
```

    name: <unnamed>
    log: /Users/connorbrennan/Library/CloudStorage/OneDrive-TheUniversityof
> fChicago/mmb/output/stepwise_regressions/y_timing_max_realrig_all.smcl
    log type: smcl
    opened on: 12 Sep 2025, 19:25:24
- capture ! rm ./output/stepwise_regressions/`depvar'`type'.pdf
= capture ! rm ./output/stepwise_regressions/y_timing_max_realrig_all.pdf
- di "*****"
> *****"
*****
> **
- di "Outcomes of bi-directional stepwise regressions with `depvar' (type = `t
> ype'))"
= di "Outcomes of bi-directional stepwise regressions with y_timing_max (type
> = realrig_all)"
Outcomes of bi-directional stepwise regressions with y_timing_max (type = real
> rig_all)
- di "*****"
> *****"
*****
> **
- ${reg_`depvar'`type'}
= poisson y_timing_max rule_g rule_itr estimated i.ntwrth#i.learning i.bnkcrd
> it#i.learning i.wlth#i.open learning if y_timing_max < 99 , vce(robust) nol
> og

> ----- begin pois
> son -----
- if _by() {
    local BY ``by `byvars'`byrc0':"
  }
- `BY' _vce_parserun poisson, mark(Exposure OFFset CLuster) : `0'
= _vce_parserun poisson, mark(Exposure OFFset CLuster) : y_timing_max rule_
> g rule_itr estimated i.ntwrth#i.learning i.bnkcrdit#i.learning i.wlth#i.op
> en learning if y_timing_max < 99 , 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 poisson, mark(Exposure OFFset CLuster) : y_timing_max ru
> le_g rule_itr estimated i.ntwrth#i.learning i.bnkcrdit#i.learning i.wlth#i
> .open learning if y_timing_max < 99 , 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 ``poisson, mark(Exposure OFFset CLuster) "" == ":" {
  sreturn local after ``"after"
  exit
}
- while ``"COLON" != ":" & ``"after" != "" {
= while ``" != ":" & ``: y_timing_max rule_g rule_itr estimated i.ntwr
> th#i.learning i.bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max
> < 99 , 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 ``:" != ":" & `` y_timing_max rule_g rule_itr estimated i.ntwr
> th#i.learning i.bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max
> < 99 , 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 `poisson, mark(Exposure OFFset CLuster) ""
- sreturn local after `""after""
= sreturn local after ` y_timing_max rule_g rule_itr estimated i.ntwrt
> h#i.learning i.bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max <
> 99 , vce(robust) nolog""

_____ end _on_colon_pa
> rse _____
- local ZERO `""s(before)""
= local ZERO `poisson, mark(Exposure OFFset CLuster) ""
- local 0 `""s(after)""
= local 0 ` y_timing_max rule_g rule_itr estimated i.ntwrth#i.learning i
> .bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max < 99 , vce(robu
> st) 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 ` y_timing_max rule_g rule_itr estimated i.ntwrth#i.learn
> i.bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max < 99 , vce(
> robust) nolog""
- local 0 `""ZERO""
= local 0 `poisson, mark(Exposure OFFset CLuster) ""
- syntax name(name=callerid="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 `` y_timing_max rule_g rule_itr estimated i.ntwrth#i.learning i
> .bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max < 99 , vce(robu
> st) 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 ``'' != "" & "" == "" {
  di as err "option vce() specified multiple times"
  exit 198
}
- if ``robust'' != "" & ``robustok'' == "" {
= if "" != "" & "" == "" {
  di as err "options vce() and robust may not be combined"
  exit 198
}
- if ``if'' != "" {
= if ``y_timing_max < 99 '' != "" {
- local if ``if (`if')''
= local if ``if (y_timing_max < 99 )''
- }
- if ``robust'' != "" {
= if "" != "" {
  local options ``robust' `options''
}
- if ``weight'' != "" {
= if "" != "" {
  local wgt [`weight'`exp']
}
- if ``options'' != "" {
= if ``nolog'' != "" {
- local opts ``:list retok options''
= local opts ``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 ``bstrap'' != "" {
= 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 `''!='' & ''!='' {
  gettoken uvceltype vcelopts : vce1, parse(" ,")
  capture unabVCETtype, `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 `"poisson `0'''
    exit
  }
- version 6, missing
- local version : di "version " string(_caller()) ":"
- if replay() {
    if "`e(cmd)'" != "poisson" {
      error 301
    }
    if _by() { error(190) }
    Display `0'
    error `e(rc)'
    exit
  }
- `version' `BY' Estimate `0'

```

```

= version 18.5: Estimate y_timing_max rule_g rule_itr estimated i.ntwrth#i
> .learning i.bnkcrdit#i.learning i.wlth#i.open learning if y_timing_max < 99
> , vce(robust) nolog

```

```

> ----- begin poisson.Estim
> ate -----
- local vv "version 8.1:"
- local mm d2
- local eval poiss_lf
- version 6, missing
- local awopt = cond(_caller()<7, "aw", "")
- syntax varlist(numeric fv ts) [fw `awopt' pw iw] [if] [in] [, IRr FROM(s
> tring) Level(cilevel) OFFset(varname numeric ts) Exposure(varname numeric ts
> ) noCONstant Robust CLuster(varname) SCore(string) noDISPLAY CRITTYPE(passth
> ru) MATAevaluator moptobj(passthru) * ]
= syntax varlist(numeric fv ts) [fw pw iw] [if] [in] [, IRr FROM(string)
> Level(cilevel) OFFset(varname numeric ts) Exposure(varname numeric ts) noCON
> stant Robust CLuster(varname) SCore(string) noDISPLAY CRITTYPE(passthru) MAT
> Aevaluator moptobj(passthru) * ]
- 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
- if "`mataevaluator'" != "" {
= if "" != "" {
local eval mopt__poisson_e2()
}
- local negh negh
- }
- if _by() {
_byoptnotallowed score ``score'""
}
- _get_diopts diopts options, `options'
= _get_diopts diopts options, vce(robust) nolog

```

```

> ----- 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, vce(robust) nolog

```

```

> ----- 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 & 17 {
  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' `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 `vce(robust) nolog'''
- }
- 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_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 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' `baselevels'
> `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coeflegend'
> `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `fvwrap
> ' `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, vce(robust) nolog


---


> 

---

 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' `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 & 17 {
    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 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 `"vce(robust) nolog"'
        - }
        - 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 `"vce(robust) nolog"'
    - }
    - 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 fvops = ``true''
- }

> _____ end _get_dio
> pts _____
- mlopts mlopts, `options'
= mlopts mlopts, vce(robust) nolog
_____
> _____ 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                                nolog
- 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 "vce(robust)" != "" {
- ml_technique, `technique' `vce' nodefault
= ml_technique, vce(robust) nodefault
}

> _____ begin ml_techni
> que _____
- version 8.0
- syntax [, TECHnique(string) VCE(passthru) noDEFault ]
- sreturn clear
- _vce_parse, argopt(CLuster) opt(Hessian OIM Opg NATIVE Robust) :, `v
> ce'
= _vce_parse, argopt(CLuster) opt(Hessian OIM Opg NATIVE Robust) :, vc
> e(robust)
}

> _____ begin _vce_pa

```

```

> rse -----
    - version 10
    - _on_colon_parse `0'
    = _on_colon_parse , argopt(CLuster) opt(Hessian OIM Opg NATIVE Robus
> t) :, vce(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 `", argopt(CLuster) opt(Hessian OIM Opg NATIVE Robust) "' ==
> ":" {
    sreturn local after ``"after"'
    exit
    }
    - while ``"COLON"' != ":" & ``"after"' != "" {
= while ``"' != ":" & `", vce(robust)'" != "" {
    - 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)'" != "" {
    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 `", argopt(CLuster) opt(Hessian OIM Opg NAT
> IVE Robust) ""
    - sreturn local after ``after``
= sreturn local after `", vce(robust)""
    }
}

> _____ end _on_colon_pa
> rse _____
    - local 0 ``s(before)``
= local 0 `", argopt(CLuster) opt(Hessian OIM Opg NATIVE Robust) ""
    - local user ``s(after)``
= local user `", vce(robust)""
    - syntax [varname(default=none)], [ OPTlist(namelist) ARGOPTlist(nam
> elist) PWallowed(namelist) OLD_is_ok NOARGOPT_is_ok ]
    - local touse `varlist'
= local touse
    - tempname o a
    - `.o' = ._optlist.new
= .__000000 = ._optlist.new
    -class {
    -instance:
    -      array options = {}
    -      array names = {}
    -}
    - local optlist : list uniq optlist
    - foreach name of local optlist {
    -   `.o'.adopt `name'
= .__000000.adopt Hessian
    }
}

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

```

```

- .options.Arrpush "`optspec'"
= .options.Arrpush "Hessian"
- .names.Arrpush "`mname'"
= .names.Arrpush "hessian"

```

```

> _____ end _optlist.add
> opt _____
- }
- .`o'.addopt `name'
= .__000000.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'
= .__000000.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'
    = .__000000.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 _____
    - }
    - .`o'.addopt `name'
    = .__000000.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
= .__000001 = ._optlist.new
- local argoptlist : list uniq argoptlist
- foreach name of local argoptlist {
- .`a'.addopt `name'
= .__000001.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 = 1/5 {
- local namelist `namelist' `.names[`i']'
= local namelist hessian
- }
- local namelist `namelist' `.names[`i']'
= local namelist hessian oim
- }
- local namelist `namelist' `.names[`i']'
= local namelist hessian oim opg
- }
- local namelist `namelist' `.names[`i']'
= local namelist hessian oim opg native
- }
- local namelist `namelist' `.names[`i']'
= local namelist hessian oim opg native robust

```

```

- }
- local namelist : list uniq namelist
- sreturn local namelist ``namelist'''
= sreturn local namelist "hessian oim opg native robust"
- class exit ``namelist'''
= class exit "hessian oim opg native robust"

> _____ end _optlist.dumpna
> mes _____
= local OPTNAMES "hessian oim opg native 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 "" != "" {
local ROBUST robust
if `:list ROBUST in OPTNAMES' {
local robust Robust

```

```

    }
    local CLUSTER cluster
    if `:list CLUSTER in ARGOPTNAMES' {
        local cluster CLuster(varname)
    }
}

- local 0 ``user'''
= local 0 `", vce(robust)'''
- syntax [fw aw pw iw] [, VCE(string asis) `robust' `cluster' ]
= syntax [fw aw pw iw] [, VCE(string asis)    ]
- if ``vce''' != "" {
= if `"robust"' != "" {
- gettoken name args : vce, parse(" ,")
- local args : list retok args
- capture ParseVCE `o' `a' , `name'
= capture ParseVCE __000000 __000001 , 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 = 1/5 {
    - local optlist `optlist' `.options['i']'
    = local optlist Hessian
    - }
    - local optlist `optlist' `.options['i']'
    = local optlist Hessian OIM
    - }
    - local optlist `optlist' `.options['i']'
    = local optlist Hessian OIM Opg
    - }
    - local optlist `optlist' `.options['i']'
    = local optlist Hessian OIM Opg NATIVE
    - }
    - local optlist `optlist' `.options['i']'
    = local optlist Hessian OIM Opg NATIVE Robust
    - }
    - local optlist : list uniq optlist
    - sreturn local optlist ``optlist'''
    = sreturn local optlist `"Hessian OIM Opg NATIVE Robust"'

```

```

- class exit ``optlist'''
= class exit ``Hessian OIM Opg NATIVE Robust'''


---


> _____ end _optlist.dumpopti
> ons _____


---


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


---


> _____ end _optlist.dumpopti
> ons _____
= local OPTSPECS ``Hessian OIM Opg NATIVE Robust CLuster'''
- local OPTNAMES ```. `0'.dumpnames' `.`A'.dumpnames'''


---


> _____ begin _optlist.dumpna
> mes _____
- version 9
- forval i = 1/`.names.arnels' {
= forval i = 1/5 {
- local namelist `namelist' `.names['i']'
= local namelist hessian
- }
- local namelist `namelist' `.names['i']'
= local namelist hessian oim
- }
- local namelist `namelist' `.names['i']'
= local namelist hessian oim opg
- }
- local namelist `namelist' `.names['i']'
= local namelist hessian oim opg native
- }
- local namelist `namelist' `.names['i']'
= local namelist hessian oim opg native robust
- }
- local namelist : list uniq namelist

```

```

- sreturn local namelist ``namelist'''
= sreturn local namelist "hessian oim opg native robust"
- class exit ``namelist'''
= class exit "hessian oim opg native robust"

> _____ end _optlist.dumpna
> mes _____
> _____ 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 OPTNAMES "hessian oim opg native robust cluster"
- syntax [, `OPTSPECS']
= syntax [, Hessian OIM Opg NATIVE Robust CLuster]
- 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 `'''' != "" {
    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", "
> cluster") {
= 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 _____
    - sreturn local vce ``r(vce)''
    = sreturn local vce `"robust"'
    - sreturn local vceargs ``r(vceargs)''
    = sreturn local vceargs `""'
    - sreturn local vceopt ``r(vceopt)''
    = sreturn local vceopt `"vce(robust)'"
    - if "`s(vce)'" == "hessian" {
    = if "robust" == "hessian" {
        sreturn local vce "oim"
        sreturn local vceopt "vce(oim)"
    }
    - if "$ML_vce" == "robust" & !inlist("`s(vce)'", "", "oim", "robust")
> {
    = if "" == "robust" & !inlist("robust", "", "oim", "robust") {
        di as err "options robust and vce(`s(vce)') may not be combined"
        exit 198
    }
    - get_technique `technique', `default'
    = get_technique , nodefault
}

> _____ begin ml_technique.get_techni
> que _____
    - syntax [anything(equalok)] [, noDEFault ]
    - local 0 ``anything''
    = local 0 `""'

```

```

- if "`default'" == "" {
= if "nodefault" == "" {
    local default_num 5
  }
- else local default_num 0
- gettoken tname rest : 0, parse(" =")
- while "`tname'" != "" {
= while "" != "" {
    local tname = lower("`tname'")
    capture confirm name `tname'
    if _rc {
      local rc = _rc
      di as err "`tname' is an invalid name in technique() option"
      exit `rc'
    }
    get_tech tech , `tname'
    gettoken num : rest, parse(" =")
    if "`num'" == "" {
      gettoken num rest : rest, parse(" =")
      gettoken num : rest, parse(" =")
    }
    capture confirm integer number `num'
    if !_rc gettoken num rest : rest, parse(" =")
    else local num `default_num'
    if `num' < 0 {
      di as err "negative integers are not allowed in technique() option"
    }
    exit 125
  }
  else if `num' > 0 {
    local techlist `techlist' `tech'
    local numlist `numlist' `num'
    local technique `technique' `tech' `num'
  }
  else if "`default'" != "" {
    local techlist `techlist' `tech'
    local numlist `numlist' `num'
    local technique `technique' `tech'
  }
  gettoken tname rest : rest, parse(" =")
}
- if "`default'`techlist'" == "" {
= if "nodefault" == "" {
    local techlist nr
    local numlist `default_num'
  }

```

```

- if "`default'" == "" {
= if "nodefault" == "" {
    sreturn local techlist `:list retok techlist'
    sreturn local numlist `:list retok numlist'
  }
- sreturn local technique `:list retok technique'
= sreturn local technique

> _____ end ml_technique.get_techni
> que _____

> _____ end ml_techni
> que _____
- if ``s(technique)'' != "" {
= if `'' != "" {
    local technique "technique(`s(technique)')"
  }
- if ``s(vce)'' != "" {
= if "robust" != "" {
- local vce ``s(vceopt)''
= local vce "vce(robust)"
- }
- sreturn local technique ``technique''
= sreturn local technique ""
- sreturn local vce ``vce''
= sreturn local vce "vce(robust)"
- }
- 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 nolog          vce(robust)
- if ``rest'' != "" {
= if "" != "" {

```

```

        c_local `rest' ``options'''
    }
}

> ----- end mlo
> pts
- local coll `s(collinear)'
= local coll
- local cns `s(constraints)'
= local cns
- local mlopts `mlopts' `crittype'
= local mlopts nolog          vce(robust)
- if ``score'''!="" {
= if `'''!="" {
    confirm new variable `score'
    local nword : word count `score'
    if `nword' > 1 {
        di in red "score() must contain the name of only " "one new variable"
        exit 198
    }
    tempvar scvar
    local scopt score(`scvar')
}
- if ``offset'''!="" & ``exposur'''!="" {
= if `'''!="" & `'''!="" {
    di in red "only one of offset() or exposure() can be specified"
    exit 198
}
- if ``constan'''!="" {
= if `'''!="" {
    local nvar : word count `varlist'
    if `nvar' == 1 {
        di in red "independent variables required with " "noconstant option"
        exit 100
    }
}
- marksample touse
- if ``cluster'''!="" {
= if `'''!="" {
    markout `touse' `cluster', strok
    local clopt cluster(`cluster')
}
- if ``exposur'''!="" {
= if `'''!="" {
    capture assert `exposur' > 0 if `touse'
    if _rc {
        di in red "exposure() must be greater than zero"
    }
}
}

```

```

    exit 459
  }
  tempvar offset
  qui gen double `offset' = ln(`exposur')
  local offvar "ln(`exposur')"
```

```

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

> _____ end tsu
> nab _____
  - local yname : subinstr local y "." "_"
  - if "`weight'"!="" {
= if ""!="" {
    if "`weight'"=="fweight" {
    local wt `[fw`exp']"'
  }
  else if "`weight'"=="iweight" {
    local wt `[iw`exp']"'

```

```

    }
    else local wt `[aw`exp']"
  }
- summarize `y' `wt' if `touse', meanonly
= summarize y_timing_max if __000000, meanonly
- if r(N) == 0 { error 2000 }
- if r(N) == 1 { error 2001 }
- if r(min) < 0 {
  di in red "`y' must be greater than or equal to zero"
  exit 459
}
- if r(min) == r(max) & r(min) == 0 {
  di in red "`y' is zero for all observations"
  exit 498
}
- tempname mean nobs
- scalar `mean' = r(mean)
= scalar __000001 = r(mean)
- scalar `nobs' = r(N)
= scalar __000002 = r(N)
- if "`display'"==" " {
= if ""==" " {
- capture assert `y' == int(`y') if `touse'
= capture assert y_timing_max == int(y_timing_max) if __000000
- if _rc {
  __New_Note `0'
}
- }
- if "`display'"==" " & "`weight'"=="aweight" {
= if ""==" " & ""=="aweight" {
  di in blu "note: you are responsible for interpretation " "of analytic w
> eights."
}
- if "`constan`cns'"==" " & "`cluster'"==" " & "`weight'"!="pweight" {
= if ""==" " & ""==" " & ""!="pweight" {
- tempname c
- if "`offset'"!=" " {
= if ""!=" " {
  SolveC `y' `offset' [`weight`exp'] if `touse', n(`nobs') mean(`mean')
  scalar `c' = r(_cons)
  if `c'>=. {
    di in blu "note: exposure = exp(`offvar') overflows;" _n "      could no
> t estimate constant-only model" _c
    if "`exposur'"==" " {
      di in blu _n "      use exposure() option for exposure = `offvar'."
    }
  }
}

```



```

        else di in blu "."
        local cfail 1
    }
    local c "`c'+`offset'"
}
- else scalar `c' = ln(`mean')
= else scalar __000003 = ln(__000001)
- if "`cfail'"==" {
= if ""==" {
- LikePois `y' `c' [weight`exp'] if `touse', n(`nobs')
= LikePois y_timing_max __000003 [] if __000000, n(__000002)

```

```

> ----- begin poisson.LikeP
> ois
- gettoken y 0 : 0
- gettoken xb 0 : 0
- syntax [fw aw pw iw] [if] , Nobs(string)
- if "`weight'"!=" {
= if ""!=" {
    if "`weight'"=="fweight" {
        local wt `[fw`exp']"
    }
    else if "`weight'"=="iweight" {
        local wt `[iw`exp']"
    }
    else local wt `[aw`exp']"
}
- tempvar lnf
- qui gen double `lnf' = `y'*(`xb')-exp(`xb')-lngamma(`y'+1) `if'
= qui gen double __000004 = y_timing_max*(__000003)-exp(__000003)-lngamm
> a(y_timing_max+1) if __000000
- summarize `lnf' `wt' `if', meanonly
= summarize __000004 if __000000, meanonly
- if r(N) != `nobs' { exit }
= if r(N) != __000002 { exit }
- if "`weight'"=="aweight" {
= if ""=="aweight" {
    ret scalar lnf = r(N)*r(mean)
}
- else ret scalar lnf = r(sum)

```

```

> ----- end poisson.LikeP
> ois
- if r(lnf)<. {
- local ll0 "lf0(1 `r(lnf)')"
```

```

= local ll0 "lf0(1 -420.2971615663282)"

```

```

- }
- else {
  di in blu "note: could not compute " "constant-only model log likelihood
> ."
  }
- }
- }
- if `fvops' {
= if 1 {
- local rmcoll "version 11: _rmcoll"
- }
- else local rmcoll _rmcoll
- if "`display'"!="" & "`weight'"=="aweight" {
= if ""!="" & ""=="aweight" {
  `rmcoll' `xvars' [iw`exp'] if `touse', `constan' `coll'
}
- else `rmcoll' `xvars' [weight`exp'] if `touse', `constan' `coll'
= else version 11: _rmcoll rule_g rule_itr estimated i.ntwrth#i.learning
> i.bnkcrdit#i.learning i.wlth#i.open learning [] if __000000,
note: 1.ntwrth#1.learning omitted because of collinearity.
note: 0.bnkcrdit#1.learning omitted because of collinearity.
note: 1.bnkcrdit#1.learning omitted because of collinearity.
- local xvars `r(varlist)'
= local xvars rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.
> learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning
> 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wl
> th#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning
- if "`from'"==" " {
= if ""==" " {
- `vv' Ipois `y' `xvars' [weight`exp'] if `touse', n(`nobs') mean(`mean'
> ) `constan' `offopt'
= version 18.5, missing: Ipois y_timing_max rule_g rule_itr estimated 0b.n
> twrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.lea
> rning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning
> 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlt
> h#1.open learning [] if __000000, n(__000002) mean(__000001)

```

```

> ----- begin poisson.Ip
> ois -----
- local vv : display "version " string(_caller()) ", missing:"
- version 6, missing
- syntax varlist(fv ts) [fw aw pw iw/] [if] , Nobs(string) Mean(string)
> [ noCONstant OFFset(string) ]
- gettoken y xvars : varlist
- tempvar xb z
- tempname b1 b2 lnf1

```

```

- if "`weight'"!="" {
= if ""!="" {
    local awt `[aw=`exp']"'
    local wt `"(`exp')*"'
    local exp `"=`exp'"
}
- quietly {
- if "`offset'"!="" {
= if ""!="" {
    tempvar ynew
    gen double `ynew' = `y' - `offset'*`mean' `if'
    local poff "+`offset'"
}
- else local ynew `y'
= else local ynew y_timing_max
- `vv' _regress `ynew' `xvars' `awt' `if', `constan'
= version 18.5, missing: _regress y_timing_max rule_g rule_itr estimate
> d 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#
> 1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.le
> arning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open
> 1.wlth#1.open learning if __000000,

> _____ 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) 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 ``level(95.00) fwrap(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 `", ""
        - 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 _____
    - 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 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')
}
- 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(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 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')
}
- 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' `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
> 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 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) PMATr
> ix(string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat
> (string) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATri
> x(string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name)
> cnszero(string) suffix(name) OFFSETlist(string asis) `beta' `standardized'
> NOCI NOPValues `dftable' `dfonly' dfci DFPValues dfmissing `groups' COEFLege
> nd selegend `fvignore' `flignore' `showginvariant' `nolabel' `nofootnote' `w
> rap' sort `pisematrix' depname(string) COEFTitle(string) coeftitle2(string)
> ptitle(string) cititle(string) NOMCLEGend noCNSReport FULLCNSReport `noclust
> report' cformat(passthru) sformat(passthru) pformat(passthru) NOFirst First
> SHOWEQns neq(integer -1) NODIPARM NOTEST SEParator(integer 0) NOSKIP OFFSETO

```

```

> NLY1 PLus NOEQCHECK EFORMALL CITYPE(string) eqselect(string) cidisjoint noEF
> ORMNOTE * ]
    = syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(string)
> ) DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) PMATr
> ix(string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat
> (string) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATri
> x(string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name)
> cnszero(string) suffix(name) OFFSETlist(string asis) Beta NOCI NOPValues
> dfci DFPValues dfmissing COEFLegend selegend sort depname(string) C
> OEFTitle(string) coeftitle2(string) ptitle(string) cititle(string) NOMCLEGen
> d noCNSReport FULLCNSReport cformat(passthru) sformat(passthru) pformat(pas
> sthru) NOFirst First SHOWEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(inte
> ger 0) NOSKIP OFFSETONLY1 PLus NOEQCHECK EFORMALL CITYPE(string) eqselect(st
> ring) 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'") tsm
> atrix("`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(s
> tring) NOCI NOPValues dfci DFPValues cidisjoint ]
    - if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`cimatr
> ix'"!=" & "`noc'"!=") {
    = if ("""=="" & ""=="" & "matrix"=="" & ""!=" & ""!=") {
        sreturn local cimatrix "empty"
    }
    - if ("`pmatrix'"=="" & "`tsmatrix'"!=" & "`e(V)'"=="" & "`cimatr

```

```

> ix'""=="" & ("`noci'""!= "" | "`nopvalues'""!= "")) {
    = if (""=="" & ""!= "" & "matrix"=="" & ""=="" & (""!= "" | ""!= "")) {
        sreturn local noci "empt"
        sreturn local nopvalues "empt"
    }
    - if ("`pmatrx'""=="" & "`tsmatrx'""=="" & "`e(V)'""=="" & "`cimatr
> ix'""!= "" & "`noci'""!= "") {
    = if (""=="" & ""=="" & "matrix"=="" & ""!= "" & ""!= "") {
        sreturn local cimatrix "empt"
    }
    - if ("`pmatrx'""!= "" & "`tsmatrx'""=="" & "`e(V)'""=="" & "`cimatr
> ix'""=="" & "`nopvalues'""!= "") {
    = if (""!= "" & ""=="" & "matrix"=="" & ""=="" & ""!= "") {
        sreturn local pmatrx "empt"
    }
    - if ("`pmatrx'""!= "" & "`tsmatrx'""=="" & "`e(V)'""=="" & "`cimatr
> ix'""=="" & "`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 ("`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:cimatix()}"
        exit 198
    }
    if ("`cidisjoint'"==" & "`e(disjoint)'"=="") {
        scalar `isvalid' = 1
        mata: _is_in_ci("`cimat'", "`isvalid'")
        if (`isvalid'==0) {
            display as error "invalid {bf:cimatix()}"
            di as txt "{p 4 4 2}"
            di as smcl as err "{bf:cimatix()}"
            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'"!=" & "`pmatix'"!="") {
= if ("matrix"==" & "'"!=" & "'"!="") {
    opts_exclusive "pmatix() `dfci'"

```

```

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

> _____ end _coef_table._novce_other_m
> ats _____
    - if ("`s(cimatix)'"=="empt") {
= if ("=="empt") {
    local cimatix ""
    }
    - if ("`s(pmatix)'"=="empt") {
= if ("=="empt") {
    local pmatix ""
    }
    - 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 po
> sted"
    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' `
> dfci' `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("pwme
> an", "regress", "")
- local is_marg = inlist("margins", "`cmd'", "`e(cmd2)'", "`caller
> '")
= local is_marg = inlist("margins", "regress", "", "")
- local OPTIONS NOADJUST BONferroni DUNCAN DUNNett SCHeffe SIDak S
> NK 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 SID
> ak 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 adjustal
> l 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 s
> vy prefix"'
            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 _____
    - 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, fwrap(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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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 fwwrapon' {
= if 0 {
  c_local fwwrapon fwwrapon(`fwwrapon')
}
}

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

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

> _____ end _get_diopts.FVW
> rap _____
- }
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fwwrap' `fwwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abbrev'
= c_local diopts fwwrap(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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 "`\lststretch' \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'

```

```

= Wraapon ,
> _____ 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, `fwwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle

```

```

> gend' `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 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, fwrap(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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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')
}
- 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 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')}"; " "nonne
> gative integer required{p_end}"
        exit 198
    }
}

> ----- end _get_diopts.FVW
> rap -----
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fwwrap' `fwwrapon' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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' `basele
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 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__) options
= _get_eformopts , eformopts() allowed(__all__) options

```

```

> _____ 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
> TRatio * ]
- local opts ``options'"
= local opts `""'
- local 0 , `eform1' `eform' `nohr' `hr' `noshr' `shr' `or' `irr'
> `rrr' `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 confirmname `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 T
> Ratio ]
    - 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 (""!="1" | `'''==``) {
- local eformdi ``s(str)'''
= local eformdi `'''
- }
- else {
  local eformdi ``eform_cons_ti'''
  local tr_item = cond(missing(``eform_cons_ti2'''), strlower(``eform
> m_cons_ti'''), ``eform_cons_ti2''')
  }
- local coefttl = cond(``eform'''==`'', ``coeftitle''', ``eformd
> i''')
= 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 options bottom plus S
> YNTAXonly * ]
- 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 as
> is) Level(cilevel) dof(passthru) syntaxonly ]
- if ``dof''' == "" & ``e(df_r)''' != "" {
= if `''' == "" & ``206''' != "" {
- capture confirm integer number `e(df_r)'
= capture confirm integer number 206
- if !c(rc) {

```

```

- local gdof dof(`e(df_r)')
= local gdof dof(206)
- }
- 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, `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, fwrap(1)

```

> _____ begin _get_diopts.GetDio

> pts _____

```

- local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASElev

```



```

> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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, fvwrap(integer)
- if `fvwrap' < 0 {
= if 1 < 0 {
    di as err "{p}invalid option {bf:fvwrap(`fvwrap')}"; " "nonne
> gative integer required{p_end}"
    exit 198
  }
}

_____  

> _____ end _get_diopts.FVW
> rap _____
- }
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 __000009 ``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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
      - syntax namelist(max=2) [, `DIOPTS' *]
      = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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' `basel

```

```

> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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)'"=="
> "cmmixlogit" | ``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)'"=="c
> mlogit" | "`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 _____
  - mat `b1' = (1/`mean')*e(b)
  = mat __000006 = (1/__000001)*e(b)
  - mat score double `xb' = `b1' `if'
  = mat score double __000004 = __000006 if __000000
  - LikePois `y' `xb'`poff' [`weight'`exp'] `if', n(`nobs')
  = LikePois y_timing_max __000004 [] if __000000, n(__000002)

_____ begin poisson.LikeP
> ois _____
  - gettoken y 0 : 0
  - gettoken xb 0 : 0
  - syntax [fw aw pw iw] [if] , Nobs(string)
  - if "`weight'"!="" {
  = if ""!="" {
    if "`weight'"=="fweight" {
      local wt `[fw`exp']"'
    }
    else if "`weight'"=="iweight" {
      local wt `[iw`exp']"'
    }
    else local wt `[aw`exp']"'
  }
  - tempvar lnf
  - qui gen double `lnf' = `y'*(`xb')-exp(`xb')-lngamma(`y'+1) `if'
  = qui gen double __000009 = y_timing_max*(__000004)-exp(__000004)-lnga
> mma(y_timing_max+1) if __000000

```



```

- summarize `lnf' `wt' `if', meanonly
= summarize __000009 if __000000, meanonly
- if r(N) != `nobs' { exit }
= if r(N) != __000002 { exit }
- if "`weight'"=="aweight" {
= if ""=="aweight" {
    ret scalar lnf = r(N)*r(mean)
  }
- else ret scalar lnf = r(sum)

```

```

> ----- end poisson.LikeP
> ois -----
- scalar `lnf1' = r(lnf)
= scalar __000008 = r(lnf)
- }
- if "`constan'"==" " {
= if ""==" " {
- SolveC `y' `xb' `poff' [`weight' `exp'] `if', n(`nobs') mean(`mean')
= SolveC y_timing_max __000004 [] if __000000, n(__000002) mean(__000001
> )

```

```

> ----- begin poisson.Sol
> veC -----
- gettoken y 0 : 0
- gettoken xb 0 : 0
- syntax [fw aw pw iw] [if] , Nobs(string) Mean(string)
- if "`weight'"=="pweight" {
= if ""=="pweight" {
    local weight "aweight"
  }
- capture confirm variable `xb'
= capture confirm variable __000004
- if _rc {
    tempvar xbnew
    qui gen double `xbnew' = (`xb') `if'
    local xb `xbnew'
  }
- summarize `xb' [`weight' `exp'] `if', meanonly
= summarize __000004 [] if __000000, meanonly
- if r(N) != `nobs' { exit }
= if r(N) != __000002 { exit }
- if r(max) - r(min) > 2*709 {
    exit
  }
- if r(max) > 709 | r(min) < -709 {
    tempname shift

```

```

        if r(max) > 709 { scalar `shift' = 709 - r(max) }
        else scalar `shift' = -709 - r(min)
        local shift "+`shift'"
    }
    - tempvar expoff
    - qui gen double `expoff' = exp(`xb'`shift') `if'
    = qui gen double __000009 = exp(__000004) if __000000
    - summarize `expoff' [`weight'`exp'], meanonly
    = summarize __000009 [], meanonly
    - if r(N) != `nobs' { exit }
    = if r(N) != __000002 { exit }
    - return scalar _cons = ln(`mean')-ln(r(mean))`shift'
    = return scalar _cons = ln(__000001)-ln(r(mean))

```

```

> _____ end poisson.Sol
> veC _____
    - tempname c
    - scalar `c' = r(_cons)
    = scalar __000009 = r(_cons)
    - if `c'<. {
    = if __000009<. {
    - local c "+`c'"
    = local c "+__000009"
    - }
    - else local c
    - LikePois `y' `xb'`poff'`c' [`weight'`exp'] `if', n(`nobs')
    = LikePois y_timing_max __000004+__000009 [] if __000000, n(__000002)

```

```

> _____ begin poisson.LikeP
> ois _____
    - gettoken y 0 : 0
    - gettoken xb 0 : 0
    - syntax [fw aw pw iw] [if] , Nobs(string)
    - if "`weight'"!="" {
    = if ""!="" {
        if "`weight'"=="fweight" {
            local wt `[fw`exp']"'
        }
        else if "`weight'"=="iweight" {
            local wt `[iw`exp']"'
        }
        else local wt `[aw`exp']"'
    }
    - tempvar lnf
    - qui gen double `lnf' = `y'*(`xb')-exp(`xb')-lngamma(`y'+1) `if'
    = qui gen double __00000A = y_timing_max*(__000004+__000009)-exp(__000

```

```

> 004+__000009)-lngamma(y_timing_max+1) if __000000
    - summarize `lnf' `wt' `if', meanonly
    = summarize __00000A if __000000, meanonly
    - if r(N) != `nobs' { exit }
    = if r(N) != __000002 { exit }
    - if "`weight'"=="aweight" {
    = if ""=="aweight" {
        ret scalar lnf = r(N)*r(mean)
    }
    - else ret scalar lnf = r(sum)



---


> ----- end poisson.LikeP
> ois -----
    - tempname lnf1c
    - scalar `lnf1c' = r(lnf)
    = scalar __00000A = r(lnf)
    - }
    - else local lnf1c .
    - capture {
    - gen double `z' = `y'*exp(-(`xb'`poff'`c'))-1 +`xb'`c' `if'
    = gen double __000005 = y_timing_max*exp(-(__000004+__000009))-1 +__0000
> 04+__000009 if __000000
    - `vv' _regress `z' `xvars' [aw=`wt'exp(`xb'`poff'`c'))] `if', `constan'
    = version 18.5, missing: _regress __000005 rule_g rule_itr estimated 0b
> .ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.l
> earning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learn
> ng 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.w
> lth#1.open learning [aw=exp(__000004+__000009)] if __000000,



---


> ----- 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("
> 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 ``level(95.00) fwrap(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 `", ''
  - 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 -----
    - 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 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 [, W0rd WIdth]
- if c(rc) {
    di as err "invalid fvwraon() 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)" != "" {
        - FVWrap, `fwwrap'
= FVWrap, fwwrap(1)


---


> 

---

 begin _get_diopts.FVW
> rap 

---


    - syntax, fwwrap(integer)
    - if `fwwrap' < 0 {
= if 1 < 0 {
        di as err "{p}invalid option {bf:fwwrap(`fwwrap')}"; " "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 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')
}
- 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 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) NOABBREV
> 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 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 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) PMATr
> ix(string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat
> (string) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATri
> x(string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name)

```

```

> cnszero(string) suffix(name) OFFSETlist(string asis) `beta' `standardized'
> NOCI NOPValues `dftable' `dfonly' dfci DFPValues dfmissing `groups' COEFLege
> nd selegend `fvignore' `flignore' `showginvariant' `nolabel' `nofootnote' `w
> rap' sort `pisematrix' depname(string) COEFTitle(string) coeftitle2(string)
> ptitle(string) cititle(string) NOMCLEGend noCNSReport FULLCNSReport `noclust
> report' cformat(passthru) sformat(passthru) pformat(passthru) NOFirst First
> SHOEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(integer 0) NOSKIP OFFSETO
> NLY1 PLus NOEQCHECK EFORMALL CITYPE(string) eqselect(string) cidisjoint noEF
> ORMNOTE * ]
      = syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(string
> ) DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) PMATr
> ix(string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFormat
> (string) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDMATri
> x(string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(name)
> cnszero(string) suffix(name) OFFSETlist(string asis) Beta NOCI NOPValues
> dfci DFPValues dfmissing COEFLegend selegend sort depname(string) C
> OEFTitle(string) coeftitle2(string) ptitle(string) cititle(string) NOMCLEGend
> d noCNSReport FULLCNSReport cformat(passthru) sformat(passthru) pformat(pas
> sthru) NOFirst First SHOEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(inte
> ger 0) NOSKIP OFFSETONLY1 PLus NOEQCHECK EFORMALL CITYPE(string) eqselect(st
> ring) cidisjoint noEFORMNOTE * ]
      - if "`wrap'" != "" {
      = if "" != "" {
        opts_exclusive `wrap(`wrap') `fvwrap'
        local fvwrap fvwrap(`wrap')
      }
      - if "`nolabel'" != "" {
      = 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'") tsm
> atrix("`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(s

```

```

> tring) NOCI NOPValues dfci DFPValues cidisjoint ]
    - if ("`pmatrx'"==" & "`tsmatrx'"==" & "`e(V)'"==" & "`cimatr
> ix'"!=" & "`noci'"!=") {
    = if ("==" & "==" & "matrix"==" & "!=" & "!=") {
        sreturn local cimatrix "empt"
    }
    - if ("`pmatrx'"==" & "`tsmatrx'"!=" & "`e(V)'"==" & "`cimatr
> ix'"==" & ("`noci'"!="|"`nopvalues'"!=")) {
    = if ("==" & "!=" & "matrix"==" & "==" & ("!="|!=")) {
        sreturn local noci "empt"
        sreturn local nopvalues "empt"
    }
    - if ("`pmatrx'"==" & "`tsmatrx'"==" & "`e(V)'"==" & "`cimatr
> ix'"!=" & "`noci'"!=") {
    = if ("==" & "==" & "matrix"==" & "!=" & "!=") {
        sreturn local cimatrix "empt"
    }
    - if ("`pmatrx'"!=" & "`tsmatrx'"==" & "`e(V)'"==" & "`cimatr
> ix'"==" & "`nopvalues'"!=") {
    = if ("!=" & "==" & "matrix"==" & "==" & "!=") {
        sreturn local pmatrx "empt"
    }
    - if ("`pmatrx'"!=" & "`tsmatrx'"==" & "`e(V)'"==" & "`cimatr
> ix'"==" & "`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 ("`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 ("`cimatrx'"!="") {
= if ("!=") {
tempname cimat isvalid
capture matrix `cimat' = `cimatrx'
local rc = _rc
if (`rc') {
display as error "invalid {bf:cimatrx()}"
exit 198
}
if ("`cidisjoint'"==" & "`e(disjoint)'"=="") {
scalar `isvalid' = 1
mata: _is_in_ci("`cimat'", "`isvalid'")
if (`isvalid'==0) {
display as error "invalid {bf:cimatrx()}"
di as txt "{p 4 4 2}"
di as smcl as err "{bf:cimatrx()}"
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'"!=" & "`cimatrx'"!=") {
= if ("matrix"==" & ""!=" & ""!=") {
  opts_exclusive "cimatrx() `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'"!=" & "`cimatrx'"!=") {
= if ("matrix"==" & ""!=" & ""!=") {
  opts_exclusive "cimatrx() `dfpvalues'"
}
}

```

```

> _____ end _coef_table._novce_other_m
> ats _____

```

```

- if ("`s(cimatrx)'"=="empty") {
= if ("=="empty") {
  local cimatrx ""
}
- if ("`s(pmatrx)'"=="empty") {
= if ("=="empty") {
  local pmatrx ""
}
- if ("`s(noci)'"=="empty") {
= if ("=="empty") {
  local noci ""
}
- if ("`s(nopvalues)'"=="empty") {
= if ("=="empty") {
  local nopvalues ""
}

```

```

    }
- if ("`e(V)'"!="" & "`cidisjoint'"!="") {
= if ("matrix"!=" & ""!="") {
    display in red "option {bf:cidisjoint} not allowed when e(V) is po
> sted"
    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 ("`noc`"!=" & "`norowci`"!=") {
= if ("!=" & "!=") {
    di as err "{bf:norowci} is not allowed with {bf:`noc`}"
    exit 198
    }
    - if "`standardized`" != "" & "`bstdmatrix`" != "" {
= if "" != "" & "" != "" {
        local beta beta
        local standardized
    }
    - if `:length local type' == 0 {
= if 0 == 0 {
        - local type `beta' `noc` `nopvalues' `dftable' `dfonly' `groups' `
> dfci' `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("pwme
> an", "regress", "")

```

```

- local is_marg = inlist("margins", "`cmd'", "`e(cmd2)'", "`caller
> '")
= local is_marg = inlist("margins", "regress", "", "")
- local OPTIONS NOADJUST BONferroni DUNCAn DUNNett SCHeffe SIDak S
> NK 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 SID
> ak 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 adjustal
> l 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 s
> vy prefix'"
    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 _____
    - 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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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')
}
- 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(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')}"; " "nonne
> gative integer required{p_end}"
        exit 198
    }
}
> ----- end _get_diopts.FVW
> rap -----
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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')
    }
- 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' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABel FVLABel NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABel FVLABel NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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, `fvwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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"
    - }
    - 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, fvwrap(1)
  _____
> _____ begin _get_diopts.GetDio
> pts _____
    - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASEleve
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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(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')}"; " "nonne
> gative integer required{p_end}"
        exit 198
    }

> _____ end _get_diopts.FVW
> rap _____
    - }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `

```

```

> fvwrap' `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 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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABel FVLABel NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo

```

```

> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 fvwraon() 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, `fvwrap'
}
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `fvwraupon' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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__) options
    = _get_eformopts , eformopts() allowed(__all__) options
}

> ----- 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
> TRatio * ]
    - local opts ``options'""
    = local opts `""
    - local 0 , `eform1' `eform' `nohr' `hr' `noshr' `shr' `or' `irr'
> `rrr' `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 T
> Ratio ]
- foreach ef of local efopts {
- local eform `eform' ``ef''
= local eform

```



```

- 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_iform)'" == "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
> m_cons_ti''), ``eform_cons_ti2'')
    }
- local coefttl = cond("`eform''==`'', ``coeftitle'', ``eformd
> i'')
= 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 S
> YNTAXonly * ]
    - 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 as
> is) Level(cilevel) dof(passthru) syntaxonly ]
- if ``dof''' == "" & ``e(df_r)''' != "" {
= if `''' == "" & `206''' != "" {
- capture confirm integer number `e(df_r)'
= capture confirm integer number 206
- if !c(rc) {
- local gdof dof(`e(df_r)')
= local gdof dof(206)
- }
- 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, `GTPTS'
gettoken diparm rest : rest, `GTPTS'
}
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, `GTPTS'
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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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')
    }
    - 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(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')}"; " "nonne
> gative integer required{p_end}"
exit 198
}

> _____ end _get_diopts.FVW
> rap _____
- }
- c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 __00000B ``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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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 ALLBASElev
> els NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYcell
> s NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string)
> sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVWRA
> PON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOABB
> REV ABBREV
      - syntax namelist(max=2) [, `DIOPTS' *]
      = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEleve
> ls ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYce
> lls EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cfo
> rmat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(pa
> ssthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(pa
> ssthru) 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 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' `basel
> evels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coefle
> gend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare' `
> fvwrap' `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)'"==
> "cmmixlogit" | "`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)'"=="c
> mlogit" | "`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 _____
- mat `b2' = e(b)
= mat __000007 = e(b)
- drop `xb'
= drop __000004
- _predict double `xb' `if'
= _predict double __000004 if __000000
- LikePois `y' `xb' `poff' [`weight' `exp'] `if', n(`nobs')
= LikePois y_timing_max __000004 [] if __000000, n(__000002)

> _____ begin poisson.LikeP
> ois _____
- gettoken y 0 : 0
- gettoken xb 0 : 0
- syntax [fw aw pw iw] [if] , Nobs(string)
- if "`weight'"!=" " {
= if ""!=" " {
    if "`weight'"=="fweight" {
        local wt "`[fw`exp']"
    }
}

```



```

        else if "`weight'"=="iweight" {
            local wt "`[iw`exp']"
        }
        else local wt "`[aw`exp']"
    }
    - tempvar lnf
    - qui gen double `lnf' = `y'*(`xb')-exp(`xb')-lngamma(`y'+1) `if'
    = qui gen double __00000B = y_timing_max*(__000004)-exp(__000004)-lnga
> mma(y_timing_max+1) if __000000
    - summarize `lnf' `wt' `if', meanonly
    = summarize __00000B if __000000, meanonly
    - if r(N) != `nobs' { exit }
    = if r(N) != __000002 { exit }
    - if "`weight'"=="aweight" {
    = if ""=="aweight" {
        ret scalar lnf = r(N)*r(mean)
    }
    - else ret scalar lnf = r(sum)
}

> _____ end poisson.LikeP
> ois _____
    - tempname lnf2
    - scalar `lnf2' = r(lnf)
    = scalar __00000B = r(lnf)
    - }
    - if _rc { local lnf2 . }
    - if `lnf1'>=. & `lnf1c'>=. & `lnf2'>=. { exit }
    = if __000008>=. & __00000A>=. & __00000B>=. { exit }
    - if `lnf2'<.&(`lnf2'>`lnf1'|`lnf1'>=.)&(`lnf2'>`lnf1c'|`lnf1c'>=.) {
    = if __00000B<.&(__00000B>__000008|__000008>=.)&(__00000B>__00000A|__000
> 00A>=.) {
    - ret matrix b0 `b2'
    = ret matrix b0 __000007
    - exit
}

> _____ end poisson.Ip
> ois _____
    - if "`r(b0)'"!="" {
    = if "matrix"!=" {
    - tempname from
    - mat `from' = r(b0)
    = mat __000004 = r(b0)
    - }
    - }
    - if "`from'"!="" {
    = if "__000004"!=" {

```

```

- local initopt `"init(`from') search(off)'"
= local initopt `"init(__000004) search(off)'"
- }
- else local initopt "search(on) maxfeas(50)"
- `vv' ml model `mm' `eval' (`yname': `y' = `xvars', `constan' `offopt') [
> `weight'`exp'] if `touse', collinear missing max nooutput nopreserve `mlopts
> ' title(Poisson regression) `scopt' `robust' `clopt' `initopt' `ll0' `negh'
> `moptobj'
= version 18.5, missing: ml model e2 poiss_lf (y_timing_max: y_timing_max
> = rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntw
> rth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o
> .learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.
> wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning, ) [] if __000000, collin
> ear missing max nooutput nopreserve nolog vce(robust) title(Poisson
> regression) init(__000004) search(off) lf0(1 -420.2971615663282) negh

> ----- begin
> ml -----
- if _caller()>=11 {
- mopt `0'
= mopt model e2 poiss_lf (y_timing_max: y_timing_max = rule_g rule_itr e
> stimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.
> ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdi
> t#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#
> 0b.open 1.wlth#1.open learning, ) [] if __000000, collinear missing max noo
> utput nopreserve nolog vce(robust) title(Poisson regression) ini
> t(__000004) search(off) lf0(1 -420.2971615663282) 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 poiss_lf (y_timing_max: y_timing
> _max = rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning
> 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrd
> it#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.ope
> n 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning, ) [] if __000000, c
> ollinear missing max nooutput nopreserve nolog vce(robust) title(Po
> isson regression) init(__000004) search(off) lf0(1 -420.2971615663282) ne
> gh

> _____ 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)) 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 `''' == "" & `206''' != "" {
    - capture confirm integer number `e(df_r)'
    = capture confirm integer number 206
    - if !c(rc) {
    - local gdof dof(`e(df_r)')
    = local gdof dof(206)
    - }
    - 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)''
- 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 ``soptions'' == '' {
  syntax [, diparm(string asis) ]
}
- if (``execute'' == '') {
= if (`` == '') {
- sreturn clear
- sreturn local options ```options''
= sreturn local options ``nolog search(off)''
- 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)''
- if `interactive' & `:length local options' {
= if 0 & 17 {
    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 313 {
  - local ++ieq
  - gettoken eq eqspec : eqspec, bind
  - if bsubstr(trim("`eq'"),1,1) == "/" {
    = if bsubstr(trim("(y_timing_max: y_timing_max = rule_g rule_itr est
> imated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.nt
> wrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#
> 0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b
> .open 1.wlth#1.open learning,  )"),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("`y_timing_max: y_timing_max = rule_g rule_itr est
> imated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.nt
> wrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#
> 0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b
> .open 1.wlth#1.open learning,  )"'')
    - if bsubstr("`p2'",1,1) == "," {
      = if bsubstr("y_timing_max: y_timing_max = rule_g rule_itr estimated
> 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1
> o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.lea
> rning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open
> 1.wlth#1.open learning,  )",1,1) == "," {
        gettoken comma p2 : p2, parse(",")
        local p2 = trim("`"p2"'')
      }
    - if "`"p1'"p2'" == "()" {
      = if `(y_timing_max: y_timing_max = rule_g rule_itr estimated 0b.nt
> wrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.lear
> ning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning
> 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth
> #1.open learning,  )"' == "()" {

```

```

        local spec : subinstr local spec "`eq'" "(eq`ieq':)", word
    }
- }
- }
- while `:length local eqspec' {
= while 3 {
- 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 __000005 if __000000 [],
- tempname eqlist
- `eqlist' = _eqlist.new, eqopts(NOCONSTANT FREEParam) eqargopts(OF
> Fset EXPosure) noneedvarlist numdepvars(0) needequal
= __000006 = _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'
= .__000006.parse (y_timing_max: y_timing_max = rule_g rule_itr est

```



```

> imated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.nt
> wrth#1o.learning 0b.bnkcrcdit#0b.learning 0b.bnkcrcdit#1o.learning 1.bnkcrcdit#
> 0b.learning 1o.bnkcrcdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b
> .open 1.wlth#1.open learning, ) []

> ----- 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 SMARKOPTLIST `'''
    - 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 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 ``(y_timing_max: y_timing_max = rule_g rule_itr estimat
> ed 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth
> #1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.l
> earning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.ope
> n 1.wlth#1.open learning, ) []'', parse(",()[]")
        - local rebind = ``1'' != "(" & ``2'' == "("
        = local rebind = ``('' != "(" & ``y_timing_max: y_timing_max = rul
> e_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0
> b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.lear
> ning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#
> 1.open 1.wlth#0b.open 1.wlth#1.open learning'' == "("
    - }
    - 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.arnels' {
    = 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(60000026e500):y_timing_max: y_timing_max
> = rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.nt
> wrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1
> o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b
> .wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning

> _____ begin _eqspec.pa
> rse _____
    - version 9.2
    - _on_colon_parse `0'
    = _on_colon_parse __key(60000026e500):y_timing_max: y_timing_max
> = rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.nt
> wrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1
> o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b
> .wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning

> _____ 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(60000026e500)` == ":" {
    sreturn local after ``after``
    exit
}
- while ``COLON`` != ":" & ``after`` != "" {
= while `` != ":" & ``y_timing_max: y_timing_max = rule_g r
> ule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.lea
> rning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning
> 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.ope
> n 1.wlth#0b.open 1.wlth#1.open learning`` != "" {
    - 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 ``:`` != ":" & ``y_timing_max: y_timing_max = rule_g r
> ule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.lea
> rning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning
> 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.ope
> n 1.wlth#0b.open 1.wlth#1.open learning`` != "" {
    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(60000026e500)''
- sreturn local after ```after'''
= sreturn local after ``y_timing_max: y_timing_max = rule_g ru
> le_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.lear
> ning 1o.ntwrth#1o.learning 0b.bnkcrcdit#0b.learning 0b.bnkcrcdit#1o.learning 1
> .bnkcrcdit#0b.learning 1o.bnkcrcdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open
> 1.wlth#0b.open 1.wlth#1.open learning''

> _____ end _on_colon_pa
> rse _____
- local ol `s(before)'
= local ol __key(60000026e500)
- 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 FREEParm
- }
- local optlist `optlist' `.options['i']'
= local optlist NOCONStant FREEParm OFFset(passthru)
- }
- local optlist `optlist' `.options['i']'
= local optlist NOCONStant FREEParm OFFset(passthru) EXPosure(
> passthru)
- }
- local optlist : list uniq optlist
- sreturn local optlist ```optlist'''
= sreturn local optlist ``NOCONStant FREEParm 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 ``y_timing_max: y_timing_max = rule_g rule_itr estimat
> ed 0b.ntwrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth
> #1o.learning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.l
> earning 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.ope
> n 1.wlth#1.open learning''
- 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("y_timing_max") == ":" {
    di as err "nothing found where equation name expected"
    exit 198
}
- if `:list sizeof REST' {
= if 19 {
- if strpos("`NAME'", ":") {
= if strpos("y_timing_max", ":") {
    capture _msparse `NAME'
    if c(rc) {
        di as err "{bf:`NAME'} invalid equation specification"
        exit 198
    }
}
- else {
- . _confirm_eqname `NAME'
= . _confirm_eqname y_timing_max
}

> _____ begin _eqspec._confirm_eqn
> ame _____
- version 9.2
- capture confirm number `0'
= capture confirm number y_timing_max
- if (! c(rc)) exit
- local hasdot = strpos("`0'", ".")
= local hasdot = strpos("y_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 "y_timing_max " == "=" {
  local REST : copy local anything
  local DEPVARs
}
- if (:length local DEPVARs' == 0 & `.n_depvars') {
= if (13 == 0 & 0) {
  di as err "nothing found where depvar expected"
  exit 198
}
- if ``"REST"' == "" {
= if "=" rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwr
> th#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.lear
> ning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning
> 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning"' == ""
> {
  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
    }
    loca 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 y_timing_max
- tsunab DEPVARs : `DEPVARs'
= tsunab DEPVARs : y_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 y_timing_max
        - local n : word count `varlist'
        = local n : word count y_timing_max
        - if `n'>=`min' & `n'<=`max' { exit }
        = if 1>=1 & 1<=32000 { exit }

> _____ end tsu
> nab _____
        - }
        - if `:list sizeof INDEPVARs' {
        = if 16 {
        - fvunab INDEPVARs : `INDEPVARs'
        = fvunab INDEPVARs : rule_g rule_itr estimated 0b.ntwrth#0b.lea
> rning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnk
> crdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit
> #1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open lea
> rning

> _____ begin fvu
> nab _____
        - version 11

```

```

- _on_colon_parse `0'
= _on_colon_parse INDEPVARs : rule_g rule_itr estimated 0b.nt
> wrth#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.lear
> ning 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning
> 1o.bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth
> #1.open learning

> ----- 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.ntwr
> th#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learni
> ng 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o
> .bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1
> .open learning"" != "" {
- 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.ntwr
> th#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learni
> ng 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o

```

```

> .bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1
> .open learning"" != "" {
    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.ntwrth
> h#0b.learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learnin
> g 0b.bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.
> bnkcrdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.
> open learning""
_____
> _____ 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.ntwrth#0b.learning
> 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#
> 0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.le
> arning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning"
> '
    - 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.ntwrth#0b.lea
> rning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnk
> crdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit
> #1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open lea
> rning

```

```

- local n : list sizeof varlist
- if (`min' <= `n' & `n' <= `max') exit
= if (1 <= 16 & 16 <= 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 = `y_timing_max'
- .depvars = ``DEPVARs''
= .depvars = `y_timing_max'
- .hasequal = `hasequal'
= .hasequal = 1
- .indepvars = ``:list retok INDEPVARs''
= .indepvars = `rule_g rule_itr estimated 0b.ntwrth#0b.learning
> 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit
> #0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.l
> earning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning
> ''

- .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 __000005
    - 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 "__000005" != "__000005" {
        `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 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 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.ntwrth#0b.learnin
> g 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdi
> t#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.
> learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learnin
> g```
      - sreturn local value ```.`value'```
      = sreturn local value ``rule_g rule_itr estimated 0b.ntwrth#0b
> .learning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b
> .bnkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcr
> rdit#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open
> learning```
      - class exit ```.`value'```
      = class exit ``rule_g rule_itr estimated 0b.ntwrth#0b.learning
> 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit
> #0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.l
> earning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning
> ```


---


> _____ end _eqspec.i
> tem _____
      - class exit ```.`s(value)'```
      = class exit ``rule_g rule_itr estimated 0b.ntwrth#0b.learning 0
> b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0
> b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.lea
> rning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning```


---


> _____ end _eqlist
> .eq _____
      = local xvarsi ``rule_g rule_itr estimated 0b.ntwrth#0b.learning 0
> b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0
> b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.lea
> rning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning```
      - 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 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 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 xvars'
        = local dim = + 16
        - if ``noconsi' == "0" {
        = if "0" == "0" {
        - local ++dim
        - }
        - }
        - sreturn local value ```dim'''
        = sreturn local value ``17''
        - class exit ```s(value)'''
        = class exit ``17''
    _____
> _____ end _eqlist.
> dim _____
        = global ML_dim = 17
        - 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 `y_timing_max'
      - sreturn local value ```.`value'```
      = sreturn local value `y_timing_max'
      - class exit ```.`value'```
      = class exit `y_timing_max'
-----

> ----- end _eqspec.i
> tem -----
      - class exit ```.s(value)```
      = class exit `y_timing_max'
-----

> ----- end _eqlist
> .eq -----
      = local depvars y_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 `y_timing_max'
- sreturn local value ``value''
= sreturn local value `y_timing_max'

```

```

        - class exit ``value''
        = class exit "y_timing_max"
    _____
> _____ end _eqspec.i
> tem _____
        - class exit ``s(value)''
        = class exit "y_timing_max"
    _____
> _____ end _eqlist
> .eq _____
    = local eq y_timing_max
    - if !`:length local eq' {
    = if !12 {
        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 y_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.ntwrth#0b.learning
> 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#
> 0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.le
> arning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning"
> '
      - sreturn local value ```.`value'```
      = sreturn local value `rule_g rule_itr estimated 0b.ntwrth#0b.l
> earning 0b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.b
> nkcrdit#0b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrd
> it#1o.learning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open l
> earning"
      - class exit ```.`value'```
      = class exit `rule_g rule_itr estimated 0b.ntwrth#0b.learning 0
> b.ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0
> b.learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.lea
> rning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning"
-----
> ----- end _eqspec.i
> tem -----
      - class exit ```.`s(value)'```
      = class exit `rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.
> ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.
> learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learn
> ing 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning"

```

```

> _____ end _eqlist
> .eq _____
      = fvexpand rule_g rule_itr estimated 0b.ntwrth#0b.learning 0b.ntwrth
> #1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b.learni
> ng 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.learning 0b
> .wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning if __0000
> 05
      - 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.ntwrth#0b.learning 0b
> .ntwrth#1.learning 1.ntwrth#0b.learning 1o.ntwrth#1o.learning 0b.bnkcrdit#0b
> .learning 0b.bnkcrdit#1o.learning 1.bnkcrdit#0b.learning 1o.bnkcrdit#1o.lear
> ning 0b.wlth#0b.open 0b.wlth#1.open 1.wlth#0b.open 1.wlth#1.open learning''
      - 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 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 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 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      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 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 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 y_timing_max
- local eq_names `eq_names' `eq'
= local eq_names y_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 = 1/16 {
- gettoken xvar TMP : TMP
- local colna ```colna' `eq':`xvar'```
= local colna `` y_timing_max:rule_g```
- }
- gettoken xvar TMP : TMP
- local colna ```colna' `eq':`xvar'```
= local colna `` y_timing_max:rule_g y_timing_max:rule_itr```
- }
- gettoken xvar TMP : TMP

```

```

        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated''
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning''
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning''
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning''
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning''
> '
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning''
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning''
        - }
        - gettoken xvar TMP : TMP
        - local colna ``colna' `eq':`xvar''
        = local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le

```

```

> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning"
    - }
    - gettoken xvar TMP : TMP
    - local colna ``"colna' `eq':`xvar'"
    = local colna ``" y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning"
    - }
    - gettoken xvar TMP : TMP
    - local colna ``"colna' `eq':`xvar'"
    = local colna ``" y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_tim
> ing_max:0b.wlth#0b.open"
    - }
    - gettoken xvar TMP : TMP
    - local colna ``"colna' `eq':`xvar'"
    = local colna ``" y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_tim
> ing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open"
    - }
    - gettoken xvar TMP : TMP
    - local colna ``"colna' `eq':`xvar'"
    = local colna ``" y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_tim
> ing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1.wlth#0b.o
> pen"
    - }
    - gettoken xvar TMP : TMP
    - local colna ``"colna' `eq':`xvar'"
    = local colna ``" y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_

```

```

> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_tim
> ing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1.wlth#0b.o
> pen y_timing_max:1.wlth#1.open''
- }
- gettoken xvar TMP : TMP
- local colna ``colna' `eq':`xvar'''
= local colna `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_tim
> ing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1.wlth#0b.o
> pen y_timing_max:1.wlth#1.open y_timing_max:learning''
- }
- 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 `` y_timing_max:rule_g y_timing_max:rule_itr y_timing_
> max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.le
> arning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning
> y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_
> timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_tim
> ing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1.wlth#0b.o
> pen y_timing_max:1.wlth#1.open y_timing_max:learning y_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 & 20 {
    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 __000007 = J(1,17,0)
- matrix colna `b0' = `colna'
= matrix colna __000007 = y_timing_max:rule_g y_timing_max:rule_itr
> y_timing_max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.n

```

```

> twrth#1.learning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o
> .learning y_timing_max:0b.bnkcrcdit#0b.learning y_timing_max:0b.bnkcrcdit#1o.l
> earning y_timing_max:1.bnkcrcdit#0b.learning y_timing_max:1o.bnkcrcdit#1o.lear
> ning y_timing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1
> .wlth#0b.open y_timing_max:1.wlth#1.open y_timing_max:learning y_timing_max:
> _cons
    - `vv' _mkvec `b0', from(`init') update first error("initial vector"
> )
    = version 18.5: _mkvec __000007, from(__000004) 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 "__000007"==" | "__000007"=="," {
        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 __000007
    - if _rc {
        di in red "matrix `matout' not found"
        exit 111
    }
    - if rowsof(matrix(`matout')) != 1 {
    = if rowsof(matrix(__000007)) != 1 {
        di in red "`matout' matrix to be updated must be " "a row vector
> "
        exit 503
    }
    - local matin `matout'
    = local matin __000007
    - }
    - if ``error'"'"!=" {
    = if `"initial vector'"!=" {
    - local error ``error': "'

```



```

= local error ``initial vector: ''
- }
- tempname mat
- if ```from'``!=`` {
= if ``__000004'``!=`` {
- From `from'
= From __000004

```

```

> _____ begin _mkvec.F
> rom _____
    - gettoken from 0 : 0, parse(",") bind
    - sret clear
    - sret local from `from'
    = sret local from __000004
    - 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 __000008 ``initial vector: '' `` __000004

```

```

> _____ begin _mkvec.Build

```

```

> Mat
- local vv : display "version " string(_caller()) ":"
- version 6
- args matall error skip
- macro shift 3
- tokenize ``*'', parse(" =")
= tokenize ``__000004'', parse(" =")
- tempname matadd
- local i 1
- while ``i''!= "" {
= while "__000004"!= "" {
- 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 __000009 __000004 ``initial vector: ''

> _____ begin _mkvec.Chk
> Mat
- args matout matin error
- capture di matrix(`matin'[1,1])
= capture di matrix(__000004[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(__000004))!=1 & colsof(__000004))!=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(__000004)) == 1 {
- mat `matout' = `matin'

```

```

= mat __000009 = __000004
- }
- else mat `matout' = `matin'
= else mat __000009 = __000004'

```

```

> _____ end _mkvec.Chk
> Mat _____
    - local i = `i' + 1
    = local i = 1 + 1
    - }
    - mat `matall' = nullmat(`matall') , `matadd'
    = mat __000008 = nullmat(__000008) , __000009
    - }
    - 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 ``y_timing_max:rule_g y_timing_max:rule_itr y_timing_max:esti
> mated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.learning y
> _timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning y_timing
> _max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_timing_m
> ax:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_timing_max:
> 0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1.wlth#0b.open y_ti
> ming_max:1.wlth#1.open y_timing_max:learning y_timing_max:_cons'""!="" {
    - `vv' ChkNames `mat' ``matin' ``colname' ``error' ``skip'
> ``first'
    = version 18.5: ChkNames __000008 "__000007" ``y_timing_max:rule_g
> y_timing_max:rule_itr y_timing_max:estimated y_timing_max:0b.ntwrth#0b.lear
> ning y_timing_max:0b.ntwrth#1.learning y_timing_max:1.ntwrth#0b.learning y_t
> iming_max:1o.ntwrth#1o.learning y_timing_max:0b.bnkcrdit#0b.learning y_timin
> g_max:0b.bnkcrdit#1o.learning y_timing_max:1.bnkcrdit#0b.learning y_timing_m
> ax:1o.bnkcrdit#1o.learning y_timing_max:0b.wlth#0b.open y_timing_max:0b.wlth

```

```

> #1.open y_timing_max:1.wlth#0b.open y_timing_max:1.wlth#1.open y_timing_max:
> learning y_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(__000008))
    - if "`first'"!="" {
    = if "first"!=" {
    - gettoken name : names
    - local eq1 = bsubstr("`name'",1,index("`name'",":"))
    = local eq1 = bsubstr("`y_timing_max:rule_g'",1,index("`y_timing
> _max:rule_g'",":"))
    - }
    - tempname new fill x
    - local k : list sizeof names
    - if "`matin'"!="" {
    = if "__000007"!=" {
    - mat `new' = `matin'
    = mat __000009 = __000007
    - }
    - else {
    mat `new' = J(1,`k',0)
    `vv' mat colnames `new' = `names'
    }
    - mat `fill' = J(1,`k',0)
    = mat __00000A = J(1,17,0)
    - local i 1
    - while `i' <= `dim' {
    = while 1 <= 17 {
    - mat `x' = `mat'[1,`i'..`i']
    = mat __00000B = __000008[1,1..1]
    - local eq : coleq `x'
    = local eq : coleq __00000B
    - local name : colname `x'
    = local name : colname __00000B
    - 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':`name'
  }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__000009,``_: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==. & ``_'=="_" & ``y_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 __00000A[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'
  }
}
}
exit 507
}
> found''

```

```

        else if !missing(`new'[1,`j']) {
            matrix `mat'[1,`i'] = `new'[1,`j']
        }
    }
- else mat `filll'[1,`j'] = 1
= else mat __00000A[1,1] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,1] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,2..2]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- 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(__000009,``_:rule_itr'')
- if `j' == . & ``oname'' != "" {
= if 2 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``spec'' != "" {
= if 2 == . & "/_:rule_itr" != "" {
  `vv' local j = colnumb(`new',``spec'')
}
- if `j'==. & ``eq''=="_" & ``eq1''!="" {
= if 2==. & ``_'=="_" & ``y_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 __00000A[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 __00000A[1,2] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,2] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,3..3]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- 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':`oname'"
}
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__000009,`_: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 == . & ```_ == "_" & ``y_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 __00000A[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 __00000A[1,3] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,3] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,4..4]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
  }
- else {
- local spec "`eq':`name'"
= local spec "_:0b.ntwrth#0b.learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:0b.ntwrth#0b.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 0b.ntwrth#0b.learning, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)''
= local oname ``0.ntwrth#0.learning''
- local ospec "`eq':`oname'"
= local ospec "_:0.ntwrth#0.learning"
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__000009,``_:0b.ntwrth#0b.lear
> ning'')
- if `j' == . & "`oname'" != "" {
= if 4 == . & "0.ntwrth#0.learning" != "" {
  `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & "`sspec'" != "" {
= if 4 == . & "/_:0b.ntwrth#0b.learning" != "" {
  `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=='_' & ``eq1'!='' {

```

```

= if 4==. & ``_""=="_ & ``y_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 __00000A[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 __00000A[1,4] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,4] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,5..5]
- local eq : coleq `x'
= local eq : coleq __00000B

```

```

- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
  }
- else {
- local spec "`eq':`name'"
= local spec "_:0b.ntwrth#1.learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:0b.ntwrth#1.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 0b.ntwrth#1.learning, 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(__000009,``_:0b.ntwrth#1.learn
> ing'')
- if `j' == . & "`oname'" != "" {
= if 5 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & "`sspec'" != "" {
= if 5 == . & "/_:0b.ntwrth#1.learning" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 5==. & ``_'""=="_" & ``y_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 __00000A[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'
> found"'
    exit 507
  }
  else if !missing(`new'[1,`j']) {
    matrix `mat'[1,`i'] = `new'[1,`j']
  }
}
- else mat `fill'[1,`j'] = 1
= else mat __00000A[1,5] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,5] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,6..6]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
  local spec "/"`name'
}
- else {
- local spec "`eq':`name'"
= local spec "_:1.ntwrth#0b.learning"

```

```

- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:1.ntwrth#0b.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1.ntwrth#0b.learning, 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(__000009,``_:1.ntwrth#0b.learn
> ing'')
- if `j' == . & ``oname' != "" {
= if 6 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec' != "" {
= if 6 == . & "/_:1.ntwrth#0b.learning" != "" {
  `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'=="_" & ``eq1'!='' {
= if 6==. & ``_'=="_" & ``y_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 __00000A[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 __00000A[1,6] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,6] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,7..7]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
  local spec "/"`name'
}
- else {
- local spec "`eq':`name'"
= local spec "_:10.ntwrth#10.learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "/_:10.ntwrth#10.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit

```

```

= capture _msparse 1o.ntwrth#1o.learning, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)''
= local oname ``1.ntwrth#1.learning''
- local ospec ``eq':`oname''
= local ospec ``_:1.ntwrth#1.learning''
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__000009,``_:1o.ntwrth#1o.lear
> ning'')
- if `j' == . & ``oname'' != "" {
= if 7 == . & "1.ntwrth#1.learning" != "" {
`vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & ``sspec'' != "" {
= if 7 == . & "/_:1o.ntwrth#1o.learning" != "" {
`vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq''=="_" & ``eq1''!="" {
= if 7==. & ``_'=="_" & ``y_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 __00000A[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 __00000A[1,7] = 1

```



```

- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,7] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,8..8]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'
}
- else {
- local spec "`eq':`name'"
= local spec "_:0b.bnkcrcdit#0b.learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "/_:0b.bnkcrcdit#0b.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 0b.bnkcrcdit#0b.learning, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname `"'r(stripe)'"
= local oname `"0.bnkcrcdit#0.learning"
- local ospec "`eq':`oname'"
= local ospec "_:0.bnkcrcdit#0.learning"

```

```

- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__000009,``_:0b.bnkcrdit#0b.le
> arning'')
- if `j' == . & ``oname' != "" {
= if 8 == . & "0.bnkcrdit#0.learning" != "" {
  `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & ``sspec' != "" {
= if 8 == . & "/_:0b.bnkcrdit#0b.learning" != "" {
  `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 8==. & ``_'=="_" & ``y_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 __00000A[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 __00000A[1,8] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,8] = __000008[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 <= 17 {
    - mat `x' = `mat'[1,`i'..`i']
    = mat __00000B = __000008[1,9..9]
    - local eq : coleq `x'
    = local eq : coleq __00000B
    - local name : colname `x'
    = local name : colname __00000B
    - if "`eq'" == "/" {
    = if "_" == "/" {
        local spec "/"`name'"
    }
    - else {
    - local spec "`eq':`name'"
    = local spec "_:0b.bnkcrdit#1o.learning"
    - if (usubstr("`eq'",1,1) != "/") {
    = if (usubstr("_",1,1) != "/") {
    - local sspec "/"`eq':`name'"
    = local sspec "/_:0b.bnkcrdit#1o.learning"
    - }
    - }
    - local oname
    - local ospec
    - capture _msparse `name', noomit
    = capture _msparse 0b.bnkcrdit#1o.learning, noomit
    - if c(rc) == 0 {
    - if r(omit) {
    - local oname ``r(stripe)'"
    = local oname ``0.bnkcrdit#1.learning'"
    - local ospec "`eq':`oname'"
    = local ospec "_:0.bnkcrdit#1.learning"
    - }
    - }
    - `vv' local j = colnumb(`new',``spec'')
    = version 18.5: local j = colnumb(__000009,`_:0b.bnkcrdit#1o.le
> arning'')
    - if `j' == . & "`oname'" != "" {
    = if 9 == . & "0.bnkcrdit#1.learning" != "" {

```

```

        `vv' local j = colnumb(`new',``ospec'')
    }
- if `j' == . & ``sspec' != "" {
= if 9 == . & "/_:#b.bnkcrcdit#1o.learning" != "" {
    `vv' local j = colnumb(`new',``sspec'')
    }
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 9==. & ``_'=="_" & ``y_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 __00000A[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 __00000A[1,9] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,9] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..'i']
= mat __00000B = __000008[1,10..10]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'
  }
- else {
- local spec "`eq':`name'"
= local spec "_:1.bnkcrcdit#0b.learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "/_:1.bnkcrcdit#0b.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1.bnkcrcdit#0b.learning, 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(__000009,``_:1.bnkcrcdit#0b.lea
> rning'')
- if `j' == . & "`oname'" != "" {
= if 10 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & "sspec'" != "" {
= if 10 == . & "/_:1.bnkcrcdit#0b.learning" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 10==. & `_"'=="_" & `y_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 __00000A[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 __00000A[1,10] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,10] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,11..11]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B

```

```

- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
}
- else {
- local spec "`eq':`name'"
= local spec "_:10.bnkcrdit#10.learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "/_:10.bnkcrdit#10.learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 10.bnkcrdit#10.learning, noomit
- if c(rc) == 0 {
- if r(omit) {
- local oname ``r(stripe)''
= local oname `1.bnkcrdit#1.learning'
- local ospec "`eq':`oname'"
= local ospec "_:1.bnkcrdit#1.learning"
- }
- }
- `vv' local j = colnumb(`new',``spec'')
= version 18.5: local j = colnumb(__000009,``_:10.bnkcrdit#10.le
> arning'')
- if `j' == . & "`oname'" != "" {
= if 11 == . & "1.bnkcrdit#1.learning" != "" {
    `vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & "`sspec'" != "" {
= if 11 == . & "/_:10.bnkcrdit#10.learning" != "" {
    `vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 11==. & ``_""=="_" & ``y_timing_max:'""!="" {
    `vv' local j = colnumb(`new',``eq1``name'')
    if `j' == . & "`oname'" != "" {
        `vv' local j = colnumb(`new',``eq1``oname'')
    }
}
- if `j'!=. {
= if 11!=. {
- if `filll'[1,`j'] {

```

```

= if __00000A[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 __00000A[1,11] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,11] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,12..12]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- 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(__000009,``_:0b.wlth#0b.open"

> )

- if `j' == . & "`oname'" != "" {
= if 12 == . & "0.wlth#0.open" != "" {
`vv' local j = colnumb(`new',``ospec'')
}
- if `j' == . & "`sspec'" != "" {
= if 12 == . & "/_:0b.wlth#0b.open" != "" {
`vv' local j = colnumb(`new',``sspec'')
}
- if `j'==. & ``eq'""=="_" & ``eq1'""!="" {
= if 12==. & ``_""=="_" & ``y_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 __00000A[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 __00000A[1,12] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,12] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..'i']
= mat __00000B = __000008[1,13..13]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- 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(__000009,``_:0b.wlth#1.open'')
- if `j' == . & ``oname''' != "" {
= if 13 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec''')
  }
- if `j' == . & ``sspec''' != "" {
= if 13 == . & "/_:0b.wlth#1.open" != "" {
    `vv' local j = colnumb(`new',``sspec''')
  }
- if `j'==. & ``eq'''=="_" & ``eq1'''!="" {
= if 13==. & ``_'=="_" & ``y_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 __00000A[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 __00000A[1,13] = 1
- mat `new'[1,`j'] = `mat'[1,`i']

```

```

= mat __000009[1,13] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,14..14]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- 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(__000009,`_:1.wlth#0b.open`)
- if `j' == . & ``oname' != "" {
= if 14 == . & "" != "" {
  `vv' local j = colnumb(`new',``ospec'`)
}
- if `j' == . & ``sspec' != "" {
= if 14 == . & "/_:1.wlth#0b.open" != "" {
  `vv' local j = colnumb(`new',``sspec'`)
}
- if `j'==. & ``eq'=="_" & ``eq1'!="" {
= if 14==. & ``_'=="_" & ``y_timing_max:'!="" {
  `vv' local j = colnumb(`new',``eq1``name'`)
  if `j' == . & ``oname' != "" {
    `vv' local j = colnumb(`new',``eq1``oname'`)
  }
}
- if `j'!=. {
= if 14!=. {
- if `fill'[1,`j'] {
= if __00000A[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 `fill'[1,`j'] = 1
= else mat __00000A[1,14] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,14] = __000008[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 <= 17 {
- mat `x' = `mat'[1,`i'..`i']
= mat __00000B = __000008[1,15..15]
- local eq : coleq `x'
= local eq : coleq __00000B
- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/"`name'"
    }
- else {
- local spec "`eq':`name'"
= local spec "_:1.wlth#1.open"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/"`eq':`name'"
= local sspec "/_:1.wlth#1.open"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse 1.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(__000009,``_:1.wlth#1.open'')
- if `j' == . & ``oname'" != "" {
= if 15 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
    }
- if `j' == . & ``sspec'" != "" {
= if 15 == . & "/_:1.wlth#1.open" != "" {
    `vv' local j = colnumb(`new',``sspec'')
    }
- if `j'==. & ``eq'"=="_" & ``eq1'"!="" {

```



```

- local name : colname `x'
= local name : colname __00000B
- if "`eq'" == "/" {
= if "_" == "/" {
    local spec "/`name'"
  }
- else {
- local spec "`eq':`name'"
= local spec "_:learning"
- if (usubstr("`eq'",1,1) != "/") {
= if (usubstr("_",1,1) != "/") {
- local sspec "/`eq':`name'"
= local sspec "_:learning"
- }
- }
- local oname
- local ospec
- capture _msparse `name', noomit
= capture _msparse learning, 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(__000009,`_:learning'')
- if `j' == . & "`oname'" != "" {
= if 16 == . & "" != "" {
    `vv' local j = colnumb(`new',``ospec'')
  }
- if `j' == . & "`sspec'" != "" {
= if 16 == . & "_:learning" != "" {
    `vv' local j = colnumb(`new',``sspec'')
  }
- if `j'==. & ``eq'=='_' & ``eq1'!='' {
= if 16==. & ``_'=='_' & ``y_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 __00000A[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 __00000A[1,16] = 1
        - mat `new'[1,`j'] = `mat'[1,`i']
        = mat __000009[1,16] = __000008[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 <= 17 {
            - mat `x' = `mat'[1,`i'..`i']
            = mat __00000B = __000008[1,17..17]
            - local eq : coleq `x'
            = local eq : coleq __00000B
            - local name : colname `x'
            = local name : colname __00000B
            - if "`eq'" == "/" {
            = if "_" == "/" {
                local spec "/"`name'"
            }
            - else {
            - local spec "`eq':`name'"
            = local spec ":_cons"
            - if (usubstr("`eq'",1,1) != "/") {

```



```

        matrix `mat'[1,`i'] = `new'[1,`j']
    }
}
- else mat `fill'[1,`j'] = 1
= else mat __00000A[1,17] = 1
- mat `new'[1,`j'] = `mat'[1,`i']
= mat __000009[1,17] = __000008[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 <= 17 {
    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 __00000A = __00000A*__00000A'
    - sret clear
    - sret local k_fill = `fill'[1,1]
    = sret local k_fill = __00000A[1,1]
    - mat `mat' = `new'
    = mat __000008 = __000009

```

```

> _____ end _mkvec.ChkNa
> mes _____
    - local fill `s(k_fill)'
    = local fill 17
    - }
    - }
    - }
    - 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(__000008))
    - if "`fill'"!="" {
    = if "17"!="" {
      - sret local k_fill `fill'
      = sret local k_fill 17
    - }
    - else sret local k_fill = colsof(matrix(`mat'))
    = else sret local k_fill = colsof(matrix(__000008))
    - matrix rename `mat' `matout', replace
    = matrix rename __000008 __000007, replace
    _____
> _____ end _mk
> vec _____
    - }
    - tempname bomit
    - matrix `bomit' = J(1,`:list sizeof colna',0)
    = matrix __000008 = J(1,17,0)
    - matrix colna `bomit' = `colna'

```

```

= matrix colna __000008 = y_timing_max:rule_g y_timing_max:rule_itr
> y_timing_max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.n
> twrth#1.learning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o
> .learning y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.l
> earning y_timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.lear
> ning y_timing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1
> .wlth#0b.open y_timing_max:1.wlth#1.open y_timing_max:learning y_timing_max:
> _cons
- _ms_omit_info `bomit'
= _ms_omit_info __000008
- if r(k_omit) | `:length local constraints' {
= if r(k_omit) | 0 {
- nobreak {
- capture noisily break {
- _b_post0 `colna'
= _b_post0 y_timing_max:rule_g y_timing_max:rule_itr y_timing_max:e
> stimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:0b.ntwrth#1.learnin
> g y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth#1o.learning y_tim
> ing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#1o.learning y_timin
> g_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.learning y_timing_m
> ax:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_max:1.wlth#0b.open y
> _timing_max:1.wlth#1.open y_timing_max:learning y_timing_max:_cons

> _____ begin _b_po
> st0 _____
- version 11
- tempname b V
- local dim : list sizeof 0
- matrix `b' = J( 1,`dim',0)
= matrix __000009 = J( 1,17,0)
- matrix `V' = J(`dim',`dim',0)
= matrix __00000A = J(17,17,0)
- matrix colnames `b' = `0'
= matrix colnames __000009 = y_timing_max:rule_g y_timing_max:rule
> _itr y_timing_max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:
> 0b.ntwrth#1.learning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrt
> h#1o.learning y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#
> 1o.learning y_timing_max:1.bnkcrdit#0b.learning y_timing_max:1o.bnkcrdit#1o.
> learning y_timing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_m
> ax:1.wlth#0b.open y_timing_max:1.wlth#1.open y_timing_max:learning y_timing_
> max:_cons
- matrix colnames `V' = `0'
= matrix colnames __00000A = y_timing_max:rule_g y_timing_max:rule
> _itr y_timing_max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:
> 0b.ntwrth#1.learning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrt
> h#1o.learning y_timing_max:0b.bnkcrdit#0b.learning y_timing_max:0b.bnkcrdit#

```

```

> 1o.learning y_timing_max:1.bnkcrcdit#0b.learning y_timing_max:1o.bnkcrcdit#1o.
> learning y_timing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_m
> ax:1.wlth#0b.open y_timing_max:1.wlth#1.open y_timing_max:learning y_timing_
> max:_cons
    - matrix rownames `V' = `0'
    = matrix rownames __00000A = y_timing_max:rule_g y_timing_max:rule
> _itr y_timing_max:estimated y_timing_max:0b.ntwrth#0b.learning y_timing_max:
> 0b.ntwrth#1.learning y_timing_max:1.ntwrth#0b.learning y_timing_max:1o.ntwrth
> h#1o.learning y_timing_max:0b.bnkcrcdit#0b.learning y_timing_max:0b.bnkcrcdit#
> 1o.learning y_timing_max:1.bnkcrcdit#0b.learning y_timing_max:1o.bnkcrcdit#1o.
> learning y_timing_max:0b.wlth#0b.open y_timing_max:0b.wlth#1.open y_timing_m
> ax:1.wlth#0b.open y_timing_max:1.wlth#1.open y_timing_max:learning y_timing_
> max:_cons
    - ereturn post `b' `V'
    = ereturn post __000009 __00000A
    _____
> _____ 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 __000009 = 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 vce(robust)
- if `:length local vce' {
= if 11 {
- _vce_parse, argoptlist(Cluster) optlist(OIM OPG Robust) old : `wt'
> , `vce'
= _vce_parse, argoptlist(Cluster) optlist(OIM OPG Robust) old : , vc
> e(robust)
_____
> _____ begin _vce_pa
> rse _____
- version 10
- _on_colon_parse `0'
= _on_colon_parse , argoptlist(Cluster) optlist(OIM OPG Robust) ol
> d : , vce(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 `"" != ":" & `": , vce(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 `":' != ":" & `", vce(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(CLister) optlist(OIM OPG R
> obust) old ""
- sreturn local after ``"after'
= sreturn local after `", vce(robust)"

----- end _on_colon_pa
> rse -----
- local 0 ``"s(before)'
= local 0 `", argoptlist(CLister) optlist(OIM OPG Robust) old ""
- local user ``"s(after)'
= local user `", vce(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
= .__00000A = ._optlist.new
- local optlist : list uniq optlist
- foreach name of local optlist {
- .`o'.addopt `name'
= .__00000A.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'
= .__00000A.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'
= .__00000A.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
    = .__00000B = ._optlist.new
    - local argoptlist : list uniq argoptlist
    - foreach name of local argoptlist {
    - .`a'.addopt `name'
    = .__00000B.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 CCluster(varname)
- }
- }
- local 0 ```user''
= local 0 `` , vce(robust)''
- 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 __00000A __00000B , 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 = 1/3 {

```

```

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

```

```

> _____ end _optlist.dumpopti
> ons _____

```

```

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

```

```

> _____ end _optlist.dumpopti
> ons _____
= local OPTSPECS ``OIM OPG Robust CLuster'''
- local OPTNAMES ```.`0'.dumpnames' `.`A'.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 _____

> _____ 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 OPTNAMES "oim opg robust cluster"
- syntax [, `OPTSPECS']
= syntax [, OIM OPG Robust CLuster]
- 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 ``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"
> , "cluster") {
= if ````' != "" & !inlist("robust","", "robust", "eim", "oim", "cluste
> r") {
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 _____
    - 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 "__000005" != "__000005" {
    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 __000005
- 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 20 {
- if `:list sizeof lf0' != 2 {
= if 2 != 2 {
  di as err "lf0(`lf0') invalid"
  exit 198
}
- global ML_k0 : word 1 of `lf0'
= global ML_k0 : word 1 of 1 -420.2971615663282
- global ML_ll_0 : word 2 of `lf0'
= global ML_ll_0 : word 2 of 1 -420.2971615663282
- confirm integer number $ML_k0
= confirm integer number 1
- confirm number $ML_ll_0
= confirm number -420.2971615663282
- if $ML_k0 < 0 {
= if 1 < 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 __000005
- 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 confirm matrix `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 (``min'' != "") {
= else if (``'' != "") {
        if (`len' > length(`min')) {
            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 poiss_lf
    - if c(rc) {
        di as err "unrecognized command:  `name' invalid command name"
        exit 199
    }
}

```

```

        - capture which `name'
        = capture which poiss_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) 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 __000005 __000005
    - 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) QTOLerance(numlist max=1 >=0) GTOLerance(numlist max=1 >=0) ITERate(numlist max=1 >=0) NOTCONCAVE(numlist max=1 >=0) ndami DIFFicult HALFSTEPonly DOOPT SEARch(name) Repeat(passthru) Bounds(string) SCore(string) noWARNing noCLEAR NOLOG LOG DOTs SHOWTOLerance SHOWNRtolerance SHOWSTEP TRace COEFDiffs GRADient

```



```

> HESSian noSKIPline noOUTput Level(cilevel) moptobj(string) * ]
    - if `:length local nrtolerance' {
    = if 0 {
      opts_exclusive "nrtolerance() `nonnrtolerance'"
    }
    - if `:length local qtolerance' {
    = if 0 {
      opts_exclusive "qtolerance() `nonnrtolerance'"
    }
    - opts_exclusive "`nonnrtolerance' `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" & "" != "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 poiss
> _lf _____
    - version 6
    - args todo b lnf g H score
    - tempvar xb
    - mlevel `xb' = `b'
    = mlevel __00000P = __00000J
    - mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
    = mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
    - if `todo' == 0 | `lnf'==. {
    = if 2 == 0 | __00000K==. {
        exit
    }
    - qui replace `score' = $ML_y1 - exp(`xb') if $ML_samp
    = qui replace __00000N = y_timing_max - exp(__00000P) if __00000
> 5
    - $ML_ec mlsun `lnf' `g' = `score'
    = * mlsun __00000K __00000L = __00000N
    - if `todo' == 1 | `lnf'==. {
    = if 2 == 1 | __00000K==. {
        exit
    }
    - mlsun `lnf' `H' = exp(`xb')
    = mlsun __00000K __00000M = exp(__00000P)

> _____ end poiss
> _lf _____

> _____ begin poiss
> _lf _____
    - version 6
    - args todo b lnf g H score
    - tempvar xb
    - mlevel `xb' = `b'
    = mlevel __00000P = __00000J

```

```

- mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
= mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
- if `todo' == 0 | `lnf'==. {
= if 0 == 0 | __00000K==. {
- exit
}

> _____ end poiss
> _lf _____

> _____ begin poiss
> _lf _____
- version 6
- args todo b lnf g H score
- tempvar xb
- mlevel `xb' = `b'
= mlevel __00000P = __00000J
- mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
= mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
- if `todo' == 0 | `lnf'==. {
= if 0 == 0 | __00000K==. {
- exit
}

> _____ end poiss
> _lf _____

> _____ begin poiss
> _lf _____
- version 6
- args todo b lnf g H score
- tempvar xb
- mlevel `xb' = `b'
= mlevel __00000P = __00000J
- mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
= mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
- if `todo' == 0 | `lnf'==. {
= if 2 == 0 | __00000K==. {
exit
}
- qui replace `score' = $ML_y1 - exp(`xb') if $ML_samp
= qui replace __00000N = y_timing_max - exp(__00000P) if __00000
> 5
- $ML_ec mlvecsum `lnf' `g' = `score'
= * mlvecsum __00000K __00000L = __00000N

```

```

- if `todo' == 1 | `lnf'==. {
= if 2 == 1 | __00000K==. {
    exit
  }
- mlmatsum `lnf' `H' = exp(`xb')
= mlmatsum __00000K __00000M = exp(__00000P)
_____
> _____ end poiss
> _lf _____
_____
> _____ begin poiss
> _lf _____
    - version 6
    - args todo b lnf g H score
    - tempvar xb
    - mlevel `xb' = `b'
    = mlevel __00000P = __00000J
    - mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
    = mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
    - if `todo' == 0 | `lnf'==. {
    = if 0 == 0 | __00000K==. {
    - exit
    _____
> _____ end poiss
> _lf _____
_____
> _____ begin poiss
> _lf _____
    - version 6
    - args todo b lnf g H score
    - tempvar xb
    - mlevel `xb' = `b'
    = mlevel __00000P = __00000J
    - mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
    = mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
    - if `todo' == 0 | `lnf'==. {
    = if 0 == 0 | __00000K==. {
    - exit
    _____
> _____ end poiss
> _lf _____
_____
> _____ begin poiss
> _lf _____

```

```

- version 6
- args todo b lnf g H score
- tempvar xb
- mlevel `xb' = `b'
= mlevel __00000P = __00000J
- mlsun `lnf' = -exp(`xb') + `xb'*$ML_y1 - lngamma($ML_y1+1)
= mlsun __00000K = -exp(__00000P) + __00000P*y_timing_max - lnga
> mma(y_timing_max+1)
- if `todo' == 0 | `lnf'==. {
= if 2 == 0 | __00000K==. {
    exit
  }
- qui replace `score' = $ML_y1 - exp(`xb') if $ML_samp
= qui replace __00000N = y_timing_max - exp(__00000P) if __00000
> 5
- $ML_ec mlevelsum `lnf' `g' = `score'
= * mlevelsum __00000K __00000L = __00000N
- if `todo' == 1 | `lnf'==. {
= if 2 == 1 | __00000K==. {
    exit
  }
- mlevelsum `lnf' `H' = exp(`xb')
= mlevelsum __00000K __00000M = exp(__00000P)
_____
> _____ end poiss
> _lf _____
_____
> _____ begin mopt_p
> ost _____
- args b V C wgt touse depvar
- if `:length local touse' {
= if 8 {
- tempname esamp
- quietly gen byte `esamp' = `touse'
= quietly gen byte __00000S = __000005
- }
- ereturn post `b' `V' `C' `wgt', esample(`esamp') depname(`depv
> ar') buildfvinfo
= ereturn post __00000P __00000Q __00000R [], esample(__00000S)
> depname(y_timing_max) buildfvinfo
_____
> _____ end mopt_p
> ost _____
- ereturn scalar k_eq_model = abs($ML_waldtest)
= ereturn scalar k_eq_model = abs(-1)
- if `:length global ML_ll_0' {

```

```

= if 18 {
- ereturn scalar ll_0 = $ML_ll_0
= ereturn scalar ll_0 = -420.2971615663282
- if `:length global ML_rank0' {
= if 0 {
    ereturn scalar rank0 = $ML_rank0
    ereturn scalar df_m = e(rank) - $ML_rank0
  }
- else ereturn scalar df_m = e(rank) - $ML_k0
= else ereturn scalar df_m = e(rank) - 1
- }
- if `:length global ML_ll_0' | inlist(e(vce), "robust", "cluster
> ", "linearized") {
= if !18 | inlist(e(vce), "robust", "cluster", "linearized") {
- if "$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 __00000S = get(Cns)
- if (c(rc)) local cns
- if "`cns'" != "" {
= if "__00000S" != "" {
- local k_autoCns = e(k_autoCns)
- if missing(`k_autoCns') {
= if missing(6) {
    local k_autoCns 0
  }
- if rowsof(`cns') == `k_autoCns' {
= if rowsof(__00000S) == 6 {
- local cns
- }
- }
- if "`svy'" != "" & "`adjust'`svyadjust'" != "" {
= if "" != "" & "" != "" {
    local adjust nosvyadjust
  }
- else local adjust
- _ms_omit_info e(b), nocons
- matrix `omit' = r(omit)
= matrix __00000T = 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 : substr 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 __00000U = r(chi2)
- scalar `df1' = r(df)
= scalar __00000V = r(df)
- scalar `p' = r(p)
= scalar __00000W = r(p)
- scalar `drop' = r(drop)
= scalar __00000X = 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 "8" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_3 8
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "9" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_4 9
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "11" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_5 11
- local ++i
- }
- while "`r(dropped_`i')'" != "" {
= while "12" != "" {
- local dropped_`i' `r(dropped_`i')'
= local dropped_6 12
- 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 "8" != "" {
- di as txt "          Constraint `dropped_`i'" dropped"
= di as txt "          Constraint 8 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "9" != "" {
- di as txt "          Constraint `dropped_`i'" dropped"
= di as txt "          Constraint 9 dropped"
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "11" != "" {
- di as txt "          Constraint `dropped_`i'" dropped"
= di as txt "          Constraint 11 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 "" != "" {
  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 __00000V ") =" as res %8.
> 2f __00000U
- di as txt _col(10) "Prob > chi2 = " as res %8.4f `p'
= di as txt _col(10) "Prob > chi2 = " as res %8.4f __00000W
- }
- 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 = __00000W
- return scalar df = `df1'
= return scalar df = __00000V
- if "`testtype'" == "F" {
= if "chi2" == "F" {
    return scalar F = `F'
    return scalar df_r = `df2'
  }
- else return scalar chi2 = `chi2'
= else return scalar chi2 = __00000U
- return scalar drop = `drop'
= return scalar drop = __00000X
- 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 "8" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_3 = 8
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "9" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_4 = 9
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "11" != "" {
- return scalar dropped_`i' = `dropped_`i''
= return scalar dropped_5 = 11
- local ++i
- }
- while "`dropped_`i'" != "" {
= while "12" != "" {
- return scalar dropped_`i' = `dropped_`i''

```

```

= return scalar dropped_6 = 12
- 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 `3'
    - while ``idrop'' != "" {
= while "3" != "" {
    - if !`omit'[1,`idrop'] {
= if !__00000T[1,3] {
    - local ++drop
    - }

```

```

- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop ``4''
- }
- while ``idrop' != "" {
= while "4" != "" {
- if !`omit'[1,`idrop'] {
= if !__00000T[1,4] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop ``7''
- }
- while ``idrop' != "" {
= while "7" != "" {
- if !`omit'[1,`idrop'] {
= if !__00000T[1,7] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop ``8''
- }
- while ``idrop' != "" {
= while "8" != "" {
- if !`omit'[1,`idrop'] {
= if !__00000T[1,8] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop ``9''
- }
- while ``idrop' != "" {
= while "9" != "" {
- if !`omit'[1,`idrop'] {
= if !__00000T[1,9] {
    local ++drop
  }
- local ++i
- local idrop ``r(dropped_`i')'''
= local idrop ``11''
- }
- while ``idrop' != "" {
= while "11" != "" {

```

```

- if !`omit'[1,`idrop'] {
= if !__00000T[1,11] {
  local ++drop
}
- local ++i
- local idrop ``r(dropped_`i')''''
= local idrop ``12''
- }
- while ``idrop' != "" {
= while "12" != "" {
- if !`omit'[1,`idrop'] {
= if !__00000T[1,12] {
  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 1 == 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
- }

```



```

- 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 __000005
  - capture drop $ML_samp
  = capture drop __000005
  - capture drop $ML_sample
  = capture drop __000005
  - 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 -----
    - if "`score'" != "" {
    = if "" != "" {
      label var `scvar' "Score index from poisson"
      rename `scvar' `score'
      est local scorevars `score'
    }
    - est scalar r2_p = 1 - e(ll)/e(ll_0)
    - est hidden local gof "poiss_g"
    - est local offset "`offvar'"
    = est local offset ""
    - est local offset1
    - est local estat_cmd "poisson_estat"
    - est hidden local marginsderiv default N IR
    - est local predict "poisso_p"
    - est local cmd "poisson"
    - global S_E_ll = e(ll)
    - global S_E_ll0 `e(ll_0)'
    = global S_E_ll0 -420.2971615663282
    - global S_E_chi2 = e(chi2)
    - global S_E_mdf = e(df_m)
    - global S_E_pr2 `e(r2_p)'
    = global S_E_pr2 .1059779745763584
    - global S_E_nobs = e(N)
    - global S_E_tdf = e(N)
    - global S_E_off `e(offset)'

```

```

= global S_E_off
- global S_E_depv `e(depvar)'
= global S_E_depv y_timing_max
- global S_E_cmd `e(cmd)'
= global S_E_cmd poisson
- if "`display'"==" " {
= if ""==" " {
- Display, `irr' level(`level') `diopts'
= Display, level(95)

```

```

> ----- begin poisson.Disp
> lay
- syntax [, Level(cilevel) IRr *]
- if "`irr'"!=" " {
= if ""!=" " {
    local eopt "eform(IRR)"
}
- _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 "'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' `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
    - 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_optlist = s(optlist)
- }
- 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 "`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')
}
- Wrapon , `fvwrapon'
= Wrapon ,

```

```

> _____ begin _get_diopts.Wra
> pon _____

```

```

- capture syntax [, W0rd WIdth]
- if c(rc) {

```

```

        di as err "invalid fwwrapon() option;"
        syntax [, WOrd 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, `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' `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' `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')
}
- 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' `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 optlist = "CLuster"
    - }
    - sreturn local `sname' = "`S_`sname'"
    = sreturn local after = ", vce(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 = "17"
    - }
    - sreturn local `sname' = "`S_`sname'"
    = sreturn local k = "17"
    - }
    _____
> _____ end _get_dio
> pts _____
    - version 9: ml di, level(`level') `eopt' nofootnote `diopts'
    = version 9: ml di, level(95) nofootnote
    _____
> _____ begin
> ml _____
    - if _caller()>=11 {

```

```

    mopt `0'
    exit
}
- version 8.2, missing
- version 6, missing
- if _caller()<6 {
    ml_5 `0'
    exit
}
- local caller : display string(_caller())
- gettoken cmd 0: 0, parse(" ,")
- local l = length("`cmd'")
= local l = length("`di'")
- if "`cmd'" == "technique" {
= if "`di'" == "technique" {
    ml_technique `0'
    exit
}
- if "`cmd'" == "clear" {
= if "`di'" == "clear" {
    ml_clear `0'
    exit
}
- if "`cmd'" == "hold" {
= if "`di'" == "hold" {
    ml_hold `0'
    exit
}
- if "`cmd'" == "unhold" {
= if "`di'" == "unhold" {
    ml_unhold `0'
    exit
}
- if "`cmd'" == "score" {
= if "`di'" == "score" {
    if "e(opt)" == "moptimize" {
        mopt score `0'
        exit
    }
    ml_score `0'
    exit
}
- if "`cmd'"=="count" {
= if "`di'"=="count" {
    ml_c_d count `0'
    exit
}

```

```

    }
- if `""cmd"" == "trace" {
= if `""di"" == "trace" {
  ml_c_d trace `0'
  exit
}
- if `""cmd""==bsubstr("graph",1,max(2,`l')) {
= if `""di""==bsubstr("graph",1,max(2,2)) {
  ml_graph `0'
  exit
}
- if `""cmd"" == "init" {
= if `""di"" == "init" {
  ml_init `0'
  exit
}
- if `""cmd"" == bsubstr("maximize",1,max(3,`l')) {
= if `""di"" == bsubstr("maximize",1,max(3,2)) {
  version `caller', missing: AllowEv ml_max `0'
  exit
}
- if `""cmd"" == bsubstr("display",1,max(2,`l')) | `""cmd"" == bsubs
> tr("mlout", 1,max(3,`l')) {
  = if `""di"" == bsubstr("display",1,max(2,2)) | `""di"" == bsubstr("mlou
> t", 1,max(3,2)) {
  - if "e(opt)" == "moptimize" {
  = if "moptimize" == "moptimize" {
  - mopt display `0'
  = mopt display , level(95) nofootnote
}
}
> _____ 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 "display" == "clear" {
  Clear
  exit
}
- if `""subcmd"" == "hold" {
= if `""display"" == "hold" {
  ml_hold `0'
  exit
}
}

```

```

- if ``subcmd''' == "unhold" {
= if "display"' == "unhold" {
  ml_unhold `0'
  exit
}
- if ``subcmd''' == "score" {
= if "display" == "score" {
  if _caller() > 12 {
    `vv' Score `0'
  }
  else {
    Score `0'
  }
  exit
}
- if ``subcmd''' == "count" {
= if "display" == "count" {
  Count `0'
  exit
}
- if ``subcmd''' == "trace" {
= if "display" == "trace" {
  Trace `0'
  exit
}
- if ``subcmd''' == bsubstr("graph",1,max(2,`lcmd')) {
= if "display" == bsubstr("graph",1,max(2,7)) {
  Graph `0'
  exit
}
- if ``subcmd''' == "init" {
= if "display" == "init" {
  Interactive 0
  `vv' Init `0'
  exit
}
- if ``subcmd''' == bsubstr("maximize",1,max(3,`lcmd')) {
= if "display" == bsubstr("maximize",1,max(3,7)) {
  Interactive 1
  `vv' MaxMin maximize `0'
  exit
}
- if ``subcmd''' == bsubstr("minimize",1,max(3,`lcmd')) {
= if "display" == bsubstr("minimize",1,max(3,7)) {
  Interactive 1
  `vv' MaxMin minimize `0'

```

```

        exit
    }
    - if "`subcmd'" == bsubstr("display",1,max(2,`lcmd')) {
    = if "display" == bsubstr("display",1,max(2,7)) {
    - Display `0'
    = Display , level(95) nofootnote
}
> ----- begin mopt.Disp
> lay -----
    - if !inlist("`e(opt)'", "ml", "moptimize") {
    = if !inlist("moptimize", "ml", "moptimize") {
        error 301
    }
    - if "`e(svyml)'" != "" | "`e(prefix)'" == "svy" {
    = if "" != "" | "" == "svy" {
        _prefix_display `0'
        exit
    }
    - syntax [, Level(cilevel) Plus noHeader First SHOWEQns NEQ(intege
> r -1) noFOOTnote NOSKIP Title(passthru) NOTEST * ]
    - if "`plus'" != "" {
    = if "" != "" {
        local footnote nofootnote
    }
    - if `neq' >= 0 {
    = if -1 >= 0 {
        local neqopt neq(`neq')
    }
    - _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 ALLBASEl
> evels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYce
> lls NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string
> ) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVW

```

```

> RAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOA
> BBREV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASELe
> vels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTY
> cells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND c
> format(string) 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')
}
- Wraon , `fvwraon'
= Wraon ,

```

```

> _____ begin _get_diopts.Wra
> pon _____

```

```

- capture syntax [, WOrd WIdth]
- if c(rc) {
  di as err "invalid fwwrapon() option;"
  syntax [, WOrd 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, `fwwrap'
      }
      - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `bas
> elevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coef
> legend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare'
> `fwwrap' `fwwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbre
> v'
      = 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_k = s(k)
- }
- local S_`sname' = s(`sname')
= local S_k_fill = s(k_fill)
- }
- local S_`sname' = s(`sname')
= local S_before = s(before)
- }
- local S_`sname' = s(`sname')
= local S_after = s(after)
- }
- local S_`sname' = s(`sname')
= local S_optlist = s(optlist)
- }
- local S_`sname' = s(`sname')
= local S_namelist = s(namelist)
- }
- local S_`sname' = s(`sname')
= local S_deriv = s(deriv)
- }
- local S_`sname' = s(`sname')
= local S_nolog = s(nolog)

```



```

- }
- sreturn clear
- GetDiopts DEFAULT

```

```

> ----- begin _get_diopts.GetDio
> pts -----
        - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBASEl
> evels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYce
> lls NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string
> ) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVW
> RAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOA
> BBREV ABBREV
        - syntax namelist(max=2) [, `DIOPTS' *]
        = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEle
> vels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTY
> cells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND c
> format(string) 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')
    }
    - 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' `bas
> elevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coef
> legend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare'
> `fvwrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbre
> v'

= 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 ALLBASEL
> evels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTYce
> lls NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(string
> ) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) FVW
> RAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) NOA
> BBREV ABBREV

- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASEle
> vels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTY
> cells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND c
> format(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(

```

```

> passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPVALUES MARKDOWN CITYTYPE(
> 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')
    }
- 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 " " 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' `bas
> elevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `coef
> legend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compare'
> `fvwrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abbre
> v'
    = 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 k = `17''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local k_fill = `17''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local before = `, argoptlist(CLister) optlist(OIM OPG
> Robust) old ''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local after = `, vce(robust)''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local optlist = `CLuster''
- }
- sreturn local `sname' = ``S_`sname''''
= sreturn local namelist = `cluster''

```

```

- }
- sreturn local `sname' = ``S_`sname''
= sreturn local deriv = ``0''
- }
- sreturn local `sname' = ``S_`sname''
= sreturn local nolog = ``nolog''
- }

```

```

> _____ end _get_dio
> pts _____
    - _get_eformopts , eformopts(`options') soptions allowed(__all__)
    = _get_eformopts , eformopts() soptions allowed(__all__)

```

```

> _____ 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 TRatio * ]
    - local opts ``options''
    = local opts ``''
    - local 0 , `eform1' `eform' `nohr' `hr' `noshr' `shr' `or' `irr
> ' `rrr' `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
> TRatio ]
- 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
- }
- 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 eform `s(eform)`
    = local eform
    - local diparm `s(options)`
    = local diparm
    - _get_diparmopts , diparmopts(`diparm`) 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
> SYNTAXonly * ]
- 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 _____
    - if "`header'"=="" {
    = if ""=="" {
    - _coef_table_header, `title'
    = _coef_table_header,

> _____ begin _coef_table_he
> der _____
    - version 9
    - if !c(noisily) {
    exit
    }

```

```

- syntax [, rclass noHeader obj(string) head1left(numlist max=1
> integer >=0) head1right(numlist max=1 integer >=0) head2left(numlist max=1 i
> nteger >=0) head2right(numlist max=1 integer >=0) head2start(numlist max=1 i
> nteger >=1) width(numlist max=1 integer >=1) * ]
- if ("`header'" != "") exit
= if (" " != " ") exit
- if ("`head2start'" != "" & "`width'" != "") {
= if (" " != " " & " " != " ") {
    opts_exclusive "head2start() width()"
    exit 198
}
- local has_obj = (`"`obj'" != "")
= local has_obj = (`"" != " ")
- if (`has_obj' == 0) {
= if (0 == 0) {
- tempname obj
- }
- if ("`rclass'" != "") {
= if (" " != " ") {
    tempname rhold
    _return hold `rhold'
}
- `.obj' = . _coef_table__header.new, `rclass' `options'
= . _000005 = . _coef_table__header.new,
- class _coef_table__header {
-     string result          = "e"
-     string newline         = "display"
-     double opt_dvheader    = 0
-     double opt_modeltest   = 0
-     double opt_cluster     = 0
-     double opt_rules       = 0
-     double opt_tvar        = 0
-     double opt_septitle    = 0
-     double linesize        = 0
-     string title           = ""
-     string title2          = ""
-     string groupvar        = ""
-     string intmethod       = ""
-     double intpoints       = -1
-     string cmd             = ""
-     string cmd2            = ""
-     string prefix          = ""
-     double is_prefix       = 0
-     double is_svy          = 0
-     double is_margins      = 0
-     double is_gsem         = 0

```

```

-      double is_gsem_group    = 0
-      double is_sem           = 0
-      double is_areg          = 0
-      double is_demand        = 0
-      double is_demandn       = 0
-      double is_bacon         = 0
-      double is_htedid        = 0
-      double is_xtvar         = 0
-      double is_cfreg         = 0
-      double is_sum           = 0
-      double is_tab           = 0
-      double is_r2_p          = 1
-      double is_multclust     = 0
-      double is_stintcox      = 0
-      double is_stmgintcox    = 0
-      double is_bmareg       = 0
-      double prop_pushtitle   = 0
-      double prop_groupvar    = 0
-      double prop_censlimits  = 0
-      double prop_censored    = 0
-      double prop_nuncensored      = 0
-      double prop_iselected   = 0
-      double prop_selected    = 0
-      double prop_groups      = 0
-      double prop_balance     = 0
-      double ablen            = 16
-      double width            = 78
-      double c1               = 1
-      double c2               = 16
-      double c2a              = 16
-      double c3               = 49
-      double c4               = 67
-      double c2wfmt           = 10
-      double c4wfmt           = 10
-      string  scheme          = "ml"
-      double lalignr          = 0
-      double ralignr          = 0
-      double lalignl          = 0
-      double ralignl          = 0
-      double lalignla         = 0
-      double ralignla         = 0
-      array   left
-      array   right
-}

```

> _____ begin _coef_table__header.

```

> new
    - version 16
    - syntax [, rclass noDVHeader noMODELtest TITLe(string asis) n
> ocluster noRULES noTVAR SEPTITLE linesize(int 0) ]
    - if "`rclass'" != "" {
= if "" != "" {
    .result = "r"
    }
    - if "`dvheader'" == "" {
= if "" == "" {
    - .opt_dvheader = 1
    - }
    - if "`modeltest'" == "" {
= if "" == "" {
    - .opt_modeltest = 1
    - }
    - if "`cluster'" == "" {
= if "" == "" {
    - .opt_cluster = 1
    - }
    - if "`rules'" == "" {
= if "" == "" {
    - .opt_rules = 1
    - }
    - if "`tvar'" == "" {
= if "" == "" {
    - .opt_tvar = 1
    - }
    - if "`septitle'" != "" {
= if "" != "" {
    .opt_septitle = 1
    }
    - .linesize = `linesize'
    = .linesize = 0
    - .title = ``title'""
    = .title = `""'
    - if ```.result'"" == "e" {
= if `e'"" == "e" {
    - .New4e

```

```

> _____ begin _coef_table__header.Ne

```

```

> w4e
    - version 16
    - .cmd = ``e(cmd)'"
    = .cmd = "poisson"
    - .cmd2 = ``e(cmd2)'"

```

```

= .cmd2 = ""
- .prefix = "`e(prefix)'"
= .prefix = ""
- if ```.prefix''' != "" {
= if `''' != "" {
    .is_prefix = 1
  }
- if ``.prefix'" == "svy" {
= if "" == "svy" {
    .is_svy = 1
  }
- if ``.cmd'" == "margins" {
= if "poisson" == "margins" {
    .is_margins = 1
  }
- if inlist(``.cmd'", "contrast", "pwcompare") {
= if inlist("poisson", "contrast", "pwcompare") {
    if ``.cmd2'" == "margins" {
      .is_margins = 1
    }
  }
- if ``.cmd'" == "gsem" | ``.cmd2'" == "gsem" {
= if "poisson" == "gsem" | "" == "gsem" {
    .is_gsem = 1
  }
- if ``.cmd'" == "sem" {
= if "poisson" == "sem" {
    .is_sem = 1
  }
- if ``.cmd'" == "areg" {
= if "poisson" == "areg" {
    .is_areg = 1
  }
- if (``.cmd'" == "demandsys" & ```e(doheader)'''=="yes") {
= if ("poisson" == "demandsys" & `'''=="yes") {
    .is_demand = 1
  }
- if (``.cmd'" == "demandsys") {
= if ("poisson" == "demandsys") {
    .is_demandn = 1
  }
- if (``.cmd'" == "xthdidregress" & ``e(inestat)'''=="") {
= if ("poisson" == "xthdidregress" & ""=="") {
    .is_htedid = 1
  }
- if (``.cmd'" == "xtvar") {

```



```

= if ("poisson" == "xtvar") {
  .is_xtvar = 1
}
- if ("`.cmd'" == "cfregress") {
= if ("poisson" == "cfregress") {
  .is_cfreg = 1
}
- if ("`.cmd'" == "stintcox" {
= if "poisson" == "stintcox" {
  .is_stintcox = 1
}
- if ("`.cmd'" == "stmgintcox" {
= if "poisson" == "stmgintcox" {
  .is_stmgintcox = 1
}
- if ("`.cmd'" == "bdecomp" {
= if "poisson" == "bdecomp" {
  .is_bacon = 1
}
- if ("`.cmd'" == "bmaregress" {
= if "poisson" == "bmaregress" {
  .is_bmareg = 1
}
- is_svysum `.cmd'
= is_svysum poisson

```

```

> ----- begin is_svy
> sum -----
  - version 9
  - local SvySum mean prop proportion ratio total
  - if `"'0'" != "" & `:list 0 in SvySum' {
= if `"poisson"' != "" & 0 {
  return scalar is_svysum = 1
}
- else return scalar is_svysum = 0

```

```

> ----- end is_svy
> sum -----
  - .is_sum = r(is_svysum)
  - if `.is_svy' {
= if 0 {
  if "`.cmd'" == "tabulate" {
    .is_tab = 1
  }
}
- if `.is_gsem' {

```

```

= if 0 {
  if ``e(groupvar)'' != "" & ``.cmd2'' == "" {
    .is_gsem_group = 1
  }
}
- local clustvars `e(clustvar)'
= local clustvars
- local kclvar : list sizeof clustvars
- .is_multclust = ((`kclvar'>1) & ("`e(kcluster)'"=="matrix"

> ))

= .is_multclust = ((0>1) & ("=="matrix"))
- if `.is_multclust' {
= if 0 {
  .prop_pushtitle = 1
}
- local c2pfix 20
- if `.is_sem' | `.is_gsem_group' {
= if 0 | 0 {
  local crlen ``e(crittype)''
  local crlen : smcllength local crlen
  if `crlen' >= `c2pfix' {
    local c2pfix = `crlen' + 2
  }
}
- if `.is_sum' {
= if 0 {
  if `.linesize' > 62 {
    local offset = `.linesize' - 62
  }
  else local offset 0
  .width = 62 + `offset'
  .c2 = 18
  .c3 = 35 + `offset'
  .c4 = 51 + `offset'
  .c2wfmt = 7
  .c4wfmt = 10
  .scheme = "compact"
}
- else if `.is_prefix' | `.is_sem' | `.is_gsem_group' {
= else if 0 | 0 | 0 {
  .width = 78
  .c2 = `c2pfix'
  .c3 = 49
  .c4 = 67
  .c2wfmt = 9
  .c4wfmt = 10

```

```

        .scheme = "svy"
    }
- if `.is_svy' {
= if 0 {
    local maxlen = `.width' - `.c2' - (`.c2wfmt' + 2) - 2 - (`.c4' - `.c3'
> ) - 2
    if `maxlen' > 19 {
    local maxlen 19
    }
    local len : display %`maxlen'.0fc e(N_pop)
    local len : list retok len
    local len : length local len
    local ++maxlen
    local ++len
    if `.c4wfmt' <= `len' & `len' <= `maxlen' {
    .c3 = `.c3' + `.c4wfmt' - `len'
    .c4 = `.c4' + `.c4wfmt' - `len'
    .c4wfmt = `len'
    }
    }
}

> _____ end _coef_table__header.Ne
> w4e _____
    - }
    - else {
    .New4r
    }
    - local props ```.Macro _coef_table_props'''
}

> _____ begin _coef_table__header.Ma
> cro _____
    - version 16
    - class exit `""`.result'(`0')'''
    = class exit `""'
}

> _____ end _coef_table__header.Ma
> cro _____
    = local props `""'
    - if `.is_sem' | `.is_gsem_group' | `:list posof "pushtitle" i
> n props' {
    = if 0 | 0 | 0 {
    .prop_pushtitle = 1
    }
    - if (`.is_demand' | `.is_htedid' | `.is_xtvar') {
    = if (0 | 0 | 0) {
    .prop_pushtitle = 1

```

```

    }
    - if (`is_bmareg') {
= if (0) {
    .prop_pushtitle = 1
    }
    - if `is_sem' | `is_gsem_group' | `:list posof "groupvar" in
> props' {
    = if 0 | 0 | 0 {
    .prop_groupvar = 1
    }
    - local list tobit cpoisson ivtobit metobit xttobit
    - local list2 metobit
    - if `:list posof ``.cmd'" in list' | `:list posof ``.cmd2'" i
> n list2' | `:list posof "censlimits" in props' {
    = if 0 | 0 | 0 {
    .prop_censlimits = !missing(e(N_unc))
    }
    - local list intreg tobit cpoisson ivtobit eintreg xtintreg xt
> tobit
    - local list2 meintreg metobit stintreg
    - if `:list posof ``.cmd'" in list' | `:list posof ``.cmd2'" i
> n list2' | `:list posof "censored" in props' {
    = if 0 | 0 | 0 {
    .prop_censored = !missing(e(N_unc))
    }
    - if `:list posof "nuncensored" in props' {
    = if 0 {
    .prop_nuncensored = 1
    }
    - local list eintreg
    - if `:list posof ``.cmd'" in list' | `:list posof "iselected"
> in props' {
    = if 0 | 0 {
    .prop_iselected = 1
    }
    - local list eregress eprobit eoprobit
    - if `:list posof ``.cmd'" in list' | `:list posof "selected"
> in props' {
    = if 0 | 0 {
    .prop_selected = 1
    }
    - if `:list posof "groups" in props' {
    = if 0 {
    .prop_groups = 1
    .groupvar = ``e(repanvar)'"
    .intmethod = ``e(reintmethod)'"

```

```

        .intpoints = e(reintpoints)
    }
    - local list xtreg xtprobit xtlogit xtcloglog xtfrontier xtint
> reg xtnbreg
    - local list `list' xtpoisson xttobit xtmlogit xtrc xtvar
    = local list xtreg xtprobit xtlogit xtcloglog xtfrontier xtint
> reg xtnbreg xtpoisson xttobit xtmlogit xtrc xtvar
    - local list2 xtlogit xtprobit xtologit xtmlogit
    - if `:list posof ``.cmd' in list' | `:list posof ``.cmd2' i
> n list2' {
    = if 0 | 0 {
        .prop_groups = 1
        .groupvar = ``e(ivar)''
        .intmethod = ``e(intmethod)''
        .intpoints = e(n_quad)
    }
    - if ``.cmd' == "xtreg" | ``.cmd2' == "xtlogit" {
    = if "poisson" == "xtreg" | "" == "xtlogit" {
        .is_r2_p = 0
    }
    - if `:list posof "balance" in props' {
    = if 0 {
        .prop_balance = 1
    }
    - .left = {}
    - .right = {}
}

> _____ end _coef_table__header.
> new _____
    - if (``rclass' != "") {
    = if ("" != "") {
        _return restore `rhold'
        .`obj'.New4r
    }
    - if (``head1left' != "") {
    = if ("" != "") {
        .`obj'.lalignla = `head1left'
    }
    - if (``head1right' != "") {
    = if ("" != "") {
        .`obj'.ralignla = `head1right'
    }
    - if (``head2left' != "") {
    = if ("" != "") {
        .`obj'.lalignr = `head2left'
    }
}

```

```

- if ("`head2right'" != "") {
= if ("" != "") {
    .`obj'.ralignr = `head2right'
}
- if ("`width'" != "") {
= if ("" != "") {
    .`obj'.width = `width'
}
- .`obj'.build
= .__000005.build

```

```

> _____ begin _coef_table__header.bu
> ild _____
    - version 16
    - if ```.title''' == "" {
    = if `''' == "" {
    - .title = ```.Macro title'''

```

```

> _____ begin _coef_table__header.Ma
> cro _____
    - version 16
    - class exit ````result'(`0')'''
    = class exit `"Poisson regression"'

```

```

> _____ end _coef_table__header.Ma
> cro _____
    = .title = `"Poisson regression"'
    - .title2 = ```.Macro title2'''

```

```

> _____ begin _coef_table__header.Ma
> cro _____
    - version 16
    - class exit ````result'(`0')'''
    = class exit `'''

```

```

> _____ end _coef_table__header.Ma
> cro _____
    = .title2 = `'''
    - }
    - if `.is_areg' {
    = if 0 {
        local absvar ``e(absvar)''
        if `:list sizeof absvar' > 1 {
        .title2 = ""
        }
        else {

```

```

        local absvar = abbrev("`absvar'", 22)
        .title2 = "Absorbed variable: {res:`absvar'}"
    }
}
- if `.prop_pushtitle' {
= if 0 {
    if `.is_svy' & ("`.cmd'" == "clogit" | "`.cmd'" == "xtmlogit"
> " | "`.cmd2'" == "xtmlogit") {
    }
    else if `.is_stintcox' | `.is_bmareg' {
        .PushLeft "`".title'"'"
        .PushLeft "`".title2'"'"
        .title = ""
        .title2 = ""
    }
    else if `.is_stmgintcox' {
        if ("`e(strata)'" == "") {
            .PushLeft "`".title'"'"
            .PushLeft "`".title2'"'"
            .title = ""
            .title2 = ""
        }
        else {
            .PushLeft "`".title2'"'"
            .title2 = ""
        }
    }
    else {
        .PushLeft "`".title'"'"
        if `.is_areg' {
            .PushLeft "`".title2'"'"
        }
        .title = ""
        .title2 = ""
    }
}
- if `.prop_groupvar' | `.is_gsem_group' {
= if 0 | 0 {
    local groupvar "`".Macro groupvar'"'"
    if ("`groupvar'" != "" & `.is_stintcox') {
        local s = cond(`:word count `groupvar'>1, "s", "")
        local grpvar = abbrev("`groupvar'",`.ablen')
        .PushLeft "Strata variable`s': {res}`grpvar'"
    }
    else if "`groupvar'" != "" {
        local grpvar = abbrev("`groupvar'",`.ablen')

```

```

        .PushLeft "Grouping variable: {res}`grpvar'"
    }
}
- if (`.is_stintcox') {
= if (0) {
    .PushLeft ""
    local ndep = e(k_dv)
    if (`ndep'==2) {
        tokenize `e(depvar)'
        local ltime `1'
        local rtime `2'
        .PushLeft "Event-time interval:"
        .PushLeft "  Lower endpoint: {res}`ltime'"
        .PushLeft "  Upper endpoint: {res}`rtime'"
        if "`e(strata)'" != "" {
            .PushLeft ""
        }
    }
    else {
        .PushLeft "ID variable: {res}`e(id)'"
        .PushLeft "Examination time: {res}`e(depvar)'"
        .PushLeft "Status indicator: {res}`e(status)'"
    }
}
- if (`.is_stmgintcox') {
= if (0) {
    .PushLeft ""
    .PushLeft "ID variable: {res}`e(id)'"
    .PushLeft "Event variable: {res}`e(event)'"
    local ndep = e(k_dv)
    if (`ndep'==2) {
        tokenize `e(depvar)'
        local ltime `1'
        local rtime `2'
        .PushLeft "Event-time interval:"
        .PushLeft "  Lower endpoint: {res}`ltime'"
        .PushLeft "  Upper endpoint: {res}`rtime'"
        if "`e(strata)'" != "" {
            .PushLeft ""
        }
    }
    else {
        .PushLeft "Examination time: {res}`e(depvar)'"
        .PushLeft "Status indicator: {res}`e(status)'"
    }
}
}

```



```

- if `.is_svy' {
= if 0 {
  .BuildLeft4svy
}
- else if `.is_prefix' {
= else if 0 {
  if !`.is_margins' {
    .Build4strata
  }
}
- if (`.is_tab' | `.is_sum') & !missing(`.result'(N_stdize)) {
= if (0 | 0) & !missing(e(N_stdize)) {
  .PushLeft "N. of std strata" " = " "as res" %`.c2wfmt'.0fc `
> .result'(N_stdize)
}
- .Build4tvar
}

> _____ begin _coef_table__header.Build4t
> var _____
- version 16
- if `.opt_tvar' == 0 {
= if 1 == 0 {
  exit
}
- if "`.Macro tvar'" == "" {
}

> _____ begin _coef_table__header.Ma
> cro _____
- version 16
- class exit ```.result'(`0')'''
= class exit `'''

> _____ end _coef_table__header.Ma
> cro _____
= if "" == "" {
- exit
}

> _____ end _coef_table__header.Build4t
> var _____
- if (`.is_stmgintcox') {
= if (0) {
  .PushRight "Number of events" " = " "as res" %`.c4wfmt'.0fc
> e(N_events)
  .PushRight "Number of subjects" " = " "as res" %`.c4wfmt'.0f
> c e(N_sub)
}

```

```

- if (!missing(`.Scalar N') & `.is_demandn'==0 & !`.is_bmareg'
> ) {
    _____
> _____ begin _coef_table__header.Sca
> lar _____
    - version 16
    - class exit ```= `.result'(`0')''''
    = class exit ``217'''
    _____
> _____ end _coef_table__header.Sca
> lar _____
    = if (!missing(217) & 0==0 & !0) {
    - .PushRight "Number of obs" " = " "as res" %`.c4wfmt'.0fc `.S
> calar N'
    _____
> _____ begin _coef_table__header.Sca
> lar _____
    - version 16
    - class exit ```= `.result'(`0')''''
    = class exit ``217'''
    _____
> _____ end _coef_table__header.Sca
> lar _____
    = .PushRight "Number of obs" " = " "as res" %10.0fc 217
    _____
> _____ begin _coef_table__header.PushRi
> ght _____
    - version 16
    - gettoken tok rest : 0
    - gettoken eq rest : rest
    - gettoken dias rest : rest
    - if `:length local rest' {
    = if 12 {
    - local direst = ustrltrim("` : di `rest'")
    = local direst = ustrltrim("      217")
    - .ralignr = max(`.ralignr', udstrlen("`direst'"))
    = .ralignr = max(0, udstrlen("217"))
    - local len : di ```tok''''
    = local len : di ``"Number of obs"'
    - local len : smcllength local len
    - .lalignr = max(`.lalignr', `len')
    = .lalignr = max(0, 13)
    - .right.Arrpush `\.C3' ``{lalign `\.lalignr':`tok'}'' `\.C4
> ' ``eq'' `dias' "{ralign `\.ralignr':`direst'}"
    = .right.Arrpush `\.C3' ``{lalign `\.lalignr':Number of obs}''
> `\.C4' " = " as res "{ralign `\.ralignr':217}"

```

```

- }
- else if `:length local tok' {
= else if 13 {
    .right.Arrpush \`.C3' ``tok'''
    }
- else {
    .right.Arrpush
    }
}

> _____ end _coef_table__header.PushRi
> ght _____

- }
- if `.is_demandn' {
= if 0 {
    if (e(N_estat_dsys)==.|``e(N_estat_dsys)''==''') {
    .PushRight "Number of obs" " = " "as res" %`.c4wfmt'.0fc `.S
> calar N'
    }
    else {
    .PushRight "Number of obs" " = " "as res" %`.c4wfmt'.0fc e(N
> _estat_dsys)
    }
    }
- if `.is_demand' {
= if 0 {
    .PushRight "Number of goods" " = " "as res" %`.c4wfmt'.0fc e
> (n_goods)
    .PushLeft "Expenditure variable: {result:``e(expenditures)}"
    }
- if `.is_demand' {
= if 0 {
    if (``e(piconstant)''!=''') {
    .PushRight "Price index constant" " = " "as res" %`.c2wfmt'.
> 4gc e(piconstant)
    .PushLeft ""
    }
    if (e(n_demos)>0) {
    local scaling = strproper("``e(demo_type)''")
    .PushLeft ``Demographic method: {text:``scaling'}"
    .PushRight "Number of demographics" " = " "as res" %`.c2wfmt
> '.0fc e(n_demos)
    }
    .PushLeft ""
    local rsquare = strproper("``e(r2_type)''")
    .PushLeft ``rsquare' R2 of model for"
    local kdem = e(k_eq_old) + 1

```

```

local labels ``e(labels)'''
forvalues i=1/`kdem' {
  if (``labels'=='') {
    .PushLeft " Good `i'" " = " "as res" %`.c2wfmt'.4f e(r2_`i'
> )

  }
  else {
    local xdm: word `i' of `labels'
    .PushLeft " `xdm'" " = " "as res" %`.c2wfmt'.4f e(r2_`i')
  }
}
.PushLeft ""
}
- if `.is_bacon' {
= if 0 {
  .PushLeft "ATET" " = " "as res" %`.c2wfmt'.0g `e(atet)'
  .PushRight "Number of groups" " = " "as res" %`.c4wfmt'.0fc
> `e(N_groups)'
  .PushRight "Number of cohorts" " = " "as res" %`.c4wfmt'.0fc
> `e(N_timing)'
}
- if `.is_bmareg' {
= if 0 {
  local heredlab = proper("`e(heredity)')')
  .PushLeft "`e(sampling_lab)'"
  if "`heredlab'" != "" {
    .PushLeft "Heredity: {txt}`heredlab'"
    if !missing(e(burnin)) {
      .PushLeft ""
    }
  }
  else if !missing(e(burnin)) {
    .PushLeft ""
    .PushLeft ""
  }
  .PushLeft ""
  .PushLeft "Priors:"
  .PushLeft " Models: {txt}`e(mprior_lab)'"
  .PushLeft " Cons.: Noninformative"
  .PushLeft " Coef.: Zellner's g"
  .PushLeft " g: {txt}`e(gprior_lab)'"
  .PushLeft " sigma2: Noninformative"
  if !missing(e(burnin)) {
    .PushLeft
    .PushLeftRalign "{help j_bmaregress_sampcorr:Sampling correl
> ation}" " = " "as res" %`.c4wfmt'.4f e(sampcorr)

```

```

    }
    .PushRight "No. of obs" " = " "as res" %`.c4wfmt'.0fc `.Scal
> ar N'
    .PushRight "No. of predictors" " = " "as res" %`.c4wfmt'.0fc
> e(p)
    .PushRightRalign "Groups" " = " "as res" %`.c4wfmt'.0fc e(p_
> groups)
    .PushRightRalign "Always" " = " "as res" %`.c4wfmt'.0fc e(p_
> always)
    .PushRight "No. of models" " = " "as res" %`.c4wfmt'.0fc e(k
> _models)
    .PushRightRalign "For CPMP >= .9" " = " "as res" %`.c4wfmt'.
> 0fc e(k_models_cpmp)
    .PushRight "Mean model size" " = " "as res" %`.c4wfmt'.3f e(
> msize_mean)
    if !missing(e(burnin)) {
    .PushRight "Burn-in" " = " "as res" %`.c4wfmt'.0fc e(burnin)
    .PushRight "MCMC sample size" " = " "as res" %`.c4wfmt'.0fc
> e(mcmcsample)
    .PushRight "Acceptance rate" " = " "as res" %`.c4wfmt'.4f e(
> arate)
    }
    else {
    if "`heredlab'" != "" {
    .PushRight
    }
    .PushRight
    }
    if !missing(e(shrinkage)) {
    .PushRight "Shrinkage, g/(1+g)" " = " "as res" %`.c4wfmt'.4f
> e(shrinkage)
    }
    else {
    .PushRight ""
    }
    if (e(sigma2) < 1e-3) {
    local sigfmt 8.4g
    }
    else {
    local sigfmt `.c4wfmt'.3f
    }
    .PushRight "Mean sigma2" " = " "as res" %`sigfmt' e(sigma2)
    }
    - if `.is_htedid' {
    = if 0 {
    .PushRight "Number of panels" " = " "as res" %`.c4wfmt'.0fc

```

```

> e(N_panels)
    }
    - if !missing(`.Scalar N_sub') & `.is_stintcox' {
    _____
> _____ begin _coef_table__header.Sca
> lar _____
    - version 16
    - class exit ``'= `.result'(`0')''''
    = class exit `".'"'
    _____
> _____ end _coef_table__header.Sca
> lar _____
    = if !missing(.) & 0 {
    .PushRight "Number of subjects" " = " "as res" %`.c4wfmt'.0fc
> c `Scalar N_sub'
    }
    - if `.is_areg' {
    = if 0 {
    local absvar "`e(absvar)'"
    if `:list sizeof absvar' == 1 {
    .PushRight "No. of categories" " = " "as res" %`.c4wfmt'.0fc
> e(k_absorb)
    }
    }
    - if `.is_sem' | `.is_gsem_group' {
    = if 0 | 0 {
    if ```.Macro groupvar'' != "" {
    .PushRight "Number of groups" " = " "as res" %`.c4wfmt'.0fc
> `Scalar N_groups'
    }
    }
    - if `.is_sem' & !`.is_svy' {
    = if 0 & !0 {
    .PushLeft `"Estimation method: {res:`.Macro method}'"
    }
    - if `.prop_selected' {
    = if 0 {
    .Build4selected
    }
    - if `.is_svy' {
    = if 0 {
    .BuildRight4svy
    if `.prop_censored' {
    .PushRight
    .Build4censobs "header"
    .PushRight

```

```

    }
  }
  - else {
  - if `.is_margins' {
= if 0 {
    .Build4subpop
  }
  - if `.prop_iselected' {
= if 0 {
    .Build4title
    .Build4iselected
  }
  - if `.prop_censored' {
= if 0 {
    .Build4title
    .Build4censobs
    if !`.is_prefix' {
    .PushRight
    }
  }
  - .Build4cluster
}

> _____ begin _coef_table__header.Build4clus
> ter _____
    - version 16
    - if `.is_multclust' {
= if 0 {
    .Build4mclust
    exit
  }
  - if missing(`.Scalar N_clust') {
    _____

> _____ begin _coef_table__header.Sca
> lar _____
    - version 16
    - class exit ```= `.result'(`0')''
    = class exit ``.``

> _____ end _coef_table__header.Sca
> lar _____
    = if missing(.) {
    - exit
    _____

> _____ end _coef_table__header.Build4clus
> ter _____
    - .Build4reps

```

```

> _____ begin _coef_table__header.Build4r
> eps _____
      - version 16
      - if !missing(`.Scalar N_reps') {
        _____
> _____ begin _coef_table__header.Sca
> lar _____
      - version 16
      - class exit ```= `.result'(`0')''''
      = class exit `".'"'
        _____
> _____ end _coef_table__header.Sca
> lar _____
      = if !missing(.) {
        .PushRight "Replications" " = " "as res" %`.c4wfmt'.0fc `.`
> Scalar N_reps'
      if `prop_censored' & !`.is_svy' {
        .PushRight
      }
    }
    _____
> _____ end _coef_table__header.Build4r
> eps _____
      - }
      - if `prop_groups' {
      = if 0 {
        .Build4groups
      }
      - if `is_stintcox' & ``e(strata)'' != "" {
      = if 0 & "" != "" {
        if `ndep' == 2 {
          .PushRight
        }
      }
      - .Build4modeltest
        _____
> _____ begin _coef_table__header.Build4modelt
> est _____
      - version 16
      - if ``.result'' != "e" {
      = if "e" != "e" {
        exit
      }
      - if `opt_modeltest' == 0 {
      = if 1 == 0 {

```



```

        exit
      }
      - if `""`.Macro k_eq_model'''' == "0" {
        _____
> _____ begin _coef_table__header.Ma
> cro _____
        - version 16
        - class exit `""`.result'(`0')''''
        = class exit `1''
        _____
> _____ end _coef_table__header.Ma
> cro _____
        = if `1'' == "0" {
          exit
          }
        - if missing(`.Scalar df_m') {
          _____
> _____ begin _coef_table__header.Sca
> lar _____
        - version 16
        - class exit `""` = `.result'(`0')''''
        = class exit `9''
        _____
> _____ end _coef_table__header.Sca
> lar _____
        = if missing(9) {
          exit
          }
        - if `""`.Macro chi2'''' != "" | `""`.Macro df_r'''' == "" {
          _____
> _____ begin _coef_table__header.Ma
> cro _____
        - version 16
        - class exit `""`.result'(`0')''''
        = class exit `",''
        _____
> _____ end _coef_table__header.Ma
> cro _____
        _____
> _____ begin _coef_table__header.Ma
> cro _____
        - version 16
        - class exit `""`.result'(`0')''''
        = class exit `""'
        _____
> _____ end _coef_table__header.Ma

```

```

> cro
    = if `".'" != "" | `""' == "" {
    - .Build4chi2test

>
> est
    - version 16
    - local type `e(chi2type)'
    = local type Wald
    - if ``type'' == "" {
    = if `Wald' == "" {
      local type Wald
    }
    - local didf = strtrim("` : di e(df_m)')
    = local didf = strtrim("9")
    - if !missing(e(chi2)) {
      .PushRight ``type' chi2({res:`didf'})" " = " "as res" %`
> .c4wfmt'.2f e(chi2)
      .PushRight "Prob > chi2" " = " "as res" %`.c4wfmt'.4f ch
> i2tail(e(df_m),e(chi2))
    }
    - else {
    - local j_robust "{help j_robustsingular##|_new:`type' chi
> 2(`didf')}"
    = local j_robust "{help j_robustsingular##|_new:Wald chi2(
> 9)}"
    - .PushRight ``j_robust'" " = " "as res" %`.c4wfmt's ".
    = .PushRight "{help j_robustsingular##|_new:Wald chi2(9)}"
> " = " "as res" %10s ".

>
> ght
    - version 16
    - gettoken tok rest : 0
    - gettoken eq rest : rest
    - gettoken dias rest : rest
    - if `:length local rest' {
    = if 9 {
    - local direst = ustrltrim("` : di `rest'")
    = local direst = ustrltrim("          .")
    - .ralignr = max(`.ralignr', udstrlen("`direst'))
    = .ralignr = max(3, udstrlen("."))
    - local len : di ``tok'
    = local len : di ``{help j_robustsingular##|_new:Wald ch
> i2(9)}"
    - local len : smcllength local len

```

```

        - .lalignr = max(`.lalignr', `len')
        = .lalignr = max(13, 12)
        - .right.Arrpush `'.C3' ``{lalign `'.lalignr':`tok'}`" \
> `.C4' ``eq' ``dias' "{ralign `'.ralignr':`direst'}`"
        = .right.Arrpush `'.C3' ``{lalign `'.lalignr':{help j_robu
> stsingular##_new:Wald chi2(9)}}" `'.C4' " = " as res "{ralign `'.ralignr':.}
> "

        - }
        - else if `:length local tok' {
        = else if 43 {
            .right.Arrpush `'.C3' ``tok'
        }
        - else {
            .right.Arrpush
        }

_____.
> _____ end _coef_table__header.PushRi
> ght _____

        - .PushRight "Prob > chi2" " = " "as res" %`.c4wfmt's ".
        = .PushRight "Prob > chi2" " = " "as res" %10s ".

_____.
> _____ begin _coef_table__header.PushRi
> ght _____

        - version 16
        - gettoken tok rest : 0
        - gettoken eq rest : rest
        - gettoken dias rest : rest
        - if `:length local rest' {
        = if 9 {
            - local direst = ustrltrim("`di `rest'")
            = local direst = ustrltrim(" ")
            - .ralignr = max(`.ralignr', udstrlen("`direst'"))
            = .ralignr = max(3, udstrlen(" "))
            - local len : di ``tok'
            = local len : di ``Prob > chi2'
            - local len : smcllength local len
            - .lalignr = max(`.lalignr', `len')
            = .lalignr = max(13, 11)
            - .right.Arrpush `'.C3' ``{lalign `'.lalignr':`tok'}`" \
> `.C4' ``eq' ``dias' "{ralign `'.ralignr':`direst'}`"
            = .right.Arrpush `'.C3' ``{lalign `'.lalignr':Prob > chi2}
> ``.C4' " = " as res "{ralign `'.ralignr':.}"
            - }
            - else if `:length local tok' {
            = else if 11 {
                .right.Arrpush `'.C3' ``tok'

```

```

        }
        - else {
            .right.Arrpush
        }
    }
}

> _____ end _coef_table__header.PushRi
> ght _____
    - }
    }
}

> _____ end _coef_table__header.Build4chi2t
> est _____
    - }
    - else {
        .Build4Ftest
    }
}

> _____ end _coef_table__header.Build4modelt
> est _____
    - .Build4r2
    }
}

> _____ begin _coef_table__header.Build
> 4r2 _____
    - version 16
    - if "`'.result'" != "e" {
    = if "e" != "e" {
        exit
    }
    - if !missing(e(r2)) {
        .PushRight "R-squared" " = " "as res" %`.c4wfmt'.4f e(r2)
    }
    - if !missing(e(r2_p)) & `.is_r2_p' {
    = if !missing(e(r2_p)) & 1 {
        - .PushRight "Pseudo R2" " = " "as res" %`.c4wfmt'.4f e(r2_p)
    }
    }
    = .PushRight "Pseudo R2" " = " "as res" %10.4f e(r2_p)
}

> _____ begin _coef_table__header.PushRi
> ght _____
    - version 16
    - gettoken tok rest : 0
    - gettoken eq rest : rest
    - gettoken dias rest : rest
    - if `:length local rest' {
    = if 15 {
        - local direst = ustrltrim("` : di `rest'")
        = local direst = ustrltrim(" 0.1060")
    }
    }
}

```

```

- .ralignr = max(`.ralignr', udstrlen("`direst'"))
= .ralignr = max(3, udstrlen("0.1060"))
- local len : di ``tok'''
= local len : di ``Pseudo R2'''
- local len : smcllength local len
- .lalignr = max(`.lalignr', `len')
= .lalignr = max(13, 9)
- .right.Arrpush `C3' ``{lalign `lalignr':`tok'}" `.`
> C4' ``eq' ``dias' "{ralign `ralignr':`direst'}"
= .right.Arrpush `C3' ``{lalign `lalignr':Pseudo R2}"" `
> .C4' " = " as res "{ralign `ralignr':0.1060}"
- }
- else if `:length local tok' {
= else if 9 {
    .right.Arrpush `C3' ``tok'''
  }
- else {
    .right.Arrpush
  }
}

> _____ end _coef_table__header.PushRi
> ght _____
- }
- if !missing(e(r2_a)) {
    .PushRight "Adj R-squared" " = " "as res" %`.c4wfmt'.4f e(
> r2_a)
    }
- if !missing(e(rmse)) {
    .PushRight "Root MSE" " = " "as res" %`.c4wfmt'.4f e(rmse)
    }
}

> _____ end _coef_table__header.Build
> 4r2 _____
- .Build4title
}

> _____ begin _coef_table__header.Build4ti
> tle _____
- version 16
- if `.opt_septitle' {
= if 0 {
    exit
  }
- if ``.title''' == "" {
= if `"Poisson regression"' == "" {
    exit
  }
}

```

```

- if `.left.arnels' {
= if 0 {
    exit
  }
- local title `.title'
= local title Poisson regression
- local title2 `.title2'
= local title2
- local len : smcllength local title
- local len2 : smcllength local title2
- if `len2' > `len' {
= if 0 > 18 {
    local len = `len2'
  }
- if `len' < (`c3' - 2) - (`c1' - 1) {
= if 18 < (49 - 2) - (1 - 1) {
- PushLeft ``.title'''
= PushLeft `"Poisson regression"'

```

```

> _____ begin _coef_table__header.PushL
> eft _____
    - version 16
    - gettoken tok rest : 0
    - gettoken eq rest : rest
    - gettoken dias rest : rest
    - if `:length local rest' {
= if 0 {
    local direst : di `rest'
    local direst = strtrim("`direst'")
    local len : smcllength local direst
    .ralignl = max(`.ralignl', `len')
    local len : di ``tok'''
    local len : smcllength local len
    .lalignl = max(`.lalignl', `len')
    .left.Arrpush \`.C1' `{lalign \`.lalignl:`tok'}" \`.C
> 2' ``eq' ``dias' "{ralign \`.ralignl:`direst'}"
    }
    - else if `:length local tok' {
= else if 18 {
    - .left.Arrpush \`.C1' ``tok'''
= .left.Arrpush \`.C1' `"Poisson regression"'
    - }
    - else {
    .left.Arrpush
    }

```

```

> _____ end _coef_table__header.PushL
> eft _____
      - .title = ""
      - if `len2' {
      = if 0 {
        PushLeft ``.title2'""
        .title2 = ""
      }
      - }
      - .opt_septitle = 1
      _____

> _____ end _coef_table__header.Build4ti
> tle _____
      - if `.prop_censlimits' {
      = if 0 {
        .Build4censlimits 1
      }
      - .c2 = `.lalignl' + 1
      = .c2 = 0 + 1
      - .c3 = `.width'-`.lalignr'-`.ralignr'-3+1
      = .c3 = 78-13-6-3+1
      - .c4 = `.c3' + `.lalignr'
      = .c4 = 57 + 13
      - if `.is_gsem' {
      = if 0 {
        if `.opt_dvheader' {
          gsem_depvar_header, left(`.left.objkey') right(`.right.objke
> y') c3(`.c3') c4(`.c4') c4wfmt(`.c4wfmt')
        }
        }
      - if `.is_xtvar' {
      = if 0 {
        local uno = (`e(rank_W)' < `e(Z_cols)') & (`e(J)' != .j)
        if ("`e(collapse)'"=="") {
          if (`uno'==0) {
          }
          else {
            local extra " {txt}"
          }
          .PushLeft "Number of moment conditions" " = " "as res" %`.c4
> wfmt'.0fc e(Z_cols) "`extra'"
        }
        else {
          if (`uno'==0) {
          }
          .PushLeft "Number of moment conditions" " = " "as res" %`.c4

```

```

> wfmt'.0fc e(Z_cols) " {txt}(collapsed)"
    }
    local uno = (`e(rank_W)' < `e(Z_cols)') & (`e(J)' != .j)
    if ("`e(collapse)'"==" & `uno'==0) {
    }
    if ("`e(collapse)'"==" & `uno'==1) {
    .PushLeft "Rank of weight matrix" " = " "as res" %`.c4wfmt'.
> 0fc e(rank_W) "{txt}{helpb j_xtvar_Jdf##lowrank:*}"
    }
    if ("`e(collapse)'"!=" & `uno'==0) {
    }
    if ("`e(collapse)'"!=" & `uno'==1) {
    .PushLeft "Rank of weight matrix" " = " "as res" %`.c4wfmt'.
> 0fc e(rank_W) "{helpb j_xtvar_Jdf##lowrank:*}          {txt}"
    }
    }
- if `.is_cfreg' {
= if 0 {
    local k = e(k_paren)
    local mtypes `e(modeltypes)'
    local looklong: list uniq mtypes
    local kt: list sizeof looklong
    forvalues ei = 1(1)`=wordcount("`e(endog)'"')' {
    local enei : word `ei' of `e(endog)'
    _ms_parse_parts `enei'
    if (`r(omit)' & "`r(type)'"=="factor") {
    continue
    }
    local endog `endog' `enei'
    }
    if (`kt'>1 | `=wordcount("`endog'")'>1) local ess s
    .PushLeft
    .PushLeft
    .PushLeft
    if ("`e(r2)'"!=") if (`e(r2)'<.) .PushLeft
    .PushLeft "Endogenous variable model`ess':"
    local largo = 0
    forvalues i=1/`kt' {
    local x: word `i' of `looklong'
    if ("`x'"=="fprobit") {
    local x "Fractional probit"
    }
    local largo1 = strlen("`x'")
    local largo = max(`largo', `largo1')
    local largo1 = `largo'
    }

```



```

forvalues j = 1/`kt' {
  local endj: word `j' of `looklong'
  forvalues i=1/`k' {
    local typo "`e(type`i)'"
    forvalues ei = 1(1) `=wordcount("`e(endogvars`i)')'" {
      local eniei : word `ei' of `e(endogvars`i)''
      _ms_parse_parts `eniei'
      if (`r(omit)' & "`r(type)'"=="factor") {
        continue
      }
      local insts`j'_'i' `insts`j'_'i'' `eniei'
    }
    if ("`endj'"=="`typo'") {
      local print`j' "`print`j'' `insts`j'_'i''
      local print`j'= strltrim("`print`j'")
    }
  }
  forvalues i=1/`kt' {
    local slng = `largo' + 6
    local x: word `i' of `looklong'
    if ("`x'"=="fprobit") {
      local x "Fractional probit"
    }
    else {
      local x = strproper("`x'")
    }
    local y "`print`i'"
    .left.Arrpush %`slng's "      `x': " "{res}`y'"
  }
  if ("`e(vcetype)'" == "HAC") {
    local ktyp `e(hac_kernel)'
    local fmt -6.0f
    local adds "s"
    if "`ktyp'" == "quadratic spectral" {
      local ktyp "Quadratic spectral"
      local fmt -6.3f
    }
    if `e(hac_lag)'==1 {
      local adds
    }
    local hn : di "{txt}HAC kernel: {res}`ktyp'" "{txt} with {re
> s}" `:di %`fmt' `e(hac_lag)'' "{txt} lag`adds'"
    .PushLeft
    .left.Arrpush %-48s "`hn'"
  }
}

```

```

    }
    - .Build4crit
  }
  > ----- begin _coef_table__header.Build4c
  > rit -----
    - version 16
    - if ``.result'' != "e" {
    = if "e" != "e" {
      exit
    }
    - local offset 1
    - local hascrit = !missing(`""e(ll)''') & !missing(`""e(crit
  > type)''')
    = local hascrit = !missing(`"-375.7549196633363"'') & !missin
  > g(`"log pseudolikelihood"')
    - if inlist(`"e(dispers)"', "mean", "constant") {
    = if inlist("", "mean", "constant") {
      if `hascrit' {
        local ++offset
        local crtype `""e(crittype)'''
        local clen : length local crtype
        .c2 = `clen' + 2
      }
      local didisp = strtrim("`": di e(dispers)')
      local DISPERS `""Dispersion: {res:`didisp'}""'
    }
    - .FillLeft `offset'
    = .FillLeft 1
  }
  > ----- begin _coef_table__header.FillL
  > eft -----
    - version 16
    - args offset
    - local kr = `.right.arnels'
    = local kr = 4
    - local kl = `.left.arnels'
    = local kl = 1
    - local diff = `kr' - `kl' - 0`offset'
    = local diff = 4 - 1 - 01
    - forval i = 1/`diff' {
    = forval i = 1/2 {
      - .left.Arrpush ""
      - }
      - .left.Arrpush ""
      - }
  }

```

```

> _____ end _coef_table__header.Fillll
> eft _____
      - if ``DISPERS''' != "" {
      = if `'''' != "" {
        .PushLeft `DISPERS'
      }
      - local crtype = upper(bsubstr(`"e(crittype)''',1,1)) + bsu
> bstr(`"e(crittype)''',2,.)
      = local crtype = upper(bsubstr(`"log pseudolikelihood"',1,1)
> ) + bsubstr(`"log pseudolikelihood"',2,.)
      - if `.is_sem' & `"e(critvalue)''' != "" & !`.is_svy' {
      = if 0 & `'''' != "" & !0 {
        .PushLeft
        .PushLeftRalign "`crtype'" " = " "as res" %10.0g e(critval
> ue)
      }
      - else if `hascrit' {
      = else if 1 {
      - if `.is_gsem_group' {
      = if 0 {
        local dill = strtrim("` : di %10.0g e(ll)''")
        .PushLeft "`crtype' = {res:`dill'}"
      }
      - else {
      - .PushLeftRalign "`crtype'" " = " "as res" %10.0g e(ll)
      = .PushLeftRalign "Log pseudolikelihood" " = " "as res" %10.
> 0g e(ll)
      _____
> _____ begin _coef_table__header.PushLeftRal
> ign _____
      - version 16
      - gettoken tok rest : 0
      - gettoken eq rest : rest
      - gettoken dias rest : rest
      - if `:length local rest' {
      = if 13 {
      - local direst : di `rest'
      = local direst : di %10.0g e(ll)
      - local direst = strtrim("`direst'")
      = local direst = strtrim("-375.75492")
      - local len : smcllength local direst
      - .ralignla = max(`.ralignla', `len')
      = .ralignla = max(0, 10)
      - local len : di ``tok'''
      = local len : di `"Log pseudolikelihood"'
      - local len : smcllength local len

```

```

        - .lalignla = max(`.lalignla', `len')
        = .lalignla = max(0, 20)
        - .left.Arrpush `\.C1' ``{lalign `\.lalignla':`tok'}`" `\.
> C2a' ``eq' ``dias' "{ralign `\.ralignla':`direst'}"
        = .left.Arrpush `\.C1' ``{lalign `\.lalignla':Log pseudolike
> lihood}`" `\.C2a' " = " as res "{ralign `\.ralignla':-375.75492}"
        - }
        - else if `:length local tok' {
        = else if 20 {
            .left.Arrpush `\.C1' ``tok'""
        }
        - else {
            .left.Arrpush
        }
    }
}

> _____ end _coef_table__header.PushLeftRal
> ign _____
        - }
        - }
    }
}

> _____ end _coef_table__header.Build4c
> rit _____
        - .c2 = `.lalignl' + 1
        = .c2 = 0 + 1
        - .c2a = `.lalignla' + 1
        = .c2a = 20 + 1
        - .c3 = `.width' - `.lalignnr' - `.ralignnr' - 3 + 1
        = .c3 = 78 - 13 - 6 - 3 + 1
        - .c4 = `.c3' + `.lalignnr'
        = .c4 = 57 + 13
    }
}

> _____ end _coef_table__header.bu
> ild _____
        - if ("`head2start'" != "") {
        = if (" " != " ") {
            .`obj'.c3 = `head2start'
            .`obj'.c4 = `head2start' + `.`obj'.lalignnr'
            .`obj'.width = `head2start' + `.`obj'.lalignnr' + `.`obj'.ralig
> nr' + 2
        }
        - if (`has_obj' == 0) {
        = if (0 == 0) {
            - .`obj'.display
            = .__000005.display
        }
    }
}

> _____ begin _coef_table__header.disp

```

```

> lay -----
      - version 16
      - .ShowRules
-----

> ----- begin _coef_table__header.ShowRu
> les -----
      - version 16
      - if "`result'" != "e" {
= if "e" != "e" {
        exit
      }
      - if `.opt_rules' == 0 {
= if 1 == 0 {
        exit
      }
      - if !inlist("`cmd'", "logistic", "logit", "probit") {
= if !inlist("poisson", "logistic", "logit", "probit") {
      - exit
-----

> ----- end _coef_table__header.ShowRu
> les -----
      - local nl = `.left.arnels'
= local nl = 4
      - local nr = `.right.arnels'
= local nr = 4
      - local K = max(`nl', `nr')
= local K = max(4,4)
      - `.newline'
= display

      - if ```.title''' != "" {
= if `''' != "" {
        display as txt ``.title'''
        if ``.title2''' != "" {
          display as txt ``.title2'''
        }
        if `K' {
          display
        }
      }
      - forval i = 1/`K' {
= forval i = 1/4 {
      - display as smcl as txt `.left[`i']' as txt `.right[`i']'
-----

> ----- begin _coef_table__header
> .C1 -----

```

```

- version 16
- class exit `"_col(`.c1')"'
= class exit `"_col(1)"'
_____
> _____ end _coef_table__header
> .C1 _____
_____
> _____ begin _coef_table__header
> .C3 _____
- version 16
- class exit `"_col(`.c3')"'
= class exit `"_col(57)"'
_____
> _____ end _coef_table__header
> .C3 _____
_____
> _____ begin _coef_table__header
> .C4 _____
- version 16
- class exit `"_col(`.c4')"'
= class exit `"_col(70)"'
_____
> _____ end _coef_table__header
> .C4 _____
= display as smcl as txt _col(1) `"Poisson regression"' as txt
> _col(57) `"{lalign 13:Number of obs}"' _col(70) " = " as res "{ralign 6:217
> }"
Poisson regression                                Number of obs =    217
- }
- display as smcl as txt `'.left['i']' as txt `'.right['i']'
_____
> _____ begin _coef_table__header
> .C3 _____
- version 16
- class exit `"_col(`.c3')"'
= class exit `"_col(57)"'
_____
> _____ end _coef_table__header
> .C3 _____
_____
> _____ begin _coef_table__header
> .C4 _____
- version 16
- class exit `"_col(`.c4')"'
= class exit `"_col(70)"'
_____

```

```

> _____ end _coef_table__header
> .C4 _____
      = display as smcl as txt  as txt _col(57) ``{lalign 13:{help j
> _robustsingular##|_new:Wald chi2(9)}}'' _col(70) " = " as res "{ralign 6:."
      Wald chi2(9) =      .
      - }
      - display as smcl as txt ``.left['i']' as txt ``.right['i']'
      _____
> _____ begin _coef_table__header
> .C3 _____
      - version 16
      - class exit ``_col(`.c3')''
      = class exit ``_col(57)''
      _____
> _____ end _coef_table__header
> .C3 _____
      _____
> _____ begin _coef_table__header
> .C4 _____
      - version 16
      - class exit ``_col(`.c4')''
      = class exit ``_col(70)''
      _____
> _____ end _coef_table__header
> .C4 _____
      = display as smcl as txt  as txt _col(57) ``{lalign 13:Prob >
> chi2}'' _col(70) " = " as res "{ralign 6:."
      Prob > chi2 =      .
      - }
      - display as smcl as txt ``.left['i']' as txt ``.right['i']'
      _____
> _____ begin _coef_table__header
> .C1 _____
      - version 16
      - class exit ``_col(`.c1')''
      = class exit ``_col(1)''
      _____
> _____ end _coef_table__header
> .C1 _____
      _____
> _____ begin _coef_table__header.
> C2a _____
      - version 16
      - class exit ``_col(`.c2a')''
      = class exit ``_col(21)''
      _____

```

```

> _____ end _coef_table_header.
> C2a _____
> _____ begin _coef_table_header
> .C3 _____
      - version 16
      - class exit `"_col(`.c3')"'
      = class exit `"_col(57)"'
> _____ end _coef_table_header
> .C3 _____
> _____ begin _coef_table_header
> .C4 _____
      - version 16
      - class exit `"_col(`.c4')"'
      = class exit `"_col(70)"'
> _____ end _coef_table_header
> .C4 _____
      = display as smcl as txt _col(1) `{lalign 20:Log pseudolikeli
> hood}' _col(21) " = " as res "{ralign 10:-375.75492}" as txt _col(57) `{la
> lign 13:Pseudo R2}' _col(70) " = " as res "{ralign 6:0.1060}"
Log pseudolikelihood = -375.75492 Pseudo R2 = 0.1060
      - }
> _____ end _coef_table_header.disp
> lay _____
      - }
      - sreturn local width_header `.`obj'.width'
      = sreturn local width_header 78
      - sreturn local head2_right `.`obj'.ralignr'
      = sreturn local head2_right 6
      - sreturn local head2_left `.`obj'.lalignr'
      = sreturn local head2_left 13
      - sreturn local head2_start `.`obj'.c3'
      = sreturn local head2_start 57
      - sreturn local head1_right `.`obj'.ralignla'
      = sreturn local head1_right 10
      - sreturn local head1_left `.`obj'.lalignla'
      = sreturn local head1_left 20
      - sreturn local head1_start `.`obj'.c1'
      = sreturn local head1_start 1
> _____ end _coef_table_h
> der _____

```



```

- di
- }
- _coef_table, `first' level(`level') `neqopt' `plus' `noskip' `sh
> oweqns' `diopts' `options' `notest'
    = _coef_table, level(95)


---


> 

---

 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 "poisson"
    - local keepmc : list cmd in mc_cmds
    - if inlist("`cmd'", "pwcompare", "pwmean") {
    = if inlist("poisson", "pwcompare", "pwmean") {
        local groups GROUPS
    }
    - if "`cmd'" == "regress" {
    = if "poisson" == "regress" {
        local beta Beta
    }
    - if "`cmd'" == "gsem" {
    = if "poisson" == "gsem" {
        local fvignore fvignore(int 0)
        local flignore flignore
    }
    - if "`cmd'" == "sem" {
    = if "poisson" == "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(st
> ring) DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) P
> MATrix(string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFo
> rmat(string) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDM
> ATrix(string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(n
> ame) cnszero(string) suffix(name) OFFSETlist(string asis) `beta' `standardiz
> ed' NOCI NOPValues `dftable' `dfonly' dfci DFPValues dfmissing `groups' COEF
> Legend selegend `fvignore' `flignore' `showginvariant' `nolabel' `nofootnote
> ' `wrap' sort `pisematrix' depname(string) COEFTitle(string) coeftitle2(stri
> ng) ptitle(string) cititle(string) NOMCLEGen noCNSReport FULLCNSReport `noc
> lustreport' cformat(passthru) sformat(passthru) pformat(passthru) NOFirst Fi
> rst SHOEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(integer 0) NOSKIP OFF
> SETONLY1 PLUS NOEQCHECK EFORMALL CITYPE(string) eqselect(string) cidisjoint
> noEFORMNOTE * ]

    = syntax [, cmdextras BMATrix(string) VMATrix(string) EMATrix(st
> ring) DFMATrix(string) EQMATrix(string) ROWMATrix(string) CIMATrix(string) P
> MATrix(string) TSMATrix(string) ROWCFormat(string) ROWPFormat(string) ROWSFo
> rmat(string) NOROWCI MMATrix(string) MVMATrix(string) MEMATrix(string) BSTDM
> ATrix(string) CNSMATrix(string) PCLASSMATrix(string) Level(cilevel) prefix(n
> ame) cnszero(string) suffix(name) OFFSETlist(string asis) NOCI NOPValues
> dfci DFPValues dfmissing COEFLegend selegend sort depname(string) C
> OEFTitle(string) coeftitle2(string) ptitle(string) cititle(string) NOMCLEGen
> d noCNSReport FULLCNSReport cformat(passthru) sformat(passthru) pformat(pas
> sthru) NOFirst First SHOEQns neq(integer -1) NODIPARM NOTEST SEPARATOR(inte
> ger 0) NOSKIP OFFSETONLY1 PLUS NOEQCHECK EFORMALL CITYPE(string) eqselect(st
> ring) 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 fvlable nofvlable
    }
    else {
        local fvlable fvlable
    }
}

- _novce_other_mats, cimatrix("`cimatrix'") pmatrix("`pmatrix'")
> tsmatrix("`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) TSMATr
> ix(string) NOCI NOPValues dfci DFPValues cidisjoint ]
        - if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`ci
> matrix'"!=" & "`noci'"!=") {
            = if ("'"=="" & "'"=="" & "matrix"=="" & "'"!=" & "'"!=") {
                sreturn local cimatrix "empt"
            }
            - if ("`pmatrix'"=="" & "`tsmatrix'"!=" & "`e(V)'"=="" & "`ci
> matrix'"=="" & ("`noci'"!="|"`nopvalues'"!=")) {
                = if ("'"=="" & "'"!=" & "matrix"=="" & "'"=="" & ("'"!="|'"!="
> )) {
                    sreturn local noci "empt"
                    sreturn local nopvalues "empt"
                }
            - if ("`pmatrix'"=="" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`ci
> matrix'"!=" & "`noci'"!=") {
                = if ("'"=="" & "'"=="" & "matrix"=="" & "'"!=" & "'"!=") {
                    sreturn local cimatrix "empt"
                }
            - if ("`pmatrix'"!=" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`ci
> matrix'"=="" & "`nopvalues'"!=") {
                = if ("'"!=" & "'"=="" & "matrix"=="" & "'"=="" & "'"!=") {
                    sreturn local pmatrix "empt"
                }
            - if ("`pmatrix'"!=" & "`tsmatrix'"=="" & "`e(V)'"=="" & "`ci
> matrix'"=="" & "`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 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) i
> s posted"
  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' `group
> s' `dfci' `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 " "

> ----- begin opts_exclus
> ive -----
- version 8.2
- args opts optname errcode optslab
- local opts `opts'
= local opts
- local 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 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 ematrixe(error)
    }
    - }
    - _get_mcompare, `options'
    = _get_mcompare,

> ----- begin _get_mcomp
> are -----
    - version 11
    - local cmd ``e(cmd)'"
    = local cmd `"poisson"
    - syntax [name(name=caller)] [, *]
    - local is_pwc = inlist("pwcompare", "`cmd'", "`caller'") | in

```

```

> list("pwmean", "`cmd'", "`caller'")
      = local is_pwc = inlist("pwcompare", "poisson", "") | inlist("
> pwmean", "poisson", "")
      - local is_marg = inlist("margins", "`cmd'", "`e(cmd2)'", "`ca
> ller'")
      = local is_marg = inlist("margins", "poisson", "", "")
      - local OPTIONS NOADJust BONferroni DUNCAn DUNNett SCHeffe SID
> ak 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' `sch
> effe' `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' `dunn
> ett' `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 adju
> stall 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 `c
> md'''
        exit 322
    }
    if ``e(prefix)''' == "svy" {
        di as err ``method `opt' is not allowed with results using t
> he svy prefix'''
        exit 322
    }
    local vce ``e(vce)'''
    if !inlist(`vce', "", "ols") {
        di as err ``method `opt' is not allowed with results using v
> ce(`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 `MET
> HOD''''

    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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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' `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' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abb
> rev'
    = 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
        - syntax namelist(max=2) [, `DIOPTS' *]
        = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
    }
    - 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' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'

    = 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV

    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND

```

```

> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
}
- 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' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'
= 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(strin
> g) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
}
- 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 fwrapon fwrapon(`fwrapon')
        }
    }
end _get_diopts.Wra
pon
    - if "`fwrap'" != "" {
    = if "" != "" {
        FVWrap, `fwrap'
    }
    - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fwrap' `fwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'
        = 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
end
begin _get_diopts.GetDio

```

```

> pts -----
      - local DIOPTS Level(cilevel) vsquish NOALLBASElevels ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
      - syntax namelist(max=2) [, `DIOPTS' *]
      = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
  }
}

```

```

- 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' `b

```

```

> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'

    = 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
}
- 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' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'
= 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__) soption

```

> s

```

= _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 RR
> r tr TRatio * ]
- local opts ``options''
= local opts ``''
- local 0 , `eform1' `eform' `nohr' `hr' `noshr' `shr' `or' `i
> rr' `rrr' `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 TRatio ]
- 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(cm
> d)'"=="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'', ``ef
> ormdi'')
= 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 pl
> us SYNTAXonly * ]
- 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(strin
> g 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' `goptio

> ns'

    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 12, 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
    - syntax namelist(max=2) [, `DIOPTS' *]
    = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFlegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
}
- 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'" 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' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvwrapon' `fvlabel' `noci' `nopvalues' `markdown' `citype' `abb
> rev'
    = 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 __000005 ``'
    - 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(stri
> ng) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV
- syntax namelist(max=2) [, `DIOPTS' *]
= syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP
> E(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')
}
- 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, `fvwrap'
      }
      - c_local `c_diopts' `levelopt' `vsquish' `allbaselevels' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fvwrap' `fvrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'

      = 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 ALLBAS
> Elevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMPTYcells EMPTY
> cells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND cformat(strin
> g) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRAP(passthru) F
> VWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYPE(passthru) N
> OABBREV ABBREV

      - syntax namelist(max=2) [, `DIOPTS' *]
      = syntax namelist(max=2) [, Level(cilevel) vsquish NOALLBASE
> levels ALLBASElevels NOBASElevels BASElevels noCNSReport FULLCNSReport NOEMP
> TYcells EMPTYcells NOOMITted OMITted NOLSTRETCH LSTRETCH COEFLegend SELEGEND
> cformat(string) sformat(string) pformat(string) CODING VERSUS COMPARE FVWRA
> P(passthru) FVWRAPON(string) NOFVLABEL FVLABEL NOCI NOPValues MARKDOWN CITYP

```

```

> E(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' `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 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' `b
> aselevels' `cnsreport' `fullcnsreport' `emptycells' `omitted' `lstretch' `co
> eflegend' `selegend' `cformat' `sformat' `pformat' `coding' `versus' `compar
> e' `fwwrap' `fwwrapon' `fvlabel' `nocl' `nopvalues' `markdown' `citype' `abb
> rev'

    = 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				
	y_timing_max	Coefficient	std. err.	z	P> z	[95% conf. inte	
> rval]							
> _____							
	rule_g	.5087735	.1631382	3.12	0.002	.1890284	.82
> 85185							
	rule_itr	.3123713	.1291169	2.42	0.016	.0593069	.56
> 54358							
	estimated	.7181505	.0941673	7.63	0.000	.5335859	.9


```

> 02715
      ntwrth#learning
      0 1 | -1.314307 .3034869 -4.33 0.000 -1.90913 -.71
> 94834
      1 0 | -.0783427 .113888 -0.69 0.492 -.3015591 .14
> 48737
      1 1 | 0 (omitted)
      bnkcrdit#learning
      0 1 | 0 (omitted)
      1 0 | .0493166 .1123367 0.44 0.661 -.1708592 .26
> 94924
      1 1 | 0 (omitted)
      wlth#open
      0 1 | .0069252 .1597525 0.04 0.965 -.3061839 .32
> 00344
      1 0 | -.3262271 .1144034 -2.85 0.004 -.5504536 -.10
> 20006
      1 1 | .167307 .2036567 0.82 0.411 -.2318529 .56
> 64668
      learning
      _cons | .5115597 .305898 1.67 0.094 -.0879894 1.1
> 11109
      _cons | .0845968 .113686 0.74 0.457 -.1382238 .30
> 74174

```

```

> -----
- if "`markdown'" != "" {
= if "" != "" {
  set linesize `lsize'
}
- if "`eqselect'" == "" & "`eformnote'" != "noeformnote" {
= if "" == "" & "" != "noeformnote" {
- if ``"eformdi"" != "" & ("`e(cmd)'" != "gnbreg" | "`e(prefix)'" ==
> "svy") {
= if ``"" != "" & ("poisson" != "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)
> ' == "cmmixlogit" | `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) a
> nd "
                        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 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(`poisson',"logit","logistic","probit") {
    CompleteLogit
  }
  - if inlist(`e(cmd)'',"mlogit","ologit","oprobit") {
  = if inlist(`poisson',"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 poisson.Disp
> lay _____
    - }
    - error `e(rc)'
    = error 0
    _____

> _____ end poisson.Estim
> ate _____
    - version 10: ereturn local cmdline `"poisson `0'"
    = version 10: ereturn local cmdline `"poisson y_timing_max rule_g rule_itr e
> stimated i.ntwrth#i.learning i.bnkcrdit#i.learning i.wlth#i.open learning
> if y_timing_max < 99 , vce(robust) nolog"
    _____

> _____ end pois
> son _____
- log close
    name: <unnamed>
    log: /Users/connorbrennan/Library/CloudStorage/OneDrive-TheUniversityo
> fChicago/mmb/output/stepwise_regressions/y_timing_max_realrig_all.smcl
    log type: smcl
    closed on: 12 Sep 2025, 19:25:24
    _____

```