The Stata Journal

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press, and Stata is a registered trademark of StataCorp LP.

The Stata Journal (2005)
5, Number 2, pp. 141–153

Exploratory analysis of single nucleotide polymorphism (SNP) for quantitative traits

Mario A. Cleves
UAMS College of Medicine, Department of Pediatrics
11219 Financial Centre Parkway, Suite 250, Little Rock, AR 72211

ClevesMarioA@uams.edu

Abstract. With the decreasing cost and the increasing ability to quickly genotype single nucleotide polymorphisms (SNP) across the human genome, large databases containing possibly hundreds of typed SNPs are becoming common in population-based studies of quantitative traits. Testing for association between individual SNPs and the quantitative trait is an important first step in the discovery of disease susceptibility SNPs. This task, however, could be time-consuming and tedious if a large number of SNPs is involved. In this article, I introduce two new commands designed to facilitate the screening and testing of multiple SNPs for possible association with quantitative traits.

Keywords: st0083, hwsnp, qtlsnp, genetic epidemiology, genetic linkage, QTL, biallelic marker, single nucleotide polymorphisms, Hardy–Weinberg

1 Introduction

Many phenotypes of medical importance can be measured quantitatively. Even qualitative diseases, such as diabetes and essential hypertension, result from variation in an underlying quantitative trait. In the last few years, there has been an increase in population-based studies that aim to identify genomic regions and subsequent genetic variants associated with many common diseases. This effort may begin a genome-wide or region-wide search for association using large numbers of single nucleotide polymorphisms (SNP), resulting in the creation of large databases with possibly hundreds of genotyped SNPs and possibly tens of quantitative traits to be examined.

The initial evaluation of these SNPs can be tedious and time-consuming. This motivated me to write two new Stata commands to facilitate rapid SNP screening: the `hwsnp` command, which tests SNPs for Hardy–Weinberg equilibrium, and the `qtlsnp` command, which uses Stata's regression commands and options to facilitate the rapid evaluation of multiple SNPs for possible association with a quantitative trait. Note that although the focus of this command and article is on SNP analysis, these commands work equally well for other biallelic markers.

2 The hwsnp command

`hwsnp` succinctly reports the results of Hardy–Weinberg equilibrium tests performed on each of multiple SNPs. `hwsnp` calls `genhw` and reports results from both asymptotic and exact Hardy–Weinberg (HW) equilibrium tests. Note that `genhw` must be installed in order for this command to work (Cleves 1999). If it is not installed, simply type in Stata `findit genhw` and follow the instructions to install `genhw`.

2.1 Syntax

`hwsnp` *SNPlist* [`if` *exp*] [`in` *range*] [`, separator(`*string*`) outfile(`*filename*`) replace`]

`by ... :` may be used with `hwsnp`; see [R] **by** ([D] **by** in Stata 9).

SNPlist may contain one or more SNPs.

`hwsnp` expects the data to be in wide form—each observation representing one subject. If the data are in long form (i.e., multiple observations per subject), `reshape` may be used to transform it to wide form; see [R] **reshape** ([D] **reshape** in Stata 9).

`hwsnp` expects each SNP in *SNPlist* to be of length = 2, where the first character (or digit) is the first allele of an individual's genotype at the SNP locus and the second character or digit is the second allele of that individual's genotype at the SNP locus (example of valid genotypes: ct, tt, 12, 22). See the `separator()` option if your data are coded differently.

2.2 Options

`separator(`*string*`)` is used to inform `hwsnp` how the SNPs are coded. By default, each SNP in *SNPlist* is assumed to be of length = 2, where the first character (or digit) is the first allele of an individual's genotype at the SNP locus and the second character or digit is the second allele of that individual's genotype at the SNP locus. `separator()` modifies this by indicating the characters used to separate alleles in the genotype. For example, if the genotype is coded as THR/SER, specify `separator("/")`.

`outfile(`*filename*`)` saves in *filename*`.dta` for resils for each SNP.

`replace` replaces an existing output file.

2.3 Example

In a recent study of individuals with congestive heart disease (CHD), we genotyped 143 CHD patients at 114 SNPs in genomic areas believed to harbor genes important in lipid metabolism. A fairly comprehensive fasting lipid profile was performed on each

patient, which included these four common lipid measurements: triglycerides (`trig`), total cholesterol (`totalchol`), LDL cholesterol (`ldl`), and HDL cholesterol (`hdl`).

Following is a list of the first ten observations and eight variables in the dataset:

```
. use lipids, clear
. list PatientID trig totalchol ldl hdl SNP1 SNP2 SNP114 in 1/10
```

	Patien~D	trig	totalc~l	ldl	hdl	SNP1	SNP2	SNP114
1.	11107	211	228	156	30	AA	CC	TT
2.	11115	176	217	147	35	AA	AA	TT
3.	11120	69	194	135	45	AA	CC	TT
4.	11135	169	189	126	29	AA	AC	TT
5.	11141	73	159	100	44	AA	AC	CC
6.	11145	462	216	107	24	AA	AC	CC
7.	11148	167	232	160	39	AA	AA	TT
8.	11149	56	158	101	46	AA	AA	CC
9.	11155	74	129	81	33	AA	AC	CT
10.	11156	238	237	156	33	AA	AA	TT

Note that the dataset is in the wide form, containing one observation per patient, as defined by `PatientID`. Because of space limitations, we only listed three of the 114 SNPs. Although in this dataset the SNPs are string variables, numeric SNP variables are also valid.

We now test SNP1 to SNP9 for Hardy–Weinberg equilibrium using `hwsnp`.

```
. hwsnp SNP1-SNP9
```

Polymorphism	Pearson chi2	P-value	LR chi2	P-value	Exact Significance
SNP 1	98.695	0.0000	55.297	0.0000	0.0000
SNP 2	0.373	0.5415	0.373	0.5416	0.6110
SNP 3	47.607	0.0000	59.514	0.0000	0.0000
SNP 4	2.725	0.0988	2.446	0.1178	0.1331
SNP 5	1.003	0.3166	1.000	0.3173	0.3026
SNP 6	0.960	0.3271	0.930	0.3348	0.3686
SNP 7	19.003	0.0000	19.381	0.0000	0.0000
SNP 8	1.440	0.2301	1.407	0.2356	0.2271
SNP 9	0.039	0.8426	0.040	0.8424	1.0000

`hwsnp` tests the null hypothesis that the SNP is in Hardy–Weinberg equilibrium. It reports Pearson's and the likelihood-ratio chi-squared statistics, as well as the exact significance probability. See Cleves (1999) for details about these tests.

Note that adjustments for multiple comparisons are not being made or reported but may need to be accounted for in the final analysis.

3 The qtlsnp command

`qtlsnp` displays summary results for SNP analysis of quantitative traits. It succinctly reports on multiple SNPs and trait variables. Optionally, `qtlsnp` reports details for each SNP analyzed.

By default, `qtlsnp` uses linear regression to compare the equality of means across genotypes, while allowing for covariate adjustment. By specifying the `median` option, `qtlsnp` uses median regression instead of linear regression (see [R] **qreg**), and by specifying `bs`, `qtlsnp` uses median regression with bootstrapped VCE (see `bsqreg` in [R] **qreg**).

By default, `qtlsnp` assumes a codominant genetic model and tests for additive and dominant effects, as well as testing that both effects are equal to zero. (This comparison is equivalent to comparing means across the three possible genotypes.)

Optionally, by specifying the `dominant` or `recessive` option, `qtlsnp` will assume a dominant or recessive genetic model of inheritance, respectively. For example, if the three possible genotypes at a given SNP are cc, ct, and tt, the `dominant` option directs `qtlsnp` to combine the cc and ct genotypes and compare the quantitative mean trait value for these combined genotypes against the mean of the tt genotype. The `recessive` option combines the ct and tt genotypes, and `qtlsnp` compares the quantitative mean trait value for these combined genotypes with the mean of the cc genotype. Note that the terms *dominant* and *recessive* as used here are arbitrary labels used only to group genotypes.

3.1 Syntax

`qtlsnp` *SNPlist* [`if` *exp*] [`in` *range*] `, traitvars(`*varlist*`)` [`siglev(`#`)`
`sumlev(`#`)` `class(`*varlist*`)` `cont(`*varlist*`)` `dominant` `recessive` `detail` `brief`
`nosummary` `means` `median` `bs` `robust` `rreg` `noasterisks` `graph` `rotate`
`outfile(`*filename*[`, replace`]`)` `effect(additive|dominant|both)` `overall`
twoway_options]

`by` ... `:` may be used with `qtlsnp`; see [R] **by** ([D] **by** in Stata 9).

SNPlist may contain one or more SNPs.

`qtlsnp` expects the data to be in wide form—each observation representing one subject. If the data are in long form (i.e., multiple observations per subject), `reshape` may be used to transform the data to wide form; see [R] **reshape** ([D] **reshape** in Stata 9).

`qtlsnp` expects each SNP in *SNPlist* to have a maximum of three and a minimum of two distinct genotypes in the data. This implies that the heterozygous genotype should be coded consistently for each SNP. For example, for a SNP with alleles C and T, we could code the heterozygous genotype as either CT or TC, but not both. Note that the SNPs in *SNPlist* can be either string or numeric. Examples of valid genotypes include ct, t/t, 12, 2-2, and THR/SER.

3.2 Options

`traitvars(`*varlist*`)` supplies names of the quantitative trait variables. At least one trait variable must be specified. Only one trait variable is allowed when `graph` or `outfile()` is specified.

`siglev(#)` and `sumlev(#)` are used to specify the significance probability used for reporting results. If neither option is specified, all results are summarized. If `sumlev()` is specified, only SNPs significant at $p \leq$ `sumlev`() will be reported. If only `siglev()` is specified, all SNPs will be summarized, but only SNPs significant at $p \leq$ `siglev`() will be `detail`ed. If `siglev()` and `graph` are specified together, a horizontal line at $y = -\log($`siglev()`$)$ will be displayed on the plot.

`class(`*varlist*`)` supplies the names of categorical variables to be used as covariates in the analyses.

`cont(`*varlist*`)` supplies the names of continuous variables to be used as covariates in the analyses.

`dominant` or `recessive` specify that heterozygous are to be combined with the homozygous wild or homozygous variant during analysis. If neither option is specified, a codominant model is assumed.

`detail` produces a detailed report of each SNP analyzed. This option uses a distilled version of Tony Brady's `reformat` command (type `findit reformat` in Stata).

`brief` is used with `detail` to suppress printing details for all covariates in the model. If `brief` is specified, only model statistics for the SNPs are reported.

`nosummary` is used with `detail` to suppress the printing of the summary table.

`means` is used with `detail` to produce tables of summary statistics by SNP genotype.

`median` specifies that comparisons be based on median regression instead of linear regression. This option calls Stata's `qreg` command. See [R] **qreg** for details.

`bs` specifies that comparisons be based on median regression with bootstrapped VCE. See [R] **qreg** for details.

`robust` specifies that the Huber/White/sandwich estimator of variance be used in place of the traditional calculation.

`rreg` specifies that comparisons be based on robust regression instead of linear regression.

`noasterisks` is used to suppress the printing of stars for significant probabilities.

`graph` produces graphical output of significance probabilities by SNP. Only one quantitative trait is allowed with the `graph` option.

`rotate` is used with `graph` to exchange the x- and y-axes on the plot.

`outfile(`*filename*`[ , replace ])` saves in *filename*`.dta` the p-values for each SNP.

`effect(additive|dominant|both)` is used with `graph`. It specifies which codominant effect to plot. If `effect()` is not specified, all three "effects" will be plotted together.

`overall` specifies that only comparisons where the overall codominant effect is significant at $p \leq$ `sumlev()` be outputted. This option is ignored when either `dominant` or `recessive` are specified.

twoway_options are most of the options documented in [G] ***twoway_options***.

3.3 Background

Assume that we have a quantitative trait Y and a candidate SNP with alleles A and B. Further, assume that the population is in Hardy–Weinberg equilibrium and that the two alleles have frequencies of p_A and $p_B = (1 - p_A)$, respectively. For this SNP, there are three possible genotypes in the population: AA, AB, and BB. Let the mean genotypic values for the three genotypes be μ_1, μ_2, and μ_3, respectively, and assume that the residual variance around these means is the same for the three genotypes. Define the additive genetic effect as

$$\alpha = 0.5(\mu_1 - \mu_3) \tag{1}$$

and the dominance genetic effect as

$$\delta = 0.5(2\mu_2 - \mu_1 - \mu_3) \tag{2}$$

The additive genetic effect, α, is the phenotypic value midway between the two homozygotes and has the interpretation that if we were to substitute allele A for allele B in the genotype, we would expect, on average, a phenotypic value to change α units. The dominance value δ measures the extent to which the mean of the heterozygote AB deviates from the average of the two homozygotes. If $\delta = 0$, there is complete additivity in the trait and the value of the heterozygote AB lies half-way between the values of the two homozygotes (Lui 1997, 377–379; Falconer and Mackay 1996, chapter 8).

Cautionary Note: The additive and dominant effects are interpreted in terms of the phenotypic means of the genotype classes. If you use median regression (`median` option), the additive and dominant effects are not interpretable as such, and in that case, perhaps only the overall test has meaning.

It is the user's responsibility that the regression assumptions regarding the quantitative trait be met. This may require that the trait measurement be transformed. In addition, it is recommended that the user verify the correctness of the functional form of all additional covariates included in the model. The remaining discussion assumes

that the trait measurement is in the proper form and that all covariates are correctly specified.

3.4 Example: Examining SNPs using linear regression

Using the same dataset as in the previous example, we will now examine the 114 SNPs for possible association with the four lipid measurements.

We begin by examining for association with serum triglycerides (`trig`).

```
. qtlsnp SNP1-SNP114, trait(trig) sumlev(0.05)
Genetic model:  Codominant
```

			Additive effect		Dominant effect		Both=0	
Trait	SNP	N	F	Prob>F	F	Prob>F	F	Prob>F
trig								
	SNP13	143	2.61	0.108	4.43	0.037**	2.25	0.109
	SNP32	143	4.30	0.040**	0.11	0.736	2.41	0.093*
	SNP39	143	0.34	0.560	5.23	0.024**	2.62	0.076*
	SNP48	143	13.12	0.000**	16.90	0.000***	8.52	0.000***
	SNP96	143	4.82	0.030**	6.65	0.011**	3.38	0.037**
	SNP99	142	6.56	0.011**	1.48	0.225	4.02	0.020**

```
*<=0.1,   **<=0.05,   ***<=0.001
```

By default, `qtlsnp` will assume a codominant genetic model and will fit a linear model after generating the appropriate indicator variables to test for additive and dominant genetic effects. Because we specified `sumlev(0.05)`, only those SNPs with at least one significant effect with $p \leq 0.05$ are summarized. In this case, only 6 of the 114 SNPs met this criterion.

(*Continued on next page*)

SNP48 looks particularly interesting. We can examine this SNP in more detail:

```
. qtlsnp SNP48, trait(trig) detail means nosummary
Genetic model:  Codominant

SNP:   SNP48                                  Quantitative trait:        trig
                                                   Genetic model:  Codominant
```

SNP 48	Summary of Triglycerides Mean	Std. Dev.	Freq.
AA	131.26531	81.554302	98
AG	107.5	57.497508	42
GG	299.33333	220.01439	3
Total	127.81119	83.255499	143

Model: Linear regression — Number of obs = 143

	N	Coef.	Std. Err.	P>|t|	[95% Conf.	Interval]
Additive effect	143	84.0340	23.2017	0.000	38.1630	129.9051
Dominant effect	143	-107.7993	26.2213	0.000	-159.6402	-55.9584
Constant	143	215.2993	23.2017	0.000	169.4283	261.1704

The first table in the output results from specifying the `means` option. It uses `tabulate, sum()` to summarize triglycerides stratified by genotype. The second table of the output summarizes the results from the regression model. Although this SNP looks like it could be related to the trait either directly or through linkage with the trait locus, we need to be cautious about this result. From the first table, we can see that there are only three homozygous GG individuals in the sample and that the mean level for the heterozygous is less than either of the homozygous individuals. These findings cast doubt on the reliability of the results originally observed.

In the above example, we specified the `nosummary` option to suppress outputting the default summary table.

The additive and dominance parameters of the quantitative genetic model, α and δ from (1) and (2), can be computed from the means reported in the summary table above. Thus the additive parameter is estimated as

$$\alpha = 0.5(\mu_1 - \mu_3) = 0.5 * (131.27 - 299.33) = -84.03$$

and the dominance parameter is estimated as

$$\delta = 0.5(2\mu_2 - \mu_1 - \mu_3) = 0.5 * (2 * 107.5 - 131.27 - 299.33) = -107.8$$

Let's now examine each SNP for association with each of the remaining three lipid measurements. We can do this in one command. To reduce the amount of output, we will specify `sumlev(0.01)`. That is, we ask `qtlsnp` to only report those SNPs with p-values less than or equal to 0.01.

```
. qtlsnp SNP1-SNP114, trait(totalchol ldl hdl) sumlev(0.01)
Genetic model:  Codominant
```

			Additive effect		Dominant effect		Both=0	
Trait	SNP	N	F	Prob>F	F	Prob>F	F	Prob>F
totalchol								
	SNP6	143	1.00	0.320	7.15	0.008**	3.78	0.025**
	SNP70	142	0.01	0.943	8.65	0.004**	4.38	0.014**
	SNP83	143	8.09	0.005**	2.05	0.155	4.06	0.019**
ldl								
	SNP24	143	6.71	0.011**	1.54	0.216	4.85	0.009**
	SNP63	142	8.58	0.004**	5.49	0.021**	4.38	0.014**
hdl								
	SNP19	143	5.92	0.016**	0.22	0.638	4.81	0.010**
	SNP24	143	5.39	0.022**	0.31	0.580	6.01	0.003**
	SNP28	142	8.62	0.004**	1.63	0.203	4.89	0.009**
	SNP85	143	3.68	0.057*	8.07	0.005**	5.15	0.007**

*<=0.1, **<=0.05, ***<=0.001

Thus far, we have assumed a codominant genetic model; alternatively, we can ask `qtlsnp` to assume either a recessive or a dominant genetic model. This is done by specifying the `recessive` or `dominant` option. We now do this and include all the SNPs and lipid measurements simultaneously in the same command. Again to cut down on the amount of output, we will specify `sumlev(0.01)`. We will also specify the option `noasterisk` to suppress printing the asterisk in the output table.

```
. qtlsnp SNP1-SNP114, trait(trig totalchol ldl hdl) sumlev(0.01) recessive
> noasterisks
Genetic model:  Recessive
```

Trait	SNP	N	F	Prob>F
trig				
	SNP48	143	14.22	0.0002
totalchol				
	SNP25	142	7.67	0.0064
ldl				
	SNP63	142	8.80	0.0035
hdl				
	SNP19	143	8.98	0.0032
	SNP24	143	9.85	0.0021

```
. qtlsnp SNP1-SNP114, trait(trig totalchol ldl hdl) sumlev(0.01) dominant
> noasterisks
Genetic model:  Dominant
```

Trait	SNP	N	F	Prob>F
trig	SNP99	142	7.65	0.0065
totalchol	SNP83	143	7.25	0.0079
ldl				
hdl				

As previously mentioned, the terms dominant and recessive are used as arbitrary labels to group genotypes. There is no a priori way to tell `qtlsnp` how to group the genotypes. However, once we have identified a SNP of interest we can use the `detail` option to examine how the genotypes were combined.

For example, we can check how the genotypes for SNP99 were combined in the above dominant model.

```
. qtlsnp SNP99, trait(trig) dominant detail noasterisks
Genetic model:  Dominant
```

Trait	SNP	N	F	Prob>F
trig	SNP99	142	7.65	0.0065

```
SNP:   SNP99                                  Quantitative trait:        trig
                                                   Genetic model:    Dominant
Model: Linear regression                        Number of obs =         142
```

	N	Coef.	Std. Err.	P>|t|	[95% Conf.	Interval]
SNP99						
AA*	25					
AG or GG	117	-49.6044	17.9391	0.006	-85.0710	-14.1378
Constant	142	168.1600	16.2836	0.000	135.9665	200.3535

```
* Reference category
```

We see that the heterozygous and the homozygous GG genotypes were combined and compared with the homozygous AA genotype.

3.5 Example: Incorporating covariates into the models

Additional patient covariates can be included into the regression models by specifying the `cont()` or `class()` options depending on whether the covariate is measured on an

interval scale or not. In our lipid dataset, we have two patient covariates: the patient's age and the patient's sex.

Let's fit our recessive model as before but include these two covariates.

```
. qtlsnp SNP1-SNP114, trait(trig totalchol ldl hdl) sumlev(0.01) recessive
> cont(age) class(sex)
Genetic model:  Recessive
```

Trait	SNP	N	F	Prob>F
trig				
	SNP48	143	13.26	0.0004***
totalchol				
ldl				
	SNP63	142	8.16	0.0049**
hdl				
	SNP19	143	9.49	0.0025**
	SNP24	143	9.30	0.0027**

*<=0.1, **<=0.05, ***<=0.001

We see that the same SNPs we previously identified, except for SNP25, remained significant at $\alpha = 0.01$ after controlling for age and sex. Let's examine the relationship between SNP25 and total cholesterol more closely.

```
. qtlsnp SNP25, trait(totalchol) recessive cont(age) class(sex) detail
Genetic model:  Recessive
```

Trait	SNP	N	F	Prob>F
totalchol				
	SNP25	142	6.43	0.0123**

*<=0.1, **<=0.05, ***<=0.001

```
SNP:   SNP25                                  Quantitative trait:   totalchol
                                                   Genetic model:   Recessive
Model: Linear regression                       Number of obs =         142
```

	N	Coef.	Std. Err.	P>|t|	[95% Conf.	Interval]
SNP25						
AA or AG*	49					
GG	93	16.2611	6.4123	0.012	3.5820	28.9402
SEX						
1*	88					
2	54	-6.5592	6.2725	0.298	-18.9618	5.8433
Age						
per unit	142	-0.0702	0.2445	0.774	-0.5537	0.4132
Constant	142	175.4632	8.1055	0.000	159.4362	191.4901

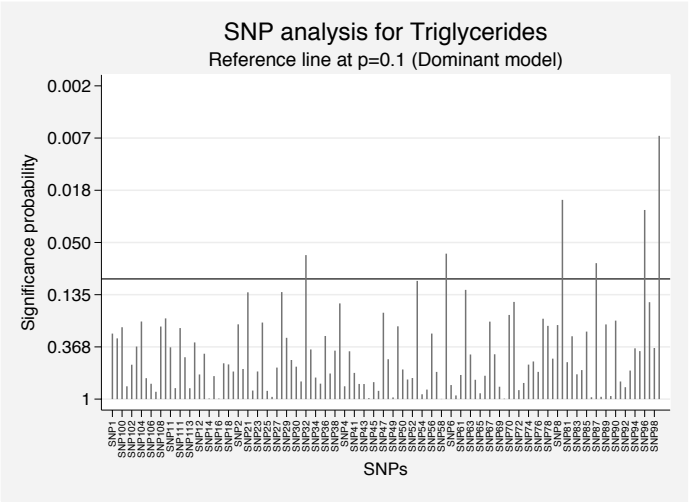
* Reference category

In the first table, we see that the adjusted p-value is now 0.0123, which is why it was not shown in our previous example. In the second table, we see that both age and sex were treated as specified. When `class(`*varlist*`)` is specified, `qtlsnp` knows to generate indicator variables for each variable in *varlist*. Note that these two options will also allow you to incorporate interaction terms into the model, although the interaction terms must be generated beforehand.

3.6 Example: Examining SNPs using the graph option

`qtlsnp`'s `graph` option is helpful for quick examination of results. The option has the limitation that only one quantitative trait can be plotted at a time. As an example, using the lipid data, we plot the results for triglycerides, assuming a dominant model.

```
. qtlsnp SNP1-SNP114, trait(trig) graph dominant
Genetic model:  Dominant
```



In this plot, the taller the line, the more significant is the SNP. By default, a horizontal reference line at $p = 0.1$ is drawn. The location of the reference line can be controlled by specifying the `siglev()` option.

4 Comments

The two commands `hwsnp` and `qtlsnp` were designed to facilitate the rapid screening of a large number of SNPs and quantitative traits simultaneously. The commands use existing Stata commands and, in that sense, are not new. Via its options, the `qtlsnp` command provides greater flexibility than that described in this article.

The `qtlsnp` command must be used with caution. Before using this command, you must be familiar with and verify the assumptions being made by the model that you are planning to estimate (e.g., normality, homoscedasticity, independence, etc.) and also be certain that you use the correct functional form of all covariates and interaction terms (i.e., do they need to be transformed to meet linearity assumption in linear regression, etc.) Additionally, be aware that the significant probabilities reported by `qtlsnp` have not been adjusted for multiple comparisons.

5 Acknowledgments

I would like to express my appreciation and acknowledge the contributions made by Dr. David C. Airey (Vanderbilt University) and Dr. Diego F. Wyszynski (Boston University) to the development of this command. This work was supported in part by Cooperative Agreement No. U50/CCU613236 from the Centers for Disease Control and Prevention (CDC), and by a grant from the National Institute of Child Health and Human Development (5R01 HD39054). The contents are solely the responsibility of the author and do not necessarily represent the official views of the CDC or NIH.

6 References

Cleves, M. A. 1999. sg110: Hardy–Weinberg equilibrium test and allele frequency estimation. *Stata Technical Bulletin* 48: 34–37. In *Stata Technical Bulletin Reprints*, vol. 8, 280–284. College Station, TX: Stata Press.

Falconer, D. S. and T. F. C. Mackay. 1996. *Introduction to Quantitative Genetics*. 4th ed. Harlow, Essex, UK: Longman.

Lui, B. H. 1997. *Statistical Genomics: Linkage, Mapping, and QTL Analysis*. Boca Raton, FL: CRC Press.

About the Author

Mario Cleves is an Associate Professor at the University of Arkansas for Medical Sciences, College of Medicine, Department of Pediatrics, and a Senior Biostatistician for the Arkansas Center for Birth Defects Research and Prevention. His current research interests focus on dissecting the genetic and environmental causes of major structural congenital malformations, particularly neural tube and congenital heart defects.