 0 {
    mata: Mopt_score(0)
    ereturn local scorevars `scores'
}
- if "$ML_preserve" != "no" {
= if "no" != "no" {
    restore
    if "`e(prefix)'" == "svy" & e(N_strata_omit) {
        tempname v
        quietly _svy2 $ML_sample if $ML_sample, type(total) svy v(`v') s
> ubpop(`e(subpop)') touse($ML_sample)
    }
    ereturn repost, esample($ML_sample)
}
- if ``moptobj'" != "" {
= if `""' != "" {
    local clear noclear
}
- if !':length local clear' {
= if !0 {
- Clear
}

> ----- begin mopt.Cl
> ear -----
    - syntax [, NOTANOPTION]
    - capture drop $ML_w
    = capture drop __00000D
    - capture drop $ML_samp
    = capture drop __00000D
    - capture drop $ML_sample
    = capture drop __00000D
    - capture drop $ML_subv
    = capture drop
    - capture drop $ML_grp
    = capture drop
    - mata: Mopt_drop_external()
    - macro drop ML_*

> ----- end mopt.Cl
> ear -----
    - }
    - else {
        capture confirm var $ML_sample
        if c(rc) {
            quietly gen byte $ML_sample = e(sample)
        }
    }

```



```

    }
    - if `:length local output' == 0 {
= if 8 == 0 {
    Display, level(`level') `options'
    }
}

> _____ end mopt.Max
> Min _____
    - }
    _____

> _____ end mopt.Mo
> del _____
    - if c(rc) {
      local rc = c(rc)
      Clear
      exit `rc'
    }
    - exit
    _____

> _____ end m
> opt _____
    - exit
    _____

> _____ end
> ml _____
    - local rc = cond(e(rc) == 430 & `reg', 0, e(rc))
    = local rc = cond(e(rc) == 430 & 0, 0, e(rc))
    - local vcetype `e(vcetype)'
    = local vcetype Robust
    - if $SGLM_estscale {
    = if 0 {
      global SGLM_ph _b[`sigma2']
    }
    - if "`vce'" == "eim" {
    = if "robust" == "eim" {
      nobreak {
        tempname esave
        version 9: estimates store `esave'
        capture noisily break {
          capture drop `eta'
          capture drop `mu'
          capture drop `v'
          capture drop `dmu'
          capture drop `W'
          capture drop `z'
          quietly predict double `eta', xb

```

```

$SGLM_L 1 `eta' `mu'
$SGLM_V 1 `eta' `mu' `v'
$SGLM_L 2 `eta' `mu' `dmu'
gen double `W' = `dmu'*`dmu'/`v' if `touse'
tempname V
version 11: quietly matrix accum `V' = `xvars' [iw=`W'*`wt'], `constant'
matrix `V' = inv(`V')
}
local rc = c(rc)
quietly version 9: estimates restore `esave'
version 9: ereturn local _estimates_name
}
if `rc' {
version 9: ereturn clear
exit `rc'
}
local vcetype "EIM"
estimates repost V = `V'
}
- if `fixme' {
= if 0 {
tempname V fix
mat `V' = e(V)
if `scale' { scalar `fix' = `scale' }
else scalar `fix' = $SGLM_ph
mat `V' = `fix'*`V'
estimates repost V = `V'
}
- if "`vcetype'" == "OPG" | "`opg'" == "opg" {
= if "Robust" == "OPG" | "" == "opg" {
local vce "opg"
}
- else if "`vcetype'" == "" | "`vce'" == "oim" {
= else if "Robust" == "" | "robust" == "oim" {
local vce oim
local vcetype "OIM"
}
- if "`unbiased'" != "" {
= if "" != "" {
tempvar hat eta mu v dmu dv d2mu z W
tempname ll
qui _predict double `eta', xb
qui $SGLM_L 1 `eta' `mu'
qui $SGLM_V 1 `eta' `mu' `v'
qui $SGLM_L 2 `eta' `mu' `dmu'
qui gen double `z' = `eta' + (`y'-`mu')/`dmu' `moffset' if `touse'

```

```

qui $SGLM_V 2 `eta' `mu' `dv'
qui $SGLM_L 3 `eta' `mu' `d2mu'
qui gen double `W' = `dmu'*`dmu'/'v' - (`mu'-`y')*(`dmu'*`dmu'*`dv'/'v'
> *`v') - `d2mu'/'v') if `touse'
nobreak {
estimates hold `ll'
`VV' qui _regress `z' `xvars' [iw=`W'*`wt'], mse1 `constant'
qui _predict double `hat' if `touse', stdp `offset'
qui replace `hat' = `hat'*`hat'*`v'
estimates unhold `ll'
}
qui replace `scvar' = `scvar' / sqrt(1-`hat')
qui _robust2 `scvar' [fweight=`fwt'] if `touse', `cll' minus(0)
local setype "Unbiased `setype'"
local setype "Unbiased Sandwich"
local vce "unbiased"
local vcetype "Unbiased"
}
- if "`opg'" != "" {
= if "" != "" {
tempname V1
mat `V1' = e(V)
local nms : colnames `V1'
local ncol = colsof(`V1')
mat `V1' = I(`ncol')
version 11: mat colnames `V1' = `nms'
version 11: mat rownames `V1' = `nms'
_robust2 `scvar' [fweight=`fwt'] if `touse', var(`V1')
mat `V1' = syminv(`V1')
estimates repost V = `V1'
local vcetype "OPG"
}
- if "`hacnam'" != "" {
= if "" != "" {
tempname V
mat `V' = e(V)
local p = colsof(`V')-1
capture drop `eta'
capture drop `mu'
capture drop `v'
capture drop `dmu'
capture drop `z'
qui _predict `eta', xb
qui $SGLM_L 1 `eta' `mu'
qui $SGLM_V 1 `eta' `mu' `v'
qui $SGLM_L 2 `eta' `mu' `dmu'

```

```

    qui gen double `z' = `dmu'*(`y'-`mu')/`v'
    sort `touse' `tvar'
    qui HAC ``hacnam''' ``haclag''' ``wt''' ``z''' ``touse''' ``constant''' ``x
> vars''' ``intlag'''
    tempname vm
    mat `vm' = r(V)
    local pm1 = `p'+1
    `hacnam' `haclag' 0 `nobs' `pm1'
    local delta = 1
    if `scale1' == 0 {
    local delta = $SGLM_ph
    }
    tempname ss
    scalar `ss' = 1/(`delta'*`delta')
    mat `V' = `ss'*`V'*`vm'*`V''
    local hac_kern ``r(setype)'''
    local vce "hac"
    local vcetype "HAC"
    estimates repost V = `V'
    }
- if ``jknife1''' != "" {
= if "" != "" {
    tempname b V
    mat `b' = e(b)
    mat `V' = e(V)
    qui JK1nr `b' `V' `y' ``xvars''' `eta' `mu' `v' `dmu' ``tvar''' `z' ``moff
> set''' ``constant''' `wt' `touse' ``weight'''
    local p = colsof(`b')
    tempname ss
    scalar `ss' = (`nobs'-`p')/`nobs'
    mat `V' = `ss'*`V'
    local vce "jackknife1"
    local vcetype "1-step JKnife"
    estimates repost V = `V'
    }
- if ``jknife''' != "" {
= if "" != "" {
    tempname b V
    mat `b' = e(b)
    mat `V' = e(V)
    noi JKnife `b' `V' `y' ``xvars''' ``tvar''' ``offopt''' ``constant''' ``argf
> am''' ``arglink''' "[`weight'`exp']" `touse' ``cluster''' `dots' ``weight'''
> ``wt'''
    local mm = `nobs'
    if ``cluster''' != "" { local mm = $SGLM_nc }
    else local mm = 1

```

```

local p = colsof(`b')
local nnn = `nobs' - $SGLM_bm
tempname ss
if "`cluster'" != "" {
  scalar `ss' = (`nnn' - `p') / `nnn'
}
else {
  scalar `ss' = (`nnn' - `p') / `nnn'
}
mat `V' = `ss' * `V'
local vcetype "Jackknife"
estimates repost V = `V'
}
- if "`bstrap'" != "" {
= if "" != "" {
  tempname b V
  mat `b' = e(b)
  mat `V' = e(V)
  noi Bstrap `b' `V' `y' "`xvars'" "`offopt'" "`constant'" "`argfam'" "" "
> `arglink'" "[`weight'`exp']" `touse' `brep' "`clopt'" `dots'
  local p = colsof(`b')
  tempname ss
  scalar `ss' = (`nobs' - `p') / (`nobs' * (`brep' - $SGLM_bm))
  mat `V' = `ss' * `V'
  local vcetype "Bootstrap"
  estimates repost V = `V'
}
- if `vfactor' != 1 {
= if 1 != 1 {
  tempname V
  mat `V' = `vfactor' * e(V)
  estimates repost V = `V'
}
- capture drop `eta'
= capture drop __000004
- capture drop `mu'
= capture drop __000005
- capture drop `v'
= capture drop __000007
- qui predict double `eta', xb
= qui predict double __000004, xb

> _____ begin pred
> ict _____
  - version 8.2, missing
  - if "`e(cmd)'" == "rocreg" & "`e(predict)'" == "" {

```

```

= if "ml" == "rocreg" & "ml_p" == "" {
  di as err "predict not allowed after nonparametric ROC"
  exit 198
}
- if "`e(mi)'"!="" & "`e(b)'"!="matrix" {
= if ""!="matrix" & "matrix"!="matrix" {
  error 321
}
- if _caller()<=5 | "`e(predict)'"==" " {
= if _caller()<=5 | "ml_p"==" " {
  _predict `0'
}
- else {
- local v : display string(_caller())
- version `v', missing
= version 7, missing
- `e(predict)' `0'
= ml_p double __000004, xb
}
}

> _____ begin m
> l_p _____
- version 9
- local version : di "version " string(_caller()) ":"
- syntax [anything] [if] [in] [, Scores * ]
- if ``scores'"" != "" {
= if `""' != "" {
  if ``e(opt)'" == "ml" {
    `version' ml score `0'
  }
  else {
    `version' mopt score `0'
  }
  exit
}
- `version' _predict `0'
= version 7: _predict double __000004, xb
}

> _____ end m
> l_p _____
- }

> _____ end pred
> ict _____
- qui $SGLM_L 1 `eta' `mu'
= qui glim_l03 1 __000004 __000005

```

```

> _____ begin glim_
> l03 _____
    - version 7
    - args todo eta mu return
    - if `todo' == -1 {
= if 1 == -1 {
    global SGLM_lt "Log"
    if "$SGLM_m" == "1" {
    global SGLM_lf "ln(u)"
    }
    else global SGLM_lf "ln(u/$SGLM_m)"
    exit
    }
    - if `todo' == 0 {
= if 1 == 0 {
    gen double `eta' = ln(`mu'/$SGLM_m)
    exit
    }
    - if `todo' == 1 {
= if 1 == 1 {
    - gen double `mu' = $SGLM_m*exp(`eta')
    = gen double __000005 = 1*exp(__000004)
    - exit
    }
}

> _____ end glim_
> l03 _____
    - qui $SGLM_mu `mu'
    = qui glim_mu 0 . __000005
}

> _____ begin glim
> _mu _____
    - version 7
    - args min max mu
    - tempname eps
    - scalar `eps' = 1e-8
    = scalar __00000D = 1e-8
    - replace `mu' = `min'+`eps' if `mu'<=`min'+`eps'
    = replace __000005 = 0+__00000D if __000005<=0+__00000D
    - replace `mu' = `max'-'`eps' if `mu'>=`max'-'`eps'
    = replace __000005 = .-__00000D if __000005>=.-__00000D
}

> _____ end glim
> _mu _____
    - qui $SGLM_V 1 `eta' `mu' `v'
    = qui glim_v6 1 __000004 __000005 __000007
}

```

```

> ----- begin glim
> _v6 -----
- version 7
- args todo eta mu return
- if `todo' == -1 {
= if 1 == -1 {
  local y "$SGLM_y"
  local m "$SGLM_m"
  local touse "`eta'"
  capture assert `y'>=0 if `touse'
  if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
  }
  capture assert `y'==int(`y') if `touse'
  if _rc {
    di as txt `"note: `y' has noninteger values"'
  }
  global SGLM_vt "Neg. Binomial"
  local SGLM_a = round($SGLM_a,0.0001)
  global SGLM_vf "u+(`SGLM_a')u^2"
  global SGLM_mu "glim_mu 0 ."
  exit
}
- if `todo' == 0 {
= if 1 == 0 {
  gen double `eta' = -ln1p($SGLM_a/`mu')
  exit
}
- if `todo' == 1 {
= if 1 == 1 {
  - gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
  = gen double __000007 = __000005+__000005*__000005*1
  - exit
}
}

> ----- end glim
> _v6 -----
- mat `b0' = e(b)
= mat __00000B = e(b)
- local dfm = e(df_m)
- capture drop `z'
= capture drop __000008
- qui gen double `z' = sum(`wt'*(`y'-`mu')^2/`v')
= qui gen double __000008 = sum(__000003*(piq_timing_max-__000005)^2/__000
> 007)
- local chi2 = `z'[_N]

```



```

= local chi2 = __000008[_N]
- local df = e(N)-`dfm'
= local df = e(N)-13
- _post_vce_rank, checksize

```

```

> _____ begin _post_vce_r
> ank _____
- syntax, [Checksize]
- if "`checksize'" != "" {
= if "checksize" != "" {
- tempname V
- capture matrix `V' = e(V)
= capture matrix __00000D = e(V)
- if _rc {
exit
}
- local cols = colsof(`V')
= local cols = colsof(__00000D)
- if `cols' == 0 {
= if 19 == 0 {
exit
}
- }
- tempname V Vi rank
- mat `V' = e(V)
= mat __00000E = e(V)
- mat `Vi' = invsym(`V')
= mat __00000F = invsym(__00000E)
- sca `rank' = rowsof(`V') - diag0cnt(`Vi')
= sca __00000G = rowsof(__00000E) - diag0cnt(__00000F)
- mata:st_numscalar("e(rank)", st_numscalar("`rank'"))
= mata:st_numscalar("e(rank)", st_numscalar("__00000G"))

```

```

> _____ end _post_vce_r
> ank _____
- local rank = e(rank)
- if $SGLM_s1 { global SGLM_ph = $SGLM_s1 }
= if 1 { global SGLM_ph = 1 }
- capture drop `z'
= capture drop __000008
- qui $SGLM_V 3 `eta' `mu' `z'
= qui glim_v6 3 __000004 __000005 __000008

```

```

> _____ begin glim
> _v6 _____
- version 7

```

```

- args todo eta mu return
- if `todo' == -1 {
= if 3 == -1 {
    local y "$SGLM_y"
    local m "$SGLM_m"
    local touse "`eta'"
    capture assert `y'>=0 if `touse'
    if _rc {
    di as err `"dependent variable `y' has negative values"'
    exit 499
    }
    capture assert `y'==int(`y') if `touse'
    if _rc {
    di as txt `"note: `y' has noninteger values"'
    }
    global SGLM_vt "Neg. Binomial"
    local SGLM_a = round($SGLM_a,0.0001)
    global SGLM_vf "u+(`SGLM_a')u^2"
    global SGLM_mu "glim_mu 0 ."
    exit
    }
- if `todo' == 0 {
= if 3 == 0 {
    gen double `eta' = -ln1p($SGLM_a/`mu')
    exit
    }
- if `todo' == 1 {
= if 3 == 1 {
    gen double `return' = `mu'+`mu'*`mu'*$SGLM_a
    exit
    }
- if `todo' == 2 {
= if 3 == 2 {
    gen double `return' = 1 + 2*`mu'*$SGLM_a
    exit
    }
- if `todo' == 3 {
= if 3 == 3 {
    - local y "$SGLM_y"
    = local y "piq_timing_max"
    - if ""y"" == "" {
    = if "piq_timing_max" == "" {
        local y "`e(depvar)'"
    }
    - gen double `return' = cond(`y'==0, 2*ln1p(`mu'*$SGLM_a)/$SGLM_a, 2*(`y
> '*ln(`y'/`mu') - (1+`y'*$SGLM_a)/$SGLM_a * ln((1+`y'*$SGLM_a)/(1+`mu'*$SGLM_

```

```

> a))))
      = gen double __000008 = cond(piq_timing_max==0, 2*ln1p(__000005*1)/1, 2*
> (piq_timing_max*ln(piq_timing_max/__000005) - (1+piq_timing_max*1)/1 * ln((1
> +piq_timing_max*1)/(1+__000005*1))))
      - exit

```

```

> _____ end glim
> _v6 _____
  - qui replace `z' = sum(`wt'*`z')
  = qui replace __000008 = sum(__000003*__000008)
  - scalar `newdev' = `z'[_N]
  = scalar __00000C = __000008[_N]
  - local disp = $SGLM_ph
  = local disp = 1
  - est scalar vf = `vfactor'
  = est scalar vf = 1
  - est scalar rc = `rc'
  = est scalar rc = 0
  - if "`oldIC'" != "" {
  = if "oldic" != "" {
  - est scalar aic = (-2*e(ll)+2*`rank')/`nobs'
  = est scalar aic = (-2*e(ll)+2*14)/217
  - }
  - else est scalar aic = (-2*e(ll)+2*`rank')
  = else est scalar aic = (-2*e(ll)+2*14)
  - }
  - else {
    if ``mlopts' != "" {
      local 0 `", `mlopts'
      syntax [, NOOPTION ]
      error 198
    }
    quietly {
      tempvar eta mu dm u v W z
      tempname Wscale b0
      if "`offset'" != "" {
        local moffset = "-`offset'"
      }
      if ``init' != "" {
        gen double `mu' = `init' if `touse'
      }
      else {
        summ `y' `wgt' if `touse', mean
        if "$SGLM_V" != "glim_v2" {
          gen double `mu' = (`y'+r(mean))/(`m'+1)
        }
      }
    }
  }

```

```

else {
gen double `mu' = `m'*(`y'+.5)/(`m'+1)
}
}
if `reg' { local iterate 1 }
capture drop `eta'
$SGLM_L 0 `eta' `mu'
tempvar dev
tempname newdev oldev
scalar `oldev' = 0
scalar `newdev' = 1
local i 1
if `iterate' == 0 { local i 0 }
`iflog' di
while abs(`newdev'-`oldev')>`ltol' & `i'<=`iterate' {
capture drop `v'
capture drop `dmu'
capture drop `W'
capture drop `z'
scalar `oldev' = `newdev'
$SGLM_V 1 `eta' `mu' `v'
$SGLM_L 2 `eta' `mu' `dmu'
gen double `z' = `eta' + (`y'-`mu')/`dmu' `moffset' if `touse'
gen double `W' = `disp'*`dmu'*`dmu'/`v' if `touse'
summ `W' `wt', meanonly
scalar `Wscale' = r(mean)
`VV' cap noisily quietly _regress `z' `xvars' [iw=`W'*`wt'/`Wscale'], ms
> e1 `constant'
if e(N)>=. {exit 459}
if _rc {exit _rc}
capture drop `eta'
capture drop `mu'
capture drop `dev'
predict double `eta' if `touse', xb
if "`offset'" != "" {
replace `eta' = `eta'+`offset'
}
$SGLM_L 1 `eta' `mu'
$SGLM_mu `mu'
$SGLM_V 3 `eta' `mu' `dev'
replace `dev' = sum(`wt'*`dev')
scalar `newdev' = `dev'[_N]/`disp'
if `i' < 10 {
local space " "
}
else {

```

```

    local space
  }
  `iflog' di as txt "Iteration `i':`space' " "Deviance = " as res %9.0g `n
> ewdev'
  if "`trace'" != "" {
    tempname beta
    `iflog' mat list e(b), noblack noheader
    `iflog' di as txt "{hline 78}"
    `iflog' di
  }
  local ic = `i'
  local i = `i'+1
  }
  if "`unbiased'`jknife1'`jknife'" != "" {
    `VV' cap noisily quietly _regress `z' `xvars' [iw=`W'*`wt'], mse1 `const
> ant'
    if e(N)>=. {exit 459}
    if _rc {exit _rc}
    tempvar h
    qui _predict double `h' if `touse', stdp
    qui replace `h' = `h'*`h'*`v'
    local minus "minus(0)"
  }
  else local h = 0
  local rc = cond(abs(`newdev'-`oldev')>`ltol' & !`reg',430,0)
  replace `z' = sum(`wt'*(`y'-`mu')^2/`v')
  local chi2 = `z'[_N]/`disp'
  local p = e(df_m)
  local df = `nobs'-`p'-(``constant'==``)
  local dispc = `chi2'/`df'
  local dispd = `newdev'/`df'
  global SGLM_ph = `dispc'
  if ``scale'==`` {
    local scale `scale1'
    if `scale1' { local delta 1 }
    else local delta `dispc'
  }
  else if `scale'==0 {
    local delta `dispc'
    if `scale1' {
      local cd "square root of Pearson"
      local cd "`cd' X2-based dispersion"
    }
  }
  else if `scale'==-1 {
    local delta `dispd'
  }

```

```

local cd "square root of deviance-based"
local cd "`cd' dispersion"
}
else {
local delta `scale'
if !`scale1' | (`scale1' & `scale'!=1) {
local cd "dispersion equal to square"
local cd "`cd' root of `delta'"
}
}
if !`scale1' | (`scale1' & `scale'!=1) {
if `scale1' { local dof 100000 }
else local dof `df'
if `delta'>=. {
local zapse "yes"
}
else {
scalar `Wscale' = `Wscale' / `delta'
}
}
tempname b V
mat `b' = e(b)
mat `V' = e(V)
local vce "eim"
local vcetype "EIM"
if "`robust'"`oim'"`hacnam'"`opg'" != "" {
capture drop `W'
capture drop `z'
capture drop `v'
capture drop `dmu'
tempvar xb
_predict double `xb', xb
$GLM_V 1 `eta' `mu' `v'
$GLM_L 2 `eta' `mu' `dmu'
tempvar ys yi
gen double `ys' = `dmu'*(`mu'-`y')/`v'
gen double `yi' = `dmu'*`dmu'
if "`oim'" != "" {
tempvar dv d2m
$GLM_V 2 `eta' `mu' `dv'
$GLM_L 3 `eta' `mu' `d2m'
replace `yi' = `yi' + `d2m'*(`mu'-`y') - `dmu'*`dv'*`ys'
}
replace `yi' = `awt' * `yi' / `v'
replace `ys' = `awt' * `ys' / sqrt(1-`h')
`VV' _regress `xb' `xvars' if `touse' [iweight=`fwt'*`yi'], `constant' m

```

```

> sel
    mat `V' = e(V)
    local vce "oim"
    local vcetype "OIM"
    if "`robust'" != "" {
        tempname vmb
        matrix `vmb' = e(V)
        if "`oim'" != "" {
            local vce "robust"
            local vcetype "Robust"
        }
        else {
            local vce "robust"
            if (`canon'==0) {
                local vcetype "Semirobust"
            }
            else {
                local vcetype "Robust"
            }
        }
        _robust2 `ys' [fweight=`fwt'] if `touse', var(`V') `clopt' `minus'
        if "`unbiased'" != "" {
            local setype "Unbiased Sandwich"
            local vce "unbiased"
            local vcetype "Unbiased"
        }
        if ``"clopt"' != "" {
            global SGLM_nc = r(N_clust)
        }
    }
    else if "`opg'" != "" {
        local nms : colnames `V'
        local ncol = colsof(`V')
        mat `V' = I(`ncol')
        version 11: mat colnames `V' = `nms'
        version 11: mat rownames `V' = `nms'
        replace `ys' = `ys' / `disp'
        _robust2 `ys' [fweight=`fwt'] if `touse', var(`V')
        mat `V' = `delta' * `delta' * syminv(`V')
        local vce "opg"
        local vcetype "OPG"
    }
    else if "`hacnam'" == "" {
        tempname ss
        scalar `ss' = `delta' * `Wscale'
        mat `V' = `ss' * `V'
    }

```

```

}
}
if "`hacnam'" != "" {
capture drop `v'
capture drop `dmu'
capture drop `z'
$SGLM_V 1 `eta' `mu' `v'
$SGLM_L 2 `eta' `mu' `dmu'
gen double `z' = `dmu'*(`y'-`mu')/`v'
sort `touse' `tvar'
qui HAC "`hacnam'" "`haclag'" "`wt'" "`z'" "`touse'" "`constant'" "`x
> vars'" "`intlag'"
tempname vm
mat `vm' = r(V)
local pm1 = `p'+1
`hacnam' `haclag' 0 `nobs' `pm1'
local hac_kern "`r(settype)'"
local vce "hac"
local vcetype "HAC"
local ddd = 1
tempname ss
scalar `ss' = 1/(`ddd'*`ddd')
mat `V' = `ss'*`V'*`vm'*`V'
}
if "`jkknife1'" != "" {
JK1irls `b' `V' `y' "`xvars'" `h' `eta' `mu' `v' `dmu' "`tvar'" `z' "`mo
> ffset'" "`constant'" `wt' `touse' "`weight'"
local p = colsof(`b')
tempname ss
scalar `ss' = (`nobs'-`p')/`nobs'
local vce "jackknife1"
local vcetype "1-step JKnife"
mat `V' = `ss'*`V'
}
if "`jkknife'" != "" {
noi JKnife `b' `V' `y' "`xvars'" "`tvar'" "`offopt'" "`constant'" "`argf
> am'" "irls" "`arglink'" "[`weight'`exp']" `touse' "`cluster'" `dots' "`weigh
> t'" "`wt'"
local mm = `nobs'
local vcetype "Jackknife"
if "`cluster'" != "" { local mm = $SGLM_nc }
else local mm=1
local p = colsof(`b')
local nnn = `nobs'-$SGLM_bm
tempname ss
scalar `ss' = (`nnn'-`p')/(`nnn'*`mm')

```



```

mat `V' = `ss'*`V'
}
if "`bootstrap'" != "" {
tempname b V
mat `b' = e(b)
mat `V' = e(V)
noi Bstrap `b' `V' `y' "`xvars'" "`offopt'" "`constant'" "`argfam'" "irl
> s" "`arglink'" "[`weight'`exp']" `touse' `brep' "`clopt'" `dots'
local vcetype "Bootstrap"
if "`cluster'" != "" { local mm = $SGLM_nc }
else local mm=1
local p = colsof(`b')
tempname ss
scalar `ss' = (`nobs'-`p')/(`nobs'*(`brep'-$SGLM_bm))
mat `V' = `ss'*`V'
}
if "`opg'`hacnam'`robust'`jknife'`jknife1'`bootstrap'" == "" {
scalar `Wscale' = 1/`Wscale'
mat `V' = `Wscale'*`V'
}
if "`zapse'"=="yes" {
local i 1
while `i'<=rowsof(`V') {
mat `V'[`i',`i'] = 0
local i=`i'+1
}
}
if `vfactor' != 1 {
mat `V' = `vfactor'*`V'
}
tempvar mysamp
local k = colsof(`b')
qui gen byte `mysamp' = `touse'
est post `b' `V' [`weight'`exp'], depname(`y') obs(`nobs') `dofopt' esam
> ple(`mysamp') buildfvinfo
if `:length local vmb' {
est matrix V_modelbased `vmb'
}
est local depvar "`y'"
est local wtype "`weight'"
est local wexp "`exp'"
est local title "Generalized linear models"
est scalar rc = `rc'
est scalar disp = `disp'
est scalar k = `k'
est hidden scalar ic = `ic'

```

```

    est hidden local crittype "deviance"
  }
  est hidden scalar noconstant = cond("`constant'" == "",0,1)
  est hidden scalar consonly = cond("`xvars'"!="",0,1)
}
- mac drop SGLM_running SGLM_nonstan
- if "`score'" != "" {
= if "" != "" {
  label var `scvar' "Score index from glm"
  rename `scvar' `score'
  est local scorevars `score'
}
- _post_vce_rank
}

> ----- begin _post_vce_r
> ank -----
  - syntax, [Checksize]
  - if "`checksize'" != "" {
= if "" != "" {
  tempname V
  capture matrix `V' = e(V)
  if _rc {
    exit
  }
  local cols = colsof(`V')
  if `cols' == 0 {
    exit
  }
}
  - tempname V Vi rank
  - mat `V' = e(V)
  = mat __00000D = e(V)
  - mat `Vi' = invsym(`V')
  = mat __00000E = invsym(__00000D)
  - sca `rank' = rowsof(`V') - diag0cnt(`Vi')
  = sca __00000F = rowsof(__00000D) - diag0cnt(__00000E)
  - mata:st_numscalar("e(rank)", st_numscalar("`rank'"))
  = mata:st_numscalar("e(rank)", st_numscalar("__00000F"))
}

> ----- end _post_vce_r
> ank -----
  - local dfm = e(rank)- (`""`constant'"=="")
  = local dfm = e(rank)- (""=="")
  - local df = `nobs'-e(rank)
  = local df = 217-e(rank)
  - est local link "$SGLM_L"

```

```

= est local link "glim_l03"
- est local varfunc "$SGLM_V"
= est local varfunc "glim_v6"
- est local m "$SGLM_m"
= est local m "1"
- if !inlist("$SGLM_m", "", "1") {
= if !inlist("1", "", "1") {
  est hidden local marginsprop atslow
}
- est hidden local a "$SGLM_a"
= est hidden local a "1"
- est hidden local msg "`cd'"
= est hidden local msg ""
- est hidden scalar canonical = `canon'
= est hidden scalar canonical = __000001
- est local cons "`constant'"
= est local cons ""
- est hidden local oim "`oim'"
= est hidden local oim "oim"
- est local clustvar "`cluster'"
= est local clustvar ""
- capture confirm number $SGLM_p
= capture confirm number 0
- if !_rc {
- est scalar power = $SGLM_p
= est scalar power = 0
- }
- else {
  version 15: ereturn hidden local power `"$SGLM_p"'
}
- if "$SGLM_V" == "glim_v2" & "$SGLM_L" == "glim_l01" & "`eform'" == "" {
= if "glim_v6" == "glim_v2" & "glim_l03" == "glim_l01" & "" == "" {
  est hidden local msg2 "Coefficients are the risk differences"
}
- if "`irls'" != "" {
= if "" != "" {
  est local opt irls
  if "`oim'" == "" {
    est local opt1 "MQL Fisher scoring"
    est local opt2 "(IRLS EIM)"
  }
  else {
    est local opt1 "MQL Newton-Raphson"
    est local opt2 "(IRLS OIM)"
  }
}
}
}

```

```

- else {
- est local opt1 "ML"
- est local opt2
- if "`vce'" == "eim" & `disp' != 1 {
= if "robust" == "eim" & 1 != 1 {
    matrix `V' = `disp'*e(V)
    est repost V = `V'
  }
- }
- if "$SGLM_bm" != "" {
= if "" != "" {
    if $SGLM_bm > 0 { est scalar Nf = $SGLM_bm }
  }
- est scalar df_m = `dfm'
= est scalar df_m = 13
- est scalar df = `df'
= est scalar df = 203
- if "$SGLM_nc" != "" {
= if "" != "" {
    est scalar N_clust = $SGLM_nc
  }
- if `brep' > 0 {
= if -1 > 0 {
    est scalar N_brep = `brep'
  }
- est scalar vf = `vfactor'
= est scalar vf = 1
- est scalar phi = `disp'
= est scalar phi = 1
- est scalar deviance = abs(`newdev')
= est scalar deviance = abs(__00000C)
- est scalar dispers = e(deviance)/e(df)
- est scalar deviance_s = e(deviance)/e(phi)
- est scalar dispers_s = e(deviance_s)/e(df)
- est scalar deviance_p = abs(`chi2')
= est scalar deviance_p = abs(62.35462627335119)
- est scalar dispers_p = e(deviance_p)/e(df)
- est scalar deviance_ps = e(deviance_p)/e(phi)
- est scalar dispers_ps = e(deviance_ps)/e(df)
- capture drop `v'
= capture drop __000007
- qui gen double `v' = sum((`y' - `mu')^2) if `touse'
= qui gen double __000007 = sum((piq_timing_max - __000005)^2) if __000002
- if "`oldIC'" != "" | "`irls'" != "" {
= if "oldic" != "" | "" != "" {
- est scalar bic = e(deviance) - `df'*log(`nobs')

```

```

= est scalar bic = e(deviance) - 203*log(217)
- }
- else est scalar bic = -2*e(ll)+log(`nobs')*`p'
= else est scalar bic = -2*e(ll)+log(217)*
- est local varfuncf "$SGLM_vf"
= est local varfuncf "u+(1)u^2"
- est local varfunct "$SGLM_vt"
= est local varfunct "Neg. Binomial"
- est local linkf "$SGLM_lf"
= est local linkf "ln(u)"
- est local linkt "$SGLM_lt"
= est local linkt "Log"
- if "`cluster'" != "" {
= if "" != "" {
    est local vce "cluster"
  }
- else {
- est local vce "`vce'"
= est local vce "robust"
- }
- est local vcetype "`vcetype'"
= est local vcetype "Robust"
- est local setype "`setype'"
= est local setype ""
- est local hac_kernel "`hac_kern'"
= est local hac_kernel ""
- est local hac_lag "`haclag'"
= est local hac_lag "215"
- est local offset "`offvar'"
= est local offset ""
- est local offset1
- est scalar k_eq_model = 0
- est local marginsok default
- est local marginsnotok stdp Anscombe Cooksd Deviance Hat Likelihood Pear
> son Response Score Working ADJusted STAndardized STUdentized MODified
- est local predict "glim_p"
- est local cmd "glm"
- capture mac drop SGLM_bm SGLM_nc
- if "$SGLM_V" == "glim_v2" {
= if "glim_v6" == "glim_v2" {
    if "$SGLM_L" == "glim_l01" | "$SGLM_L" == "glim_l03" {
      CheckAdmiss
    }
  }
- est scalar nbml = `"$SGLM_ml"' == "ml"
= est scalar nbml = `""' == "ml"

```

```

- if "`display'" == "" {
= if "" == "" {
- Display, `eform' level(`level') `header' `table' `diopts'
= Display, level(95)
}

> ----- begin glm.Disp
> lay
- if "`e(prefix)'" == "svy" {
= if "" == "svy" {
    _prefix_display `0'
    exit
  }
- di

- syntax [, noHEADer noTABLE Level(cilevel) eform *]
- _get_diopts diopts, `options'
= _get_diopts diopts,

> ----- begin _get_dio
> pts
- version 11
- syntax namelist(max=2) [, *]
- gettoken name1 namelist : namelist
- gettoken name2 namelist : namelist
- GetDiopts diopts options, `options'
= GetDiopts diopts options,

> ----- begin _get_diopts.GetDio
> pts
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels
> NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NO
> OMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sfor
> mat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(
> string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels A
> LLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells
> EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat
> (string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passth
> ru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passth
> ru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

> ----- begin opts_exclus

```

```

> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``emptycells' `noemptycells'
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
        }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``fvlabel' `nofvlabel'
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {

```



```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus

```

```

> ive
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

>
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> ive
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "

>
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> ive
    - opts_exclusive "`versus' `compare'"

```

```

= opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> _____ end opts_exclus
> ive _____
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'

```

```

= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
}

```

```

- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth("`pformat'") > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat "`pformat'"
    local pformat `pformat(`pformat')'
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraapon , `fvwraapon'
= Wraapon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
    di as err "invalid fvwraapon() option;"
    syntax [, W0rd WIdth]
    exit 198
}
- opts_exclusive "`word' `width'" fvwraapon
= opts_exclusive " " fvwraapon

```

> _____ begin opts_exclus

> ive _____

```

- version 8.2
- args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> _____ end opts_exclus
> ive _____
- local fwwrapon `word' `width'
= local fwwrapon
- if `:length local fwwrapon' {
= if 0 {
    c_local fwwrapon fwwrapon(`fwwrapon')
}

> _____ end _get_diopts.Wra
> pon _____
- if ``fwwrap' != "" {
= if "" != "" {
    FVWrap, `fwwrap'
}
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevel
> s' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend
> ' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fwr
> ap' `fwwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
= c_local diopts
- if `K' == 2 {
= if 2 == 2 {
- c_local `c_opts' ``options'
= c_local options `"'
- }
- sreturn local coding `coding'
= sreturn local coding
- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> _____ end _get_diopts.GetDio
> pts _____
    - c_local `name1' ```diopts''
    = c_local diopts `''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)
    - }
    - local S_`sname' = s(`sname')
    = local S_nolog = s(nolog)
    - }
    - local S_`sname' = s(`sname')
    = local S_deriv = s(deriv)
    - }
    - local S_`sname' = s(`sname')
    = local S_namelist = s(namelist)
    - }
    - local S_`sname' = s(`sname')
    = local S_after = s(after)
    - }
    - local S_`sname' = s(`sname')
    = local S_before = s(before)
    - }
    - local S_`sname' = s(`sname')
    = local S_k_fill = s(k_fill)
    - }
    - local S_`sname' = s(`sname')
    = local S_k = s(k)
    - }
    - sreturn clear
    - GetDiopts DEFAULT
    _____

> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels
> NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NO
> OMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sfor
> mat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(
> string) NOFVLAbel FVLAbel NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels A
> LLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells
> EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat
> (string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passth

```



```

> ru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPVALUES MARKDOWN CITYPE(passth
> ru) NOABBREV ABBREV *)
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive ``emptycells' `noemptycells'
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive ``fvlabel' `nofvlabel'
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2

```

```

- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0

```

```

- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind

```

```

    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "
}
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
= opts_exclusive " "
}
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
    -----
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit
    }
    -----
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
    -----
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit
    }
    -----
> ----- end opts_exclus

```

```

> ive
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {

```

```

    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat ``cformat''
    local cformat `cformat(`cformat')''
    }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth(`sformat') > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
        }
    sreturn local sformat ``sformat''
    local sformat `sformat(`sformat')''
    }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth(`pformat') > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
        }
    sreturn local pformat ``pformat''
    local pformat `pformat(`pformat')''
    }
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
    }
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
    }
- Wrapon , `fvwrapon'
= Wrapon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
    di as err "invalid fvwrapon() option;"
    syntax [, W0rd WIdth]
    exit 198
    }
- opts_exclusive ``word' `width' fvwrapon

```



```

= opts_exclusive " " fvwrapon
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
    _____
> _____ end _get_diopts.Wra
> pon _____
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevel
> s' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend
> ' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwr
> ap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DEFAULT
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options'
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus

```

```

- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> _____ end _get_diopts.GetDio
> pts _____
- if ``name2''' == '' {
= if `''' == '' {
- GetDiopts DUPS, `options'
= GetDiopts DUPS,

```

```

> _____ begin _get_diopts.GetDio
> pts _____
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevels
> NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells NO
> OMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sfor
> mat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAPON(
> string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBREV
> ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels A
> LLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells
> EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat
> (string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passth
> ru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passth
> ru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

```

```

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts''' != '' {
= while `''' != '' {
local ++n
gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive
    - version 8.2

```

```

- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0

```

```

- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind

```

```

        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells
    - local omitted `omitted' `noomitted'
    = local omitted
    - local lstretch `lstretch' `nolstretch'
    = local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
    = local fvlabel
    - local abbrev `abbrev' `noabbrev'
    = local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
    = if "" == "" {
        - if c(showbaselevels) == "all" {
            local allbaselevels allbaselevels
        }
    - else if c(showbaselevels) == "on" {

```



```

    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"
  local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth("`pformat'") > 5 {
    local pformat "%5.3f"

```

```

di "{txt}note: invalid pformat(), using default"
}
sreturn local pformat ``pformat''
local pformat `pformat(pformat)''
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraupon , `fvwraupon'
= Wraupon ,

> _____ begin _get_diopts.Wra
> pon _____
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwraupon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive ``word' `width'" fvwraupon
    = opts_exclusive " " fvwraupon

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit

> _____ end opts_exclus
> ive _____
    - local fvwraupon `word' `width'

```

```

        = local fvwrapon
        - if `:length local fvwrapon' {
        = if 0 {
            c_local fvwrapon fvwrapon(`fvwrapon')
        }
    }
}

> _____ end _get_diopts.Wra
> pon _____
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselevel
> s' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend
> ' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwr
> ap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DUPS
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options'
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95
}

> _____ end _get_diopts.GetDio
> pts _____
    - }
    - else {
        GetDiopts DUPS options, `options'
        c_local `name2' ``options'
    }
    - local DUPS : list DUPS - DEFAULT
    - gettoken DUPS : DUPS
    - if ``DUPS' != "" {
    = if `"' != "" {
        if (strpos(`DUPS', "(")) {
            gettoken DUPS : DUPS, parse("(")
            local DUPS `DUPS'(")'"
        }
        di as err "duplicate {bf:`DUPS'} option not allowed"
    }
}

```

```

        exit 198
    }
    - sreturn clear
    - foreach sname of local snames {
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local level = ``95''
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local nolog = ``nolog''
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local deriv = ``0''
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local namelist = ``cluster''
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local after = ``", robust''
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local before = ``", argoptlist(CLuster) optlist(OIM OPG Robus
> t) old ""
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local k_fill = ``19''
    - }
    - sreturn local `sname' = ``S_`sname''''
    = sreturn local k = ``19''
    - }
    _____
> _____ end _get_dio
> pts _____
    - if ``eform'' != "" {
    = if "" != "" {
        Eform
        local eform ``r(eform)''
    }
    - if ``eform'' != "" {
    = if "" != "" {
        local eopt "eform(`eform)''
    }
    - if ``header'' == "" { Head }
    = if "" == "" { Head }
    _____
> _____ begin glm.H
> ead _____

```

```

- di as txt "Generalized linear models" _col(51) "Number of obs" _col(
> 67) "=" _col(69) as res %10.0gc e(N)
Generalized linear models                                Number of obs    =          217
- di as txt "Optimization      : " as res "`e(opt1)'" as txt _col(51) "
> Residual df" _col(67) "=" _col(69) as res %10.0gc e(df)
= di as txt "Optimization      : " as res "ML" as txt _col(51) "Residua
> l df" _col(67) "=" _col(69) as res %10.0gc e(df)
Optimization      : ML                                Residual df      =          203
- di as res _col(20) "`e(opt2)'" as txt _col(51) "Scale parameter" _co
> l(67) "=" _col(70) as res %9.0g e(phi)
= di as res _col(20) "" as txt _col(51) "Scale parameter" _col(67) "="
> _col(70) as res %9.0g e(phi)
Scale parameter =          1
- di as txt "Deviance" _col(18) "=" as res _col(20) %12.0g e(deviance)
> as txt _col(51) "(1/df) Deviance" _col(67) "=" as res _col(70) %9.0g e(dis
> ers)
Deviance          =    51.2971855                      (1/df) Deviance =    .2526955
- di as txt "Pearson" _col(18) "=" as res _col(20) %12.0g e(deviance_p
> ) as txt _col(51) "(1/df) Pearson" _col(67) "=" as res _col(70) %9.0g e(dis
> ers_p)
Pearson           =    62.35462627                     (1/df) Pearson  =    .3071656
- di

- if `"$SGLM_ml" == "ml" {
= if `"" == "ml" {
di as txt "Variance function: " as res "V(u) = " as res _col(27) "`e
> (varfunc)'" _col(51) as txt "[" as res "`e(varfunc)'" as txt "]"
}
- else {
- di as txt "Variance function: " as res "V(u) = " as res _col(27) "`e
> (varfunc)'" _col(51) as txt "[" as res "`e(varfunc)'" as txt "]"
= di as txt "Variance function: " as res "V(u) = " as res _col(27) "u+
> (1)u^2" _col(51) as txt "[" as res "Neg. Binomial" as txt "]"
Variance function: V(u) = u+(1)u^2                    [Neg. Binomial]
- }
- di as txt "Link function      : " as res "g(u) = " as res _col(27) "`e
> (linkf)'" _col(51) as txt "[" as res "`e(linkt)'" as txt "]"
= di as txt "Link function      : " as res "g(u) = " as res _col(27) "ln
> (u)" _col(51) as txt "[" as res "Log" as txt "]"
Link function      : g(u) = ln(u)                    [Log]
- if "`e(hac_kernel)'" != "" {
= if "" != "" {
di
di as txt "HAC kernel (lags): " e(hac_kernel) " ({yellow:`e(hac_lag)
> '})" _c
local di di

```

```

    }
    - else if "`e(settype)'" != "" {
= else if "" != "" {
    di
    di as txt "Standard errors : " as res "`e(settype)'" _c
    local di di
    }
    - else local cr _n
    - if "`e(ll)'" != "" {
= if "-420.9775440505756" != "" {
    - local cr
    - di

    - local crtype = upper(bsubstr("`e(crittype)'",1,1)) + bsubstr("`e(
> crittype)'",2,.))
    = local crtype = upper(bsubstr("`log pseudolikelihood'",1,1)) + bsubst
> r("`log pseudolikelihood'",2,.))
    - local crlen = max(18,length("`crtype'")) + 2)
    = local crlen = max(18,length("`Log pseudolikelihood'")) + 2)
    - di as txt _col(51) "{help j_glmic##_new:AIC}" _col(67) "=" as res _
> col(70) %9.0g e(aic)

                                AIC                                =    4.00901

    - di as txt "`crtype'" _col(`crlen') "=" as res %12.0g e(ll) _c
    = di as txt "Log pseudolikelihood" _col(22) "=" as res %12.0g e(ll) _
> c
Log pseudolikelihood = -420.9775441 - }
    - else if "`e(dis)'" != "" & "`e(dis)'" != "1" {
= else if "" != "" & "" != "1" {
    local cr
    `di'
    di
    di as txt "Quasi-likelihood model with dispersion: " as res `e(dis)
> ' _c
    }
    - if "`e(ll)'" != "" {
= if "-420.9775440505756" != "" {
    - di as txt `cr' _col(51) "{help j_glmic##_new:BIC}" _col(67) "=" as
> res _col(70) %9.0g e(bic)
    = di as txt _col(51) "{help j_glmic##_new:BIC}" _col(67) "=" as res
> _col(70) %9.0g e(bic)

                                BIC                                =    -1040.822

    - }
    - else {
    di as txt `cr' _col(51) "BIC" _col(67) "=" as res _col(70) %9.0g e(b
> ic)
    }

```

```

- di

> _____ end glm.H
> ead _____
- if "`table'" != "" { exit }
= if "" != "" { exit }
- _coef_table , level(`level') `eopt' first `diopts'
= _coef_table , level(95) first

> _____ begin _coef_ta
> ble _____
- version 11
- local vv : di "version " string(max(11,_caller())) ", missing:"
- if (!c(noisily) & c(coeftabresults) == "off") {
  exit
}
- _check_eclass
- syntax [, BMATrix(passthru) VMATrix(passthru) * ]
- if "`e(b)'" == "" & "`e(V)'" == "" & "`bmatrix'`vmatrix'""=="" {
= if "matrix" == "" & "matrix" == "" & ""=="" {
  exit
}
- local mc_cmds contrast margins pwcompare pwmean
- local cmd "`e(cmd)'"
= local cmd "glm"
- local keepmc : list cmd in mc_cmds
- if inlist("`cmd'", "pwcompare", "pwmean") {
= if inlist("glm", "pwcompare", "pwmean") {
  local groups GROUPS
}
- if "`cmd'" == "regress" {
= if "glm" == "regress" {
  local beta Beta
}
- if "`cmd'" == "gsem" {
= if "glm" == "gsem" {
  local fvignore fvignore(int 0)
  local flignore flignore
}
- if "`cmd'" == "sem" {
= if "glm" == "sem" {
  local standardized STANDARDIZED
  local showginvariant SHOWGINvariant
  local nolabel NOLABEL LABEL NOFVLABEL FVLABEL
  local nofootnote NOFOOTnote

```

```

        local wrap wrap(numlist max=1) fvwrap(passthru)
    }
    - if "`e(mi)'"=="mi" {
= if ""=="mi" {
    local dftable DFTable
    local dfonly DFONLY
    local noclustreport NOCLUSTReport
    local pisematrix PISEMATrix(string)
}
    - syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(string)
> DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) PMATrix
> (string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat(s
> tring) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATrix(
> string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name) c
> nszero(string) suffix(name) OFFSETlist(string asis) `beta' `standardized' NO
> CI NOPValues `dftable' `dfonly' dfci DFPValues dfmissing `groups' COEFLegend
> selegend `fvignore' `flignore' `showginvariant' `nolabel' `nofootnote' `wra
> p' sort `pisematrix' depname(string) COEFTitle(string) coeftitle2(string) pt
> itle(string) cititle(string) NOMCLEgend noCNSReport FULLCNSReport `noclustre
> port' cformat(passthru) sformat(passthru) pformat(passthru) NOFirst First SH
> OWEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(integer 0) NOSKIP OFFSETONL
> Y1 Plus NOEQCHECK EFORMALL CITYPE(string) eqselect(string) cidisjoint noEFOR
> MNOTE * ]
    = syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(string)
> DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) PMATrix
> (string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat(s
> tring) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATrix(
> string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name) c
> nszero(string) suffix(name) OFFSETlist(string asis) NOCI NOPValues dfci
> DFPValues dfmissing COEFLegend selegend sort depname(string) COEFTit
> le(string) coeftitle2(string) ptitle(string) cititle(string) NOMCLEgend noCN
> SReport FULLCNSReport cformat(passthru) sformat(passthru) pformat(passthru)
> NOFirst First SHOWEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(integer 0)
> NOSKIP OFFSETONLY1 Plus NOEQCHECK EFORMALL CITYPE(string) eqselect(string)
> cidisjoint noEFORMNOTE * ]
    - if "`wrap'" != "" {
= if "" != "" {
    opts_exclusive `wrap(`wrap') `fvwrap'
    local fvwrap fvwrap(`wrap')
}
    - if "`nolabel'`label'" != "" {
= if "" != "" {
    opts_exclusive "`nolabel' `label'"
    opts_exclusive "`nofvlabel' `fvlabel'"
    opts_exclusive "`nolabel' `fvlabel'"
    opts_exclusive "`nofvlabel' `label'"

```



```

        if "`nolabel'" != "" {
            local fvlablel nofvlabel
        }
        else {
            local fvlablel fvlablel
        }
    }
    - _novce_other_mats, cimatrix("`cimatrix'") pmatrix("`pmatrix'") tsmat
> rix("`tsmatrix'") `noci' `nopvalues' `dfci' `dfpvalues' `cidisjoint'
    = _novce_other_mats, cimatrix("") pmatrix("") tsmatrix("")

_____  

> _____ begin _coef_table._novce_other_m
> ats _____
    - syntax [anything], [ CIMATrix(string) PMATrix(string) TSMATrix(str
> ing) NOCI NOPValues dfci DFPValues cidisjoint ]
    - if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"!=" & "`noci'"!="") {
        = if ("'"=="" & "'"=="" & "matrix"=="" & '"!=" & '"!="") {
            sreturn local cimatrix "empt"
        }
        - if ("`pmatrix'"=="" & "`tsmatrix'"!=" & "`e(V)'"=="" & "`cimatrix
> '"=="" & ("`noci'"!="|"`nopvalues'"!=")) {
            = if ("'"=="" & '"!=" & "matrix"=="" & "'"=="" & ("'"!="|'"!=")) {
                sreturn local noci "empt"
                sreturn local nopvalues "empt"
            }
            - if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"!=" & "`noci'"!="") {
                = if ("'"=="" & "'"=="" & "matrix"=="" & '"!=" & '"!="") {
                    sreturn local cimatrix "empt"
                }
                - if ("`pmatrix'"!=" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"=="" & "`nopvalues'"!="") {
                    = if ("'"!=" & "'"=="" & "matrix"=="" & "'"=="" & '"!="") {
                        sreturn local pmatrix "empt"
                    }
                    - if ("`pmatrix'"!=" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatrix
> '"=="" & "`noci'"!="") {
                        = if ("'"!=" & "'"=="" & "matrix"=="" & "'"=="" & '"!="") {
                            sreturn local noci "empt"
                        }
                        - if ("`pmatrix'"!=") {
                            = if ("'"!=") {
                                tempname pmat invalid
                                capture matrix `pmat' = `pmatrix'
                                local rc = _rc

```

```

if (`rc') {
display as error "invalid {bf:pmatrix()}"
exit 198
}
else {
local k: rowsof `pmat'
if (`k'>1 | `k'==0) {
display as error "{bf:pmatrix()} should be a row vector"
exit 198
}
}
scalar `isvalid' = 1
mata: _is_in_0_1void("`pmat'", "`isvalid'")
if (`isvalid'==0) {
display as error "invalid {bf:pmatrix()}"
di as txt "{p 4 4 2}"
di as smcl as err "{bf:pmatrix()}"
di as smcl as err "should contain values inside"
di as smcl as err "the interval [0,1]."
di as smcl as err "{p_end}"
exit 198
}
}
- if ("`tsmatrix'"!="") {
= if ("!=") {
tempname tmat
capture matrix `tmat' = `tsmatrix'
local rc = _rc
if (`rc') {
display as error "invalid {bf:tsmatrix()}"
exit 198
}
else {
local k: rowsof `tmat'
if (`k'>1 | `k'==0) {
display as error "{bf:tsmatrix()} should be a row vector"
exit 198
}
}
}
- if ("`cimatix'"!="") {
= if ("!=") {
tempname cimat isvalid
capture matrix `cimat' = `cimatix'
local rc = _rc
if (`rc') {

```

```

display as error "invalid {bf:cimatrix()}"
exit 198
}
if ("`cidisjoint'"=="" & "`e(disjoint)'"=="") {
  scalar `isvalid' = 1
  mata: _is_in_ci("`ciamat'", "`isvalid'")
  if (`isvalid'==0) {
    display as error "invalid {bf:cimatrix()}"
    di as txt "{p 4 4 2}"
    di as smcl as err "{bf:cimatrix()}"
    di as smcl as err "lower limit should be"
    di as smcl as err "smaller than the upper"
    di as smcl as err "limit."
    di as smcl as err "{p_end}"
    exit 198
  }
}
}
}
- if ("`e(V)'"=="" & "`dfci'"!="" & "`pmatrx'"!="") {
= if ("matrix"=="" & ""!="" & ""!="") {
  opts_exclusive "pmatrx()" `dfci'"
}
- if ("`e(V)'"=="" & "`dfci'"!="" & "`tsmatrx'"!="") {
= if ("matrix"=="" & ""!="" & ""!="") {
  opts_exclusive "tsmatrx()" `dfci'"
}
- if ("`e(V)'"=="" & "`dfci'"!="" & "`cimatrix'"!="") {
= if ("matrix"=="" & ""!="" & ""!="") {
  opts_exclusive "cimatrix()" `dfci'"
}
- if ("`e(V)'"=="" & "`dfpvalues'"!="" & "`pmatrx'"!="") {
= if ("matrix"=="" & ""!="" & ""!="") {
  opts_exclusive "pmatrx()" `dfci'"
}
- if ("`e(V)'"=="" & "`dfpvalues'"!="" & "`tsmatrx'"!="") {
= if ("matrix"=="" & ""!="" & ""!="") {
  opts_exclusive "tsmatrx()" `dfpvalues'"
}
- if ("`e(V)'"=="" & "`dfpvalues'"!="" & "`cimatrix'"!="") {
= if ("matrix"=="" & ""!="" & ""!="") {
  opts_exclusive "cimatrix()" `dfpvalues'"
}
}

```

```
> _____ end _coef_table._novce_other_m
```

```
> ats _____
- if ("`s(cimatrix)'"=="empt") {
```

```

= if (""=="empt") {
  local cimatix ""
}
- if ("`s(pmatrix)""=="empt") {
= if (""=="empt") {
  local pmatrix ""
}
- if ("`s(noci)""=="empt") {
= if (""=="empt") {
  local noci ""
}
- if ("`s(nopvalues)""=="empt") {
= if (""=="empt") {
  local nopvalues ""
}
- if ("`e(V)""!=" & "`cidisjoint""!=") {
= if ("matrix"!=" & ""!=") {
  display in red "option {bf:cidisjoint} not allowed when e(V) is post
> ed"
  exit 198
}
- else local fvlabel `nofvlabel' `fvlabel'
= else local fvlabel
- local type `coeflegend' `selegend'
= local type
- opts_exclusive "`type' `standardized'"
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
  - version 8.2
  - args opts optname errcode optslab
  - local opts `opts'
  = local opts
  - local n 0
  - while "`opts'" != "" {
  = while `""' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
  - if `n' < 2 {
  = if 0 < 2 {
  - exit

> _____ end opts_exclus
> ive _____

```

```

- if (``rowmatrix'""=="" & ``norowci'""!="") {
= if (``'""=="" & '""!="") {
    di as err "{bf:norowci} requires {bf:rowmatrix()}"
    exit 198
}
- if (``dftable'""!="" & ``norowci'""!="") {
= if (``'""!="" & '""!="") {
    di as err "{bf:norowci} is not allowed with {bf:`dftable'}"
    exit 198
}
- if (``dfonly'""!="" & ``norowci'""!="") {
= if (``'""!="" & '""!="") {
    di as err "{bf:norowci} is not allowed with {bf:`dfonly'}"
    exit 198
}
- if (``noci'""!="" & ``norowci'""!="") {
= if (``'""!="" & '""!="") {
    di as err "{bf:norowci} is not allowed with {bf:`noci'}"
    exit 198
}
- if ``standardized'"" != "" & ``bstdmatrix'"" != "" {
= if "" != "" & "" != "" {
    local beta beta
    local standardized
}
- if `:length local type' == 0 {
= if 0 == 0 {
- local type `beta' `noci' `nopvalues' `dftable' `dfonly' `groups' `df
> ci' `dfpvalues'
    = local type
    - }
    - opts_exclusive ``type'
    = opts_exclusive ""
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }

```

```

- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive
- opts_exclusive "`first' `nofirst'"
= opts_exclusive "first "
}

> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts first
- local n 0
- while ``opts' != "" {
= while `"first" != "" {
- local ++n
- gettoken item`n' opts : opts, bind
= gettoken item1 opts : opts, bind
- }
- while ``opts' != "" {
= while `"' != "" {
- local ++n
- gettoken item`n' opts : opts, bind
- }
- if `n' < 2 {
= if 1 < 2 {
- exit
}

> ----- end opts_exclus
> ive
- opts_exclusive "`first' `showeqns'"
= opts_exclusive "first "
}

> ----- begin opts_exclus
> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts first
- local n 0
- while ``opts' != "" {
= while `"first" != "" {
- local ++n

```

```

- gettoken item`n' opts : opts, bind
= gettoken item1 opts : opts, bind
- }
- while ```opts'`` != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 1 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- local cnsreport `cnsreport' `fullcnsreport'
= local cnsreport
- opts_exclusive ``cnsreport'
= opts_exclusive ""

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'`` != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- if ```showeqns'`` != "" {
= if `'' != "" {
    local nofirst nofirst
    }
- if ```ematrix'`noeqcheck'`` == "" {
= if `'' == "" {
- if ```e(error)'`` == "matrix" {
= if `'' == "matrix" {
    local ematrix e(error)

```

```

    }
  - }
  - _get_mcompare, `options'
  = _get_mcompare,

```

```

> ----- begin _get_mcomp
> are -----
    - version 11
    - local cmd ``e(cmd)'''
    = local cmd ``glm'''
    - syntax [name(name=caller)] [, *]
    - local is_pwc = inlist("pwcompare", ``cmd'", ``caller'") | inlist("
> pwmean", ``cmd'", ``caller'")
    = local is_pwc = inlist("pwcompare", "glm", "") | inlist("pwmean", "
> glm", "")
    - local is_marg = inlist("margins", ``cmd'", ``e(cmd2)'", ``caller'")
> )
    = local is_marg = inlist("margins", "glm", "", "")
    - local OPTIONS NOADJust BONferroni DUNCAn DUNNett SCHeffe SIDak SNK
> TUKey ADJustall
    - if `is_pwc' {
    = if 0 {
        syntax [name] [, `OPTIONS' MCOMPare(string) *]
    }
    - else {
    - syntax [name] [, MCOMPare(string) *]
    - local suboption mcompare()
    - }
    - sreturn clear
    - sreturn local options ``options'''
    = sreturn local options `'''
    - if `is_pwc' {
    = if 0 {
        local METHOD `noadjust' `bonferroni' `duncan' `dunnett' `scheffe'
> `sidak' `snk' `tukey'
        local METHOD : list uniq METHOD
        opts_exclusive ``METHOD'''
        local ALL `adjustall'
        opts_exclusive ``noadjust' `ALL'''
    }
    - local 0 `", `mcompare'''
    = local 0 `", ''
    - capture syntax [, `OPTIONS']
    = capture syntax [, NOADJust BONferroni DUNCAn DUNNett SCHeffe SIDak
> SNK TUKey ADJustall]
    - if c(rc) {

```



```

        di as err "invalid mcompare() option;"
        di as err "method ``mcompare'' not allowed"
        error 198
    }
    - local METHOD `METHOD' `noadjust' `bonferroni' `duncan' `dunnett' `
> scheffe' `sidak' `snk' `tukey'
    = local METHOD
    - local METHOD : list uniq METHOD
    - local ALL `ALL' `adjustall'
    = local ALL
    - local ALL : list uniq ALL
    - opts_exclusive ``METHOD' ``suboption'
    = opts_exclusive "" mcompare()

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - if ``METHOD' == "" {
    = if "" == "" {
    - if ``ALL' != "" {
    = if "" != "" {
        di as err "multiple comparison method required with the adjustall
> option"
        exit 198
    }
    - local METHOD noadjust
    - local noadjust noadjust
    - }
    - opts_exclusive ``ALL' `noadjust' ``suboption'
    = opts_exclusive " noadjust" mcompare()

```

```

> ----- begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts  noadjust
    - local n 0
    - while ```opts'`` != "" {
    = while `"noadjust" != "" {
    - local ++n
    - gettoken item`n' opts : opts, bind
    = gettoken item1 opts : opts, bind
    - }
    - while ```opts'`` != "" {
    = while `"" != "" {
      local ++n
      gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 1 < 2 {
    - exit

> ----- end opts_exclus
> ive
    - local EXCLUDE duncan dunnett snk tukey
    - if `:list METHOD in EXCLUDE' {
    = if 0 {
      if `is_pwc' {
        local opt : copy local METHOD
      }
      else {
        local opt "mcompare(`METHOD')"
      }
      local cmdlist anova manova regress mvreg pwmean
      if ``cmd' == "pwcompare" {
        local cmd ``e(cmd_est)''
      }
      if `is_marg' {
        local cmd margins
      }
      if !`:list cmd in cmdlist' | !`is_pwc' {
        di as err `"method `opt' is not allowed with results from `cmd'"
        exit 322
      }
      if ``e(prefix)'' == "svy" {
        di as err `"method `opt' is not allowed with results using the svy

```

```

> prefix''
    exit 322
    }
    local vce ``e(vce)''
    if !inlist("`vce'", "", "ols") {
di as err ``method `opt' is not allowed with results using vce(`vc
> e')''
    exit 322
    }
    }
    - local EXCLUDE duncan dunnett snk tukey scheffe
    - if `:list METHOD in EXCLUDE' {
= if 0 {
    if ``ALL'' != "" {
di as err ``option adjustall is not allowed with method `METHOD''
exit 198
    }
    }
    - sreturn local method ``METHOD''
= sreturn local method `noadjust'
    - sreturn local adjustall ``ALL''
= sreturn local adjustall `''



---


> ----- end _get_mcomp
> are -----
    - local method ``s(method)''
= local method `noadjust'
    - local all ``s(adjustall)''
= local all `''
    - local options ``s(options)''
= local options `''
    - if "`method'" != "noadjust" {
= if "noadjust" != "noadjust" {
    local keepmc 1
    }
    - opts_exclusive "`all' `groups'"
= opts_exclusive " "



---


> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ``opts'' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - if "`method'" == "dunnett" {
= if "noadjust" == "dunnett" {
    opts_exclusive "`method' `groups'"
}
- _get_diopts diopts options, `options' `fvwrap' `fvlabel'
= _get_diopts diopts options,
}

> ----- begin _get_dio
> pts -----
    - version 11
    - syntax namelist(max=2) [, *]
    - gettoken name1 namelist : namelist
    - gettoken name2 namelist : namelist
    - GetDiopts diopts options, `options'
= GetDiopts diopts options,
}

> ----- begin _get_diopts.GetDio
> pts -----
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nbaselevels'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nbaselevels'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n

```

```

        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    end opts_exclus
>
> ive
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive

```



```

- opts_exclusive "`nocl' `nopvalues'"
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

> _____ end opts_exclus
> ive _____
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

> _____ end opts_exclus
> ive _____
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``abbrev' `noabbrev'
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells

```

```

- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {

```

```

= if 0 {
    confirm numeric format `sformat'
    if fmtwidth(`""sformat'") > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat `""sformat'""
    local sformat `sformat(`sformat')""
}
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth(`""pformat'") > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat `""pformat'""
    local pformat `pformat(`pformat')""
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraupon , `fvwraupon'
= Wraupon ,

```

> _____ begin _get_diopts.Wra

```

> pon _____
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive ""`word' `width'"" fvwrapon
= opts_exclusive " " fvwrapon

```

> _____ begin opts_exclus

```

> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

        = local opts
        - local n 0
        - while ``opts'' != "" {
        = while `'' != "" {
            local ++n
            gettoken item`n' opts : opts, bind
        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }

> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }

> _____ end _get_diopts.Wra
> pon _____
    - if ``fvwrap'' != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts
    - if `K' == 2 {
    = if 2 == 2 {
        - c_local `c_opts' ``options''
        = c_local options `''
        - }
        - sreturn local coding `coding'
        = sreturn local coding
        - sreturn local versus `versus'
        = sreturn local versus
        - sreturn local compare `compare'
        = sreturn local compare
        - sreturn local level `level'
        = sreturn local level 95

> _____ end _get_diopts.GetDio

```

```

> pts
    - c_local `name1' ``diopts'''
    = c_local diopts `'''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)
    - }
    - local S_`sname' = s(`sname')
    = local S_method = s(method)
    - }
    - sreturn clear
    - GetDiopts DEFAULT

```

```

> _____ begin _get_diopts.GetDio
> pts
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive ``allbaselevels' `noallbaselevels'
    = opts_exclusive " "

```

```

> _____ begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts''' != "" {
    = while `''' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {

```

```

        - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____

```

```

- opts_exclusive "`emptycells' `noemptycells'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'` != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit


---


> 

---

 end opts_exclus
> ive 

---


- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'` != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit


---


> 

---

 end opts_exclus
> ive 

---


- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "


---


> 

---

 begin opts_exclus

```



```

> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

>
> end opts_exclus
> ive
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "

>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'

```

```

= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``coeflegend' `selegend'
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive ``noci' `nopvalues'
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'"" != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind

```

```

        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells
    - local omitted `omitted' `noomitted'
    = local omitted
    - local lstretch `lstretch' `nolstretch'
    = local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
    = local fvlabel
    - local abbrev `abbrev' `noabbrev'
    = local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
    = if "" == "" {
    - if c(showbaselevels) == "all" {
        local allbaselevels allbaselevels

```

```

    }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')'
  }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'

```

```

        if fmtwidth("`pformat'") > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
        }
        sreturn local pformat "`pformat'"
        local pformat `pformat(`pformat')'
    }
    - if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
        syntax namelist(max=2) [, `DIOPTS']
    }
    - if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
        local levelopt level(`level')
    }
    - Wrapon , `fvwrapon'
= Wrapon ,

> ----- begin _get_diopts.Wra
> pon -----
    - capture syntax [, WOrd WIdth]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, WOrd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while "`opts'" != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
        - exit

> ----- end opts_exclus

```

```

> ive -----
      - local fvwrapon `word' `width'
      = local fvwrapon
      - if `:length local fvwrapon' {
      = if 0 {
          c_local fvwrapon fvwrapon(`fvwrapon')
      }
-----

> ----- end _get_diopts.Wra
> pon -----
      - if "`fvwrap'" != "" {
      = if "" != "" {
          FVWrap, `fvwrap'
      }
      - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
      = c_local DEFAULT
      - if `K' == 2 {
      = if 1 == 2 {
          c_local `c_opts' ``options''
      }
      - sreturn local coding `coding'
      = sreturn local coding
      - sreturn local versus `versus'
      = sreturn local versus
      - sreturn local compare `compare'
      = sreturn local compare
      - sreturn local level `level'
      = sreturn local level 95
-----

> ----- end _get_diopts.GetDio
> pts -----
      - if ``name2'' == "" {
      = if ``options'' == "" {
          GetDiopts DUPS, `options'
      }
      - else {
      - GetDiopts DUPS options, `options'
      = GetDiopts DUPS options,
-----

> ----- begin _get_diopts.GetDio
> pts -----
      - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells

```

```

> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit


---


> 

---

 end opts_exclus
> ive 

---


    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind

```



```

        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
}

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`lstretch' `nolstretch'"

```

```

= opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive ``cnsreport' `fullcnsreport'
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive ``coeflegend' `selegend'
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {

```

```

= while `"" != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
}
- }
- if "`lstretch'" == "" {

```

```

= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth(`""cformat'') > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat `""cformat''
  local cformat `"cformat(`cformat')"'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth(`""sformat'') > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat `""sformat''
  local sformat `"sformat(`sformat')"'
}
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth(`""pformat'') > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat `""pformat''
  local pformat `"pformat(`pformat')"'
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
  syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
  local levelopt level(`level')
}
- Wraon , `fvwraon'
= Wraon ,

```

```

> _____ begin _get_diopts.Wra
> pon _____
    - capture syntax [, WOrd Width]
    - if c(rc) {
        di as err "invalid fvwrapon() option;"
        syntax [, WOrd Width]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwrapon
    = opts_exclusive " " fvwrapon
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit
    }
    _____

> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
    _____

> _____ end _get_diopts.Wra
> pon _____
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'

```



```

= c_local DUPS
- if `K' == 2 {
= if 2 == 2 {
- c_local `c_opts' ``options'''
= c_local options `'''
- }
- sreturn local coding `coding'
= sreturn local coding
- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> _____ end _get_diopts.GetDio
> pts _____
- c_local `name2' ``options'''
= c_local options `'''
- }
- local DUPS : list DUPS - DEFAULT
- gettoken DUPS : DUPS
- if ``DUPS''' != "" {
= if `''' != "" {
  if (strpos(`DUPS''', "(")) {
    gettoken DUPS : DUPS, parse("(")
    local DUPS `DUPS'()'
  }
  di as err "duplicate {bf:`DUPS'} option not allowed"
  exit 198
}
- sreturn clear
- foreach sname of local snames {
- sreturn local `sname' = ``S_`sname''''
= sreturn local level = `95''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local method = `noadjust''
- }

```

```

> _____ end _get_dio
> pts _____
- local legend coeflegend selegend
- local legend : list legend & diopts
- local diopts : list diopts - legend
- local lstretch lstretch nolstretch

```

```

- local lstretch : list lstretch & diopts
- local diopts : list diopts - lstretch
- local markdown markdown
- local markdown : list markdown & diopts
- local diopts : list diopts - markdown
- _get_diopts ignore, `diopts' `cformat' `sformat' `pformat'
= _get_diopts ignore,

```

```

> ----- begin _get_dio
> pts -----
- version 11
- syntax namelist(max=2) [, *]
- gettoken name1 namelist : namelist
- gettoken name2 namelist : namelist
- GetDiopts diopts options, `options'
= GetDiopts diopts options,

```

```

> ----- begin _get_diopts.GetDio
> pts -----
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'`` != "" {
= while `""' != "" {
local ++n
gettoken item`n' opts : opts, bind

```

```

        }
        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nbaselevels'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nbaselevels'"
    = opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`omitted' `noomitted'"

```

```

= opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

```

```

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
    }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`abbrev` `noabbrev`"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while ``opts`` != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels` `noallbaselevels`
= local allbaselevels
    - local baselevels `baselevels` `nobaselevels`
= local baselevels
    - local emptycells `emptycells` `noemptycells`
= local emptycells
    - local omitted `omitted` `noomitted`
= local omitted
    - local lstretch `lstretch` `nolstretch`
= local lstretch
    - local fvlabel `fvlabel` `nofvlabel`
= local fvlabel
    - local abbrev `abbrev` `noabbrev`
= local abbrev
    - if "`allbaselevels` `baselevels`" == "" {

```



```

= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')'
  }
}

```

```

- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth(`"'pformat'"') > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat `"'pformat'"'
  local pformat `pformat(`pformat')"'
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
  syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
  local levelopt level(`level')
}
- Wrapon , `fvwrapon'
= Wrapon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
  di as err "invalid fvwrapon() option;"
  syntax [, W0rd WIdth]
  exit 198
}
- opts_exclusive "'word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

> _____ begin opts_exclus

> ive _____

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while `"'opts'"' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    _____
> _____ end opts_exclus
> ive _____
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
    _____

> _____ end _get_diopts.Wra
> pon _____
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts
    - if `K' == 2 {
    = if 2 == 2 {
    - c_local `c_opts' ``options''
    = c_local options `''
    - }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95
    _____

> _____ end _get_diopts.GetDio
> pts _____
    - c_local `name1' ``diopts''
    = c_local ignore `''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)
    - }
    - local S_`sname' = s(`sname')

```

```

        = local S_method = s(method)
        - }
        - sreturn clear
        - GetDiopts DEFAULT
    }
}

> ----- begin _get_diopts.GetDio
> pts -----
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `"'opts'"' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nobaselevels'"
    = opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n

```

```

        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts' != "" {
= while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive ``coeflegend' `selegend'
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive ``noci' `nopvalues'
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____

```



```

- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive ``coding' `compare'
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'`` != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

> _____ end opts_exclus
> ive _____
- opts_exclusive ``versus' `compare'
= opts_exclusive " "

> _____ begin opts_exclus
> ive _____
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'`` != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

> _____ end opts_exclus
> ive _____
- opts_exclusive ``abbrev' `noabbrev'
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `'' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit

> ----- end opts_exclus
> ive -----
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels
    - local baselevels `baselevels' `nobaselevels'
    = local baselevels
    - local emptycells `emptycells' `noemptycells'
    = local emptycells
    - local omitted `omitted' `noomitted'
    = local omitted
    - local lstretch `lstretch' `nolstretch'
    = local lstretch
    - local fvlabel `fvlabel' `nofvlabel'
    = local fvlabel
    - local abbrev `abbrev' `noabbrev'
    = local abbrev
    - if "`allbaselevels'`baselevels'" == "" {
    = if "" == "" {
    - if c(showbaselevels) == "all" {
        local allbaselevels allbaselevels
    }
    - else if c(showbaselevels) == "on" {
        local baselevels baselevels
    }
    - }
    - if "`emptycells'" == "" {
    = if "" == "" {
    - if c(showemptycells) == "off" {
        local emptycells noemptycells

```

```

    }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
      di "{txt}note: invalid cformat(), using default"
    }
    sreturn local cformat "`cformat'"
    local cformat `cformat(`cformat')"'
  }
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth("`sformat'") > 8 {
      local sformat "%8.2f"
      di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat "`sformat'"
    local sformat `sformat(`sformat')"'
  }
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth("`pformat'") > 5 {
      local pformat "%5.3f"
      di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat "`pformat'"
    local pformat `pformat(`pformat')"'
  }
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {

```

```

        syntax namelist(max=2) [, `DIOPTS']
    }
    - if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
        local levelopt level(`level')
    }
    - Wrapon , `fvwrapon'
= Wrapon ,

> _____ begin _get_diopts.Wra
> pon _____
        - capture syntax [, W0rd WIdth]
        - if c(rc) {
            di as err "invalid fvwrapon() option;"
            syntax [, W0rd WIdth]
            exit 198
        }
        - opts_exclusive "`word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

> _____ begin opts_exclus
> ive _____
        - version 8.2
        - args opts optname errcode optslab
        - local opts `opts'
= local opts
        - local n 0
        - while ``opts' != "" {
= while `"' != "" {
            local ++n
            gettoken item`n' opts : opts, bind
        }
        - if `n' < 2 {
= if 0 < 2 {
            - exit

> _____ end opts_exclus
> ive _____
        - local fvwrapon `word' `width'
= local fvwrapon
        - if `:length local fvwrapon' {
= if 0 {
            c_local fvwrapon fvwrapon(`fvwrapon')
        }

> _____ end _get_diopts.Wra

```

```

> pon
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DEFAULT
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options''
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'
    = sreturn local compare
    - sreturn local level `level'
    = sreturn local level 95

> _____ end _get_diopts.GetDio
> pts
    - if ``name2'' == "" {
    = if ``' == "" {
    - GetDiopts DUPS, `options'
    = GetDiopts DUPS,

> _____ begin _get_diopts.GetDio
> pts
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive ``allbaselevels' `noallbaselevels''

```

```

= opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`allbaselevels' `nbaselevels'"
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`baselevels' `nbaselevels'"
    = opts_exclusive " "
> _____ begin opts_exclus
> ive _____

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `"' != "" {
  local ++n

```



```

        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    _____
> _____ end opts_exclus
> ive _____
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____
> _____ end opts_exclus
> ive _____
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "
    _____
> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

```

```

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""opts'" != "" {
    = while `"" != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while `""opts'" != "" {
    = while `"" != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - local allbaselevels `allbaselevels' `noallbaselevels'
    = local allbaselevels

```

```

- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
    }
    di "{txt}note: invalid cformat(), using default"
  }
}

```

```

    sreturn local cformat ``cformat'''
    local cformat `cformat(`cformat')''
}
- if `:length local sformat' {
= if 0 {
    confirm numeric format `sformat'
    if fmtwidth(`sformat') > 8 {
        local sformat "%8.2f"
        di "{txt}note: invalid sformat(), using default"
    }
    sreturn local sformat ``sformat'''
    local sformat `sformat(`sformat')''
}
- if `:length local pformat' {
= if 0 {
    confirm numeric format `pformat'
    if fmtwidth(`pformat') > 5 {
        local pformat "%5.3f"
        di "{txt}note: invalid pformat(), using default"
    }
    sreturn local pformat ``pformat'''
    local pformat `pformat(`pformat')''
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraon , `fvwraon'
= Wraon ,

```

> _____ begin _get_diopts.Wra

> pon _____

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
    di as err "invalid fvwraon() option;"
    syntax [, W0rd WIdth]
    exit 198
}
- opts_exclusive ``word' `width'" fvwraon
= opts_exclusive " " fvwraon

```

> _____ begin opts_exclus

```

> ive
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

> ----- end opts_exclus
> ive
- local fwwrapon `word' `width'
= local fwwrapon
- if `:length local fwwrapon' {
= if 0 {
  c_local fwwrapon fwwrapon(`fwwrapon')
}

> ----- end _get_diopts.Wra
> pon
- if "`fwwrap'" != "" {
= if "" != "" {
  FWWrap, `fwwrap'
}
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fwwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
= c_local DUPS
- if `K' == 2 {
= if 1 == 2 {
  c_local `c_opts' ``options'
}
- sreturn local coding `coding'
= sreturn local coding
- sreturn local versus `versus'
= sreturn local versus
- sreturn local compare `compare'
= sreturn local compare
- sreturn local level `level'

```

```

        = sreturn local level 95
    }
}
> ----- end _get_diopts.GetDio
> pts
    - }
    - else {
        GetDiopts DUPS options, `options'
        c_local `name2' ``options'
    }
    - local DUPS : list DUPS - DEFAULT
    - gettoken DUPS : DUPS
    - if ``DUPS' != "" {
= if `'' != "" {
    if (strpos(`DUPS', "(")) {
        gettoken DUPS : DUPS, parse("(")
        local DUPS `DUPS'()'
    }
    di as err "duplicate {bf:`DUPS'} option not allowed"
    exit 198
}
    - sreturn clear
    - foreach sname of local snames {
        - sreturn local `sname' = ``S_`sname'
= sreturn local level = `95'
    }
    - sreturn local `sname' = ``S_`sname'
= sreturn local method = `noadjust'
    - }
}
> ----- end _get_dio
> pts
    - local cformat ``s(cformat)'
= local cformat `''
    - local sformat ``s(sformat)'
= local sformat `''
    - local pformat ``s(pformat)'
= local pformat `''
    - _get_eformopts , eformopts(`options') allowed(__all__) soptions
= _get_eformopts , eformopts() allowed(__all__) soptions
}
> ----- begin _get_eformo
> pts
    - version 8.0
    - syntax [, eformopts(string asis) soptions ALLOWed(string) ]
    - local 0 , `eformopts'
= local 0 ,

```

```

- local EFALL NOHR hr NOSHR shr or IRr RRr tr TRatio
- if "`soptions'" != "" {
= if "soptions" != "" {
- syntax [, EForm1(passthru) EForm `EFALL' * ]
= syntax [, EForm1(passthru) EForm NOHR hr NOSHR shr or IRr RRr tr T
> Ratio * ]
- local opts ``options'
= local opts `''
- local 0 , `eform1' `eform' `nohr' `hr' `noshr' `shr' `or' `irr' `r
> rr' `tr' `tratio'
= local 0 ,
- }
- if "`allowed'" == "__all__" {
= if "__all__" == "__all__" {
- local allowed `EFALL'
= local allowed NOHR hr NOSHR shr or IRr RRr tr TRatio
- }
- foreach ef of local allowed {
- capture confirm name `ef'
= capture confirm name NOHR
- if _rc {
di as err "`ef' is not a valid name"
exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr
- }
- capture confirm name `ef'
= capture confirm name hr
- if _rc {
di as err "`ef' is not a valid name"
exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr
- }
- capture confirm name `ef'
= capture confirm name NOSHR
- if _rc {
di as err "`ef' is not a valid name"
exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr
- }
- capture confirm name `ef'

```



```

= capture confirm name shr
- if _rc {
  di as err "`ef' is not a valid name"
  exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr
- }
- capture confirm name `ef'
= capture confirm name or
- if _rc {
  di as err "`ef' is not a valid name"
  exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or
- }
- capture confirm name `ef'
= capture confirm name IRr
- if _rc {
  di as err "`ef' is not a valid name"
  exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or irr
- }
- capture confirm name `ef'
= capture confirm name RRr
- if _rc {
  di as err "`ef' is not a valid name"
  exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or irr rrr
- }
- capture confirm name `ef'
= capture confirm name tr
- if _rc {
  di as err "`ef' is not a valid name"
  exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or irr rrr tr
- }
- capture confirm name `ef'
= capture confirm name TRatio

```

```

- if _rc {
  di as err "'ef' is not a valid name"
  exit 198
}
- local efopts `efopts' `=lower("`ef'")'
= local efopts nohr hr noshr shr or irr rrr tr tratio
- }
- local efopts : list uniq efopts
- syntax [, EForm1(string) EForm `allowed' ]
= syntax [, EForm1(string) EForm NOHR hr NOSHR shr or IRr RRr tr TRa
> tio ]

- foreach ef of local efopts {
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- local eform `eform' ``ef''
= local eform
- }
- if ```eform1'''"=="Odds ratio" {
= if `""'"=="Odds ratio" {
  local eform_cons_ti = "Odds"
  local eform_cons_note = "odds ratios"
}
- else if ```eform1'''"=="Haz. ratio" {
= else if `""'"=="Haz. ratio" {

```

```

    local eform_cons_ti = "Hazard"
    local eform_cons_note = "hazard ratios"
  }
- else if ``eform1``=="Time ratio" {
= else if ``""=="Time ratio" {
    local eform_cons_ti = "Time"
    local eform_cons_note = "time ratios"
  }
- else if ``eform1``=="IRR" {
= else if ``""=="IRR" {
    local eform_cons_ti = "Inc. rate"
    local eform_cons_note = "incidence-rate ratios"
    local eform_cons_ti2 = "incidence rate"
  }
- else if ``eform1``=="Risk ratio" {
= else if ``""=="Risk ratio" {
    local eform_cons_ti = "Risk"
    local eform_cons_note = "risk ratios"
  }
- else if ``eform1``=="Hlth ratio" {
= else if ``""=="Hlth ratio" {
    local eform_cons_ti = "Health"
    local eform_cons_note = "health ratios"
  }
- local k : list sizeof eform
- if `k' {
= if 0 {
    opts_exclusive ``eform''
    if `:length local eform1' {
      opts_exclusive "eform() `eform'"
    }
  }
- if `k' {
= if 0 {
    if (``eform'=="eform") local eform1 = "exp(b)"
    else if (``eform'=="hr") {
      local eform1 = "Haz. ratio"
      local eform_cons_ti = "Hazard"
      local eform_cons_note = "hazard ratios"
    }
    else if (``eform'=="shr") {
      local eform1 = "SHR"
      local eform_cons_ti = "Subhazard"
      local eform_cons_note = "subhazard ratios"
    }
    else if (``eform'=="tr") {

```

```

    local eform1 = "Time ratio"
    local eform_cons_ti = "Time"
    local eform_cons_note = "time ratios"
  }
  else if ("`eform'"=="tratio") {
    local eform1 = "Time ratio"
    local eform_cons_ti = "Time"
    local eform_cons_note = "time ratios"
  }
  else if ("`eform'"=="or") {
    local eform1 = "Odds ratio"
    if ("`e(cmd)'"=="asclogit" | "`e(cmd)'"=="cmclogit" | "`e(cmd)'"=="
> "cmmixlogit" | "`e(cmd)'"=="cmxtmixlogit" ) {
      local eform_cons_ti = "Rel. risk"
      local eform_cons_note = "relative-risk ratios"
      local eform_cons_ti2 = "relative risk"
    }
    else {
      local eform_cons_ti = "Odds"
      local eform_cons_note = "odds ratios"
    }
  }
  else if ("`eform'"=="irr") {
    local eform1 = "IRR"
    local eform_cons_ti = "Inc. rate"
    local eform_cons_note = "incidence-rate ratios"
    local eform_cons_ti2 = "incidence rate"
  }
  else if ("`eform'"=="rrr") {
    local eform1 = "RRR"
    local eform_cons_ti = "Rel. risk"
    local eform_cons_note = "relative-risk ratios"
    local eform_cons_ti2 = "relative risk"
  }
  else if ("`eform'"!="nohr") & ("`eform'"!="noshr") {
    local eform1 = upper("`eform'")
  }
}

- sreturn clear
- sreturn local options ``opts'''
= sreturn local options `'''
- sreturn local opt `eform'
= sreturn local opt
- sreturn local str `eform1'
= sreturn local str
- if ``eform1''' != "" {

```

```

= if `""` != "" {
  sreturn local eform eform(`""eform1'`)
}
- else sreturn local eform ""
- sreturn local eform_cons_ti `eform_cons_ti'
= sreturn local eform_cons_ti
- sreturn local eform_cons_note `eform_cons_note'
= sreturn local eform_cons_note
- sreturn local eform_cons_ti2 `eform_cons_ti2'
= sreturn local eform_cons_ti2

```

```

> ----- end _get_eformo
> pts -----
- local tr_item = ``s(eform_cons_note)''
= local tr_item = ``''
- if inlist("`bmatrix'", "", "e(b)") & "`e(k_eform)'" == "0" {
= if inlist("", "", "e(b)") & "" == "0" {
  local eform
}
- else local eform ``s(str)''
= else local eform ``''
- local eform_cons_ti ``s(eform_cons_ti)''
= local eform_cons_ti ``''
- local eform_cons_ti2 ``s(eform_cons_ti2)''
= local eform_cons_ti2 ``''
- if (`e(consonly)''!="1" | ``eform_cons_ti'')=="" ) {
= if ("0"!="1" | ``'')=="" ) {
- local eformdi ``s(str)''
= local eformdi ``''
- }
- else {
  local eformdi ``eform_cons_ti''
  local tr_item = cond(missing("`eform_cons_ti2'"), strlower("`eform_
> cons_ti''), ``eform_cons_ti2''))
}
- local coefttl = cond(`eform'=='', ``coeftitle'', ``eformdi'
> '')
= local coefttl = cond(`'=='', `', `')
- local options ``s(options)''
= local options ``''
- _get_diparmopts, diparmopts(`options') level(`level')
= _get_diparmopts, diparmopts() level(95)

```

```

> ----- begin _get_diparmo
> pts -----
- if _caller() < 8.2 {

```

```

        _get_diparmopts_8 `0'
        exit
    }
    - version 9
    - local version : di "version " string(_caller()) ":"
    - syntax [, diparmopts(string asis) EXECute soptions bottom plus SYN
> TAXonly * ]
    - local goptions `options'
    = local goptions level(95)
    - SpliceGlobals junk , `goptions'
    = SpliceGlobals junk , level(95)

> _____ begin _get_diparmopts.SpliceGlob
> als _____
    - syntax name(id="macro name" name=c_diparm) [, diparm(string asis
> ) Level(cilevel) dof(passthru) syntaxonly ]
    - if ```dof''' == "" & ```e(df_r)''' != "" {
    = if `''' == "" & `''' != "" {
        capture confirm integer number `e(df_r)'
        if !c(rc) {
            local gdof dof(`e(df_r)')
        }
        capture
    }
    - else local gdof ```dof'''
    = else local gdof `'''
    - local glevel `level'
    = local glevel 95
    - local lopt level(`glevel')
    = local lopt level(95)
    - if (```diparm''' == "") exit
    = if (``''' == "") exit

> _____ end _get_diparmopts.SpliceGlob
> als _____
    - local 0 `", `diparmopts'''
    = local 0 `", ""
    - local diparmopts
    - local k 0
    - syntax [, diparm(string asis) * ]
    - while ```diparm''' != "" {
    = while `''' != "" {
        local ++k
        local single 1
        local after
        capture _on_colon_parse `diparm'

```

```

if !c(rc) {
  local single 0
  local diparm ``s(before)''
  local after ``s(after)''
}
capture
SpliceGlobals diparm , diparm(`diparm') `syntaxonly' `goptions'
local diparm`k' ``diparm''
local diparmopts ``diparmopts' : `diparm''
if `single' | ``:list retok after'' == '' {
  local 0 `", `options''
  syntax [, diparm(string asis) * ]
}
else {
  local diparm ``after''
}
}
- if ``soptions'' == '' {
= if ``'' == '' {
- syntax [, diparm(string asis) ]
- }
- if ("`execute" == "") {
= if ("" == "") {
- sreturn clear
- sreturn local options ``options''
= sreturn local options ``''
- sreturn local diparm ``diparmopts''
= sreturn local diparm ``''
- forval i = 1/`k' {
= forval i = 1/0 {
  sreturn local diparm`i' ``diparm`i''
}
- sreturn local k `k'
= sreturn local k 0
- exit

```

> _____ end _get_diparmo

> pts

```

- local NODIPARM : length local nodiparm
- if `NODIPARM' {
= if 0 {
  local options
}
- local GTOPTS parse(":") bind quotes
- local k 0
- if !`NODIPARM' {

```

```

= if !0 {
- if ``e(diparm)'' != '' {
= if `'' != '' {
    local i 0
    local diparm ``e(diparm)''
  }
- else if ``e(diparm1)'' != '' {
= else if `'' != '' {
    local i 1
    local diparm ``e(diparm1)''
  }
- while `:length local diparm' {
= while 0 {
  gettoken diparm rest : diparm, `GTOPTS'
  while `:length local rest' {
    local ++k
    local diparm`k' : copy local diparm
    gettoken COLON rest : rest, `GTOPTS'
    gettoken diparm rest : rest, `GTOPTS'
  }
  local ++k
  local diparm`k' : copy local diparm
  local ++i
  local diparm ``e(diparm`i)''
  }
- }
- if `:length local options' {
= if 0 {
  local 0 `", `options''
  syntax [, diparm(string asis) *]
  if `k' & `:length local diparm' {
    local ++k
    local diparm`k' __sep__
  }
  while `:length local diparm' {
    gettoken diparm rest : diparm, `GTOPTS'
    while `:length local rest' {
      local ++k
      local diparm`k' : copy local diparm
      gettoken COLON rest : rest, `GTOPTS'
      gettoken diparm rest : rest, `GTOPTS'
    }
    local ++k
    local diparm`k' : copy local diparm
    local 0 `", `options''
    syntax [, diparm(string asis) *]
  }
}

```



```

    }
    }
    - if "`markdown'" != "" {
= if "" != "" {
    local lsize = c(linesize)
    set linesize 255
    }
    - `vv' mata: _coef_table()
= version 11, missing: mata: _coef_table()

```

```

> _____ begin _get_dio
> pts _____
    - version 11
    - syntax namelist(max=2) [, *]
    - gettoken name1 namelist : namelist
    - gettoken name2 namelist : namelist
    - GetDiopts diopts options, `options'
= GetDiopts diopts options,

```

```

> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
    - opts_exclusive "`allbaselevels' `noallbaselevels'"
    = opts_exclusive " "

```

```

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts'" != "" {
    = while `""' != "" {

```

```

        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`allbaselevels' `nbaselevels'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
= if 0 < 2 {
    - exit
}
}
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nbaselevels'"
= opts_exclusive " "
}
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
= local opts
    - local n 0
    - while ```opts'`` != "" {
= while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {

```

```

        = if 0 < 2 {
        - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus
> ive -----
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "
> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'`` != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    }
> ----- end opts_exclus

```

```

> ive -----
    - opts_exclusive "`omitted' `noomitted'"
    = opts_exclusive " "
-----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
-----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch' `nolstretch'"
    = opts_exclusive " "
-----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ```opts'"" != "" {
    = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
-----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport' `fullcnsreport'"
    = opts_exclusive " "
-----

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend' `selegend'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`noci' `nopvalues'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`versus' `compare'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts

```

```

- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `'' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'

```

```

= local abbrev
- if "`allbaselevels'"`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"

```



```

    local sformat ``sformat(`sformat')''
  }
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth(`"pformat"') > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat ``pformat''
  local pformat `pformat(`pformat')''
}
- if `K' == 1 & `:length local options' {
= if 2 == 1 & 0 {
  syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
  local levelopt level(`level')
}
- Wrapon , `fvwrapon'
= Wrapon ,

```

```

> ----- begin _get_diopts.Wra
> pon -----

```

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
  di as err "invalid fvwrapon() option;"
  syntax [, W0rd WIdth]
  exit 198
}
- opts_exclusive ``word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

```

> ----- begin opts_exclus
> ive -----

```

```

- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts'' != "" {
= while `""' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}

```

```

        - if `n' < 2 {
        = if 0 < 2 {
        - exit
    }
}

> ----- end opts_exclus
> ive -----
    - local fvwrapon `word' `width'
    = local fvwrapon
    - if `:length local fvwrapon' {
    = if 0 {
        c_local fvwrapon fvwrapon(`fvwrapon')
    }
}

> ----- end _get_diopts.Wra
> pon -----
    - if "`fvwrap'" != "" {
    = if "" != "" {
        FVWrap, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbrev'
    = c_local diopts
    - if `K' == 2 {
    = if 2 == 2 {
        - c_local `c_opts' ``options''
        = c_local options ``''
        - }
        - sreturn local coding `coding'
        = sreturn local coding
        - sreturn local versus `versus'
        = sreturn local versus
        - sreturn local compare `compare'
        = sreturn local compare
        - sreturn local level `level'
        = sreturn local level 95
    }

> ----- end _get_diopts.GetDio
> pts -----
    - c_local `name1' ``diopts''
    = c_local __00000D ``''
    - local snames : s(macros)
    - foreach sname of local snames {
    - local S_`sname' = s(`sname')
    = local S_level = s(level)

```

```

- }
- local S_`sname' = s(`sname')
= local S_k = s(k)
- }
- sreturn clear
- GetDiopts DEFAULT

```

```

> ----- begin _get_diopts.GetDio
> pts -----
- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass
> thru) NOABBREV ABBREV *]
- opts_exclusive "`allbaselevels' `noallbaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ```opts'"" != "" {
= while `""' != "" {
local ++n
gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- opts_exclusive "`allbaselevels' `nobaselevels'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`baselevels' `nobaselevels'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`fvlabel' `nofvlabel'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`lstretch` `nolstretch`"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport` `fullcnsreport`"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}

```

```

- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`coeflegend' `selegend'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

```

```

> _____ end opts_exclus
> ive _____
    - local K : list sizeof namelist
    - gettoken c_diopts c_opts : namelist
    - opts_exclusive "`coding' `compare'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
    _____

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____

```



```

- opts_exclusive "`abbrev' `noabbrev'"
= opts_exclusive " "

```

```

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
  local ++n
  gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit

```

```

> ----- end opts_exclus
> ive -----
- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
  local allbaselevels allbaselevels
}
- else if c(showbaselevels) == "on" {
  local baselevels baselevels
}
- }
- if "`emptycells'" == "" {
= if "" == "" {

```

```

- if c(showemptycells) == "off" {
  local emptycells noemptycells
}
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
  local omitted noomitted
}
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
  local lstretch nolstretch
}
- }
- if `:length local cformat' {
= if 0 {
  confirm numeric format `cformat'
  if fmtwidth("`cformat'") > 9 {
    local cformat "%9.0g"
    di "{txt}note: invalid cformat(), using default"
  }
  sreturn local cformat "`cformat'"
  local cformat `cformat(`cformat')'
}
- if `:length local sformat' {
= if 0 {
  confirm numeric format `sformat'
  if fmtwidth("`sformat'") > 8 {
    local sformat "%8.2f"
    di "{txt}note: invalid sformat(), using default"
  }
  sreturn local sformat "`sformat'"
  local sformat `sformat(`sformat')'
}
- if `:length local pformat' {
= if 0 {
  confirm numeric format `pformat'
  if fmtwidth("`pformat'") > 5 {
    local pformat "%5.3f"
    di "{txt}note: invalid pformat(), using default"
  }
  sreturn local pformat "`pformat'"
  local pformat `pformat(`pformat')'
}

```

```

- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
    syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
    local levelopt level(`level')
}
- Wraapon , `fvwraapon'
= Wraapon ,

> ----- begin _get_diopts.Wra
> pon -----
    - capture syntax [, W0rd WIdth]
    - if c(rc) {
        di as err "invalid fvwraapon() option;"
        syntax [, W0rd WIdth]
        exit 198
    }
    - opts_exclusive "`word' `width'" fvwraapon
    = opts_exclusive " " fvwraapon

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
        - exit

> ----- end opts_exclus
> ive -----
    - local fvwraapon `word' `width'
    = local fvwraapon
    - if `:length local fvwraapon' {
    = if 0 {
        c_local fvwraapon fvwraapon(`fvwraapon')
    }

```

```

> _____ end _get_diopts.Wra
> pon _____
      - if "`fvwrap'" != "" {
      = if "" != "" {
        FVWrap, `fvwrap'
      }
      - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
      = c_local DEFAULT
      - if `K' == 2 {
      = if 1 == 2 {
        c_local `c_opts' ``options''
      }
      - sreturn local coding `coding'
      = sreturn local coding
      - sreturn local versus `versus'
      = sreturn local versus
      - sreturn local compare `compare'
      = sreturn local compare
      - sreturn local level `level'
      = sreturn local level 95
      _____

> _____ end _get_diopts.GetDio
> pts _____
      - if ``name2'' == "" {
      = if ``' == "" {
      - GetDiopts DUPS, `options'
      = GetDiopts DUPS,
      _____

> _____ begin _get_diopts.GetDio
> pts _____
      - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElevel
> s NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcells
> NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string) sf
> ormat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRAP0
> N(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABBRE
> V ABBREV
      - syntax namelist(max=2) [, `DIOPTS' *]
      = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASElevels
> ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcell
> s EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cform
> at(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pass
> thru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pass

```

```

> thru) NOABBREV ABBREV *]
      - opts_exclusive "`allbaselevels' `noallbaselevels'"
      = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


      - version 8.2
      - args opts optname errcode optslab
      - local opts `opts'
      = local opts
      - local n 0
      - while ```opts'`` != "" {
      = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
      }
      - if `n' < 2 {
      = if 0 < 2 {
      - exit


---


> 

---

 end opts_exclus
> ive 

---


      - opts_exclusive "`allbaselevels' `nobaselevels'"
      = opts_exclusive " "


---


> 

---

 begin opts_exclus
> ive 

---


      - version 8.2
      - args opts optname errcode optslab
      - local opts `opts'
      = local opts
      - local n 0
      - while ```opts'`` != "" {
      = while `""' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
      }
      - if `n' < 2 {
      = if 0 < 2 {
      - exit


---


> 

---

 end opts_exclus
> ive 

---


      - opts_exclusive "`baselevels' `nobaselevels'"
      = opts_exclusive " "


---



```

```

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`emptycells' `noemptycells'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    -----

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`fvlabel' `nofvlabel'"
    = opts_exclusive " "
    -----

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab

```

```

- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`omitted' `noomitted'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
  }
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`lstretch' `nolstretch'"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while ``opts' != "" {

```

```

= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
- if `n` < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`cnsreport` `fullcnsreport`"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}
    - if `n` < 2 {
= if 0 < 2 {
    - exit
}

> ----- end opts_exclus
> ive -----
    - opts_exclusive "`coeflegend` `selegend`"
= opts_exclusive " "
}

> ----- begin opts_exclus
> ive -----
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts`
= local opts
    - local n 0
    - while `""opts`"" != "" {
= while `""` != "" {
    local ++n
    gettoken item`n` opts : opts, bind
}

```



```

- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- opts_exclusive "`noci' `nopvalues'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {
- exit
}

> ----- end opts_exclus
> ive -----
- local K : list sizeof namelist
- gettoken c_diopts c_opts : namelist
- opts_exclusive "`coding' `compare'"
= opts_exclusive " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local n 0
- while "`opts'" != "" {
= while `"' != "" {
    local ++n
    gettoken item`n' opts : opts, bind
}
- if `n' < 2 {
= if 0 < 2 {

```

```

        - exit
    end opts_exclus
>
> ive
    - opts_exclusive "`versus' `compare'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive
    - opts_exclusive "`abbrev' `noabbrev'"
    = opts_exclusive " "
>
> begin opts_exclus
> ive
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    end opts_exclus
>
> end opts_exclus
>
> ive

```

```

- local allbaselevels `allbaselevels' `noallbaselevels'
= local allbaselevels
- local baselevels `baselevels' `nobaselevels'
= local baselevels
- local emptycells `emptycells' `noemptycells'
= local emptycells
- local omitted `omitted' `noomitted'
= local omitted
- local lstretch `lstretch' `nolstretch'
= local lstretch
- local fvlabel `fvlabel' `nofvlabel'
= local fvlabel
- local abbrev `abbrev' `noabbrev'
= local abbrev
- if "`allbaselevels'`baselevels'" == "" {
= if "" == "" {
- if c(showbaselevels) == "all" {
    local allbaselevels allbaselevels
  }
- else if c(showbaselevels) == "on" {
    local baselevels baselevels
  }
- }
- if "`emptycells'" == "" {
= if "" == "" {
- if c(showemptycells) == "off" {
    local emptycells noemptycells
  }
- }
- if "`omitted'" == "" {
= if "" == "" {
- if c(showomitted) == "off" {
    local omitted noomitted
  }
- }
- if "`lstretch'" == "" {
= if "" == "" {
- if c(lstretch) == "off" {
    local lstretch nolstretch
  }
- }
- if `:length local cformat' {
= if 0 {
    confirm numeric format `cformat'
    if fmtwidth("`cformat'") > 9 {
      local cformat "%9.0g"
    }
  }
}

```

```

di "{txt}note: invalid cformat(), using default"
}
sreturn local cformat ``cformat'''
local cformat `cformat(`cformat')''
}
- if `:length local sformat' {
= if 0 {
confirm numeric format `sformat'
if fmtwidth(`sformat') > 8 {
local sformat "%8.2f"
di "{txt}note: invalid sformat(), using default"
}
sreturn local sformat ``sformat'''
local sformat `sformat(`sformat')''
}
- if `:length local pformat' {
= if 0 {
confirm numeric format `pformat'
if fmtwidth(`pformat') > 5 {
local pformat "%5.3f"
di "{txt}note: invalid pformat(), using default"
}
sreturn local pformat ``pformat'''
local pformat `pformat(`pformat')''
}
- if `K' == 1 & `:length local options' {
= if 1 == 1 & 0 {
syntax namelist(max=2) [, `DIOPTS']
}
- if reldif(`level', c(level)) > 1e-3 {
= if reldif(95, c(level)) > 1e-3 {
local levelopt level(`level')
}
- Wraupon , `fvwraupon'
= Wraupon ,

```

```

> _____ begin _get_diopts.Wra
> pon _____

```

```

- capture syntax [, W0rd WIdth]
- if c(rc) {
di as err "invalid fvwrapon() option;"
syntax [, W0rd WIdth]
exit 198
}
- opts_exclusive ``word' `width'" fvwrapon
= opts_exclusive " " fvwrapon

```

```

> _____ begin opts_exclus
> ive _____
    - version 8.2
    - args opts optname errcode optslab
    - local opts `opts'
    = local opts
    - local n 0
    - while ``opts' != "" {
    = while `"' != "" {
        local ++n
        gettoken item`n' opts : opts, bind
    }
    - if `n' < 2 {
    = if 0 < 2 {
    - exit
    _____

> _____ end opts_exclus
> ive _____
    - local fwwrapon `word' `width'
    = local fwwrapon
    - if `:length local fwwrapon' {
    = if 0 {
        c_local fwwrapon fwwrapon(`fwwrapon')
    }
    _____

> _____ end _get_diopts.Wra
> pon _____
    - if "`fwwrap'" != "" {
    = if "" != "" {
        FWWrap, `fwwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `baselev
> els' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflege
> nd' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fv
> wrap' `fwwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
    = c_local DUPS
    - if `K' == 2 {
    = if 1 == 2 {
        c_local `c_opts' ``options'
    }
    - sreturn local coding `coding'
    = sreturn local coding
    - sreturn local versus `versus'
    = sreturn local versus
    - sreturn local compare `compare'

```

```

= sreturn local compare
- sreturn local level `level'
= sreturn local level 95

```

```

> ----- end _get_diopts.GetDio
> pts
- }
- else {
  GetDiopts DUPS options, `options'
  c_local `name2' ``options''
}
- local DUPS : list DUPS - DEFAULT
- gettoken DUPS : DUPS
- if ``DUPS'' != "" {
= if ``' != "" {
  if (strpos(`DUPS', "(")) {
    gettoken DUPS : DUPS, parse("(")
    local DUPS `DUPS'()'
  }
  di as err "duplicate {bf:`DUPS'} option not allowed"
  exit 198
}
- sreturn clear
- foreach sname of local snames {
- sreturn local `sname' = ``S_`sname''
= sreturn local level = `95'
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local k = `0'
- }

```

```

> ----- end _get_dio
> pts

```

			Robust				
		Coefficient	std. err.	z	P> z	[95% conf. interv	
piq_timing_max							
> all]							
> -----							
rule_g		.2016407	.1232634	1.64	0.102	-.0399511	.4432
> 326							
rule_itr		.0458037	.1160217	0.39	0.693	-.1815947	.2732
> 021							
estimated		.8078875	.094814	8.52	0.000	.6220555	.9937
> 195							

wlth#open							
0 1	-.2016537	.1950234	-1.03	0.301	-.5838926	.1805	
> 852							
1 0	-.116462	.0814518	-1.43	0.153	-.2761046	.0431	
> 805							
1 1	0	(omitted)					
cb_authors_ext	.5911701	.0913197	6.47	0.000	.4121867	.7701	
> 536							
ln_neq	-.1086846	.0727957	-1.49	0.135	-.2513616	.0339	
> 924							
ntwrth#bnkcrdit							
0 1	-.4937633	.2070435	-2.38	0.017	-.8995611	-.0879	
> 654							
1 0	0	(omitted)					
1 1	0	(omitted)					
ntwrth	-.5329849	.154731	-3.44	0.001	-.836252	-.2297	
> 178							
open	.3592447	.1773659	2.03	0.043	.011614	.7068	
> 754							
bnkcrdit	.6298037	.1876556	3.36	0.001	.2620055	.9976	
> 019							
vint_early	.8755441	.2189822	4.00	0.000	.4463469	1.304	
> 741							
vint_mid	.4043989	.1314693	3.08	0.002	.1467238	.6620	
> 741							
_cons	-.1490742	.2049272	-0.73	0.467	-.5507241	.2525	
> 757							

```

> —
- if "`markdown'" != "" {
= if "" != "" {
  set linesize `lsize'
}
- if "`eqselect'" == "" & "`eformnote'" != "noeformnote" {
= if "" == "" & "" != "noeformnote" {
- if "`eformdi'" != "" & ("`e(cmd)'" != "gnbreg" | "`e(prefix)'" == "svy")
> {
= if `""' != "" & ("glm" != "gnbreg" | "" == "svy") {
  local k_eform `r(k_eform)'
  if ("`k_eform'" == "") {
    local k_eform = 1
  }
}

```

```

    local k_eq `e(k_eq)'
    if ("`k_eq'"=="") {
        local k_eq = 1
    }
    if (`k_eform'<`k_eq' & `k_eform') {
        if ("`e(cmd2)'" == "stintreg" | "`e(cmd2)'" == "streg") & ("`e(cmd)'"
> " == "weibull" | "`e(cmd)'" == "gompertz") {
        }
        else {
            _eform_multeq_note efnote : `k_eform' ``tr_item'''
            di as txt "`efnote'"
        }
    }
    - if (``eformdi'''!=" & "`e(noconstant)'"=="0" & ``eform_cons_ti'''
> !=" & "`e(consonly)'"!="1") {
        = if (``'''!=" & "0"=="0" & ``'''!=" & "0"!="1") {
            if ("`e(cmd)'"=="asclogit" | "`e(cmd)'"=="cmclogit" | "`e(cmd)'"=="c
> mmixlogit" | "`e(cmd)'"=="cmxtmixlogit" ) {
                local asvar = ``e(ranvars)``e(fixvars)``e(indvars)''' != ""
                if ``e(casevars)''' != "" & `asvar' {
                    local note1 "Exponentiated coefficients represent odds ratios for "
                    local note2 "alternative-specific variables (first equation) and "
                    local note3 "{help j_cmrisk###_new:relative-risk ratios} "
                    local note4 "for case-specific variables."
                    di as txt "{p 0 6 2}Note: `note1'`note2'`note3'`note4'`p_end}"
                }
            }
            if "`eform_cons_ti'" == "Inc. rate" {
                local eform_cons_note "incidence rate"
            }
            else if "`eform_cons_ti'" == "Rel. risk" {
                local eform_cons_note "relative risk"
            }
            else if "`eform_cons_ti'" == "Health" {
                local eform_cons_note "health (probability of no disease)"
            }
            else {
                local eform_cons_note = strlower("`eform_cons_ti'")
            }
            local eform_cons_note "baseline `eform_cons_note'"
            if ("`e(cmd2)'"=="") {
                local cmd `e(cmd)'
            }
            else {
                local cmd `e(cmd2)'
            }

```



```

    }
    local f2 = udsustr("`cmd'",1,2)
    local is_re = ("`f2'"=="xt" | "`f2'"=="me")
    if `is_re' {
        local extranote " (conditional on zero random effects)"
    }
    di as txt "{p 0 6 2}Note: {bf:_cons} estimates " "`eform_cons_note'"
> extranote'" _c
    if ("`e(cmd)'"=="mlogit" | "`e(cmd)'"=="asclogit" | "`e(cmd)'"=="cmc
> logit" | "`e(cmd)'"=="cmmixlogit" | "`e(cmd)'"=="cmxtmixlogit" ) {
    di as txt " for each outcome" _c
    }
    di as txt ".{p_end}"
    }
- }
- return add
- if !`keepmc' {
= if !0 {
- return local mcmethod
- return local mctitle
- }
}

> _____ end _coef_ta
> ble _____
- if "`e(msg)'" != "" {
= if "" != "" {
    noi di as txt "(Standard errors scaled using `e(msg)').)"
}
- if "`e(msg2)'" != "" {
= if "" != "" {
    noi di as txt "`e(msg2)'."
}
- if "`e(adm_warn)'" != "" {
= if "" != "" {
    noi di "{help j_glmadmiss##|_new:Warning: Parameter estimates produce
> }" _c
    noi di "{help j_glmadmiss##|_new:inadmissible mean estimates in one or
> }"
    noi di "{help j_glmadmiss##|_new:          more observations.}"
    }
- if "`e(Nf)'" != "" {
= if "" != "" {
    if `e(Nf)' > 1 { local s "s" }
    noi di as txt "(" as res `e(Nf)' as txt " model`s' failed to converge)
> "
    }
}

```

```

- if `e(nbml)' {
= if 0 {
  di as txt "{p 0 6 0 79}Note: Negative binomial "
  di as txt "parameter estimated via ML and treated "
  di as txt "as fixed once estimated."
}
- _prefix_footnote

```

```

> _____ begin _prefix_footn
> ote _____
  - version 9
  - local is_st = "`e(cmd2)'" == "streg"
  = local is_st = "" == "streg"
  - if `is_st' {
  = if 0 {
    local tropt TR
  }
  - syntax [, deff `tropt' linesize(int 79) * ]
  = syntax [, deff linesize(int 79) * ]
  - if `is_st' & ``e(noeform)'" != "" & ``tr'" != "" {
  = if 0 & `""' != "" & `""' != "" {
    di as smcl as txt "{p 0 6 2}" "Note: Option {res:tr} ignored; " "not
> appropriate with option {res:strata()} or {res:ancillary()}." "{p_end}"
  }
  - local footnote ``e(footnote)'"
  = local footnote `""'
  - gettoken cmd args : footnote, parse(" ,")
  - capture which `cmd'
  = capture which
  - if !c(rc) {
    if `:length local options' {
      `footnote' , `options'
    }
    else {
      `footnote'
    }
  }
  - capture
  - if ``e(prefix)'" == "svy" {
  = if "" == "svy" {
    _svy_fpc_note "`deff'" `linesize'
    _svy_subpop_note, linesize(`linesize')
    _svy_singleton_note, linesize(`linesize')
  }
  - if inlist(`e(cmd)'"',"logit","logistic","probit") {
  = if inlist(`glm'"',"logit","logistic","probit") {

```

```

        CompleteLogit
    }
    - if inlist("`e(cmd)'", "mlogit", "ologit", "oprobit") {
    = if inlist("`glm'", "mlogit", "ologit", "oprobit") {
        CompleteOlogit
    }
    - if "`e(missing)'" == "missing" & inlist("`e(vce)'", "brr", "bootstra
> p", "jackknife") {
    = if "" == "missing" & inlist("robust", "brr", "bootstrap", "jackknife
> ") {
        if "`e(vce)'" == "brr" {
            local vcetype BRR
        }
        else local vcetype `e(vce)'
        if e(N_misreps) == 1 {
            local reps replicate
        }
        else {
            local reps replicates
        }
        if "`e(N_misreps)'" != "" {
            local the `e(N_misreps)'
        }
        else local the "some of the"
        di as txt "{p 0 6 0 `linesize'}" "Note: One or more parameters could
> not be estimated in " "`the' `vcetype' `reps'; " "standard-error estimates
> include only complete replications.{p_end}"
    }
    - if inlist("`e(opt)'", "ml", "moptimize") | !missing(e(converged))
> {
    = if inlist("`moptimize'", "ml", "moptimize") | !missing(e(converged))
> {
        - ml_footnote
    }
}
> ----- begin ml_footn
> ote -----
    - version 9
    - syntax
    - _check_e_rc
}
> ----- begin _check_e
> _rc -----
    - version 11
    - syntax [, ignorerc]
    - if e(rc) == 504 & inlist("`e(opt)'", "ml", "moptimize") {
    = if e(rc) == 504 & inlist("moptimize", "ml", "moptimize") {

```

```

        di as err "{p 0 0 2}variance matrix missing because `e(user)' fa
> iled" "to compute scores or computed scores with missing values{p_end}"
        exit 504
    }
    - if (`""ignorerc""=="" & !missing(e(rc)) & e(rc) != 0) error e(r
> c)
        = if (`""=="" & !missing(e(rc)) & e(rc) != 0) error e(rc)
        _____
> _____ end _check_e
> _rc _____
    - if ``"e(converged)"" == "0" {
    = if ``"1"" == "0" {
        di as txt "Warning: Convergence not achieved."
    }
    _____
> _____ end ml_footn
> ote _____
    - }
    - _check_e_rc
    _____
> _____ begin _check_e
> _rc _____
    - version 11
    - syntax [, ignorerc]
    - if e(rc) == 504 & inlist("`e(opt)"" , "ml", "moptimize") {
    = if e(rc) == 504 & inlist("moptimize", "ml", "moptimize") {
        di as err "{p 0 0 2}variance matrix missing because `e(user)' fail
> ed" "to compute scores or computed scores with missing values{p_end}"
        exit 504
    }
    - if (`""ignorerc""=="" & !missing(e(rc)) & e(rc) != 0) error e(rc)
    = if (`""=="" & !missing(e(rc)) & e(rc) != 0) error e(rc)
    _____
> _____ end _check_e
> _rc _____
    _____
> _____ end _prefix_footn
> ote _____
    _____
> _____ end glm.Disp
> lay _____
    - }
    - error `e(rc)'
    = error 0
    _____
> _____ end glm.gl

```

```

> m_7
- if _rc {
  if "$SGLM_running" ~= "" & "$SGLM_nonstan" ~= "" {
    di as err "data not suitable for nonstandard family-link combination"
    mac drop SGLM_*
    exit 459
  }
  else {
    mac drop SGLM_*
    exit _rc
  }
}
- version 10: ereturn local cmdline `"glm `0'"
= version 10: ereturn local cmdline `"glm piq_timing_max rule_g rule_itr est
> imated i.wlth#i.open cb_authors_ext ln_neq i.ntwrth#i.bnkcrdit ntwrth op
> en bnkcrdit vint_early vint_mid if piq_timing_max < 99 , family(nbinomial) l
> ink(log) vce(robust) nolog"
- if "$SGLM_running" == "" & "$SGLM_nonstan" == "" {
= if "" == "" & "" == "" {
- mac drop SGLM_*
- }
}

> _____ end
> glm
- log close
  name: <unnamed>
  log: /Users/connorbrennan/Library/CloudStorage/OneDrive-TheUniversityo
> fChicago/mmb/output/stepwise_regressions/piq_timing_max_all.smcl
  log type: smcl
  closed on: 12 Sep 2025, 19:25:33

```
