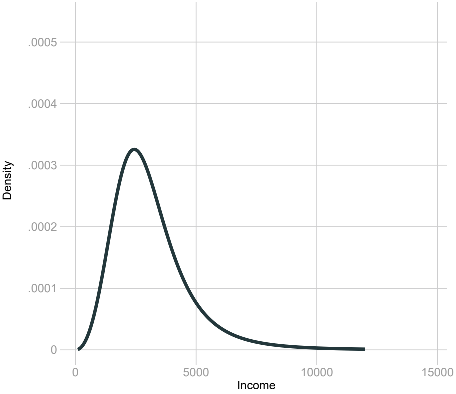
Conditional likelihood models for distributional regression analysis

Philippe Van Kerm
University of Luxembourg and LISER

2020 Swiss Stata Conference — November 19, 2020

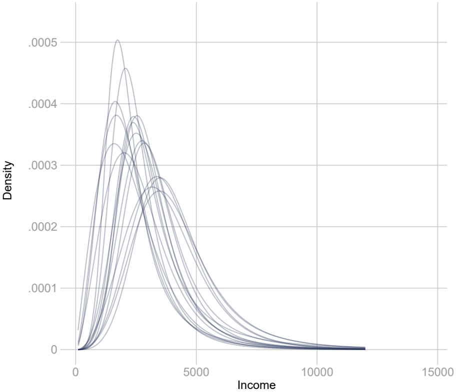
Conditional likelihood models in a nutshell

- Fit a parametric distribution function $f_\theta(y)$ …
  - θ is a small vector of parameters (typically, say, 2–4 parameters)
  - e.g., a (log-)normal, a gamma, a beta distribution, etc.
- … conditioning on vector of covariates, $f_{\theta(X)}(y)$
- … by specifying a parametric relationship between X and θ
  - For example, $\theta(X) = X\beta$ (or $\theta(x) = \exp(X\beta)$ if $\theta(X)$ must be > 0)



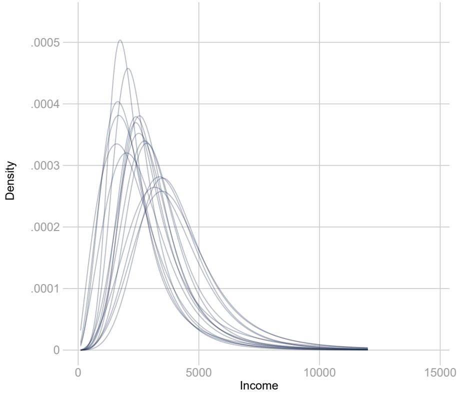
Conditional likelihood models in a nutshell

- Fit a parametric distribution function $f_\theta(y)$ …
  - θ is a small vector of parameters (typically, say, 2–4 parameters)
  - e.g., a (log-)normal, a gamma, a beta distribution, etc.
- … conditioning on vector of covariates, $f_{\theta(X)}(y)$
- … by specifying a parametric relationship between X and θ
  - For example, $\theta(X) = X\beta$ (or $\theta(x) = \exp(X\beta)$ if $\theta(X)$ must be > 0)



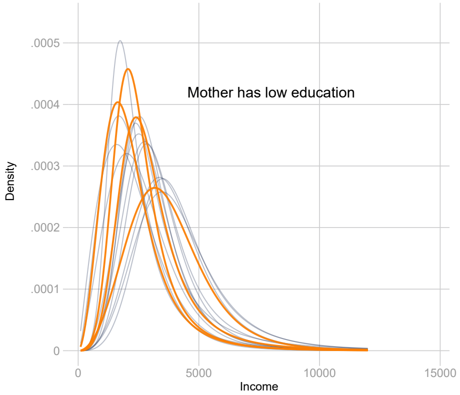
Conditional likelihood models in a nutshell

- Fit a parametric distribution function $f_\theta(y)$ ...
  - θ is a small vector of parameters (typically, say, 2–4 parameters)
  - e.g., a (log-)normal, a gamma, a beta distribution, etc.
- ... conditioning on vector of covariates, $f_{\theta(X)}(y)$
- ... by specifying a parametric relationship between X and θ
  - For example, $\theta(X) = X\beta$ (or $\theta(x) = \exp(X\beta)$ if $\theta(X)$ must be > 0)



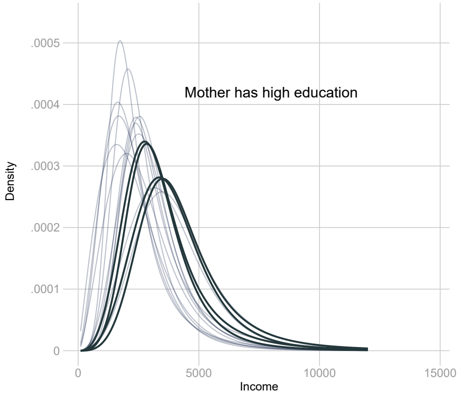
Conditional likelihood models in a nutshell

- Fit a parametric distribution function $f_\theta(y)$ ...
  - θ is a small vector of parameters (typically, say, 2–4 parameters)
  - e.g., a (log-)normal, a gamma, a beta distribution, etc.
- ... conditioning on vector of covariates, $f_{\theta(X)}(y)$
- ... by specifying a parametric relationship between X and θ
  - For example, $\theta(X) = X\beta$ (or $\theta(x) = \exp(X\beta)$ if $\theta(X)$ must be > 0)



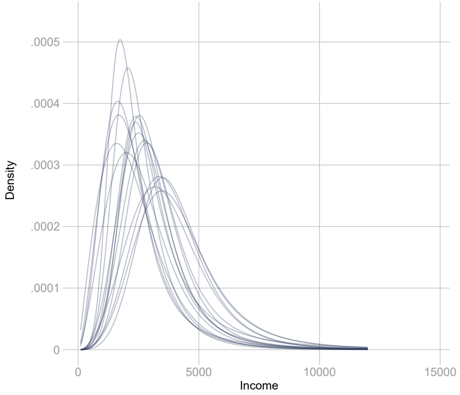
Conditional likelihood models in a nutshell

- Fit a parametric distribution function $f_\theta(y)$ ...
  - θ is a small vector of parameters (typically, say, 2–4 parameters)
  - e.g., a (log-)normal, a gamma, a beta distribution, etc.
- ... conditioning on vector of covariates, $f_{\theta(X)}(y)$
- ... by specifying a parametric relationship between X and θ
  - For example, $\theta(X) = X\beta$ (or $\theta(x) = \exp(X\beta)$ if $\theta(X)$ must be > 0)



Uses of conditional likelihood models

- Functional outcomes (Biewen and Jenkins, 2005)
- Quantile regression… without running quantile regression (Noufaily and Jones, 2013)
- Censored data (Jenkins et al., 2011)
- Endogenous selection (Van Kerm, 2013)
- Instrumental variables (Briseño Sanchez et al., 2020)
- Marginalisation and counterfactual distributions (Van Kerm et al., 2017)



Array of models for conditional distributions F_X

Many models and estimators available, more or less parametrically restricted, e.g.,

- quantile regression (Koenker and Bassett, 1978)
- distribution regression (Foresi and Peracchi, 1995, Chernozhukov et al., 2013, Van Kerm, 2016)
- duration models (Donald et al., 2000, Royston, 2001)
- conditional likelihood models (Biewen and Jenkins, 2005, Van Kerm et al., 2017)

1. Quantile regression
2. Distribution regression
3. Conditional likelihood models

Linear quantile regression model

Assume a particular relationship (linear) between conditional quantile and x:

$$Q_\tau(y|x) = x\beta_\tau$$

(Or equivalently $y_i = x_i\beta_\tau + u_i$ where $F^{-1}_{u_i|x_i}(\tau) = 0$)

$$\hat{\beta}_\tau = \arg\min_\beta \sum_i \rho_\tau(y_i - x_i\beta)$$

(Koenker and Bassett, 1978)

Estimate of the conditional quantile (given linear model):

$$\hat{Q}_\tau(y|x) = x\hat{\beta}_\tau$$

$\hat{\beta}_\tau$ can be interpreted as the marginal change in the τ conditional quantile for a marginal change in x

Recovering $\upsilon(F_x)$

Estimation of $\hat{Q}_\tau(y|x)$ for a continuum of τ in $(0,1)$ provides a model for the entire conditional quantile function of Y given X (the quantile 'process'–See Blaise Melly's presentation and `qrprocess` for fast implementation)

After estimation of the quantile process $(0,1)$, estimation of the distributional statistic conditional on X is relatively easy by simulation:

- a set of predicted conditional quantile values $\{x_i\hat{\beta}_\theta\}_{\theta\in(0,1)}$ is a pseudo-random draw from F_x (if grid for θ is equally-spaced) (Autor et al., 2005)
- so, a simple estimator for υ from unit-record data can be used to estimate $\upsilon(F_{X_i})$

Disadvantage?

Linearity of the model $Q_\tau(y|x) = x\beta_\tau$ may possibly be problematic in some situations

- discontinuities (e.g. minimum wage)
- quantile crossing within the support of X (Simple solution is re-arrangement of quantile predictions (Chernozhukov et al., 2009))

1. Quantile regression
2. Distribution regression
3. Conditional likelihood models

'Distribution regression'

$F_x(y) = \Pr\{y_i \leqslant y|x\}$ is a binary choice model once y is fixed (dependent variable is $1(y_i < y)$)

Estimate $F_x(y)$ on a grid of values for y spanning the domain of definition of Y by running repeated standard binary choice models, e.g. a logit:

$$\begin{aligned} F_x(y) &= \Pr\{y_i \leqslant y|x\} \\ &= \Lambda(x\beta_y) \\ &= \frac{\exp(x\beta_y)}{1 + \exp(x\beta_y)} \end{aligned}$$

or a probit $F_x(y) = \Phi(x\beta_y)$ or else ...

‘Distribution regression’

- Estimate distributional process by repeating estimation at different values of y—makes little assumptions about the overall shape of distribution
- Discontinuities are handled without difficulties
- Estimation of these models is well-known and straightforward (`probit`, `logit`)
- Faster to run than quantile regression
- Evidence that provides better fit to conditional quantile processes than quantile regression (Rothe and Wied, 2013, Van Kerm et al., 2017)

Disadvantage

Drawback: Conditional statistic $\upsilon(F_x)$ often less easy to recover from the $\hat{F}_X$ predictions than with quantile regression

- invert the predicted F_x to obtain predicted quantiles
- proceed as with quantiles predicted from quantile regression (see above)

1. Quantile regression
2. Distribution regression
3. Conditional likelihood models

Conditional likelihood models

Assume that the conditional distribution has a particular parametric form: e.g., (log-)normal (2 parameters – quite restrictive), Gamma (2 params), Singh-Maddala (3 param.), Dagum (3 param.), GB2 (4 param.), ... or any other distribution that is likely to fit the data at hand (think domain of definition, fatness of tails, modality)

Let parameters (say vector θ) depend on x in a particular fashion, typically linearly (up to some transformation satisfyng range of variation of pthe arameters), e.g., $\theta_X^1 = \exp(x\beta_1)$, $\theta_X^2 = \exp(x\beta_2)$ and $\theta_X^3 = x\beta_3$

This gives a fully specified parametric model which can be estimated using maximum likelihood ($\Longrightarrow$ inference is straightforward).

Functionals derived from conditional likelihood models

- With parameter estimates $\hat{\theta}_X$, we can recover conditional quantiles, CDF, PDF and all sort of functionals $\upsilon(F_x)$ (means, dispersion measures, etc.) often from closed-from expressions
- Typically much less computationally expensive than estimating full quantile/distributional processes
- Price to pay is stronger parametric assumptions! (Look at goodness-of-fit statistics (KS, KL, of predicted dist – contrast with non-parametric fit also useful; see (Rothe and Wied, 2013))
- User-written commands in Stata do these estimations for many models (Stephen Jenkins, Nick Cox and colleagues): `smfit`, `dagumfit`, `gb2fit`, `lognfit`, `paretofit`, `fiskfit`, `gammafit`, `betafit`, `gevfit`, `invgammafit`, `weibullfit`) – and relatively easy to program new distributions

Likelihood framework makes several important extensions easy

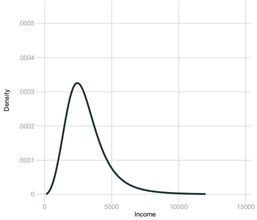
- Censoring (e.g., top-coding in income data, minimum wage)
  - Involves minor modification to likelihood contribution for censored observations ($1 - F(y)$ instead of $f(y)$)
- Endogenous selection
  - Standard selection model à la Heckman (joint normal) (relatively) easily extended to other distributional assumptions in likelihood framework using copula-based representations (Van Kerm, 2013) Details
- Multivariate distributions Details

Example: Modelling income with a Singh-Maddala distribution

Household income in Luxembourg, by educational achievement of father and mother (cf. *inequality of opportunity* analysis)

3-parameters Singh-Maddala distribution often provides good fit to income distributions

- Constrained version of 4-parameter GB2; similar to a Dagum distribution
- Stephen Jenkins' `smfit`
  - (Using here home-brewed `smfit2`—log-linear in covariates)
- Closed-form expressions available for PDF, CDF, percentiles, mode, Gini coefficient, etc. (see `help smfit`)



Fitting a model with no covariates

```
. smfit2 eqinc  , svy stats

initial:       log pseudolikelihood = -4075915.3
alternative:   log pseudolikelihood = -3094364.2
rescale:       log pseudolikelihood = -2875478.5
rescale eq:    log pseudolikelihood = -2514467.8
Iteration 0:   log pseudolikelihood = -2514467.8  (not concave)
Iteration 1:   log pseudolikelihood = -2323390.4
Iteration 2:   log pseudolikelihood = -2229316.3
Iteration 3:   log pseudolikelihood = -2227868.2
Iteration 4:   log pseudolikelihood = -2226987.1
Iteration 5:   log pseudolikelihood = -2226983.8
Iteration 6:   log pseudolikelihood = -2226983.8

ML fit of Singh-Maddala distribution

pweight:  wfinal                                   Number of obs     =      7400
Strata:   <one>                                    Number of strata  =         1
PSU:      <observations>                           Number of PSUs    =      4331
                                                   Population size   = 256957.34
                                                   F(   0,    4331)  =         .
                                                   Prob > F          =         .

------------------------------------------------------------------------------
       eqinc |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
a            |
       _cons |   1.165815   .0510336    22.84   0.000     1.065763    1.265867
-------------+----------------------------------------------------------------
b            |
       _cons |    8.07241   .0652616   123.69   0.000     7.944464    8.200356
-------------+----------------------------------------------------------------
q            |
       _cons |   .2963603   .1372053     2.16   0.031     .0273676    .5653531
------------------------------------------------------------------------------
```

Fitting a model with no covariates

```
. smfit2 eqinc  , svy stats

initial:       log pseudolikelihood = -4075915.3
alternative:   log pseudolikelihood = -3094364.2
rescale:       log pseudolikelihood = -2875478.5
rescale eq:    log pseudolikelihood = -2514467.8
Iteration 0:   log pseudolikelihood = -2514467.8  (not concave)
Iteration 1:   log pseudolikelihood = -2323390.4
Iteration 2:   log pseudolikelihood = -2229316.3
Iteration 3:   log pseudolikelihood = -2227868.2
Iteration 4:   log pseudolikelihood = -2226987.1
Iteration 5:   log pseudolikelihood = -2226983.8
Iteration 6:   log pseudolikelihood = -2226983.8

ML fit of Singh-Maddala distribution

pweight:  wfinal                                   Number of obs    =      7400
Strata:   <one>                                    Number of strata =         1
PSU:      <observations>                           Number of PSUs   =      4331
                                                   Population size  = 256957.34
                                                   F(   0,    4331) =         .
                                                   Prob > F         =         .
```

eqinc	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
a						
_cons	1.165815	.0510336	22.84	0.000	1.065763	1.265867
b						
_cons	8.07241	.0652616	123.69	0.000	7.944464	8.200356
q						
_cons	.2963603	.1372053	2.16	0.031	.0273676	.5653531

	Quantiles	Cumulative shares of total eqinc (Lorenz ordinates)		
1%	697.48948	0.00168		
5%	1.16e+03	0.01396		
10%	1.47e+03	0.03494		
20%	1.88e+03	0.08824		
25%	2.05e+03	0.11933		
30%	2.21e+03	0.15304		
40%	2.52e+03	0.22787	Mode	2.44e+03
50%	2.83e+03	0.31250	Mean	3.16e+03
60%	3.18e+03	0.40753	Std. Dev.	1.75e+03
70%	3.60e+03	0.51448		
75%	3.85e+03	0.57332	Variance	3.07e+06
80%	4.16e+03	0.63659	Half CV^2	0.15358
90%	5.14e+03	0.78172	Gini coeff.	0.27412
95%	6.19e+03	0.87021	p90/p10	3.50079
99%	9.22e+03	0.96184	p75/p25	1.87852

Fitting a model with no covariates

Recover functionals with closed form expressions: `nlcom`

```
.         nlcom exp(_b[b:_cons])

       _nl_1:  exp(_b[b:_cons])
```

eqinc	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_nl_1	3204.817	209.1515	15.32	0.000	2794.887	3614.746

```
.         nlcom (1 - (exp(lngamma(exp(_b[q:_cons])))*exp(lngamma(2*exp(_b[q:_cons]) - 1/exp(_b[a:_cons]))) / (
> exp(lngamma(exp(_b[q:_cons])-1/exp(_b[a:_cons])))*exp(lngamma(2*exp(_b[q:_cons]))) )))

       _nl_1:  1 - (exp(lngamma(exp(_b[q:_cons])))*exp(lngamma(2*exp(_b[q:_cons]) - 1/exp(_b[a:_cons]))) / ( ex
> p(lngamma(exp(_b[q:_cons])-1/exp(_b[a:_cons])))*exp(lngamma(2*exp(_b[q:_cons]))) ))
```

eqinc	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_nl_1	.2741153	.0058528	46.83	0.000	.262644	.2855866

Fitting a model *with* covariates

```
. smfit2 eqinc  , svy a(`vars' ) b(`vars'  ) q(`vars' ) iterate(100) nolog

ML fit of Singh-Maddala distribution

pweight:  wfinal                                 Number of obs     =      7400
Strata:   <one>                                  Number of strata =         1
PSU:      <observations>                         Number of PSUs    =      4331
                                                 Population size  = 256957.34
                                                 F(   6,    4325)  =      1.45
                                                 Prob > F          =    0.1928

------------------------------------------------------------------------------
       eqinc |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
a            |
F_educ_med~m |  -.1364453   .0822718    -1.66   0.097    -.2977403    .0248496
 F_educ_high |   .0213813   .0919567     0.23   0.816    -.1589009    .2016635
   F_educ_99 |  -.5430838   .2265472    -2.40   0.017    -.9872324   -.0989352
M_educ_med~m |   -.183148   .0753087    -2.43   0.015    -.3307915   -.0355045
 M_educ_high |  -.0172072   .1048687    -0.16   0.870    -.2228034    .1883891
   M_educ_99 |   .1134547   .2575129     0.44   0.660    -.3914024    .6183119
       _cons |   1.441756   .0559656    25.76   0.000     1.332035    1.551477
-------------+----------------------------------------------------------------
b            |
F_educ_med~m |   .4197169   .0874385     4.80   0.000     .2482927    .5911412
 F_educ_high |   .3077951   .0919526     3.35   0.001     .1275208    .4880693
   F_educ_99 |   .3242196   .3384133     0.96   0.338    -.3392436    .9876829
M_educ_med~m |   .3441702    .097583     3.53   0.000     .1528576    .5354827
 M_educ_high |   .2360692   .1090313     2.17   0.030      .022312    .4498264
   M_educ_99 |  -.2715757   .1519467    -1.79   0.074     -.569469    .0263176
       _cons |   7.716722   .0501402   153.90   0.000     7.618421    7.815022
-------------+----------------------------------------------------------------
q            |
F_educ_med~m |   .4565336   .2063705     2.21   0.027     .0519417    .8611255
 F_educ_high |  -.0381287   .2099373    -0.18   0.856    -.4497134    .3734559
   F_educ_99 |   .9756004   .6154183     1.59   0.113    -.2309346    2.182135
M_educ_med~m |   .4495618   .2124156     2.12   0.034     .0331184    .8660052
 M_educ_high |   .1248801   .2415592     0.52   0.605    -.3486995    .5984597
   M_educ_99 |  -.4907478   .4019395    -1.22   0.222    -1.278755    .2972593
       _cons |  -.1584176   .1191991    -1.33   0.184     -.392109    .0752737
------------------------------------------------------------------------------
```

Average marginal effects `margins`

Fitting a model *with* covariates

```
. smfit2 eqinc  , svy a(`vars' ) b(`vars'  ) q(`vars' ) iterate(100) nolog

ML fit of Singh-Maddala distribution

pweight:  wfinal                                Number of obs     =      7400
Strata:   <one>                                 Number of strata =          1
PSU:      <observations>                        Number of PSUs    =      4331
                                                Population size  = 256957.34
                                                F(   6,    4325)  =      1.45
                                                Prob > F          =    0.1928

------------------------------------------------------------------------------
       eqinc |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
a            |
F_educ_med~m |  -.1364453   .0822718    -1.66   0.097    -.2977403    .0248496
 F_educ_high |   .0213813   .0919567     0.23   0.816    -.1589009    .2016635
   F_educ_99 |  -.5430838   .2265472    -2.40   0.017    -.9872324   -.0989352
M_educ_med~m |   -.183148   .0753087    -2.43   0.015    -.3307915   -.0355045
 M_educ_high |  -.0172072   .1048687    -0.16   0.870    -.2228034    .1883891
   M_educ_99 |   .1134547   .2575129     0.44   0.660    -.3914024    .6183119
       _cons |   1.441756   .0559656    25.76   0.000     1.332035    1.551477
-------------+----------------------------------------------------------------
b            |
F_educ_med~m |   .4197169   .0874385     4.80   0.000     .2482927    .5911412
 F_educ_high |   .3077951   .0919526     3.35   0.001     .1275208    .4880693
   F_educ_99 |   .3242196   .3384133     0.96   0.338    -.3392436    .9876829
M_educ_med~m |   .3441702    .097583     3.53   0.000     .1528576    .5354827
 M_educ_high |   .2360692   .1090313     2.17   0.030      .022312    .4498264
   M_educ_99 |  -.2715757   .1519467    -1.79   0.074     -.569469    .0263176
       _cons |   7.716722   .0501402   153.90   0.000     7.618421    7.815022
-------------+----------------------------------------------------------------
q            |
F_educ_med~m |   .4565336   .2063705     2.21   0.027     .0519417    .8611255
 F_educ_high |  -.0381287   .2099373    -0.18   0.856    -.4497134    .3734559
   F_educ_99 |   .9756004   .6154183     1.59   0.113    -.2309346    2.182135
M_educ_med~m |   .4495618   .2124156     2.12   0.034     .0331184    .8660052
 M_educ_high |   .1248801   .2415592     0.52   0.605    -.3486995    .5984597
   M_educ_99 |  -.4907478   .4019395    -1.22   0.222    -1.278755    .2972593
       _cons |  -.1584176   .1191991    -1.33   0.184     -.392109    .0752737
------------------------------------------------------------------------------
```

Average marginal effects `margins`

```
. // Quantile effects
. loc ax exp(predict(equation(a)))

. loc bx exp(predict(equation(b)))

. loc qx exp(predict(equation(q)))

. foreach quant of numlist .1(.2).9 {
  2.         margins , expression(`bx'*((1-`quant')^(-1/`qx') - 1)^(1/`ax')) dydx(*)
  3. }

Average marginal effects

Number of strata   =          1                 Number of obs     =      7,400
Number of PSUs     =      4,331                 Population size   = 256,957.34
Model VCE      : Linearized                     Design df         =      4,330

Expression     : exp(predict(equation(b)))*((1-.1)^(-1/exp(predict(equation(q)))) -
                 1)^(1/exp(predict(equation(a))))
dy/dx w.r.t. : F_educ_medium F_educ_high F_educ_99 M_educ_medium M_educ_high M_educ_99

------------------------------------------------------------------------------
             |            Delta-method
             |      dy/dx   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
F_educ_medium |   328.2016   72.46092     4.53   0.000     186.1411    470.2621
  F_educ_high |     534.26    91.4912     5.84   0.000     354.8904    713.6296
    F_educ_99 |  -480.3538   144.3838    -3.33   0.001    -763.4199   -197.2877
M_educ_medium |   161.4287   63.85339     2.53   0.012     36.24334     286.614
  M_educ_high |   305.6255   96.29058     3.17   0.002     116.8467    494.4044
    M_educ_99 |   -98.4401   241.5729    -0.41   0.684    -572.0466    375.1664
------------------------------------------------------------------------------
```

SM fit vs quantile regression

```
. margins , expression(`bx'*((1-0.5)^(-1/`qx') - 1)^(1/`ax')) dydx(*)  post

Average marginal effects

Number of strata   =          1                 Number of obs     =      7,400
Number of PSUs     =      4,331                 Population size   = 256,957.34
Model VCE     : Linearized                      Design df         =      4,330

Expression    : exp(predict(equation(b)))*((1-0.5)^(-1/exp(predict(equation(q)))) -
                1)^(1/exp(predict(equation(a))))
dy/dx w.r.t. : F_educ_medium F_educ_high F_educ_99 M_educ_medium M_educ_high M_educ_99
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
F_educ_medium	704.2529	61.35307	11.48	0.000	583.9695	824.5363
F_educ_high	925.7618	100.7973	9.18	0.000	728.1475	1123.376
F_educ_99	-191.9853	303.2901	-0.63	0.527	-786.5892	402.6186
M_educ_medium	484.5383	65.9422	7.35	0.000	355.2579	613.8188
M_educ_high	544.5631	107.7364	5.05	0.000	333.3446	755.7816
M_educ_99	-249.8355	180.6226	-1.38	0.167	-603.9483	104.2774

```
. estimates store smed

. qreg  eqinc  `vars'   [pw=wfinal] , vce(robust) quant(.5) nolog

Median regression                                Number of obs =      7,400
  Raw sum of deviations 1.54e+08 (about 2797.8333)
  Min sum of deviations 1.39e+08                 Pseudo R2     =     0.1003
```

eqinc	Coef.	Robust Std. Err.	t	P>|t|	[95% Conf.	Interval]
F_educ_medium	752.1333	66.21129	11.36	0.000	622.3403	881.9263
F_educ_high	1073.583	116.4441	9.22	0.000	845.3189	1301.846
F_educ_99	-258.0278	129.1406	-2.00	0.046	-511.1802	-4.875464
M_educ_medium	493.3584	72.26787	6.83	0.000	351.6928	635.024
M_educ_high	682.8489	130.1273	5.25	0.000	427.7623	937.9355
M_educ_99	-280.3833	161.583	-1.74	0.083	-597.132	36.36543
_cons	2337.133	33.57903	69.60	0.000	2271.309	2402.958

```
. estimates table smed qrmed ,  b(%5.2f) se(%5.2f)
```

Variable	smed	qrmed
F_educ_med~m	704.25	752.13
	61.35	66.21
F_educ_high	925.76	1073.58
	100.80	116.44
F_educ_99	-191.99	-258.03
	303.29	129.14
M_educ_med~m	484.54	493.36
	65.94	72.27
M_educ_high	544.56	682.85
	107.74	130.13
M_educ_99	-249.84	-280.38
	180.62	161.58
_cons		2337.13
		33.58

legend: b/se

Marginal effects on other outcome functionals

Marginal effect on conditional distribution dispersion as measured by Gini coefficient (a "Gini regression"?)

Marginal effects on other outcome functionals

Marginal effect on conditional distribution dispersion as measured by Gini coefficient (a “Gini regression”?)

```
. margins , expression(1 - (exp(lngamma(`qx'))*exp(lngamma(2*`qx' - 1/`ax')) / ( exp(lngamma(`qx'-1/`ax'))*exp(
> lngamma(2*`qx')) )))  dydx(*)

Average marginal effects

Number of strata   =         1                Number of obs     =      7,400
Number of PSUs     =     4,331                Population size   = 256,957.34
Model VCE     : Linearized                    Design df         =      4,330

Expression    : 1 - (exp(lngamma(exp(predict(equation(q)))))*exp(lngamma(2*exp(predict(equation(q))) -
                1/exp(predict(equation(a))))) / (
                exp(lngamma(exp(predict(equation(q)))-1/exp(predict(equation(a)))))*exp(lngamma(2*exp(predict(
                > equation(q))))) ))
dy/dx w.r.t. : F_educ_medium F_educ_high F_educ_99 M_educ_medium M_educ_high M_educ_99
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
F_educ_medium	-.0164227	.0119007	-1.38	0.168	-.0397543	.0069088
F_educ_high	-.0011366	.0169179	-0.07	0.946	-.0343043	.0320311
F_educ_99	.0280777	.0458904	0.61	0.541	-.0618909	.1180464
M_educ_medium	-.0039178	.0153331	-0.26	0.798	-.0339784	.0261429
M_educ_high	-.009545	.0186855	-0.51	0.610	-.0461783	.0270882
M_educ_99	.0259968	.0411598	0.63	0.528	-.0546976	.1066911

Allowing for censoring is (almost) trivial

```
. smfit2 eqinc2  , svy a(`vars' ) b( `vars' ) q( `vars')  censored(censored)

initial:       log pseudolikelihood = -4033348.5
alternative:   log pseudolikelihood = -3045664.8
rescale:       log pseudolikelihood =   -2821319
rescale eq:    log pseudolikelihood = -2466588.1
Iteration 0:   log pseudolikelihood = -2466588.1  (not concave)
Iteration 1:   log pseudolikelihood = -2371433.1  (not concave)
Iteration 2:   log pseudolikelihood = -2221615.1
Iteration 3:   log pseudolikelihood = -2166750.9
Iteration 4:   log pseudolikelihood = -2160448.8
Iteration 5:   log pseudolikelihood = -2160316.9
Iteration 6:   log pseudolikelihood = -2160315.4
Iteration 7:   log pseudolikelihood = -2160315.4

ML fit of Singh-Maddala distribution

pweight:  wfinal                                  Number of obs     =      7421
Strata:   <one>                                   Number of strata  =         1
PSU:      <observations>                          Number of PSUs    =      4347
                                                  Population size   = 257330.02
                                                  F(   6,    4341)  =      2.49
                                                  Prob > F          =    0.0208

------------------------------------------------------------------------------
      eqinc2 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
a            |
F_educ_med~m |  -.1573756   .0824665    -1.91   0.056     -.319052    .0043008
 F_educ_high |   .0223444   .0975184     0.23   0.819    -.1688414    .2135302
   F_educ_99 |   -.492445   .2498827    -1.97   0.049    -.9823426   -.0025475
M_educ_med~m |  -.2224373   .0705865    -3.15   0.002    -.3608229   -.0840518
 M_educ_high |   .0165667   .1134237     0.15   0.884    -.2058015     .238935
   M_educ_99 |   .0529433   .3137487     0.17   0.866    -.5621641    .6680508
       _cons |   1.449801   .0544394    26.63   0.000     1.343072     1.55653
-------------+----------------------------------------------------------------
b            |
F_educ_med~m |   .4576982   .1019152     4.49   0.000     .2578924    .6575041
 F_educ_high |   .2799387    .122373     2.29   0.022     .0400252    .5198522
   F_educ_99 |   .1887622   .3347119     0.56   0.573    -.4674439    .8449682
M_educ_med~m |   .4688615   .1129831     4.15   0.000      .247357    .6903661
 M_educ_high |   .2011779   .1326776     1.52   0.130    -.0589379    .4612938
   M_educ_99 |  -.1775786   .2151357    -0.83   0.409    -.5993543    .2441971
       _cons |   7.703581    .048204   159.81   0.000     7.609076    7.798085
-------------+----------------------------------------------------------------
q            |
```

Comparison of P90 quantile coefficient censored/uncensored

Allowing for censoring is (almost) trivial

```
. smfit2 eqinc2  , svy a(`vars' ) b( `vars' ) q( `vars')  censored(censored)

initial:       log pseudolikelihood = -4033348.5
alternative:   log pseudolikelihood = -3045664.8
rescale:       log pseudolikelihood =   -2821319
rescale eq:    log pseudolikelihood = -2466588.1
Iteration 0:   log pseudolikelihood = -2466588.1  (not concave)
Iteration 1:   log pseudolikelihood = -2371433.1  (not concave)
Iteration 2:   log pseudolikelihood = -2221615.1
Iteration 3:   log pseudolikelihood = -2166750.9
Iteration 4:   log pseudolikelihood = -2160448.8
Iteration 5:   log pseudolikelihood = -2160316.9
Iteration 6:   log pseudolikelihood = -2160315.4
Iteration 7:   log pseudolikelihood = -2160315.4

ML fit of Singh-Maddala distribution

pweight:   wfinal                                   Number of obs     =      7421
Strata:    <one>                                    Number of strata  =         1
PSU:       <observations>                           Number of PSUs    =      4347
                                                    Population size   = 257330.02
                                                    F(    6,    4341)  =      2.49
                                                    Prob > F           =    0.0208

------------------------------------------------------------------------------
      eqinc2 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
a            |
F_educ_med~m |  -.1573756   .0824665    -1.91   0.056     -.319052    .0043008
 F_educ_high |   .0223444   .0975184     0.23   0.819    -.1688414    .2135302
   F_educ_99 |   -.492445   .2498827    -1.97   0.049    -.9823426   -.0025475
M_educ_med~m |  -.2224373   .0705865    -3.15   0.002    -.3608229   -.0840518
 M_educ_high |   .0165667   .1134237     0.15   0.884    -.2058015     .238935
   M_educ_99 |   .0529433   .3137487     0.17   0.866    -.5621641    .6680508
       _cons |   1.449801   .0544394    26.63   0.000     1.343072     1.55653
-------------+----------------------------------------------------------------
b            |
F_educ_med~m |   .4576982   .1019152     4.49   0.000     .2578924    .6575041
 F_educ_high |   .2799387    .122373     2.29   0.022     .0400252    .5198522
   F_educ_99 |   .1887622   .3347119     0.56   0.573    -.4674439    .8449682
M_educ_med~m |   .4688615   .1129831     4.15   0.000      .247357    .6903661
 M_educ_high |   .2011779   .1326776     1.52   0.130    -.0589379    .4612938
   M_educ_99 |  -.1775786   .2151357    -0.83   0.409    -.5993543    .2441971
       _cons |   7.703581    .048204   159.81   0.000     7.609076    7.798085
-------------+----------------------------------------------------------------
q            |
```

Comparison of P90 quantile coefficient censored/uncensored

```
. estimates table nocen cen ,  b(%7.2f) se(%7.2f)
```

Variable	nocen	cen
F_educ_med~m	838.21	969.85
	146.95	177.06
F_educ_high	1271.79	1638.26
	219.65	281.58
F_educ_99	-274.93	-114.24
	512.80	568.50
M_educ_med~m	498.81	567.39
	150.80	179.11
M_educ_high	719.07	920.49
	205.68	279.96
M_educ_99	-268.21	-282.34
	418.42	509.10

A sample selection model: earnings distributions with endogenous LM partici-pation

```
. selsmfit py010g  [pw=wfinal]    ,  a(`vars' ) b( `vars' ) q( `vars') m(atwork = `vars' bothparents) robust cl
> uster(hid)

initial:       log pseudolikelihood =     -<inf>  (could not be evaluated)
feasible:      log pseudolikelihood = -3264348.3
rescale:       log pseudolikelihood = -2961568.9
rescale eq:    log pseudolikelihood = -2673775.5
Iteration 0:   log pseudolikelihood = -2673775.5  (not concave)
Iteration 1:   log pseudolikelihood = -2672623.7  (not concave)
Iteration 2:   log pseudolikelihood = -2550380.8  (not concave)
Iteration 3:   log pseudolikelihood =   -2511135
Iteration 4:   log pseudolikelihood = -2508669.5  (not concave)
Iteration 5:   log pseudolikelihood =   -2477948  (not concave)
Iteration 6:   log pseudolikelihood = -2449200.9  (not concave)
Iteration 7:   log pseudolikelihood = -2445995.5  (not concave)
Iteration 8:   log pseudolikelihood = -2444528.7  (not concave)
Iteration 9:   log pseudolikelihood = -2443819.7  (not concave)
Iteration 10:  log pseudolikelihood = -2443404.4
Iteration 11:  log pseudolikelihood = -2443388.1
Iteration 12:  log pseudolikelihood = -2442068.4  (not concave)
Iteration 13:  log pseudolikelihood = -2441707.1
Iteration 14:  log pseudolikelihood = -2441560.9
Iteration 15:  log pseudolikelihood = -2441322.4
Iteration 16:  log pseudolikelihood = -2441298.3
Iteration 17:  log pseudolikelihood = -2441296.5
Iteration 18:  log pseudolikelihood = -2441296.5

ML fit of Singh-Maddala distribution with endogenous sample selection

                                                Number of obs     =      7,390
                                                Wald chi2(6)      =      10.02
Log pseudolikelihood = -2441296.5               Prob > chi2       =     0.1238

                                     (Std. Err. adjusted for 4,339 clusters in hid)
```

	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
a						
F_educ_medium	.008689	.120015	0.07	0.942	-.2265361	.243914
F_educ_high	-.0579292	.1384985	-0.42	0.676	-.3293813	.213523
F_educ_99	.3224045	.317147	1.02	0.309	-.2991922	.9440012
M_educ_medium	.1241578	.0982026	1.26	0.206	-.0683158	.3166314
M_educ_high	.1270647	.1742204	0.73	0.466	-.214401	.4685304
M_educ_99	.1415868	.527333	0.27	0.788	-.8919668	1.17514
_cons	.6053633	.1244153	4.87	0.000	.3615139	.8492127
b						
F_educ_medium	.7549241	.3435027	2.20	0.028	.0816711	1.428177
F_educ_high	.7792893	.4685174	1.66	0.096	-.1389881	1.697567
F_educ_99	-.6508057	.4384438	-1.48	0.138	-1.51014	.2085283
M_educ_medium	-.1031399	.3576386	-0.29	0.773	-.8040986	.5978188
M_educ_high	-.1623631	.5355654	-0.30	0.762	-1.212052	.8873257
M_educ_99	-.486774	.5764455	-0.84	0.398	-1.616586	.6430383
_cons	10.75241	.4715018	22.80	0.000	9.828283	11.67654
q						
F_educ_medium	.6947501	.4597264	1.51	0.131	-.206297	1.595797
F_educ_high	.5023466	.6519717	0.77	0.441	-.7754945	1.780188
F_educ_99	-.8657182	.6306449	-1.37	0.170	-2.101759	.3703231
M_educ_medium	-.2181679	.4958463	-0.44	0.660	-1.190009	.753673
M_educ_high	-.1593864	.7391883	-0.22	0.829	-1.608169	1.289396
M_educ_99	-.3879608	.9091663	-0.43	0.670	-2.169894	1.393973
_cons	.1033944	.6389945	0.16	0.871	-1.149012	1.355801
m						
F_educ_medium	-.0782107	.0574784	-1.36	0.174	-.1908663	.034445
F_educ_high	-.0971069	.0956444	-1.02	0.310	-.2845665	.0903528
F_educ_99	.0606312	.1636486	0.37	0.711	-.2601142	.3813766
M_educ_medium	-.1614078	.0680914	-2.37	0.018	-.2948646	-.027951
M_educ_high	-.2042567	.1196805	-1.71	0.088	-.4388262	.0303129
M_educ_99	-.1729523	.2001426	-0.86	0.388	-.5652247	.21932
bothparents	-.1354852	.0492244	-2.75	0.006	-.2319633	-.039007
_cons	-.5604291	.0560934	-9.99	0.000	-.6703702	-.450488
theta						
_cons	-13.57081	3.75177	-3.62	0.000	-20.92414	-6.217475

More complex likelihood function (with 5 equations), but same use

A sample selection model: earnings distributions with endogenous LM participation

```
. selsmfit py010g  [pw=wfinal]    ,  a(`vars' ) b( `vars' ) q( `vars') m(atwork = `vars' bothparents) robust cl
> uster(hid)

initial:       log pseudolikelihood =     -<inf>  (could not be evaluated)
feasible:      log pseudolikelihood = -3264348.3
rescale:       log pseudolikelihood = -2961568.9
rescale eq:    log pseudolikelihood = -2673775.5
Iteration 0:   log pseudolikelihood = -2673775.5  (not concave)
Iteration 1:   log pseudolikelihood = -2672623.7  (not concave)
Iteration 2:   log pseudolikelihood = -2550380.8  (not concave)
Iteration 3:   log pseudolikelihood =   -2511135
Iteration 4:   log pseudolikelihood = -2508669.5  (not concave)
Iteration 5:   log pseudolikelihood =   -2477948  (not concave)
Iteration 6:   log pseudolikelihood = -2449200.9  (not concave)
Iteration 7:   log pseudolikelihood = -2445995.5  (not concave)
Iteration 8:   log pseudolikelihood = -2444528.7  (not concave)
Iteration 9:   log pseudolikelihood = -2443819.7  (not concave)
Iteration 10:  log pseudolikelihood = -2443404.4
Iteration 11:  log pseudolikelihood = -2443388.1
Iteration 12:  log pseudolikelihood = -2442068.4  (not concave)
Iteration 13:  log pseudolikelihood = -2441707.1
Iteration 14:  log pseudolikelihood = -2441560.9
Iteration 15:  log pseudolikelihood = -2441322.4
Iteration 16:  log pseudolikelihood = -2441298.3
Iteration 17:  log pseudolikelihood = -2441296.5
Iteration 18:  log pseudolikelihood = -2441296.5

ML fit of Singh-Maddala distribution with endogenous sample selection

                                                Number of obs     =      7,390
                                                Wald chi2(6)      =      10.02
Log pseudolikelihood = -2441296.5               Prob > chi2       =     0.1238

                                   (Std. Err. adjusted for 4,339 clusters in hid)
```

	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
a						
F_educ_medium	.008689	.120015	0.07	0.942	-.2265361	.243914
F_educ_high	-.0579292	.1384985	-0.42	0.676	-.3293813	.213523
F_educ_99	.3224045	.317147	1.02	0.309	-.2991922	.9440012
M_educ_medium	.1241578	.0982026	1.26	0.206	-.0683158	.3166314
M_educ_high	.1270647	.1742204	0.73	0.466	-.214401	.4685304
M_educ_99	.1415868	.527333	0.27	0.788	-.8919668	1.17514
_cons	.6053633	.1244153	4.87	0.000	.3615139	.8492127
b						
F_educ_medium	.7549241	.3435027	2.20	0.028	.0816711	1.428177
F_educ_high	.7792893	.4685174	1.66	0.096	-.1389881	1.697567
F_educ_99	-.6508057	.4384438	-1.48	0.138	-1.51014	.2085283
M_educ_medium	-.1031399	.3576386	-0.29	0.773	-.8040986	.5978188
M_educ_high	-.1623631	.5355654	-0.30	0.762	-1.212052	.8873257
M_educ_99	-.486774	.5764455	-0.84	0.398	-1.616586	.6430383
_cons	10.75241	.4715018	22.80	0.000	9.828283	11.67654
q						
F_educ_medium	.6947501	.4597264	1.51	0.131	-.206297	1.595797
F_educ_high	.5023466	.6519717	0.77	0.441	-.7754945	1.780188
F_educ_99	-.8657182	.6306449	-1.37	0.170	-2.101759	.3703231
M_educ_medium	-.2181679	.4958463	-0.44	0.660	-1.190009	.753673
M_educ_high	-.1593864	.7391883	-0.22	0.829	-1.608169	1.289396
M_educ_99	-.3879608	.9091663	-0.43	0.670	-2.169894	1.393973
_cons	.1033944	.6389945	0.16	0.871	-1.149012	1.355801
m						
F_educ_medium	-.0782107	.0574784	-1.36	0.174	-.1908663	.034445
F_educ_high	-.0971069	.0956444	-1.02	0.310	-.2845665	.0903528
F_educ_99	.0606312	.1636486	0.37	0.711	-.2601142	.3813766

Comparison of median regression with/without selection correction

```
. estimates table nosel sel ,  b(%5.2f) se(%5.2f)
```

Variable	nosel	sel
F_educ_med~m	13731.41	15045.43
	1549.90	2095.02
F_educ_high	18464.21	22146.73
	2661.19	3294.31
F_educ_99	-4331.88	-9.68
	7317.12	8746.21
M_educ_med~m	6855.53	3621.08
	1683.84	2216.25
M_educ_high	3855.99	-1411.96
	2579.26	3458.65
M_educ_99	-1902.05	-1.0e+04
	16080.64	9637.90
bothparents		(omitted)

legend: b/se

Marginalisation: deriving unconditional distributions

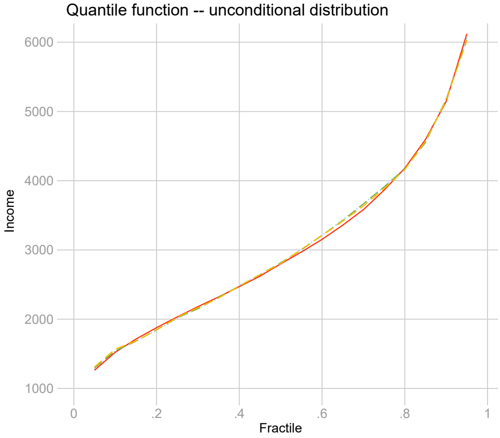
1. Fit the model (possibly allowing for censoring, selection)
2. Generate (equally-spaced), say, 99 predicted quantiles from the model
3. Vectorize the $N \times 99$ predicted quantiles into V (`reshape` or some simple Mata operations)
4. Calculate quantiles of V (or CDF or whatever functional)

Procedure does not depend on specific conditional distribution model used.

(Can easily be used to generate counterfactual distributions. (Not shown today.))

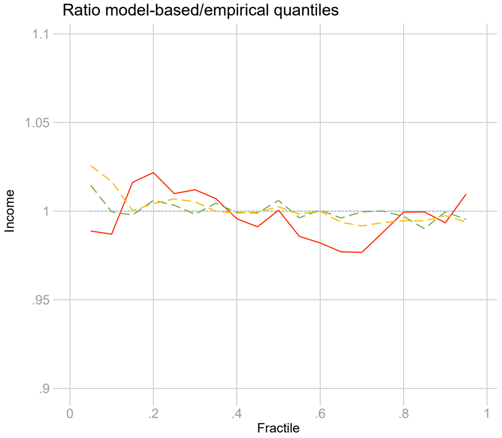
Marginalisation: comparison with different conditional quantile prediction models

- conditional Singh-Maddala
- quantile regression
- distribution regression



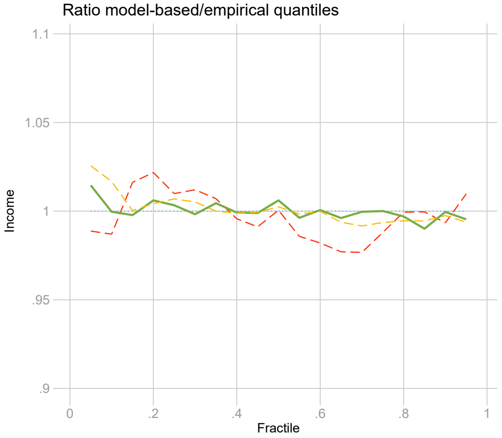
Marginalisation: comparison with different conditional quantile prediction models

- conditional Singh-Maddala
- quantile regression
- distribution regression



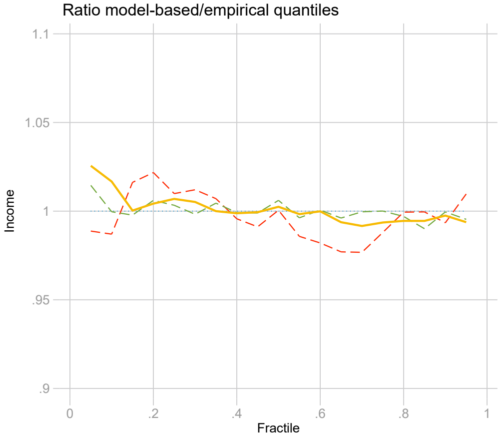
Marginalisation: comparison with different conditional quantile prediction models

- conditional Singh-Maddala
- quantile regression
- distribution regression



Marginalisation: comparison with different conditional quantile prediction models

- conditional Singh-Maddala
- quantile regression
- distribution regression



Envoi

1. Conditional likelihood models are *easy*
2. ... and already packaged in a collection of user-written commands on SSC
3. `margins`, `nlcom`, `predictnl` are essential here
4. Combine advantages of quantile regression and distribution regression...
5. ... at the cost of imposing parametric restrictions (whose credibility is often an empirical question)
6. Interest in handling censoring, selection, joint distributions with simple, familiar estimators

References

Autor, D. H., Katz, L. F. and Kearney, M. S. (2005), Rising wage inequality: The role of composition and prices, NBER Working Paper 11628, National Bureau of Economic Research, Cambridge MA, USA.

Biewen, M. and Jenkins, S. P. (2005), 'A framework for the decomposition of poverty differences with an application to poverty differences between countries', *Empirical Economics* **30**(2), 331–358. **URL:** *http://dx.doi.org/10.1007/s00181-004-0229-1*

Briseño Sanchez, G., Hohberg, M., Groll, A. and Kneib, T. (2020), 'Flexible instrumental variable distributional regression', *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **183**(4), 1553–1574.

References ii

Chernozhukov, V., Fernández-Val, I. and Galichon, A. (2009), 'Improving point and interval estimators of monotone functions by rearrangement', *Biometrika* **96**(3), 559–575.

Chernozhukov, V., Fernandez-Val, I. and Melly, B. (2013), 'Inference on counterfactual distributions', *Econometrica* **81**(6), 2205–2268.
URL: *http://dx.doi.org/10.3982/ECTA10582*

Donald, S. G., Green, D. A. and Paarsch, H. J. (2000), 'Differences in wage distributions between Canada and the United States: An application of a flexible estimator of distribution functions in the presence of covariates', *Review of Economic Studies* **67**(4), 609–633.

Foresi, S. and Peracchi, F. (1995), 'The conditional distribution of excess returns: An empirical analysis', *Journal of the American Statistical Association* **90**(430), 451–466.

Jäntti, M., Sierminska, E. M. and Van Kerm, P. (2015), Modeling the joint distribution of income and wealth, *in* T. Garner and K. Short, eds, 'Measurement of Poverty, Deprivation, and Economic Mobility', number 23 *in* 'Research on Economic Inequality', Emerald Group Publishing Limited, pp. 301–327.

References iii

Jenkins, S. P., Burkhauser, R. V., Feng, S. and Larrimore, J. (2011), 'Measuring inequality using censored data: a multiple imputation approach', *Journal of the Royal Statistical Society: Series A (Statistics in Society)* **174**(1), 63–81.

Koenker, R. and Bassett, G. (1978), 'Regression quantiles', *Econometrica* **46**(1), 33–50. **URL:** *http://www.jstor.org/stable/1913643*

Noufaily, A. and Jones, M. C. (2013), 'Parametric quantile regression based on the generalized gamma distribution', *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **62**(5), 723–740. **URL:** *http://dx.doi.org/10.1111/rssc.12014*

Rothe, C. and Wied, D. (2013), 'Misspecification testing in a class of conditional distributional models', *Journal of the American Statistical Association* **108**(501), 314–324.

Royston, P. (2001), 'Flexible alternatives to the Cox model, and more', *Stata Journal* (1), 1–28.

Van Kerm, P. (2013), 'Generalized measures of wage differentials', *Empirical Economics* **45**(1), 465–482. (published online, DOI:10.1007/s00181-012-0608-y).

References iv

Van Kerm, P. (2016), Distribution regression made easy, United Kingdom Stata Users' Group Meetings 2016 13, Stata Users Group.
URL: *https://ideas.repec.org/p/boc/usug16/13.html*

Van Kerm, P., Yu, S. and Choe, C. (2017), 'Decomposing quantile wage gaps: a conditional likelihood approach', *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **65**(4), 507–527.
URL: *http://onlinelibrary.wiley.com/doi/10.1111/rssc.12137/pdf*

Conditional likelihood models with endogenous selection

Let s denote binary participation (outcome y only observed if $s = 1$). Assume $s = 1$ if $s^* > 0$ and $s = 0$ otherwise. s^* is latent propensity to be observed.

Assume pair (y, s^*) is jointly distributed H and express H using its copula formulation

$$H(y, s^*) = \Psi(F(y), G(s^*))$$

where F is outcome distribution, G is latent participation distribution (typically Gaussian), and Ψ is a parametric copula function.

Everything is parametric (need to select a copula) and can be estimated using maximum likelihood (Van Kerm, 2013)

Derivation of conditional functionals (incl., quantiles) from $\hat{F}$ remains trivial

Conditional multivariate likelihood models

The same modelling approach can be used to build conditional multivariate models

Assume pair (y, z) is jointly distributed H and express H using its copula formulation

$$H(y, z) = \Psi(F(y), G(z))$$

where F and G are outcome distributions (of the same or different family) and Ψ is a copula function.

Everything is parametric and can be estimated using maximum likelihood (see Jäntti et al. (2015) for a model of the joint distribution of income and wealth)