Using Projection and 2D Plots to Visually Reveal Genetic Mechanisms of Complex Human Disorders

Boonthanome Nouanesengsy*
Battelle Center for Mathematical Medicine
Nationwide Children's Hospital
& The Ohio State University

Sang-Cheol Seok†
Battelle Center for Mathematical Medicine
Nationwide Children's Hospital

Han-Wei Shen‡
The Ohio State University

Veronica J Vieland§
Battelle Center for Mathematical Medicine
Nationwide Children's Hospital
& The Ohio State University

ABSTRACT

Gene mapping is a statistical method used to localize human disease genes to particular regions of the human genome. When performing such analysis, a genetic likelihood space is generated and sampled, which results in a multidimensional scalar field. Researchers are interested in exploring this likelihood space through the use of visualization. Previous efforts at visualizing this space, though, were slow and cumbersome, only showing a small portion of the space at a time, thus requiring the user to keep a mental picture of several views. We have developed a new technique that displays much more data at once by projecting the multidimensional data into several 2D plots. One plot is created for each parameter that shows the change along that parameter. A radial projection is used to create another plot that provides an overview of the high dimensional surface from the perspective of a single point. Linking and brushing between all the plots are used to determine relationships between parameters. We demonstrate our techniques on real world autism data, showing how to visually examine features of the high dimensional space.

Keywords: Visualization, Multidimensional data, Linkage Analysis, Posterior Probability of Linkage, PPL, PPLD, LD analysis, Linkage disequilibrium, Autism

1 INTRODUCTION

Genetic linkage and/or linkage disequilibrium (LD) analysis [5] is a class of statistical methods used to associate functionality of genes to their locations on chromosomes. It is commonly used to map the genes responsible for genetic diseases. These analyses are based on the fact that genes which are located close to each other on a chromosome will tend to be inherited together by offspring. Thus, if a disease gene is being transmitted in a family, and a causal gene is close to a physically identifiable location on a particular chromosome (or a genetic marker), then the result will be observable co-segregation of the disease with the genetic marker through the family, which can be modeled using statistical (linkage) methods. A class of models for mapping and modeling genes for complex disorders, called the Posterior Probability of Linkage (PPL), has shown striking results in the field of statistical human genetics over the last decade [10][11][13][14]. The PPL statistic has been developed as a method of rigorous accumulation of evidence for or against linkage and/or LD. It can be calculated from human pedigree or case-control data. For more details, please refer to Vieland [11].

One feature of the PPL framework is the representation of a genetic model via likelihoods. This genetic model takes several parameters, each one having several possible values. The genetic model is sampled over a grid of varying parameter values, and integrated over the parameter space. The output of this genetic model is the *genetic likelihood ratio*, or *GLR*. The GLR represents the likelihood of the data allowing for linkage and/or LD relative to the likelihood assuming no linkage and no LD. The final integral value, which is called the Bayes Ratio (BR), is used to calculate the PPL value. The PPL is a representation of the evidence for or against a disease gene at a chromosome position. A genome-wide scan in which the PPL is calculated at each position is performed. Once linkage analysis identifies possible disease genes, one of the follow-up research questions of great interest is to get insight into the underlying genetic mechanisms. Thus, visualization of the high dimensional scalar field associated with this position is required.

There are many reasons researchers want to explore this high dimensional space. One is to see how greatly a particular parameter affects the GLR value. Some parameters may affect the model substantially, or may not affect it at all. Also, examining the shape of the high dimensional surface provides qualitative information about the amount of evidence for linkage. In particular, the shape and slope of peaks of the surface are of great interest. Usually, only the global maximum of the space is used in analysis. For example, a standard statistical approach would be to consider the global maximum of the space in order to find the best supported (maximum likelihood) parameter values [1][3][12]. There could be other peaks, though, that have a close GLR value to the global maximum which may also deserve examination. In general, we wish to learn about the shape and slope of local maxima, and not just rely on the global maximum. This includes the combination of parameters and their values around local maxima. Not just genetics, but many areas of Statistics also share this problem of wanting to know the support interval of a multidimensional likelihood. Other reasons of interest include finding dependent relationships between parameters. For example, two parameters could have an inverse relationship, whereby as one parameter increases and the other decreases, the resulting GLR remains relatively the same.

There have been previous attempts at visualizing this genetic likelihood space [6][7]. Those attempts included fixing the value of all parameters except two, and plotting the resulting 2D slice as a height map, with the log(GLR) used as the elevation. The value of the fixed parameters can be changed in order to observe the effects of a parameter. Unfortunately, it is difficult to perceive higher order interactions between parameter values using this technique. There have been other techniques of visualizing a multidimensional scalar function. Many of these techniques limit the number of dimensions seen at once, for example showing 2D slices of the space in a matrix

*e-mail: nouanese@cse.ohio-state.edu
†e-mail:Sang-Cheol.Seok@nationwidechildrens.org
‡e-mail:hwshen@cse.ohio-state.edu
§e-mail:Veronica.Vieland@nationwidechildrens.org

IEEE Symposium on Visual Analytics Science and Technology
October 12 - 13, Atlantic City, New Jersey, USA
978-1-4244-5283-5/09/$25.00 ©2009 IEEE

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

format [9] or using multiple volume renderings to display 3D slices at a time [6]. All previously mentioned techniques share the disadvantage of requiring the user to keep a mental picture of several views in order to gain a sufficient understanding of the space.

We wanted a technique that displayed more than two or three dimensions at once, while giving some intuition about the curvature of the surface and the shape of important features, such as peaks. This is accomplished by creating several plots using different projection methods, and then using linking and brushing to explore the space. From the multidimensional data, line segments are extracted. A line segment represents one step in one dimension from a point in the high dimensional space. By plotting these segments in different ways, several unique views of the data can be achieved. One way to plot these segments is to use one parameter's value as the x-axis, and the GLR as the y-axis. This is done for every parameter. These *parameter plots* show an overview of how each parameter is affecting the GLR. One more 2D plot is created that plots line segments based on their distance to a certain point. This *distance plot* gives us what the space looks like from one point, and provides a high-level overview of the space. Guided by the distance plots, interesting features of the space are selected, and line segments in that space are highlighted over all plots, leading to insights of the relationship between parameters.

2 Related Work

Over the years, there have been many techniques developed to visualize a multidimensional scalar function. A multidimensional scalar function is defined as a function that can be denoted as $F = f(x_1, x_2, ..., x_N)$, where F is a scalar value. The function is expressed as a multidimensional array of sampled values. For low values of N, visualization is straightforward. When $N = 1$, a line graph is sufficient. For $N = 2$, a height map or a colored heat map can display the data. There is a set of standard visualization techniques when $N = 3$, which includes volume rendering and isosurfacing. For $N = 4$, possible techniques include treating one dimension as time and animating a volume rendering or isosurface, or showing multiple renderings side by side as time changes. A disadvantage to this, though, is that not all dimensions are treated equally. Anything past four dimensions becomes very difficult to visualize because such spaces are beyond the physical world and the human mind has little intuition about such high dimensional spaces.

One of the first techniques to explore high dimensional spaces was introduced by Fiener and Beshers [2]. Called *Worlds within Worlds*, it creates a hierarchy of displays. At each level of the hierarchy, the user could select up to three parameter values. The user continues by choosing a point in the selected parameter space, then choosing more parameters for the next level, creating another "world" within that display. This process continues until all dimensions have been selected.

The HyperSlice method, introduced by van Wijk and van Liere [9], is another method to visualize a multidimensional scalar field. It shows all 2D orthogonal slices of a subspace of the data. Each slice is obtained by fixing all parameters except two to a certain value, and varying the values over two parameters. This is done for all possible pairs of parameters. The slices, displayed as heat maps, are then laid out similarly to a scatter plot matrix. The user can navigate through the space by clicking and dragging on slices. A problem with the HyperSlice method is that the user can get lost while navigating the high dimensional space.

In 1991 Mihalisin [4] introduced the hierarchical axis method. In it, axes are laid out horizontally in a nested hierarchy. Each axis has a certain "speed", with axes having higher speed being nested in a repeating fashion inside other axes of lower speed. Thus each point along the horizontal axis is mapped to a unique point in the high dimensional space. The function is plotted using the vertical axis as the value of the scalar of each point. One disadvantage to this technique is that the screen can become very cluttered when the number of parameters is high.

There have been two previous attempts at visualizing a genetic likelihood space. The first one introduced a program called LiViT [7] (Likelihood Visualization Tool). It displayed 2D slices of 6D space by fixing four parameter values, and keeping the remaining two parameters free. The resulting 2D slice was displayed as a height map. It allowed the user to interactively change which parameters were fixed and plotted, and also let the user change the value of the fixed parameters. The second attempt [6] was a visualization based on Worlds within Worlds, which used color and filtering to assist the user.

3 Data Parameters and Definitions

As described earlier, a genetic likelihood function having several independent variables is generated from pedigree data, and values of this function are sampled over a grid of varying parameter values. The number of actual parameters varies depending on the type of analysis done. For example, an analysis can assume linkage equilibrium (LE) or linkage disequilibrium (LD). If the analysis is LD, then another parameter, D', must be considered. The number of parameters typically ranges from four to seven. The range and step size for each parameter is different, chosen mainly because of the underlying genetics behind them.

The following are a list of the possible parameters and their description:

- D' – A parameter that is used during LD analysis. Its range is [-1, 1], and is ordinarily sampled with a step size of 0.1.
- θ – This parameter is the measure of the distance from the gene locus to the genetic marker in recombination units. Its range is [0.0, 0.5], with a normal step size of 0.01. This parameter is omitted for some types of analysis.
- α – This parameter is a mixture parameter designed to vary the impact of individual pedigrees to the likelihood, since it is possible not all pedigrees are linked at the same locus. If only one pedigree is being used, then this parameter is not needed. The range of α is [0, 1], and is usually sampled from 0.05 to 1.00 with a normal step size of 0.05.
- *gene frequency* (gf) – The frequency of a disease variant or allele (say, "*D*") in certain populations. It has a possible range of [0.0, 1.0]. It is normally only sampled at six values: 0.001, 0.01, 0.1, 0.3, 0.5, 0.8.
- *DD*, *Dd*, and *dd* – These parameters, called penetrance values, are the probability a person with this genotype ("*DD*" or "*Dd*" or "*dd*" alleles) becomes affected by the disease. The range for these parameters is [0.0, 1.0), with a normal step size of 0.1.

4 Two-Dimensional Projection of Line Segments

For a multidimensional grid, one can view traversing through the data as walking on a multidimensional surface. At each point, one step can be taken forwards or backwards in any dimension to reach another point. Thus each point has $2N$ neighbors, unless the current point is at the boundary of one or more parameters. When moving from one point to the next, the GLR value will change, which can be thought of as a change in elevation. The slope can be determined by taking the line from the original point to the new point. Thus, because the slope of the surface is a large part of what we want to visualize, our approach is to concentrate on visualizing these high dimensional line segments. A *line segment* is defined as a line formed by two endpoints, in which the values for the endpoints only differ in one dimension, and in that one dimension the values are consecutively sampled values of that dimension. The first step in our method is to extract all line segments from a data

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

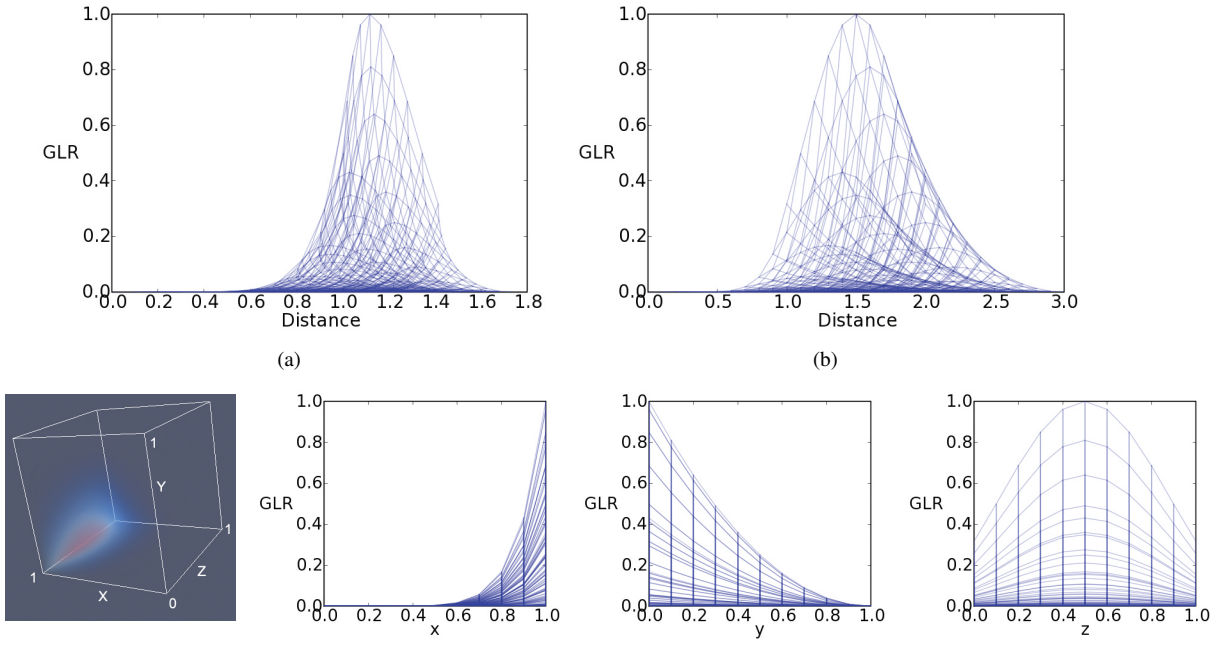

Figure 1: An example 3D function, $f(\mathbf{x}) = x^8(1-y)^2(1-(z-0.5)^2)^4$, which was sampled from 0 to 1 with a step size of 0.1 in all dimensions. (a) The distance plot of the function, with distance point at (0, 0, 0), using euclidean distance. (b) The distance plot of the function, with distance point at (0, 0, 0), using manhattan distance. (c) A volume rendering of the function. The color scale ranges from blue (low) to red (high). (d)-(f) The parameter plots for the x, y, z parameters, respectively. The volume rendering shows that as x increases, the function value increases. This same trend is reflected in the x parameter plot.

set. As long as the function is sampled along a grid, this is easily done. To visualize these line segments, we focused on techniques that will project these line segments to 2D plots.

4.1 Parameter Plots

One way to project these line segments is to create a 2D plot where the x-axis is the value of one parameter, and the y-axis is the GLR. This type of graph is called a *parameter plot*. One can think of this projection as viewing a 2D function as a height map, and setting the view direction parallel to the x-axis or z-axis. We do this for every parameter. Note that this is done for all line segments, not just the line segments that change value in that particular dimension. Any line segment that is not a step in that dimension becomes projected as a vertical line in the graph. These vertical lines serve as visual indicators of the values at which that parameter was sampled. The parameter plots of a 3D function are shown in Figure 1(d)-(f).

These parameter plots are useful because they show how the high-dimensional space changes with respect to one parameter. Parameter plots that have segments with high slope indicate that the function varies greatly when the value of this parameter changes. On the other hand, if the segments are flat, that means the GLR changes very little over different values if this parameter, and the function does not care so much about this parameter.

4.2 Distance Plot

Another technique that we have developed to project these high dimensional line segments is to display the line segments as they would seem from the point of view of one point in space. First, we choose a *distance point*, which could be any point in the high dimensional space. Let p_d be the distance point, and let p_{e_1} and p_{e_2} be the endpoints for one line segment. We calculate the distance from p_d to each endpoint. Let d_1 be the distance from p_d to p_{e_1}, and d_2 be the distance from p_d to p_{e_2}. A d_1 and d_2 are calculated for every line segment in the data set. We create a *distance plot*, which is a 2D plot where the x-axis plots the distances, and the y-axis is the GLR value.

This type of projection, showing line segments as they would seem from a point in space, can also be thought of as a type of radial projection. For a 2D function over the domain $[0,1]^2$, assuming the distance point is at the origin, the result of this projection is the equivalent of taking the height map of the function and doing a radial sweep from the y-z plane and ending at the y-x plane, with the pivot being the y-axis. One characteristic of this projection is that segments and features that are equidistant to the distance point end up overlapped, even if those features are far away from each other. For functions higher than 2D, it is more difficult to visualize what the projection is doing, but this property will always hold.

The distance plot gives a general overview of the space from the view of one point. For each point on the plot, all possible paths to its neighbors are shown, along with the change in GLR when moving to those neighbors. This property is important, because the parameter plots only show changes to the space as one parameter changes. The distance plot will show how the GLR changes from one point as all possible parameter values change. Interesting structures that involve value changes in multiple parameters can be discerned. In general, the user is guided by the distance plot in the exploration of the high dimensional space by investigating these structures. Another benefit of the distance plot is that it compactly displays the total change around one point, or set of points. For example, if the user wanted to determine the general slope around a peak without a distance plot, every parameter plot would have to be scanned, and the highest point would need to be found. Then the user would need to observe and study what is happening to this point as values are changing in every parameter plot. With the distance plot, all paths from the maximum can be seen at once, so determining the general slope can be done much more efficiently.

Care must be taken when interpreting this plot, because distances between two unconnected points in the plot are not accurate. Points that are close to each other only mean that their distance to the

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

distance point is similar. They are not necessarily close to each other. When looking at a distance plot and studying a peak, there is a tendency to concentrate on the shape the outline of the peak makes to characterize its slope. This is not the correct way to view the plot. To reasonably distinguish the slope of a peak, one needs to visually take into account the slope of *all* line segments within the area of the peak, and not just its outline.

When generating the distance plot, there is the option of which exact distance metric to use. We experimented with different distance metrics, most notably euclidean distance and manhattan distance. For a point $(x_1, x_2, ..., x_N)$ and a point $(y_1, y_2, ..., y_N)$, the manhattan distance between the two points is defined as

$$distance_{manhattan} = \sum_{i=1}^{N} |x_i - y_i| \quad (1)$$

Both the euclidean and manhattan distance metrics have their own pros and cons. The euclidean distance is more intuitive, and the distances assigned to a point have some meaning, e.g. this point has a distance of 0.5 from the global max. The disadvantage of euclidean distance is that the appearance of line segments partly depends on the distance from the distance point. For example, a line segment that has a high slope, but is relatively far away from the distance point may appear as a vertical line, thus distorting the actual slope of the segment. Fortunately this is not a problem when using manhattan distance, where a line segment's displacement (the amount the segment spans over the distance axis) on the distance plot is always the same as its step size, since line segments only cross over in one dimension. Thus, a line segment that represents a step of 0.1 in one dimension will have a displacement of 0.1 in the distance plot. So using the manhattan distance metric results in a plot that accurately represents the slope of all line segments. Using manhattan distance has the disadvantage of the distances assigned to a line segment's endpoints being unintuitive. For example, if one endpoint of a line segment is assigned a distance of 3 from the distance point, very little can be inferred from it. In practice, the resulting images from the two distance metrics are similar. Figure 1(a) and (b) show distance plots using the two distance metrics. When using our technique, we usually use both distance metrics to see the data from different perspectives.

One major consideration in generating a distance plot is the location of the distance point, which can dramatically affect the resulting plot. Figure 2 illustrates how the function featured in Figure 1 appears using different distance points. The choice of the distance point can affect whether peaks can be seen, or whether peaks are obscured by other overlapping features. One useful location for the distance point is the maximum point of a peak. This way, line segments close to zero can generally be assumed to be part of the peak. As the distance increases, the segments show how the slope of the peak changes. Another strategy in deciding the location of the distance point is to choose a point such that all values are either the minimum or maximum sampled value of that parameter. This prevents two problems. First, a poorly chosen distance point could have parts of features "reflect" back. Figure 4 shows how a distance point that lies in the middle of the peak will reflect back any line segments that are part of the peak and are behind the distance point relative to the max point of the peak. The second problem is a related issue whereby a segment can "wrap around" if the distance point intersects the segment. For example, in a 3D space, let the distance point be (0.5, 0.0, 0.0). Assume a line segment exists that have endpoints of (0.0, 0.0, 0.0) and (1.0, 0.0, 0.0). Both line endpoints of the line segment are 0.5 units away from the distance point. It will be mapped as a vertical line, which is a false representation of this line segment. Choosing a distance point at one of the corners of the multidimensional grid will avoid both of these problems.

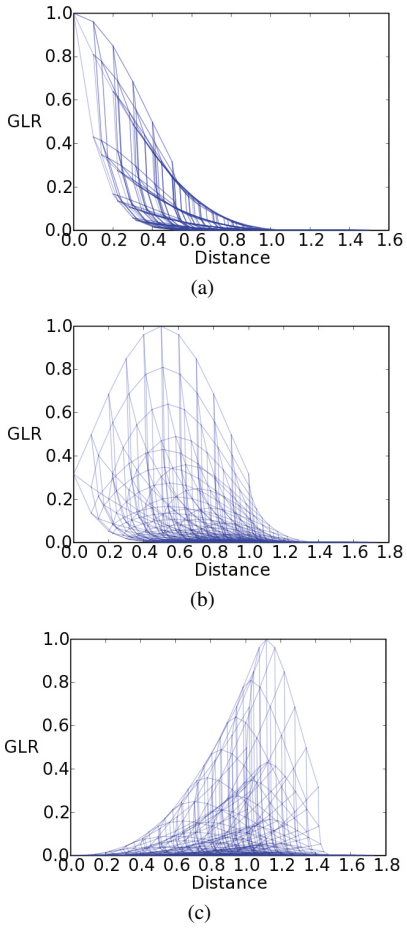

Figure 2: How different distance points affect the distance plot of the function featured in Figure 1. (a) The distance point at the global maximum, (1, 0, 0.5). (b) The distance point at (1, 0, 0). (c) The distance point at (1, 1, 1).

5 Interaction Techniques

By itself, the distance plot gives information about the possible paths of the space, and how the GLR value changes along these paths. There is no visual indication, however, of the exact parameter values of a line segment. There is also no way to infer which dimension the line segment moves in. The plot may be able to show a peak and other features, but it is unknown where exactly in the high dimensional space the features lie in. We have developed interaction techniques with our plots that solve these shortcomings.

5.1 Highlighting Lines

Our main interaction technique is to allow the user to select and highlight segments of interest. Highlighted lines are shown as a different color, larger width, and increased opacity in order for them to stand out. For all the figures in this paper, highlighted lines are colored red. Once a line segment is highlighted in one plot, that segment is also highlighted in all other plots. Since all line segments will be projected as a vertical line in every parameter plot except for one, these vertical lines serve as visual markers to indicate the exact parameter value in each dimension. For example, suppose a 3D grid with dimensions x, y, z, with which each dimension is sampled using step size of 0.1. If a line segment with endpoints (0.1, 0.1, 0.3) and (0.2, 0.1, 0.3) is selected, then in the x dimension parameter plot a line that stretches from 0.1 to 0.2 will be highlighted, while in the other parameter plots a vertical line will appear at 0.1 and 0.3 for the y and z parameter plots, respectively.

The user selects lines by dragging and designating a rectangular

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

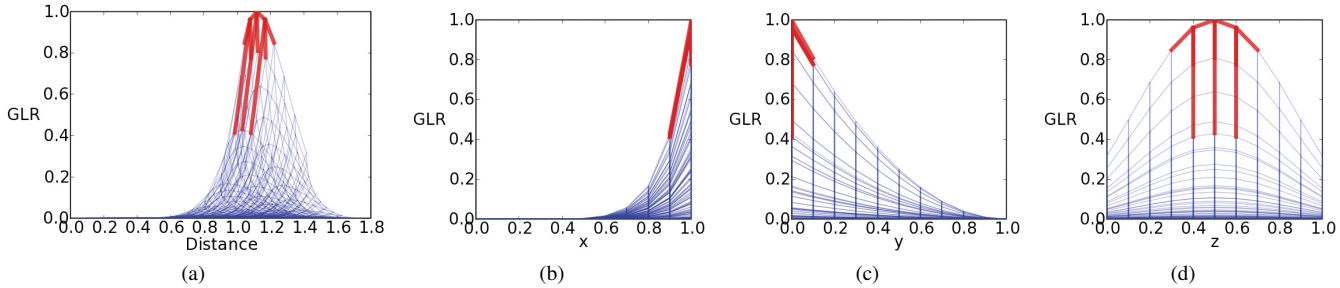

Figure 3: Line segments being highlighted after selecting the top portion of the peak. (a) The distance plot. (b)-(d) parameter plots for the x, y, z parameters, respectively.

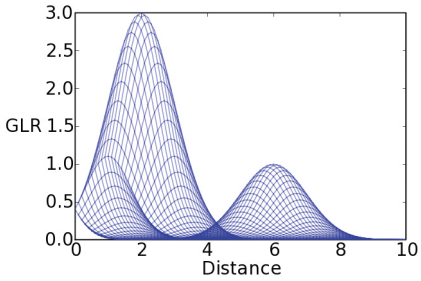

Figure 4: A distance plot of 2D Gaussians. The left part of the large peak seems to "reflect" back on itself because the distance point lies on a point on the peak.

region on the plot. Any segments that lie in that region are selected. Because lines may overlap in the plots, it may be difficult for the user to select a specific line. Fortunately, there are options to zoom and pan the current view. So if a user wants to select a particular line but there are several overlapping lines making it difficult, then the user can simply zoom in to an unobstructed part of the line.

Selecting and highlighting lines this way allows the user to select features of interest in any of the available plots, and then have a visual indication of where that feature lies in the space, and the range of the different parameter values present in that feature. Figure 3 displays highlighted lines after selecting the top portion of the peak.

5.2 Other Interaction Techniques

Another option that is afforded to the user is the ability to change the distance point to any set of values. This can be useful for finding interesting features that would have otherwise been hidden because of overlapping line segments. The user can also switch between using euclidean or manhattan distance for the distance plot. Our visualization program also supports plotting individual points onto the distance plot and the parameter plots. This is useful to indicate the location of local maxima or local minima.

6 Filtering

One problem when trying to visualize high dimensional data sampled from a grid is that the number of points can quickly become exponentially large. The worst case scenario for our problem is an analysis that uses all seven available parameters, with normal grid range and step size as specified in Section 3. Thus, in the worst case the total number of points is approximately 39 million points. The number of total line segments is even greater than the number of points. This many line segments is too much to render at once while trying to keep the display interactive. Even when that many lines are rendered, the amount of overlap makes it impossible to distinguish any line segments apart. In these situations, we reduce the total number of points used by filtering. Depending on the specific task, different filtering methods can be applied.

One option is to perform a radial filter around a point of interest, keeping only line segments that lie within a certain distance of one or more points of interest. This option is particularly helpful when wanting to investigate an already known feature. For example, keeping only line segments within a certain threshold of a local maximum will retain line segments that are part of that peak.

Another way to reduce the number of points in the data is to perform a threshold on the GLR value. This type of filter works especially well for our problem because of the nature of our data. For many of our data sets, most of the surface is relatively flat and has a low GLR, except for a handful of large peaks. For example, one of our data sets has a GLR range of 0 to 250,000. The average value of all the points, though, is a GLR value of 40. This indicates that performing a threshold will cull away a large percentage of the points. In this case, keeping only the points with a GLR value above 10,000 resulted in removing 95% of the data points.

Using a coarser grid is another way to reduce the number of points in the data set. Reducing the number of samples along one dimension by half, by doubling its step size, will halve the number of total points. Applying a coarser grid has the disadvantage that the coarser grid will remove fine features from the data, and could miss a peak entirely. A coarser grid could be used to help identify an interesting subspace, which can then be sampled with a denser grid for further inspection.

7 Case Studies

The following sections detail specific case studies of our visualization technique using real world genetic data. The data include families in which at least two children have autism. A linkage analysis performed on this data set indicated a certain chromosome position of interest. The multidimensional genetic likelihood information associated with this position was extracted for each individual pedigree. Thus, there is a separate high-dimensional scalar field for each pedigree in the data set. This was done in an effort to see how each individual pedigree affected the final PPL value. The data have five dimensions: α, gf, dd, Dd, and DD. Because we are only looking at a single pedigree at time, α can be dropped, since this parameter controls the degrees of influence of different pedigrees. This results in four dimensions with a total of 1,650 points and 10,500 line segments to visualize. The case studies in the following sections detail the task of investigating a high dimensional peak, determining the trait model of a peak, and comparing the likelihoods of two different pedigrees.

7.1 Investigating a Peak

One of the goals in our visualization of this high dimensional space is to understand the slope around peaks, and to see which parameters affect the GLR value the most. Figures 5 and 6 show the resulting plots of one of the pedigrees in our autism dataset. From looking at the distance plots, it can be seen that a ridge runs along the top part of the highest peak. The result of selecting the line segments that lie along this ridge can be seen in Figure 5. From looking at the parameter plots, it can be determined that the ridge is formed by line segments which span all values of gf, while another series of line segments span all values of DD.

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

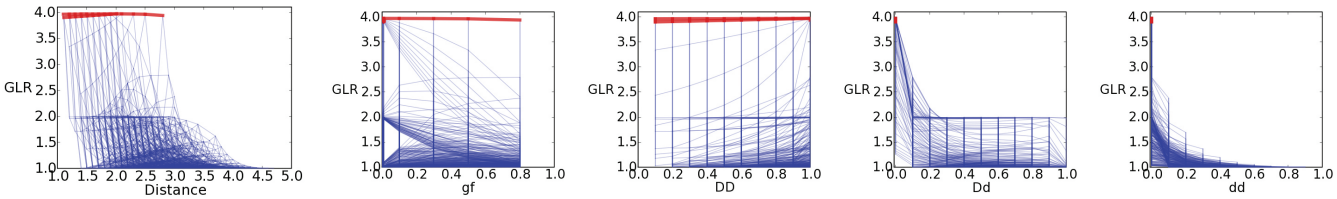

Figure 5: The plots produced from one pedigree of a four dimensional, real world data set. Line segments making up a ridge containing the global max are highlighted. (a) The distance plot of the data, with distance point at the origin, (0, 0, 0, 0). (b) The parameter plot of gf. The highlighted lines indicate that the function is not greatly affected by gf near the global max. (c) The parameter plot of DD. The highlighted lines indicate that the function is also not greatly affected by DD near the global max. (d)-(e) Parameter plots of Dd and dd, respectively. The only highlighting present is vertical lines at position 0.0, indicating that all selected line segments are within this subspace.

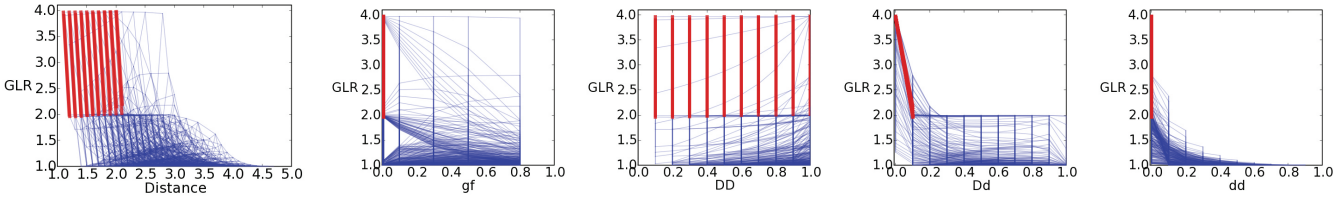

Figure 6: The same plots as in Figure 5, with steep line segments near the peak highlighted. (a) The distance plot of the data, with distance point at the origin, (0, 0, 0, 0). (b) The parameter plot of gf. (c) The parameter plot of DD. The multiple vertical lines indicate that the segments have varying values of DD. (d) The parameter plot of Dd. The highlighted lines indicate that all selected segments vary their values of Dd from 0.0 to 0.1. (e) The parameter plot of dd.

Another feature that can be observed from the distance plot is the set of very steep line segments that start from the top ridge of the peak. The user can select these lines to investigate them further. Figure 6 shows the result of selecting these line segments. It can be seen from the parameter plots that all these segments with large slope are steps in Dd, specifically from 0.0 to 0.1. Also, it can be determined that the only difference in values between segments are varying values of DD, since there is only one vertical line highlighted in each of the gf and dd parameter plots.

Using the information gathered from the previous line selections, it can be determined that around the global maximum, changing the values of gf and DD will keep the GLR value relatively the same. At any point along this ridge, though, if a change in the value of Dd occurs, then a sharp drop in the GLR occurs. Thus the likelihood cares greatly about the value of Dd around this peak.

7.2 Determining the Trait Model

An important piece of information about the likelihood model of a pedigree is whether it favors a dominant trait model or a recessive trait model. This is usually done by looking at the values of DD, Dd, and dd at the global maximum. The ratio DD/Dd and Dd/dd are calculated. If the ratio DD/Dd is closer to 1, then the data favor a dominant trait model. If on the other hand the ratio Dd/dd is closer to 1, then the data favor a recessive trait model. But performing this on the global max only tests the trait model of the highest peak. Other peaks in the data may favor another trait model, or may not favor any particular model. Using visualization, peaks in the data can be analyzed to see what trait model they favor. The line segments at the top of the peak are selected, and then the parameter plots of DD, Dd, and dd are inspected. If the Dd values of the highlighted lines are close to the DD values of the highlighted lines, then the peak favors a dominant model. On the other hand, if values of Dd are closer to values of dd, then the peak favors the recessive model.

Figure 7 shows the resulting plots from a pedigree in our autism data set. In it, the large peak can be seen to be dominant. This is because the Dd values of the selected line segments are close to 1, as can be seen in Figure 7(d), while DD values of the selected segments are also close to 1 (Figure 7(c)). Note that dd values lie at the opposite end, close to 0 (Figure 7(e)). The same pedigree is displayed in Figure 8, with the difference being a different distance point is selected for the distance plot. From this particular view, a smaller peak can be seen, which was occluded before. We can use the same method to determine what trait model this peak favors. Figure 8 shows the result of selecting the lines segments of the peak, revealing that the peak favors a recessive trait model.

7.3 Comparing Pedigrees

One reason to examine each pedigree data separately is to see how each individual pedigree affects the final PPL value. Having a separate data set for each pedigree allows us to see the differences between pedigrees. Information about how the space generated from one pedigree is different compared to another pedigree's space can be useful to a researcher. For example, in the presence of heterogeneity, this can be used to find the subset of pedigrees most likely to be "linked" at the location of interest.

The initial plots obtained from the high dimensional data can be compared to each other to look for similarity. If two likelihoods are very different, then their respective plots will look different. If the plots are similar, then the two likelihoods may be similar, but because of the nature of the projection, similar plots do not guarantee that the actual spaces are alike. This is because different structures could overlap, and result in similar looking plots even though they are actually different features.

One feature that was added to specifically address comparing two data sets is the ability to select the exact same line segments in two different data sets. The same line segments refers to line segments that have endpoints with the same parameter values, but not necessarily the same GLR values associated with them. Currently, this can only be done with data sets that were sampled using the same grid. To determine differences between pedigrees, a user can select interesting line segments from one pedigree, and then have the same segments highlighted in the other data set. This assures the fact that the same area in the high dimensional space is being examined. For this case study, we used two different pedigree data

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

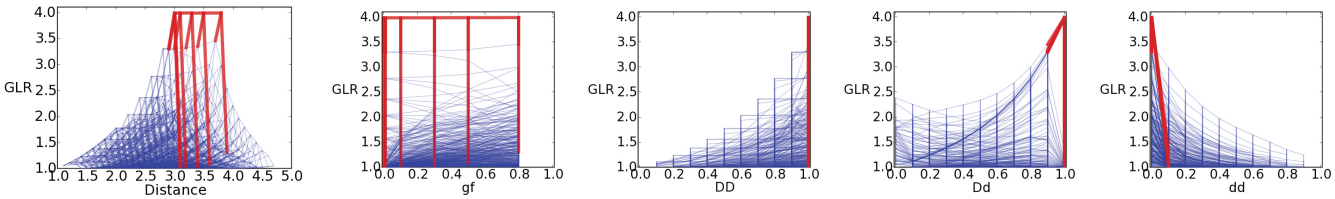

Figure 7: Plots of the pedigree data discussed in Section 7.2. (a) The distance plot, with distance point at the origin, (0, 0, 0, 0). Line segments near the peak are highlighted. (b)-(e) The parameter plots for gf, DD, Dd, and dd, respectively. The highlighted lines of the parameter plots of DD and Dd have the same range of values, indicating that the peak favors a dominant trait model.

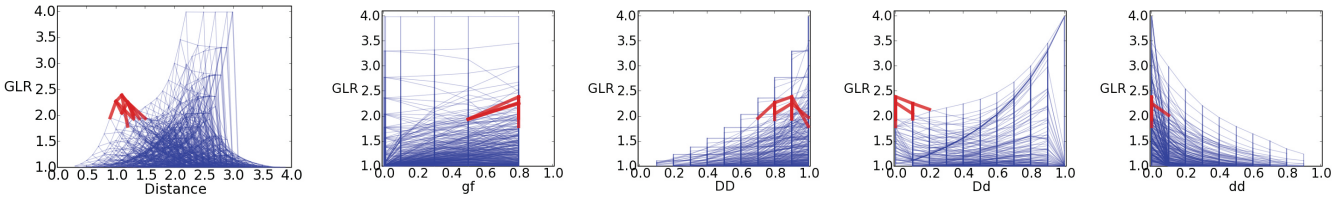

Figure 8: Plots of the pedigree data discussed in Section 7.2. (a) The distance plot, with distance point at (1, 0, 0, 0). A second peak can now be seen. The lines at the top of this peak are selected. (b)-(e) The parameter plots for gf, DD, Dd, and dd, respectively. The highlighted lines of the parameter plots of Dd and dd have the same range of values, indicating that this peak favors a recessive trait model.

from our autism dataset. The first one, ped_1, is the same one studied in Section 7.1. The other pedigree, ped_2, is another pedigree from our dataset that has a very similar pedigree structure to ped_1. The distance plots of each data set are shown in Figure 9, with the distance point being (0, 0, 0, 0). From looking at these plots, there seems to be some similar structures, mainly on the left side.

The first comparison examines the large peak in ped_1. Line segments comprising the ridge that includes the global max are manually selected, along with some of the steep line segments that come off of this ridge. The same line segments are then selected in ped_2. The result of these operations are displayed in Figure 9. As can be seen in Figure 9, the right side of the peak has sunk.

The next comparison deals with the question of where in the high dimensional space the evidence for linkage has changed. GLR is a value relating to the amount of linkage evidence, so a high GLR indicates greater evidence for linkage. The GLR can also signal evidence *against* linkage, if the GLR value is less than 1. For ped_1, the minimum GLR value is 1, indicating that there is only evidence for linkage in this data set. On the other hand, ped_2 has regions where the GLR dips below 1. A question to ask is, for ped_2's region below 1, how does that region map to ped_1. The answer can be found by highlighting the entire region in ped_2 that lies below 1, then selecting those same segments in ped_1. The results, shown in Figure 10, illustrates that much of the right side of the distance plot of ped_1 is in the same region that is below 1 in ped_2. The parameter plots indicate the coordinates of the subspace that this region lies in. The highlighted region spans all possible parameter values of gf, Dd, and dd. Note that the highlighted range of DD is restricted to the higher values of DD (0.7 to 1.0). Thus, a conclusion can be formed that much of the subregion of the high dimensional space in which there are high values of DD exhibits evidence for linkage in ped_1, but in ped_2 indicates evidence against linkage.

8 Discussion

We have shown a technique for visualizing and exploring a multi-dimensional genetic likelihood space. Even though we were able to gain insight into our real world data, this approach is not without its share of disadvantages. One disadvantage is that line segments which end up lying close to each other in the distance plot may actually lay far apart in the high dimensional space. The user can

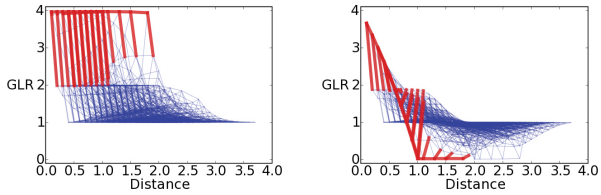

Figure 9: Selecting and highlighting the peak to see how it changes from one pedigree to the next. (a) ped_1's peak is highlighted. (b) The same line segments are highlighted in ped_2. The right half of the main peak has sunk. (See Section 7.3)

always select them and see where they are in the space, though. Another issue is the problem of overlapping lines. A user may wish to select segments comprising a peak, but will probably not want to select other extraneous segments that are not part of the structure, but were still projected to the same area. Currently there are few good ways around this problem, but fortunately our data has only a handful of peaks in the GLR space (we think this will usually be true in genetic applications). Thus, viewing the data using a couple of different distance points will usually reveal all the prominent peaks in a data set. If this technique was applied to high dimensional data that had many more peaks, it may be very difficult to distinguish individual peaks. New interaction techniques would have to be developed to help the user. Scalability, in terms of the number of parameters, is also another concern. For each additional parameter, the number of points increases by a factor of the number of samples taken in that parameter. As the number of points increases, the density of the plots also swells. This results in more lines overlapping and occluding each other, thus making it difficult to discern any patterns. In our experience, we have found that plots of up to seven dimensions are explorable. After that point, the plots become largely imperceivable, especially the distance plot.

Because highlighting lines is a crucial element of understanding exactly where structures lie in the high dimensional space, the response time is an important part of the user experience. The response time of our program does well but will slow down when dealing with a large amount of data. Besides speed, memory usage is also an issue. As previously mentioned, the number of points in

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.

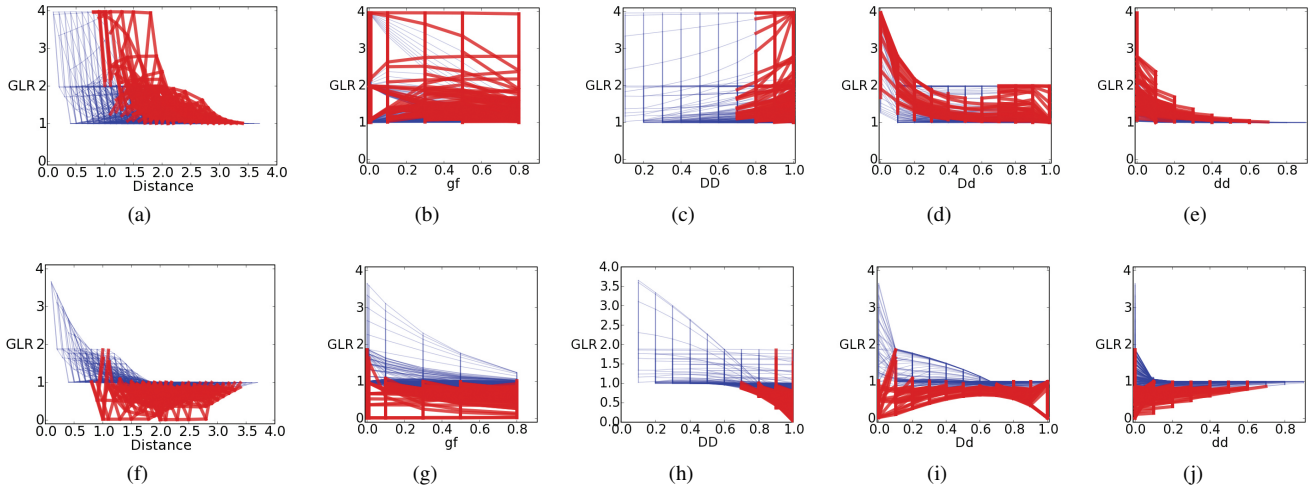

Figure 10: Selecting the region in ped_2 that has a GLR below 1, and seeing how that region maps to ped_1. (a) The distance plot of ped_1. (b)-(e) The parameter plots for ped_1, using parameters gf, DD, Dd, and dd, respectively. (f) The distance plot of ped_2. (g)-(j) The parameter plots for ped_2, using parameters gf, DD, Dd, and dd, respectively. (See Section 7.3)

a dataset can reach up to 39 million. Adding the fact that each line segment is duplicated once for each of the $N+1$ plots, the amount of memory usage can be very large. There have been several occurrences of not being able to load all parameter plots due to lack of memory for some data sets. The filtering techniques discussed in Section 6 are applied in these cases.

9 Conclusion and Future Work

A new method for visually exploring multidimensional genetic likelihood spaces has been presented. It includes many improvements over previous visualization attempts. It displays a much larger region of the high dimensional space compared to earlier methods. A distance plot utilizing a radial projection is used to provide a high level overview of the entire space. The main features we are interested in, peaks in the data, can be easily identified from this plot. By highlighting line segments, the exact values of where these features occur can be determined from the parameter plots. Also, parameter plots indicate if individual parameters hold a considerable sway on the likelihood. Other tasks such as determining the trait model of a pedigree and comparing the likelihoods of two pedigrees can be accomplished using our new method. This technique has already helped shed some light into the underlying genetic mechanism of complex disorders, such as autism.

For future work, we plan to add more interaction techniques to help the user more easily explore the high dimensional space. A selection filter based on slope could be helpful. One technique that might be useful is to have selected line segments "move" along a dimension, interactively changing their value along one dimension and showing the user the resulting change. Dealing with the problems incurred in overlapping segments is another area that needs to be addressed. One method that might alleviate this problem is to change the opacity of line segments based on their distance to another point. Recently, work has been done to sample the likelihood over a dynamic grid instead of a regular grid [8]. Modifications will be required to visualize this multidimensional dynamic grid. The PPL framework itself scales well and currently handles genome scans based on general pedigree structures and moderately complex models involving epistasis [11]. Further applications of our visualization technique will focus on extensions to these types of models.

Acknowledgements

The authors wish to thank Dr. Peter Szatmari and Dr. Steve Scherer for providing the autism data.

References

[1] R. C. Elston. Man bites dog? The validity of maximizing LOD scores to determine mode of inheritance. *Am J Med Genet*, 34(4):487–488, 1989.

[2] S. K. Feiner and C. Beshers. Worlds within worlds: metaphors for exploring n-dimensional virtual worlds. In *UIST '90: Proceedings of the 3rd annual ACM SIGGRAPH symposium on User interface software and technology*, pages 76–83, New York, NY, USA, 1990. ACM.

[3] D. A. Greenberg. Inferring mode of inheritance by comparison of LOD scores. *Am J Med Genet*, 34(4):480–486, 1989.

[4] T. Mihalisin, J. Timlin, and J. Schwegler. Visualizing multivariate functions, data, and distributions. *IEEE Comput. Graph. Appl.*, 11(3):28–35, 1991.

[5] J. Ott. *Analysis of Human Genetic Linkage*. The Johns Hopkins University Press, Baltimore, MA, USA, 3rd edition, 1999.

[6] J. W. Park, J. F. Cremer, and A. M. Segre. Visual exploration of genetic likelihood space. In *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, pages 1335–1340, 2006.

[7] J. W. Park, M. Logue, J. Ni, J. Cremer, A. Segre, and V. J. Vieland. Scientific visualization of multidimensional data: Genetic likelihood visualization. In *Current Trends in High Performance Computing and Its Applications*, pages 403–408. Springer Berlin Heidelberg, 2005.

[8] S. Seok, M. Evans, and V. J. Vieland. Fast and accurate calculation of a computationally intensive statistic for mapping disease genes. *J Comput Bio*, 16(5):659–676, 2009.

[9] J. J. van Wijk and R. van Liere. Hyperslice: visualization of scalar functions of many variables. In *VIS '93: Proceedings of the 4th conference on Visualization '93*, pages 119–125, 1993.

[10] V. J. Vieland. Bayesian linkage analysis, or: How I learned to stop worrying and love the posterior probability of linkage. *Am J Hum Genet*, 64(4):947–954, 1998.

[11] V. J. Vieland. Thermometers: Something for statistical geneticists to think about. *Hum Hered*, 61:144–156, 2006.

[12] V. J. Vieland and S. E. Hodge. The problem of ascertainment for linkage analysis. *Am J Hum Genet*, 58(5):1072–1084, 1996.

[13] V. J. Vieland, Y. Huang, C. Bartlett, T. Davies, and Y. Tomer. A multilocus model of the genetic architecture of autoimmune thyroid disorder with clinical implications. *Am J Hum Genet*, 82:1349–1356, 2008.

[14] N. S. Wratten, H. Memoli, Y. Huang, A. M. Dulencin, P. G. Matteson, M. A. Cornacchia, M. A. Azaro, J. Messenger, J. E. Hayter, A. S. Basset, S. Buyske, J. H. Millonig, V. J. Vieland, and L. Brzustowicz. Identification of a schizophrenia associated functional non-coding variant in NOS1AP. *Am J Psychiatry*, 2009.

Authorized licensed use limited to: The Ohio State University. Downloaded on August 17,2010 at 02:52:39 UTC from IEEE Xplore. Restrictions apply.