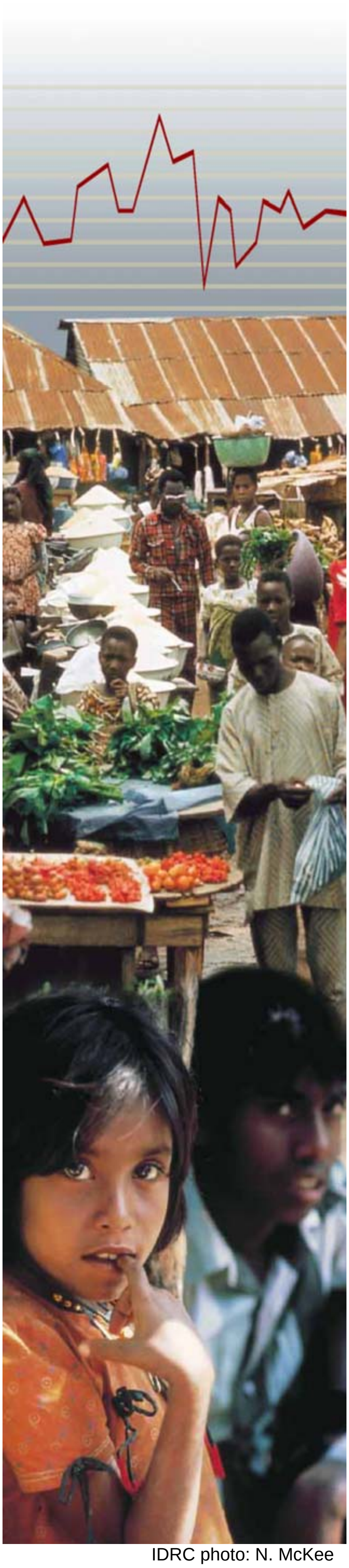

IDRC photo: N. McKee

Politiques Economiques et Pauvreté
Poverty and Economic Policy

PMMA Working Paper
2006-10

DAD: A Software for Poverty and Distributive Analysis

Abdelkrim Araar
Jean-Yves Duclos

May 2006

Araar, Abdelkarim (Université Laval)
aabd@ecn.ulaval.ca
Duclos, Jean-Yves (Université Laval)
jyves@ecn.ulaval.ca

Abstract

DAD[1] is designed to facilitate the analysis and the comparisons of social welfare, inequality, poverty and equity across distributions of living standards and using disaggregated data. It is freely distributed. *DAD*'s features include the estimation of a large number of indices and curves that are useful for distributive comparisons as well as the provision of various statistical tools to enable statistical inference. Many of the features are useful for estimating the impact of programs (and reforms to these programs) on poverty and equity.

Keywords: Poverty, equity, inequality, statistical inference, software.

JEL Classification: L86, I30, I32, D63, C12.

[1]The software *DAD* was conceived by Jean-Yves Duclos* and Araar Abdelkrim** and was programmed in Java by Araar Abdelkrim and Carl Fortin.

*Département d'économique and CIRPÉE, Pavillon De Sève, Université Laval, Sainte-Foy, Québec, Canada, G1K 7P4; Email: jyves@ecn.ulaval.ca; fax: 1-418-656-7798; phone: 1-418-656-7096

**Département d'économique and CIRPÉE, Pavillon De Sève, Université Laval, Sainte-Foy, Québec, Canada, G1K 7P4; Email: aabd@ecn.ulaval.ca; fax: 1-418-656-7798;

1. Introduction

DAD – which stands for "Distributive analysis/Analyse distributive" – is designed to facilitate the analysis and the comparisons of social welfare, inequality, poverty and equity using micro (or disaggregated) data. It is freely distributed and its use does not require purchasing any commercial software. *DAD*'s features include the estimation of a large number of indices and curves that are useful for distributive comparisons. It also provides various statistical tools to enable statistical inference. Many of *DAD*'s features are useful for estimating the impact of programs (and reforms to these programs) on poverty and equity.

The first version of *DAD* was launched in September 1998. It initially came to life following a request by the Canadian International Development Research Centre (IDRC) to Université Laval to support research then carried out in Africa in the context of the IDRC's programme on the Micro Impacts of Macro-economic and Adjustment Policies (MIMAP). Improved versions of *DAD* subsequently appeared as errors and bugs were corrected and as attempts were made to make it more reliable, more flexible and broader in scope. The current version is 4.4.1 – version 4.4.8 should be launched in June 2006.

Several factors motivated us in the process of building *DAD*. First, there seemed to be an ever increasing need for developing-country analysts to carry out poverty and inequality "profiles". Much of development policy is indeed now assessed through poverty criteria, and this is carried out among other things through the elaboration of poverty assessments, poverty reduction strategy papers (the now well-known *PRSP*'s), poverty and social impact analyses, *etc*.. Much of these distributive assessments had earlier typically been done by foreign consultants and by international organizations' technical staff. Little was left in the form of national capacity building and local empowerment following these largely external exercises. Local researchers and national policy analysts typically felt alienated by these poverty assessments which they often did not understand and which they could not usually influence. To break that segregation between foreign experts and local policy makers and analysts, it seemed useful to introduce tools that would benefit developing country analysts pedagogically and operationally.

Second, micro-data accessibility was increasingly becoming less of a problem to developing-country researchers. This followed what had occurred in more developed countries some 15 to 20 years earlier when data tapes and records started to circulate widely in research centers and universities. This was made possible in large part by the amazing increase in storage and processing speed that the computer revolution was creating. Developing-country analysts were gaining from the same advances, though with some lag due to tighter resource constraints. Furthermore, in addition to the computing and technical demands that handling large data sets involved, developing country analysts often had to deal with data accessibility difficulties. This meant *inter alia* having to face skepticism and rent-seeking behavior from statistical agencies and international organization staff when requesting access to data that were supposed in principle to be public. That problem had also become less severe by the end of the 1990's, in part due to outside pressure. To process and analyze these data then typically became the next barrier to break.

Third, much of distributive analysis was (and is still) handled as if it was not subject to statistical imprecision. Indeed, a considerable amount of energy and resources seems to be wasted in discussions of poverty and inequality "results" that cannot be trusted on formal statistical grounds. Even changes in poverty headcounts of around 4% or 5% are often statistically insignificant within the usual statistical precision criteria. Needless to say, the efforts deployed by analysts and policy makers to account for variations of less than 1% or 2% (as often occurs) in poverty rates are typically a pure loss of resources. This unfortunate state of affairs needed to be remedied by a much greater use of appropriate statistical techniques. Though conceptually relatively simple, the use of these techniques nevertheless required reading through some technical literature as well as writing tedious computer programmes. *DAD* was in large part written to help bypass these hurdles. Achieving this meant clearing the ground of statistically insignificant results and leaving more time and resources for the interpretation of those distributive findings that were statistically significant.

DAD was thus conceived to help policy analysts and researchers analyze poverty and equity using disaggregated data. An overriding operational objective was to try to make *DAD*'s environment as accessible and as user friendly as possible. Carl Fortin, our co-author, convincingly argued from the start that we should program *DAD* in the Java programming language. An object-oriented language, Java created a new paradigm of platform independence: once written, Java applications could run on any operating system as well as on the internet. Conceived by Sun in 1995, Java could still be considered in 1998 to be an infant programming language[2]. By now, however, it has become an important pillar of the programming and internet industry. To make *DAD* completely free of charge, we also chose not to tie its use to statistical commercial softwares such as Excel, SPSS or STATA. We therefore opted to design *DAD* from scratch using some of Java's packages as building blocks.

To make *DAD* as user friendly as possible, we use pop-up application windows and spreadsheets as the main working tools. This enables users to visualize a lot of information at a glance, and to manage that information easily. Most of the relevant variables and options needed for running applications can be selected from single application windows. *DAD*'s use

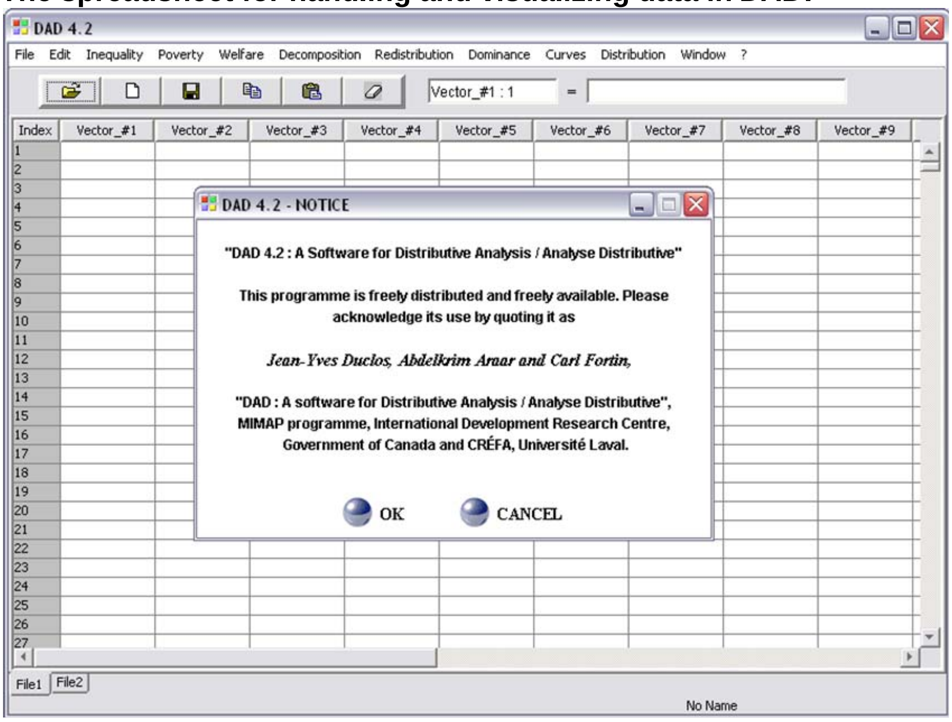
[2]For further information on Java's development and structure, see Deitel and Deitel (2003)'s introductory book, or Chapter 1 of Lewis and Loftus (2000).

of spreadsheets has the advantage of displaying the entire data sets to be used. Small data sets can easily be entered manually. Changes to cell values can be made directly on the spreadsheet. The results of operations on data vectors can be checked easily. *DAD* also allows loading two data bases simultaneously, and to display each of these two data bases alternatively on the spreadsheet. This makes it easy to carry out applications with either one or two data bases. That structure also enables *DAD* to account for whether the data bases are independent when it comes to computing standard errors on distributive estimators that use information from two samples.

2. Loading, editing and saving databases in *DAD*

DAD's databases are displayed on spreadsheets similar to those of SPSS, STATA, or Microsoft's Excel – see Figure 1. Every line in a sheet represents one observation or one data "record". Typically, an observation consists of one of the sampling or statistical units that were drawn into a survey. In distributive analyses, a sampling unit is often a household since it is households that are typically the last sampling units in surveys. When observations represent households, there will thus be as many lines or observations in the data as there are households drawn into the household survey. The statistical units (or units of interest) are usually (for ethical reasons) the individuals. Even though the sampling units originally drawn into the survey may have been the households, data sets are sometimes re-organized in such a way that each individual in a household is assigned its own line of data. There will then be as many observations in a data set as there are individuals found in the households.

Figure 1. The spreadsheet for handling and visualizing data in *DAD*.



A database used in *DAD* is then a matrix (a set of columns) whose length is the number of observations discussed above and whose width is the number of variables contained in the database. Each column displays the values of a variable. A variable has as many values as there are observations in the database. All columns in *DAD* are therefore of the same length. Variable values can have a float format – indicating, for example, the level of household income – or an integer format – showing for instance the socio-economic category to which a household belongs.

There are several options for entering data into *DAD*. The first one is to create a new database in *DAD* and then enter the variable values manually. This can be useful for exploratory or pedagogical purposes. Clearly, however, this option not convenient for entering large databases into *DAD*. A second option for reading existing data bases into *DAD* is done by using well-known copy/paste facilities. Before doing this, however, a new data base must be created in *DAD* and then assigned a number of observations (or size) that corresponds to the length of the variables that will be copied/pasted.

The third possibility for entering data into *DAD* is typically more reliable (and also faster) than the first two and involves two steps. The first step saves the database in an ASCII (or a text) format. The way in which this is done in practice depends on the software in which the data were previously handled. *DAD*'s Users Manual gives examples of such output procedures for several common commercial softwares. One fast alternative to this is offered by the use of STAT/TRANSFER (note however that this requires buying a license), which transforms databases rapidly from the most popular formats into an ASCII format. Once the database is in ASCII format, it can easily be imported using *DAD*'s Data Import Wizard. The wizard ensures *inter alia* that the imported database does not contain missing or unreadable values. Once the data are read in *DAD*, they can be submitted to a number of arithmetical and logical operations, variable names can be added or changed, and new variables can be created. Databases can subsequently be saved in *DAD*'s preferred ASCII format (identified by the extension *.daf*).

As already mentioned, many of *DAD*'s applications can use simultaneously two databases. To use a second database, the user should first activate a second file by clicking on the button **File2**, and then follow the same procedures as for loading a first file.

3. Inputting the sampling design information

The process of generating random surveys usually displays four important characteristics:

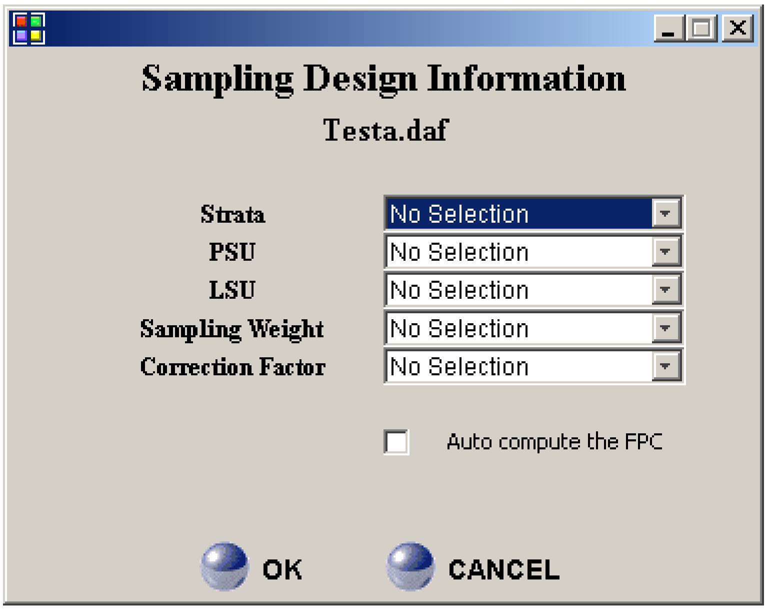
- the base of sampling units (the base from which sample observations are drawn) is stratified;

- sampling is multi-staged, generating clusters of observations;
- observations come with sampling weights, also called inverse probability weights;
- observations may have been drawn with or without replacement;
- observations often provide aggregate information on a number of units of interest (such as the different individuals that live in a household).

Recent versions of *DAD* enable taking that structure into account in the estimation of the various distributive statistics as well as in the computation of the sampling distributions of these statistics.

When a data file is first read or typed into *DAD*, the survey design assigned to it by default is Simple Random Sampling. This supposes that the observations were independently selected from a large base of sampling units. This, however, is rarely how surveys are designed and implemented. Once the data are loaded, the exact sampling design structure can nevertheless be easily specified. This is done using the Set Sample Design dialogue box. Specifying the sample design structure can involve letting *DAD* know about (up to) 5 vectors (see Figure 2).

Figure 2. The Set Sample Design window in *DAD*.



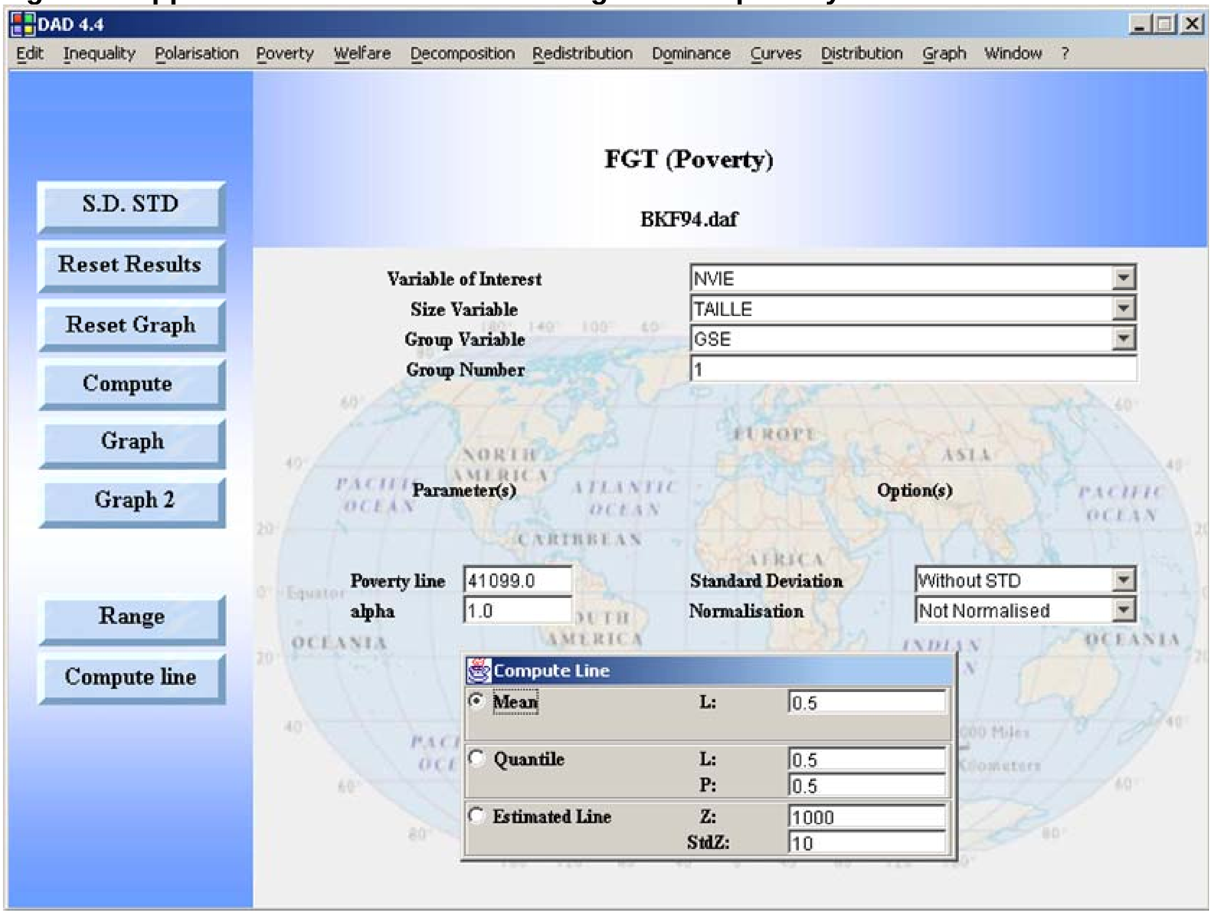
- STRATA: this specifies the name of the variable (in an integer format) that contains the Stratum identifiers.
- PSU: this specifies the name of the variable (in an integer format) that contains the identifiers for the Primary Sampling Units.
- LSU: this specifies the name of the variable (in an integer format) that contains the identifiers for the Last Sampling Units.
- SAMPLING WEIGHT: this specifies the name of the Sampling Weights variable.

- CORRECTION FACTOR: this provides *DAD* with a Finite Population Correction variable.

4. Applications in *DAD*: basic procedures

Once data have been read into *DAD* and that the sampling design has been specified, the field is wide open for the estimation of distributive statistics and for performing distributive tests. For every application programmed in *DAD*, there is a specific application window that facilitates the specification of variables, parameters and options to generate the desired distributive statistics. For example, Figure 3 shows the specific application window for computing the FGT poverty index with one distribution. There is a separate specific window for the case of two distributions. The list of all applications available in *DAD*'s current version 4.4 appears in Tables 3 and 4 in Annex A.

Figure 3. Application window for estimating the FGT poverty index – one distribution.



Most application windows, including that of Figure 3, are divided into three panels. The first panel is used to specify the relevant database variables needed for the estimation. The second panel (generally at the bottom of the application window) specifies the parameter values and options to be used by the estimator – examples include the level of inequality aversion, the value of the poverty line and the percentile to be considered as well as whether indices should be normalized and whether statistical inference should be performed. The third panel activates buttons in order that various types of results may be generated. Some application windows can also generate popping-up dialogue boxes. One example of this can be found when clicking on the **Compute line** button in the Poverty|FGT application window. This serves to specify the manner in which the poverty line should be (or was) estimated.

The following basic variables are typically required for carrying out *DAD*'s computations.

- VARIABLE OF INTEREST. This is the variable that usually captures living standards. It can represent, for instance, expenditures per adult equivalent, calorie intake, normalized height-for-age scores for children, *etc*..
- SIZE VARIABLE. This refers to the "ethical" of physical size of the observation. For the computation of many distributive statistics, we will indeed wish to take into account how many relevant individuals (or statistical units) are found in a given observation. We might, for instance, wish to estimate inequality across individuals or the proportion of children who are poor. Individuals and children will then be respectively the statistical units of interest. Households do differ, however, in their size or in the number of children they contain. *DAD* takes this into account through the use of the SIZE VARIABLE. When an observation represents a household, computing inequality across individuals requires specifying household size as the SIZE VARIABLE, whereas computing poverty among children requires putting the number of children in the household as the SIZE VARIABLE. If the statistics of interest were the proportion of households in poverty, then no SIZE VARIABLE would be needed.
- GROUP VARIABLE. (This should be used in combination with GROUP NUMBER.) It is often useful to limit some distributive analysis to some population subgroup. We might for example wish to estimate poverty within a country's rural area or within the group of public workers. One way to do this is to set SIZE VARIABLE to zero for all of the observations that fall outside these groups of interest. Another way is by defining a GROUP VARIABLE whose values will allow *DAD* to identify which are the observations of interest.
- GROUP NUMBER. GROUP NUMBER tells *DAD* on which value of the GROUP VARIABLE to condition the computation of some distributive statistics. The value for GROUP NUMBER should be an integer. For example, rural households might be assigned a value of 1 for some variable denoted as *region*. Setting GROUP VARIABLE to *region* and GROUP NUMBER to 1 makes *DAD* know that we wish the distributive statistics to be computed only within the group the rural households. For some of the more frequently used applications, it is also possible to choose simultaneously all of the group numbers and to perform at once the appropriate estimation for the total population as well as for each group. These applications deal with the S-Gini, the Atkinson and the FGT indices.

- SAMPLING WEIGHT. Sampling weights are the inverse of the sampling rate. They are best specified once and for all using the **Set Sample Design** window (as discussed above). Distributive statistics (but not necessarily their sampling distribution and standard errors) will be left unchanged, however, if no variable is given for Sample Weight (in **Set Sample Design** window) and if the product of the sampling weight and size variables is subsequently specified as the SIZE VARIABLE in the relevant application windows.

DAD's applications with two distributions can be launched after having loaded two databases. Each time one launches an application that can support two distributions, the dialog box, shown in Figure 4, opens to allow the user to specify the desired number of distributions to be used as well as the name of the databases for these distributions. The application window for two distributions is very similar to that for one. The main difference is the addition of a second panel to specify the relevant variables to be used for the second distribution. The application for two distributions generally serves to compute distributive differences across the two distributions. For curve applications with two distributions, for instance, differences between the curves of the two distributions can usually be drawn. In Annex A, we list most of *DAD*'s applications. Note that the modules currently developed for the popular Stata software (see Annex B for some more information) cover only a small part of the distributive analysis provided by *DAD* ; in most cases, the options of the STATA modules are quite limited [3].

Figure 4. Choosing between configurations of one or two distributions.

[3] Abdelkrim Araar has started developing a new comprehensive Stata package for distributive analysis (the 'DASP' package). Some modules of the package have already been published.

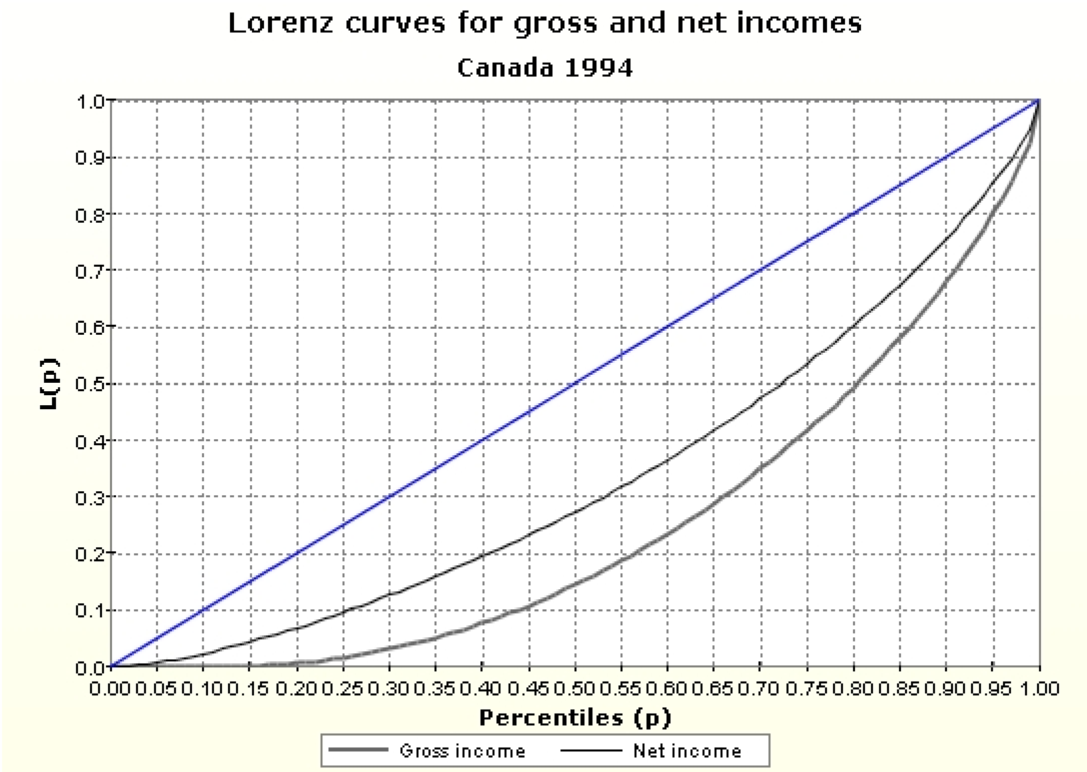
5. Curves

DAD has built-in tools that facilitate the use of curves to display distributive information. Say, for instance, that we wish to graph a Lorenz curve. We can compare it to the 45° line to observe by much income shares differ from population shares. This is done by following these steps:

- From the main menu, select the submenu : Curve | Lorenz. Indicate that the number of distributions equals one.
- After choosing the application variables, click on the button **Graph** to draw the first Lorenz curve.
- If you would like to draw another Lorenz curve for another variable of interest, return to the Lorenz window application, re-initialize the variable of interest and click again on the button **Graph**.
- When the graph window appears, click on the button **Draw all** to plot all of the curves.
- If you wish to draw the 45° line, select (from the main menu of the graph window) Tools | Properties, and activate the option DRAW THE 45° LINE

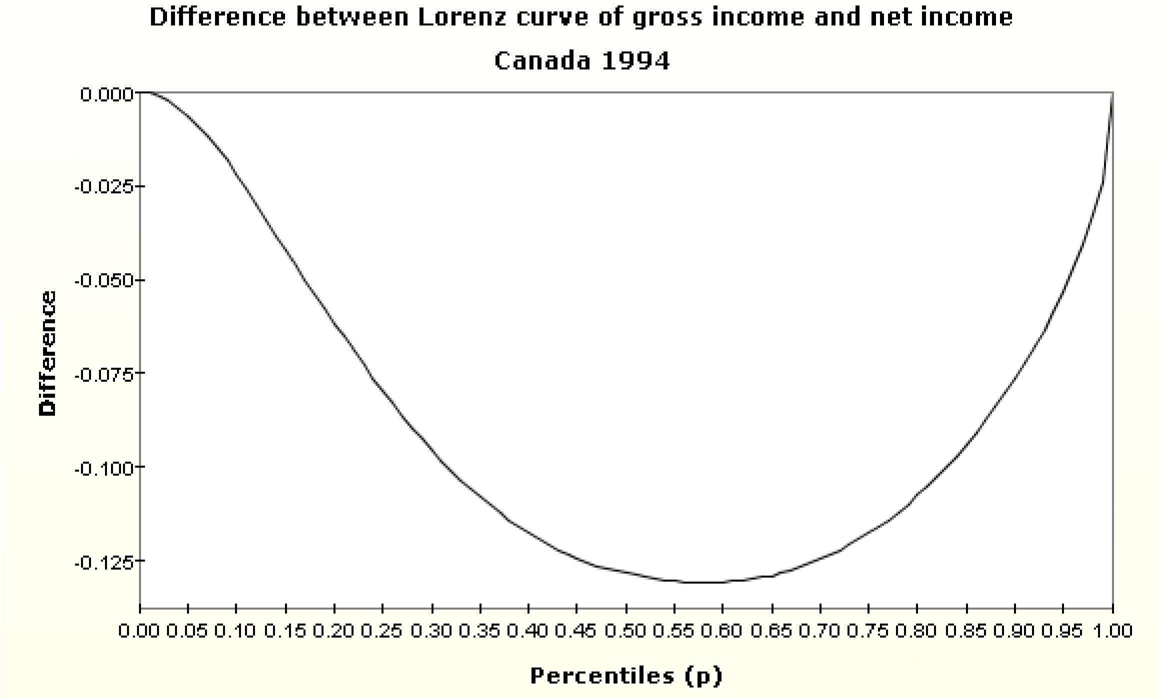
Figure 5 shows an example of Lorenz curves drawn by *DAD*.

Figure 5. Lorenz curves for two distributions



We can also compare two Lorenz curves to test for inequality dominance of one distribution over the other. For this, we choose again the application Curves|Lorenz, but this time with two distributions.

Figure 6. Differences in Lorenz curves drawn by *DAD*.



DAD can also usually draw curves that show how the levels of some distributive statistics vary with ethical parameters – such as inequality or poverty aversion parameters. Take for instance the Atkinson index of inequality. It may be informative to check how fast it varies as a function of ε, its parameter of inequality aversion. To do this, follow these steps:

- From the main menu, select the submenu : Inequality|Atkinson. Indicate that the number of distribution equals one.
- After setting the application variables, click on the button **Range** and specify the desired range for the parameter ε.
- Click on the button **Graph** to draw the curve that shows the Atkinson index against the risk aversion parameter ε.
- When the graph window appears, click on the button **Draw** to plot the curve.

6. Graphs

Recent versions of *DAD* are quite flexible in terms of editing, saving and printing graphs. On most application windows, a button **Graph** is available to draw graphs instantly. The type of graphs drawn depends on the application and on the type of **Graph** buttons selected. There are for instance two **Graph** buttons in the Poverty|FGT Index application window. Clicking on the GRAPH button plots estimates of the FGT index for a range of alternative poverty lines. Clicking on the GRAPH2 button draws instead estimates of the equally-distributed poverty that is equivalent to the estimated FGT poverty index, and this for a range of poverty aversion parameters α .

Most of the options for editing *DAD*'s graphs can be accessed from the Graph Properties dialogue box – see Figure 7. *DAD*'s graphs can also be saved in a variety of formats. Table 1 lists some of them. The coordinates of *DAD*'s curves can be easily edited in a new sheet or saved in ASCII format. With such ASCII files, one can reproduce *DAD* graphs with other softwares with graphical capabilities, such as Excel and the like.

Curves are useful tools to check various types of distributive dominance. Table 2 sums up some of the links between some of the applications and curves found in *DAD* and the tests for various orders of social welfare, poverty and inequality dominance.

Table 1. Available format to save *DAD*'s graph.

Extension	Description
*.png	Portable Network Graphic
*.pmb	Bitmat Image file
*.tif	Tag Image File Format
*.jpg	JPEG File Interchange Format
*.pdf	Portable Document Format
*.ps	Postscript

Figure 7. The dialogue box for graphical options

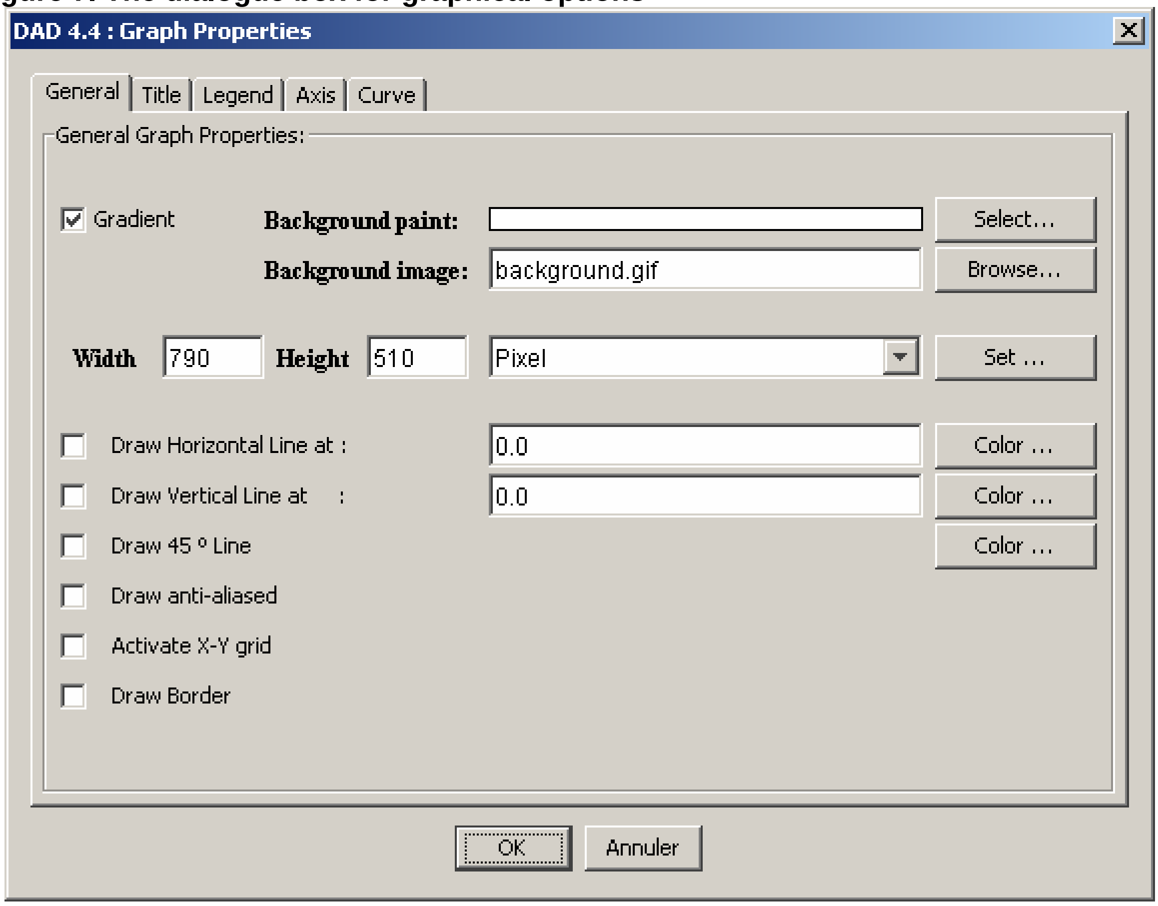


Table 2. Curves and stochastic dominance.

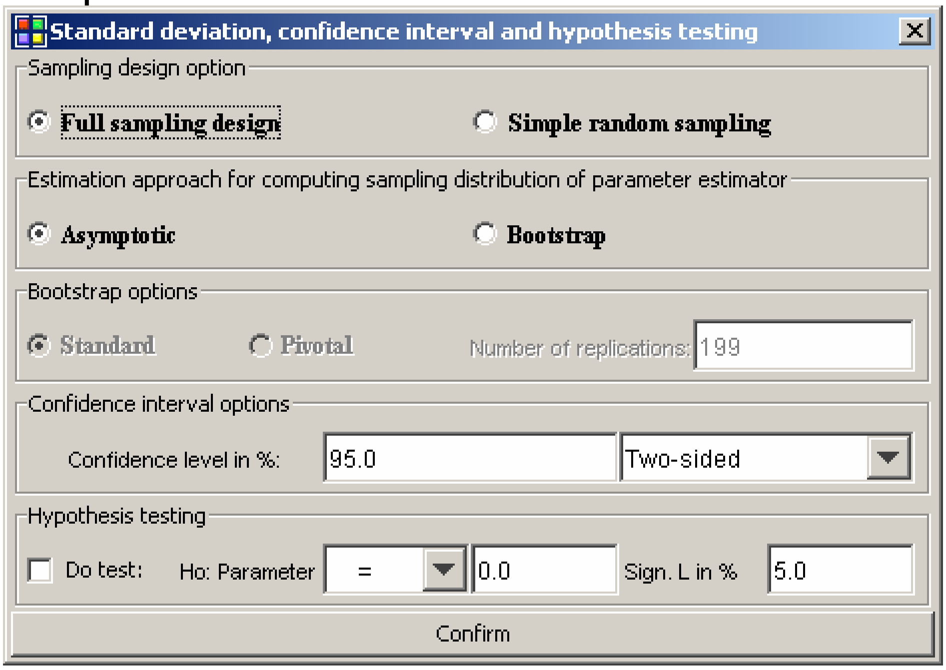
Order	Primal approach	Dual approach
	Social welfare	
1	Distribution|Distribution function	Curves|Quantile
2	Dominance|Poverty Dominance $s=2$	Curves|Generalized Lorenz
s	Dominance|Poverty Dominance	
	Poverty	
1	Dominance|Poverty Dominance $s=1$	Curves|Poverty Gap
2	Dominance|Poverty Dominance $s=2$	Curves|CPG
s	Dominance|Poverty Dominance	
	Inequality	
1	Dominance|Inequality Dominance $s=1$	Curves|Normalised Quantile
2	Dominance|Inequality Dominance $s=2$	Curves|Lorenz
s	Dominance|Inequality Dominance	

7. Statistical inference: standard deviation, confidence intervals and hypothesis testing

DAD facilitates statistical inference in a number of original ways:

- DAD readily provides asymptotic standard errors on a large number of estimators of distributive statistics, including estimators of inequality and social welfare indices, normalized/un-normalized poverty indices, poverty indices with deterministic/estimated poverty lines, poverty indices with absolute/relative poverty lines, equally-distributed-equivalent incomes and poverty gaps, quantiles, density functions, non-parametric regressions, points on a large number of curves, crossing points of curves, critical poverty lines, differences in indices and curves, ratios of various statistics, various income/price/population impacts and elasticities, distributive decompositions into demographic/factor components, progressivity, redistribution and equity indices, dominance statistics, etc.. It can be (and has typically formally been) shown that all of these estimators are asymptotically normally distributed.
- DAD can calculate the sampling distribution of most of these estimators taking into account the sometimes complex design of the survey. This is done as indicated in Section 3. Existing (commercial) softwares can sometimes take this design into account, but only for a sample number of relatively simple distributive statistics (such as simple sums and ratios).

Figure 8. STD option.



- *DAD* can provide at the click of a button estimates of confidence intervals as well as test statistics and *p*-values for various symmetric and asymmetric hypothesis tests of interest.
- *DAD* can be used to simulate numerically the finite-sample sampling distribution of most of the above-mentioned estimators using bootstrap procedures. The bootstrap can be performed on the ordinary estimators or on (asymptotically) pivotal transforms of them. It is well known that bootstrapping on pivotal statistics leads to faster rates of convergence to the true sampling distribution than bootstrapping on untransformed non-pivotal statistics. Pivotal bootstrapping is, however, usually more costly in time and resources since it requires estimates of the asymptotic distribution of the estimators. This is not a problem for *DAD,* however, since the (sometimes complex) asymptotic standard errors of these estimators are already programmed into it. Moreover, as mentioned above, the asymptotic standard errors and the pivotal statistics derived from them can be sample-design corrected, providing one more degree of superior accuracy for the bootstrap procedures available in *DAD.*

The Standard deviation, confidence interval and hypothesis testing dialogue box is the main tool for telling *DAD* what to do in terms of statistical inference. This box is shown on Figure 8.

8. Web site, users' manual and pedagogical documents

The web site for the *DAD* software [4] contains sufficient information for new users to start using *DAD*. Among items on this site, we can find:

- A downloadable theoretical manual for *DAD*.
- A users' manual for recent versions of the software.
- Projected developments for future versions.

References

Deitel, H. M. and P. J. Deitel (2003): *Java How to Program*, New York: Prentice Hall.
Lewis, J. and W. Loftus (2000): *Java, Software Solutions*, USA: Addsion Wesley, second ed.

[4] http://www.mimap.ecn.ulaval.ca

ANNEX A: *DAD*'s Applications

Table 3. *DAD*'s applications (Version 4.4)

Main menu	*Applications*
Inequality	Atkinson Index
	Gini / S-Gini Index
	Atkinson-Gini Index
	Entropy Index
	Quantiles Ratio
	Interquantile Ratio
	Coefficient of Variation
	Logarithmic Variance
	Variance of Logarithms
	Relative Mean Deviation
	Conditional Mean Ratio
	Share Ratio
	Income-Component Proportional Growth
Polarisation	Wolfson Index
	Duclos, Esteban and Ray Index
Poverty	FGT Index
	Bounded Income Index
	Watts Index
	S-Gini Index
	CHU Index
	Sen Index
	Bidimensional FGT index
	Impact of Price Change
	Impact of Tax Reform
	Lump-sum Targeting
	Inequality-neutral Targeting
	FGT Elasticity
	Income-Component Proportional Growth
	Impact of Demographic Change
Dominance	Poverty Dominance
	Inequality Dominance
	Indirect Tax Dominance
Welfare	Atkinson Index
	S-Gini Index
	Atkinson-Gini Index
	ATK:Impact of Price Change
	ATK:Impact of Tax Reform
	ATK:Impact of Income-component Growth

Table 4. *DAD*'s applications (version 4.4 — continued)

Main menu	*Applications*
Decomposition	FGT: Decomposition by groups
	FGT: Decomposition for 2 groups
	FGT: Decomposition by sources
	FGT: Growth & Redistribution
	FGT: Sectoral
	FGT: Transient & Chronic
	Transition Matrix
	S-Gini: Decomposition by groups
	S-Gini: Decomposition by sources
	Squared CV: Decomposition by sources
	Entropy: Decomposition by groups
	Atkinson: Social Welfare
Redistribution	Tax or Transfer
	Tax/Transfer vs Tax/Transfer
	Transfer vs Tax
	Horizontal Inequity
	Redistribution
	Coefficient of Concentration
	HI:Duclos & Lambert (1999)
	HI:Duclos, Jalbert & Araar (2003)
Curves	Lorenz
	Generalised Lorenz
	Concentration
	Generalised Concentration
	Quantile
	Normalised Quantile
	Poverty Gap
	CPG
	C-Dominance
	Relative deprivation
	Bi-polarisation
	Pro-poor
Distribution	Density Function
	Joint Density Function
	Distribution Function
	Joint Distribution Function
	Plot Scatters XY
	Non-Parametric Regression
	Non-Parametric Derivative
	Conditional Standard Deviation
	Descriptive Statistics
	Statistics
	Confidence Interval
	Group Information

ANNEX B: EXISTING STATA MODULES

sepov.ado

- provides estimates of the Foster-Greer-Thorbecke (FGT) class of poverty indices, as well as estimates of the sampling variance of these indices.
- http://www.stata.com/stb/stb51/sg117/sepov.hlp
- does not produce correct variance estimates for subpopulations in many cases

poverty.ado

- computes a series of poverty measures based on the (income) distribution described by varname
- http://www.stata.com/stb/stb48/sg108/poverty.hlp
- does not produce correct variance estimates for subpopulations in many cases

povdeco.ado

- povdeco estimates three poverty indices: FGT(0), FGT(1) and FGT(2)
- Optionally provided are decompositions across groups
- http://www.stata.com/stb/stb48/sg104/povdeco.hlp
- Variance estimates are not produced.

gidecomposition.ado

- The change in poverty is decomposed into three components(growth, distribution & residue)
- Michael M. Lokshin and Martin Ravallion, 2002, The World Bank
- Cannot weight differently the two data files
- Variance estimates are not computed
- Shapley decomposition procedure not included

secdecomposition.ado

- The change in poverty is decomposed into three components (intra-sectoral, inter-sectoral and residue)
- Michael M. Lokshin and Martin Ravallion, 2004, The World Bank
- Cannot weight differently the two data files
- Variance estimates are not computed
- Shapley decomposition procedure not included

gicurve.ado

- gicurve produces the growth incidence curve and calculates a measure of the rate of pro-poor growth.
- Michael M. Lokshin and Martin Ravallion, 2004, The World Bank
- Cannot weight differently the two data files
- Variance estimates are not computed

pov_robust.ado

- Graphs poverty incidence, deficit and severity curves
- Michael M. Lokshin and Martin Ravallion, 2002-2004, The World Bank
- Cannot weight differently the two data files
- Intersection points (critical poverty lines) are not computed

Inequality & Wellbeing Stata Modules

glcurve.ado

- Draws the Generalised Lorenz/Lorenz curve and/or generates two new variables containing the Generalised Lorenz/Lorenz ordinates for x
- http://www.stata-journal.com/software/sj4-4
- Cannot use a list of variables like: income1 income2…

ineqerr.ado

- Computes three indices of inequality - Gini coefficient, Theil entropy measure and Variance of Logs - and bootstrap estimates of their sampling variances. The program offers three variations of the bootstrap variance estimates
- http://www.stata.com/stb/stb51/sg115/ineqerr.hlp
- Analytical variance estimates are not computed
- Cannot perform the computation for a specific group

Ineqdeco.ado

- Provides an exact decomposition of the inequality of total income into inequality contributions from each of the factor components of total income. (Shorrocks 1982 approach)
- http://www.stata.com/stb/stb48/sg104/ineqdeco.hlp
- Variance estimates are not computed

Rspread.ado

- Produces several measures of relative dispersion or spread or inequality for each variable in the varlist.
- http://www.stata.com/stb/stb23

- Variance estimates are not computed
- Cannot compute for a specific group or tabulate for all groups

Atkinson.ado
Inequal.ado
Lorenz.ado
Relsgini.ado

- These four ado-files provide a variety of measures of inequality. Atkinson computes the Atkinson inequality index using the inequality aversion parameter specified in the parameter list. inequal displays the following measures: relative mean deviation, coefficient of variation, standard deviation of logs, Gini index, Mehran index, Piesch index, Kakwani index, Theil entropy index, and mean log deviation. lorenz displays a Lorenz curve. relsgini computes the Donaldson-Weymark relative S-Gini using the distributional sensitivity parameters specified in the parameter list.
- http://www.stata.com/stb/stb23/sg30/inequal.hlp
- Variance estimates are not computed

ineqdeco.ado

- Estimates a range of inequality and related indices commonly used by economists, plus decompositions of a subset of these indices by population subgroup
- http://www.stata.com/stb/stb48/sg104/ineqdeco.hlp
- Variance estimates are not computed.