Defence Research and Development Canada
Recherche et développement pour la défense Canada

ChromaBlast – a Data Visualization Tool

Barry N Ford, Yimin Shei, and Stephen Bjarnason
Defence R&D Canada – Suffield

Catharine Richardson
Webgenii Consulting

Technical Memorandum
DRDC Suffield TM 2006-049
June 2006

Canada

ChromaBlast – a Data Visualization Tool

Barry N Ford, Yimin Shei, and Stephen Bjarnason
Defence R&D Canada – Suffield

Catharine Richardson
Webgenii Consulting

Defence R&D Canada – Suffield

Technical Memorandum

DRDC Suffield TM 2006-049

June 2006

Author

B.N. Ford

Approved by

L.P. Nagata

HCBDS

Approved for release by

P.A. D'Agostino

DRP Chair

Abstract

Experimental systems in biology now have the capability to produce massive amounts of numerical data. Large scale analysis of such data is facilitated by costly integrated software packages. Often however, such packages have limited or no data reduction or manipulation tools, such as basic spreadsheet functionality. Thus the researcher is compelled to utilize a competent spreadsheet package, such as Microsoft Excel, then export the data set to analysis software. Reformatting data or reducing it for analysis must then be redone in the spreadsheet after each analysis run, until the final dataset is appropriately formatted for the analysis package. In order to facilitate the use of the capabilities of the spreadsheet software and perform data reduction or basic analysis without having to switch back and forth between software units, we are building a set of analysis tools in Excel. One of the most useful of the tool set is ChromaBlast, which normalizes columnar data, sorts the data into user-selectable range-driven bins, develops a colour heat map from the data, and outputs the heat map and bin assortment for review. Using intrinsic tools in the spreadsheet software, output data can be filtered and sorted to emphasize data patterns, and facilitate rapid data review.

Résumé

Les systèmes expérimentaux actuellement disponibles en biologie ont la capacité de produire des données numériques en quantités massives. L'analyse à grande échelle de telles données est facilitée par des progiciels intégrés coûteux. De tels progiciels n'ont cependant qu'une capacité limitée de dépouillement de données ou d'outils de manipulation telle que la fonctionnalité élémentaire des tableurs. Le chercheur est donc astreint à utiliser des progiciels tableurs qui soient compétents tels que Microsoft Excel, puis à exporter l'ensemble des données dans un logiciel d'analyse. Restructurer ou dépouiller les données pour leur analyse doit être effectué à nouveau dans le tableur après chaque analyse, jusqu'à ce que l'ensemble final des données soit restructuré de manière appropriée pour le progiciel d'analyse. Pour faciliter l'utilisation des capacités du logiciel tableur et effectuer le dépouillement des données ou bien l'analyse élémentaire sans avoir à faire le va et vient entre les unités de logiciels, nous avons construit un ensemble d'outils d'analyse dans Excel. Un des outils les plus utiles est ChromaBlast qui normalise les données en colonnes, trie les données dans les fichiers définis par des plages paramétrables par l'utilisateur, développe une carte thermographique en couleur et exécute une sortie de la carte thermographique et la sélection des fichiers pour l'examen. L'utilisation d'outils intrinsèques au logiciel tableur permet de filtrer les données de sortie et de les trier pour mettre en évidence les schémas de données et faciliter l'examen rapide des données.

This page intentionally left blank.

Executive summary

Background: Recent efforts in genomics and microarray-based gene expression analysis have led to massive data sets which require enhanced review and analysis tools. Many commercial or open source data analysis packages do not have have adequate spreadsheet capabilities. For example, Statistica, a widely used comprehensive statistical package, does not have the capabilities to resort data, cut and paste, or search and replace cells. Spreadsheet software, which has excellent data manipulation features, usually has only basic charting and statistics, often requires hand entry of formulas, and multiple steps for basic analysis. User-defined macros are a useful shortcut system, but usually have insufficient flexibility for repetitive, similar but different data sets. Typical analyses of large data sets involve multiple iterations of data formatting, pruning, resorting, and reduction.

In order to begin to defeat these mixed shortcomings, we are developing a set of data reduction and analysis tools which exploit the capabilities of existing spreadsheet software, while enabling data review and analysis. ChromaBlast is a component of this effort.

Results: A data visualization tool, ChromaBlast, was developed to facilitate rapid and intuitive data display, and to assist in reducing data complexity for interpretation. This tool has been used to analyze genomic microarray data as well as other data types.

Significance: Well documented data visualization tools which are easy to use are lacking within the general area of bioinformatics. Using the Visual Basic for Applications environment for Microsoft Excel, such a tool has been developed with application in bioinformatics and other areas where visualization and rapid interpretation of complex data sets are required.

Future Directions: Enhancements to the existing software will include default color maps which can be user modified, and simpler extraction of alphabetic data representations.

Ford, B.N., Shei, Y., Bjarnason, S., Richardson, C. 2006. ChromaBlast – A Data Visualization Tool. DRDC Suffield TM 2006-049. Defence R&D Canada – Suffield.

Sommaire

Contexte : Les efforts récents en génomique et en analyses d'expression génétique à base de microréseaux ont abouti à des ensembles massifs de données requérant un examen approfondi et des outils d'analyse améliorés. Beaucoup de progiciels d'analyse de données commerciaux ou de source non secrète ne possèdent pas de capacités en tableur. Statistica, par exemple, un progiciel de statistique compréhensif très utilisé, ne possède pas la capacité de retrier les données, de couper et coller ou de rechercher et remplacer les cases. Les logiciels tableurs qui possèdent des caractéristiques excellentes de manipulation des données, n'ont normalement que des organigrammes et statistiques de base et exigent souvent d'entrer manuellement les formules et d'effectuer de multiples étapes pour les analyses élémentaires. Les macros configurées par l'utilisateur sont un système de raccourcis utiles mais qui ne sont pas suffisamment souples pour les ensembles de données qui sont répétitifs et similaires tout en étant différents. Les analyses ordinaires d'ensembles importants de données comportent des itérations de formatage, de coupures, retriage et de dépouillement de données.

Pour être en mesure de commencer à combler ces différentes sortes de lacunes, nous sommes en voie de mettre au point un ensemble d'outils de dépouillement de données et d'analyse qui exploite les capacités des logiciels tableurs existants tout en examinant et analysant les données. ChromaBlast est une composante de cet effort.

Résultats : Un outil de visualisation de données, ChromaBlast a été mis au point pour faciliter l'affichage rapide et intuitif des données et pour aider à réduire la complexité des données et mieux les interpréter. Cet outil a été utilisé pour analyser les données de microréseaux génomiques ainsi que d'autres types de données.

Portée des résultats : Le domaine général de la bioinformatique manque d'outils de visualisation de données qui aient été bien documentés et qui soient faciles à utiliser. Un tel outil a été mis au point, en utilisant l'environnement Visual Basic d'application de Microsoft Excel, ayant une application en bioinformatique et autres domaines où la visualisation et l'interprétation rapide d'ensembles complexes de données sont requises.

Orientations futures : L'amélioration des logiciels existants inclura des cartes en couleurs par défaut qui pourront être modifiées par l'utilisateur et une extraction simplifiée des représentations des données alphabétiques.

Ford, B.N., Shei, Y., Bjaranson, S., Richardson, C. 2006. ChromaBlast – A Data Visualization Tool. DRDC Suffield TM 2006-049. R & D pour la défense Canada – Suffield.

Table of contents

List of figures

Acknowledgements

Genomic fingerprinting data shown here were collected in part by Ms J. Bamforth during a Co-Op term, using DNA provided by Dr John Cherwonogrodzky and Nicole Stady from the DRDC Suffield BSL3 facility. Gene expression microarray data were prepared by M. Crichton, M. McWilliams, and R. Garneau from Canada West Biosciences.

This page intentionally left blank.

Introduction

Data collection from biological experimental work is no longer limited to a few replicates of a single enzymatic assay. Current laboratory efforts can produce gene expression data from tens of thousands of genes and dozens of replicates in a few days. The main work of biological data has shifted from laboratory effort, towards analysis and computational effort. Unfortunately, software tools for this work have not in general, kept pace with technological progress. What tools are available commercially are costly, minimally functional, and seem to remain in a permanent state of beta development, with poorly behaving interfaces, overly complex controls, and nonexistent customer support. Conversely, software developed for generic functions, such as spreadsheets, tend to have poorly documented algorithms for analysis, unvalidated statistical tools, and insufficient capacity for large datasets.

Examples of these problems can be found in locally deployed software. Microsoft Excel is a full function, high quality spreadsheet system for generic data. In a first pass gene expression experiment, 23,000 genes were analyzed in multiple replicates, at multiple time points. The data set is comprised of some 360 individual arrays of 23,000 intensity signals. The default installation of Excel cannot contain the entire dataset in one spreadsheet. Development of an enhancement to Excel using VBA (not documented here) was required in order to support the entire dataset. J-Express, a "fully functioned" gene expression microarray package, is able to load the entire dataset, but none of the analysis or display elements function when the entire dataset is loaded.

The usual solution to these issues is the employment of multiple software packages, attempting to exploit their capabilities, while working around the deficiencies. In order to begin to solve these issues for ongoing work, development of data visualization and analysis tools using the programming interface of MS Excel has been undertaken. ChromaBlast, a tool for normalization and visualization of data has been developed. ChromaBlast is a stand alone component designed for microarray work, but which can be used to look at various data types. This report describes the functions, interface, and output of ChromaBlast.

Materials and Methods

Software Code

ChromaBlast was coded in Visual Basic for Applications (VBA)[1,2], which is a programming language designed to connect the macro programming functions of Excel with relatively sophisticated manual coding in a development environment. Preliminary macros (keystrokes and program functions recorded while being used) can be used to quickly template a specific piece of work. Using the macro record as a framework, generalization, functional options, and user input can be implemented. Within VBA, a multitude of functions can also be added which cannot be accessed via the macro recording process. Thus the programmer can quickly template a concept with macros, then add function and usability inside the VBA environment. VBA can also be used to develop code from scratch, like any typical programming language.

Through a number of iterations involving the scientists who use the final product, a relatively simple tool was developed, which nevertheless has enormous functionality, and incidentally has properties during analysis which were not obvious in the design phase. ChromaBlast is either installed alone or as a component of a larger suite called BioTools.XLA. The code for ChromaBlast is at Annex 1. The code presented uses the BioTools front menu, but the ChromaBlast functionality is contained in the included script.

To use ChromaBlast, the user preselects (highlights) the range of cells for the subroutine, then selects the number and colour of bins to be applied. With that information, the subroutine then:

1. labels the selected range as a range, called *Analysis*.
2. adds the sheet "binsetup" from the BIOTOOLS addin.
3. reassigns the RGB value of the 56 available colours within the current file. (Note: the new colours become part of the information held within the file).
4. counts the number of bins assigned by the user.
5. creates an array containing the alphabetical labels assigned by the user.
6. determines the number of *Columns* and *Rows* in the *Analysis* range.
7. moves to first column within the range, selects the data in the first column.
8. determines the minimum and maximum values within each column.
9. using the minimum and maximum values creates the number of bins specified by the user on the "binsetup" sheet.
10. assigns each data value to a bin.
11. assigns a letter value to each data value.
12. pastes the letter value for each data cell into a new column to the right of the current data column.
13. loads the data letter value into an array
14. pastes the array of letters as a block two columns to the right of the last data column.
15. colours each cell in the array.
16. moves the array of letters to the right, leaving behind the block of coloured cells.

Results

ChromaBlast was designed to facilitate rapid comparison of microarray data, without the prior requirement for statistical analysis. It proves to be useful for analyzing disparate data types. Multiple numerical data models ranging from microarray data, population statistics, and global temperature data have been analyzed here with ChromaBlast. Comparison of large value ranges within columns is straightforward, since the binning strategy is intrinsically normalizing between columns. Notably, data with letter values or large numbers of null values or zeros are not appropriate for direct analysis with ChromaBlast. Conversion of nulls and letters to numeric representative values (which could represent one extreme value) could be a useful formatting strategy.

Figure 1 represents a data excerpt from a microarray genomic fingerprinting experiment, attempting to discriminate between various bacterial species and strains. Within the five main columns are represented hybridization patterns of two species, B. anthracis (left most) and E. coli. The primary differentiation at this level is whether or not the samples exhibit different hybridization patterns, or very similar ones. No attempt to present statistical support for the analysis is given at this point. 2000 data points from each of 5 arrays are represented. Even though the columnar values range widely (e.g. column 1 ranges from 1 to 59,717, column 3 from 1 to 29,178), comparison of the normalized data differentiates Bacillus anthracis from Escherichia species without difficulty. Column E representing chip feature numbers 1601-2000 is a region where the Bacillus sp. exhibits a very different pattern from the E.coli strains. Other areas such as Column A do not show very different patterns of hybridization. IT is clearly necessary to be able to see a large amount of the data set at once in order to discern the main areas of difference. Simple examination of the numerical values would not be a practical comparison strategy. ChromaBlast in this case allows an extreme reduction in data complexity for simple inspection.

Figure 2 illustrates the application of ChromaBlast to a dataset of yearly crime data from the United States Department of Justice (http://www.ojp.usdoj.gov/bjs/dtdata.htm). Represented are data from 1960 to 2003, including total census population estimates. The left most column indicates the population data for the range of years, and effectively recapitulates the color pattern in the heat map. In the ChromaBlast processed color map, it is apparent from the left most color column (map position corresponds to spreadsheet cell position) that the population is increasing with time, and essentially reproduces the defined heat map. The peak population occurs in bins for years 2001-3. Conversely, it can be seen that crime patterns do not match the population growth. Property crime actually reached its peak incidence and rate in the years 1980-81, while the murder incidence and rates peaked in 1991, and has been declining in both absolute frequency and rate ever since. Indeed, all the crime rates shown have been in decline since the early 1990s. These data could be displayed graphically, but representing all of these sources on a single chart would be rather confusing. The ChromaBlast representation is simple and compelling.

As a further example of the usefulness of this tool, we analysed global meteorological data for temperature during the period 1880 to 2004 (Figure 3). This particular dataset is currently highly politicized and controversial in the media, and is of broad public interest. in Figure 3, the heat map color pattern is recapitulated in the a column (population), increasing towards the bottom (bright yellow).

Figure 1. *ChromaBlast of first 2000 features of a genotyping microarray dataset. Data are unfiltered.*

Column	*Feature Numbers*
A	*1-400*
B	*401-800*
C	*801-1200*
D	*1201-1600*
E	*1601-2000*

In this figure, the data columns are genotype fingerprints from (left to right) : B.anth. RP42, E.coli DH5α , E.coli JM108, E.coli JM109, E.coli TOP10.

color	bin	max intensity	count	percentile
	G	56672	34	99.8
	F	48514	212	99
	E	40472	1023	95
	D	32377	3277	80
	C	24287	5147	58
	B	16189	6685	30
	A	8094	6861	

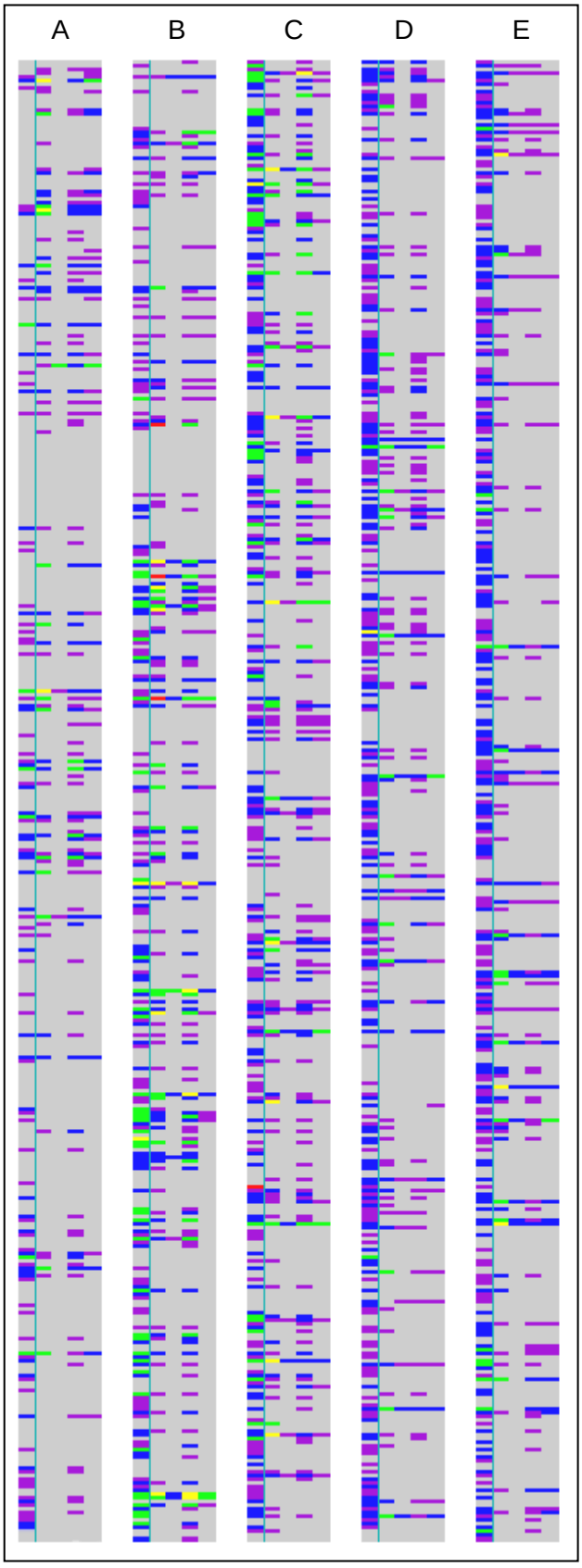


Figure 2. *Crime data for United States, 1960-2003. Source: http://bjsdata.ojp.usdoj.gov/dataonline/Search/Crime/State/statebystatelist.cfm. The left most column (population) recapitulates the heat map pattern.*

Column	Frequencies	Column	Rates
a	population	k	violent crime total
b	violent crime total	l	murder
c	murder	m	rape
d	rape	n	robbery
e	robbery	o	assault
f	assault	p	property crime total
g	property crime total	q	burglary
h	burglary	r	larceny
i	larceny	s	vehicle
j	vehicle		

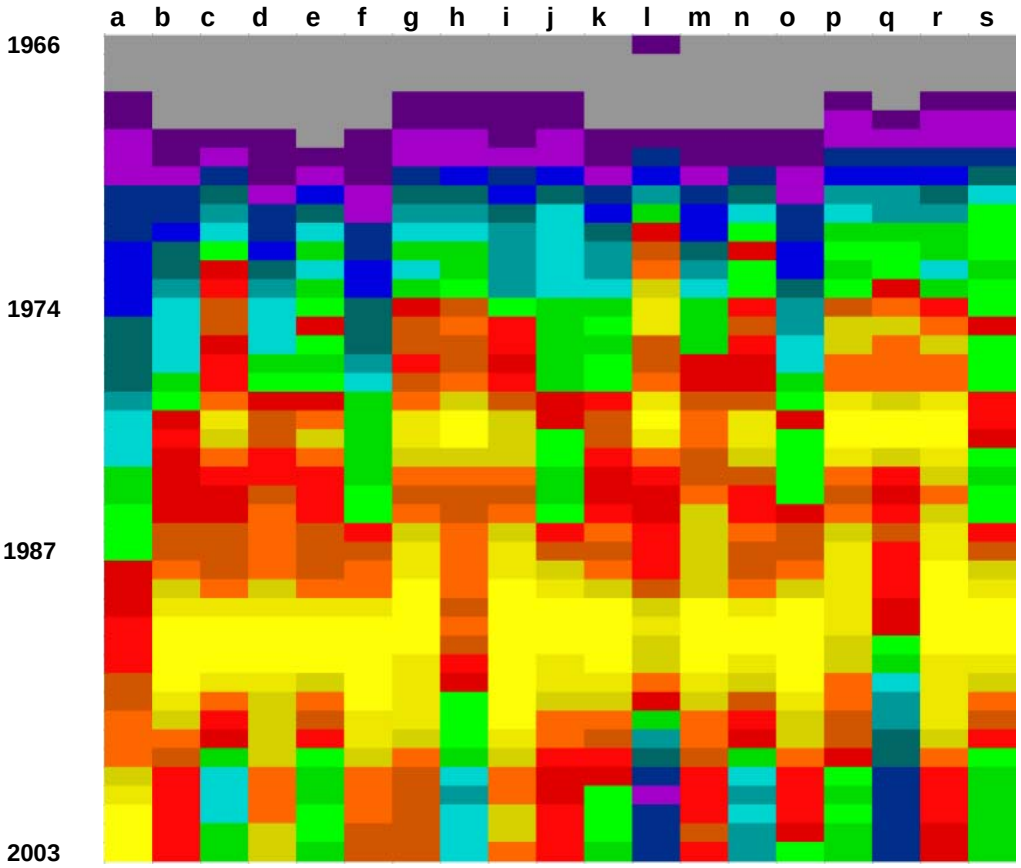
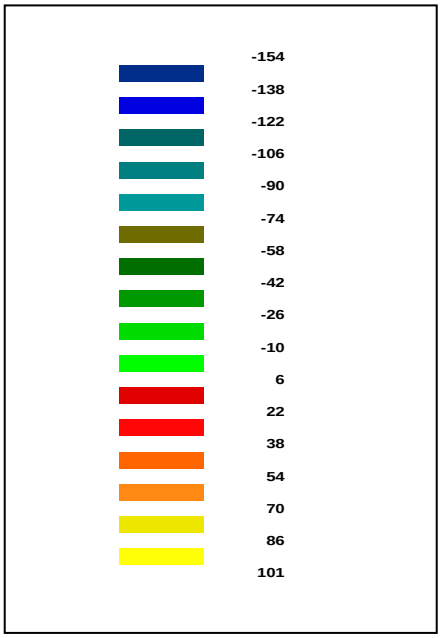
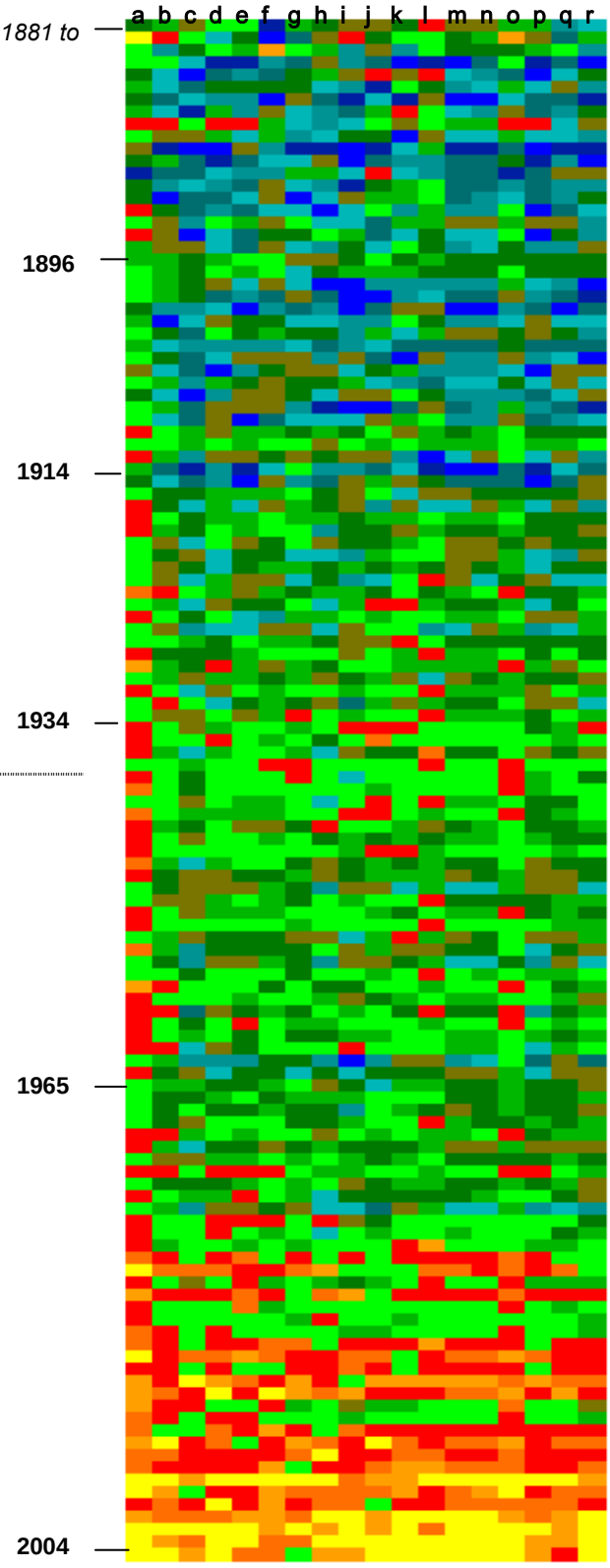


Figure 3. *Global temperature anomalies from 1881 to 2004, versus base period 1951-1980.*
Source : http://data.giss.nasa.gov/gistemp/tabledata/GLB.Ts.txt.
Values are in 0.01 °C increments.

Col	Period
a	Jan
b	Feb
c	Mar
d	Apr
e	May
f	Jun
g	Jul
h	Aug
i	Sep
j	Oct
k	Nov
l	Dec
m	Jan-Dec year average
n	Dec-Nov year average
o	Dec-Jan-Feb average
p	Mar-Apr-May average
q	Jun-Jul-Aug average
r	Sep-Oct-Nov average





Discussion

Using microarray technology, one can assay thousands of features at once, in each sample. A microarray in this application is a microscope slide onto which are spotted several thousand individual DNA probe sequences, each one of which can detect unique fragments of labelled DNA. Using DNA probes specific to known microbial sequences, one can identify with high confidence the species and probably the strain of organism under examination. Such a tool is a useful complement to existing PCR, RFLP, or AFLP technologies. Unfortunately, the data from microarray experiments is difficult to work with, often containing thousands of values, different scaling between experiments (i.e. between microarrays), relatively noisy variation within and between experiments, and complex patterns of expression [3,4]. Rapid and facile analysis tools are required with which one can perform simple data review and comparison [5]. ChromaBlast, running under the Windows™ environment in Excel™, is part of the answer to this problem.

ChromaBlast assigns a color and letter code to values within each column, dividing the columnar data into bins of equal size, distributed evenly over the range of column values. The maximum and minimum values are used to establish the range of values in each bin. The effect of this strategy is that wide ranges of values in adjacent columns will end up with color maps which are automatically scaled for the total range of the data in the column. Thus datasets which are scaled differently (e.g. from different instruments or data logging units) can be compared very easily without employing scaling or normalization functions.

The letter code simply represents an alphabetic coding for the bins. The alphabetic code could in principal be entirely arbitrary. The alphabetic code was developed for future exploitation in analysis using existing algorithms for comparing and aligning alpha datasets, such as Needleman-Wunsch [6], the basis for fast comparisons in genomic databases.

ChromaBlast is useful for genomic or gene expression microarray data review. The binning strategy intrinsically normalizes the data, enabling comparisons between quite different microarray sets. An unanticipated benefit of ChromaBlast is the effect of using bins which are asymmetrically distributed. This is achieved by simply assigning the same color to adjacent bins (color are not mutually exclusive). This proves to be useful when datasets contain an abundance of near-background values, or where values across a certain proportion of the data are over dispersed, often observed in microarray datasets. Because ChromaBlast assigns bin value ranges based on the upper and lower values in the column, datasets with frequent zero or small values will tend to have a large frequency of low-heat color assignments. If the dataset is deficient in middle range values, assigning more than one bin the same color in the middle range can reduce the data complexity substantially.

This tool was explicitly designed for microarray analysis, but can be readily applied to any data which have a similar pattern of sampling (e.g. iterative sampling over time in multiple replicates). Digitized data from graphical displays (such as spectrometers) could be compared in this way. Digital data collected over long time periods can be easily summarized on a single graphical display. The pattern of crime statistics shown in Figure 2 are an example of this.

ChromaBlast also has subtle display properties which may reveal issues within datasets. The global temperature anomaly data in Figure 3 is an interesting example (source http://data.giss.nasa.gov/gistemp/tabledata/GLB.Ts.txt.). A 20-year periodicity of positive temperature anomalies is apparent, with rising global maxima around 1900, 1920, 1940 and 1960, which subside within 2-4 years. Another global maximum might be predicted around 1980. Notably, the values recorded around 1980 are indeed 20 year maxima based on the prior years, but do not appear to decline again within a short period. Indeed, the temperature anomalies after 1980 appear to continue to increase. Also discernible is that certain months of the year seem to anticipate future trends. The temperature anomalies recorded in January reflect the annual global temperature anomalies, but also exhibit a long-range rising trend, which is recapitulated in annual averages with a multi-year lag.

It is apparent that ChromaBlast has value in complexity reduction and intuitive review of a diversity of data types. Future improvements will include a wider range of bins (currently ChromaBlast is limited to 26 bins), automatic bin assignment (with user modification optional) using generally optimized color maps, and streamlined output of the alphabetic values for other analysis methods.

References

1. Getz, G. and Gilbert,M. (2000) *VBA Developers Handbook*, 2nd ed. Sybex, Alameda,CA.

2. Bullen, S., Bovey, R., And Green, J. (2005) *Professional Excel Development: The Definitive Guide to Developing Applications Using Microsoft Excel and VBA*. Addison Wesley Professional, Boston, MA.

3. Kuo, W., Jenssen, T-K., Butte, A., Ohno-Machado, L., Kohne, I. (2002) Analysis of matched RNA measurements from two different microarray technologies. Bioinformatics 18;405-412.

4. Yang,Y., Dudoit S., Luu P., Lin D., Peng V., Ngai J., Speed T. (2002) Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. Nucleic Acids Res., 30(e15); 1-10.

5. Wruck, W., Griffiths, H., Steinfath, M., Lehrach, H., Radelof, U., O'Brien, J. (2002) Xdigitise: visualizaton of hybridzation experiments. Bioinformatics, 18;757-760.

6. Needleman, S.B., Wunsch, C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. J. Mol. Biol., 48;443-453.

Annex A

ChromaBlast Code

```
Attribute VB_Name = "ChromaBlast"
Option Base 1

Dim strMessage As String, strButtons As String, strTitle As
String
Dim intResponse As Integer
Const APPNAME = "ChromaBlast!"

Sub StartChromaBlast()
Dim intColourIndex As Integer

Dim strMsg As String
Dim strAns As String
   strMsg = "Have you selected the data you wish to
ChromaBlast?"
   strAns = MsgBox(strMsg, vbQuestion + vbYesNo, APPNAME)
   If strAns = vbNo Then Exit Sub


On Error Resume Next
ActiveWorkbook.Names.Item("Analysis").Delete 'delete the range
if it already exists

'create the range
    ActiveWorkbook.Names.Add Name:="Analysis",
RefersTo:=Selection
'Add binsetup sheet from BIOTOOLS Addin
Application.DisplayAlerts = False
ChromaBlastV3.Sheet1.Copy _
    After:=ActiveWorkbook.Sheets(ActiveWorkbook.Sheets.Count)
    Sheets("binsetup").Select
'    ActiveWindow.SelectedSheets.Delete
Application.DisplayAlerts = True

'set Workbook Colours with pretty display
'White to Black
intColourIndex = 1
For intColourIndex = 1 To 56
ActiveWorkbook.Colors(intColourIndex) = RGB(0, 0, 0)
 If intColourIndex < 56 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(77, 77, 77)
 End If
 If intColourIndex < 55 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(119, 119, 119)
```

```
 End If
 If intColourIndex < 54 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(150, 150, 150)
 End If
 If intColourIndex < 53 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(192, 192, 192)
 End If
 If intColourIndex < 52 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(221, 221, 221)
 End If
  If intColourIndex < 51 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 255, 255)
 End If
Next intColourIndex
'Violet
intColourIndex = 43
For intColourIndex = 43 To 49
ActiveWorkbook.Colors(intColourIndex) = RGB(93, 0, 126)
 If intColourIndex < 49 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(115, 0, 156)
 End If
 If intColourIndex < 48 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(164, 0, 202)
 End If
 If intColourIndex < 47 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(204, 0, 255)
 End If
 If intColourIndex < 46 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(210, 121, 255)
 End If
 If intColourIndex < 45 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(215, 175, 255)
 End If
  If intColourIndex < 44 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(232, 209, 255)
 End If
Next intColourIndex
'Indigo
intColourIndex = 35
For intColourIndex = 35 To 42
ActiveWorkbook.Colors(intColourIndex) = RGB(0, 46, 138)
 If intColourIndex < 42 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(0, 0, 182)
 End If
 If intColourIndex < 41 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(0, 0, 224)
 End If
 If intColourIndex < 40 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(73, 73, 255)
 End If
 If intColourIndex < 39 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(129, 129, 255)
 End If
 If intColourIndex < 38 Then
```

```
  ActiveWorkbook.Colors(intColourIndex) = RGB(171, 171, 255)
 End If
  If intColourIndex < 37 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(217, 217, 255)
 End If
Next intColourIndex
'Blue
intColourIndex = 28
For intColourIndex = 28 To 35
ActiveWorkbook.Colors(intColourIndex) = RGB(0, 102, 102)
 If intColourIndex < 35 Then

 ActiveWorkbook.Colors(intColourIndex) = RGB(0, 128, 128)
 End If
 If intColourIndex < 34 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(0, 153, 153)
 End If
 If intColourIndex < 33 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(0, 214, 209)
 End If
 If intColourIndex < 32 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(3, 255, 255)
 End If
 If intColourIndex < 31 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(129, 255, 255)
 End If
  If intColourIndex < 30 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(197, 255, 255)
 End If
Next intColourIndex
'Green
intColourIndex = 21
For intColourIndex = 21 To 28
ActiveWorkbook.Colors(intColourIndex) = RGB(0, 110, 0)
 If intColourIndex < 28 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(0, 153, 0)
 End If
 If intColourIndex < 27 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(0, 220, 0)
 End If
 If intColourIndex < 26 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(0, 254, 0)
 End If
 If intColourIndex < 25 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(141, 255, 141)
 End If
 If intColourIndex < 24 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(187, 255, 187)
 End If
  If intColourIndex < 23 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(225, 225, 225)
 End If
Next intColourIndex
'Yellow
```

```
intColourIndex = 14
For intColourIndex = 14 To 21
ActiveWorkbook.Colors(intColourIndex) = RGB(110, 107, 0)
 If intColourIndex < 21 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(182, 178, 0)
 End If
 If intColourIndex < 20 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(214, 209, 0)
 End If
 If intColourIndex < 19 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(238, 232, 0)
 End If
 If intColourIndex < 18 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 255, 7)
 End If
 If intColourIndex < 17 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 255, 167)
 End If
  If intColourIndex < 16 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 255, 205)
 End If
Next intColourIndex
'Orange
intColourIndex = 7
For intColourIndex = 7 To 14
ActiveWorkbook.Colors(intColourIndex) = RGB(210, 85, 0)
 If intColourIndex < 14 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(255, 102, 0)
 End If
 If intColourIndex < 13 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 137, 19)
 End If
 If intColourIndex < 12 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 162, 69)
 End If
 If intColourIndex < 11 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 183, 111)
 End If
 If intColourIndex < 10 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 206, 157)
 End If
  If intColourIndex < 9 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 238, 221)
 End If
Next intColourIndex
'Red
intColourIndex = 1
For intColourIndex = 1 To 7
ActiveWorkbook.Colors(intColourIndex) = RGB(118, 0, 0)
 If intColourIndex < 7 Then
 ActiveWorkbook.Colors(intColourIndex) = RGB(168, 0, 0)
 End If
 If intColourIndex < 6 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(224, 0, 0)
```

```
 End If
 If intColourIndex < 5 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 7, 7)
 End If
 If intColourIndex < 4 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 107, 121)
 End If
 If intColourIndex < 3 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 159, 161)
 End If
  If intColourIndex < 2 Then
  ActiveWorkbook.Colors(intColourIndex) = RGB(255, 213, 221)
 End If
Next intColourIndex
End Sub

Sub PreviewAssignedColours()
Dim intCellCount As Integer, intCounter As Integer
Dim ArrayColourBins As Variant

'count cells in Bin range to determine the number of bins
Range("bins").Select
intCellCount = 0
For Each xCell In Selection
 If xCell.Value > 0 Then intCellCount = intCellCount + 1
Next xCell

'create array of Labels
On Error Resume Next 'incase no bins were selected

ReDim ArrayColourBins(intCellCount, 2)
intCounter = 1
For Each xCell In Selection
 If xCell.Value > 0 Then 'if the cell has a value
  ArrayColourBins(intCounter, 1) = xCell.Offset(-1, 0).Value
'put the label into the array
  ArrayColourBins(intCounter, 2) = xCell.Value 'put the colour
value into the array
  xCell.Interior.ColorIndex = ArrayColourBins(intCounter, 2)
'colour the cell
  intCounter = intCounter + 1 'increment the counter
 Else
 xCell.ClearFormats
 End If
Next xCell

'dump the array values into cells to prove they were picked up
 '   Range("a20").Select
 '   Range(ActiveCell.Offset(0, 0),
ActiveCell.Offset(intCellCount - 1, 1)).Value = ArrayColourBins


End Sub
```

```
Sub CCBins()
Dim intRowCount As Integer, intColCount As Integer, intBinCount
As Integer
Dim BinArray 'hold bin values
Dim SequenceArray As Variant ' hold sequence values
Dim intCounter2 As Integer ' counter to move from column to
column
Dim intCounter As Integer
Dim ArrayColourBins As Variant
Dim varValue As Variant
Dim varCheckBin As Variant 'compares array values



'count cells in Bin range to determine the number of bins
Range("bins").Select
intBinCount = 0
For Each xCell In Selection
 If xCell.Value > 0 Then intBinCount = intBinCount + 1
Next xCell

'create array of Labels
On Error Resume Next 'in case no bins were selected

ReDim ArrayColourBins(intBinCount, 2)
intCounter = 1
For Each xCell In Selection
 If xCell.Value > 0 Then 'if the cell has a value
  ArrayColourBins(intCounter, 1) = xCell.Offset(-1, 0).Value
'put the label into the array
  ArrayColourBins(intCounter, 2) = xCell.Value 'put the colour
value into the array
  xCell.Interior.ColorIndex = ArrayColourBins(intCounter, 2)
'colour the cell
  intCounter = intCounter + 1 'increment the counter
 Else
 xCell.ClearFormats
 End If
Next xCell

'dump the array values into cells to prove they were picked up
'test to this point to make sure it works
 '   Range("a4").Select
  '  Range(ActiveCell.Offset(0, 0),
ActiveCell.Offset(intBinCount - 1, 1)).Value = ArrayColourBins



On Error Resume Next
ActiveWorkbook.Names.Item("myRange").Delete
ActiveWorkbook.Names.Item("CurrentRange").Delete
```

```
'go to Analysis range
Application.Goto reference:="Analysis"
'count range dimensions
    intRowCount = Selection.Rows.Count - 1 '-1 because of
offset
    intColCount = Selection.Columns.Count
'set counter2 for use with column looping
    intCounter2 = 1
'Redim create sequencearray to size of selection
    ReDim SequenceArray(intRowCount + 1, intColCount + 1)
'go to first column in selection, insert column
    Range(ActiveCell.Offset(0, 1),
ActiveCell.Offset(intRowCount, 1)).Select
    ActiveCell.EntireColumn.Insert
    Selection.Columns.ColumnWidth = 4
'loop through columns
For intCounter2 = 1 To intColCount
    'select the range
    Range(ActiveCell.Offset(0, -1),
ActiveCell.Offset(intRowCount, -1)).Select
'create the range
    ActiveWorkbook.Names.Add Name:="myRange",
RefersTo:=Selection
' find the min and max value of myRange
    myMin = WorksheetFunction.Min(Range("myRange"))
    myMax = WorksheetFunction.Max(Range("myRange"))
' create the bin values for myRange
    ReDim BinArray(intBinCount, 1)
    intCounter = 1
    For intCounter = 1 To intBinCount
     BinArray(intCounter, 1) = (((myMax - myMin) / intBinCount)
* intCounter) + myMin
    Next intCounter

'select the column beside active selection and call it Current
Range
    Range(ActiveCell.Offset(0, 1),
ActiveCell.Offset(intRowCount, 1)).Select
    ActiveWorkbook.Names.Add Name:="CurrentRange",
RefersTo:=Selection
    Set rng = Range("CurrentRange")
'fill each cell with bin formula
    For Each xCell In Selection
     varValue = xCell.Offset(0, -1).Value 'check the value of
the cell to the left
         intCounter = 1 'set the counter
     varCheckBin = BinArray(intCounter, 1) 'set the initial
value before the loop begins
     Do
     'sets the cell value to first label, then loops through
checking the varValue against
     'varCheckBin array until the value is less than or equal.
        xCell.Value = ArrayColourBins(intCounter, 1)
        If intCounter > intBinCount Then Exit Do
```

```
        varCheckBin = BinArray(intCounter, 1)
        intCounter = intCounter + 1
     Loop Until varValue <= varCheckBin
    Next xCell


'place values in Current Range into the SequenceArray looping
through each cell
For intCounter3 = 1 To intRowCount + 1
    SequenceArray(intCounter3, intCounter2) =
rng.Cells(intCounter3, 1).Value
Next intCounter3

'repeat for next range

    Range(ActiveCell.Offset(0, 2),
ActiveCell.Offset(intRowCount, 2)).Select
    ActiveCell.EntireColumn.Insert
    Selection.Columns.ColumnWidth = 4


Next intCounter2
'move over and fill with Array info
    Range(ActiveCell.Offset(0, 1)).Select
    Range(ActiveCell.Offset(0, 0),
ActiveCell.Offset(intRowCount, intColCount)).Value =
SequenceArray
    Range(ActiveCell.Offset(0, 0),
ActiveCell.Offset(intRowCount, intColCount - 1)).Select
'set column widths
    ActiveWorkbook.Names.Add Name:="CurrentRange",
RefersTo:=Selection
    Range("CurrentRange").Columns.ColumnWidth = 4
'colour the cells
    For Each xCell In Selection
     varValue = xCell.Value
         intCounter = 1 'set the counter
     varCheckBin = ArrayColourBins(intCounter, 1) 'set the
initial value before the loop begins
     xCell.Interior.ColorIndex = ArrayColourBins(intCounter, 2)
     Do
     'sets the cell value to first label, then loops through
checking the varValue against
     'varCheckBin array until the value is equal.
        If varValue = varCheckBin Then Exit Do
        varCheckBin = ArrayColourBins(intCounter, 1)
        xCell.Interior.ColorIndex = ArrayColourBins(intCounter,
2)
        intCounter = intCounter + 1
     Loop Until varValue = varCheckBin
    Next xCell
 'clear the contents of the cells
    Selection.ClearContents
'place SequenceArray beside Coloured Bins (as per Y's request)
```

```
'move over and fill with Array info
    'Range(ActiveCell.Offset(0, (2 * intColCount) + 1)),
ActiveCell.Offset(intRowCount, -1)).Select
    Range(ActiveCell.Offset(0, intColCount + 1),
ActiveCell.Offset(intRowCount, (2 * intColCount))).Select
    Range(ActiveCell.Offset(0, 0),
ActiveCell.Offset(intRowCount, intColCount)).Value =
SequenceArray
    Range(ActiveCell.Offset(0, 0),
ActiveCell.Offset(intRowCount, intColCount - 1)).Select
'set column widths
    ActiveWorkbook.Names.Add Name:="CurrentRange",
RefersTo:=Selection
    Range("CurrentRange").Columns.ColumnWidth = 4
End Sub
Sub CleanWorkbook()

'clean up workbook
On Error Resume Next
Application.DisplayAlerts = False
ActiveWorkbook.Names.Item("myRange").Delete
ActiveWorkbook.Names.Item("CurrentRange").Delete
ActiveWorkbook.Names.Item("Analysis").Delete
Sheets("binsetup").Delete
Application.DisplayAlerts = True


End Sub
```

List of symbols/abbreviations/acronyms/initialisms

AFLP	amplified fragment length polymorphism
DNA	deoxyribonucleic acid
PCR	polymerase chain reaction
RFLP	restriction fragment length polymorphism
RNA	ribonucleic acid

Glossary

fingerprint	a collection of signal intensity scores, digitized from an image of a hybridization of genomic DNA to a microarray spotted with DNA fragments. The fingerprint of a given species and strain is unique from that of other species or strains.
gene	a DNA sequence which encodes a single genetic trait which is inherited by offspring
genomic DNA	the DNA which comprises the genetic material of an cell, and is inherited by the progeny of the cell. The so-called blueprint of life. The sequence of nucleotides in the genomic DNA comprises the genes, and determines the properties of the microbe. For many microbes, the sequence of the genomic DNA is in the public domain.
hybridization	sample DNA (or RNA) is tagged with a fluorescent dye, then applied to the surface of the microarray. Under controlled conditions, sequences in the sample DNA which correspond to sequences in the microarray features, will bind to the features (hybridize). Hybridization often refers to the entire process from labeling to binding, to post incubation washing.
microarray	a microscope slide, filter membrane or other solid surface, onto which DNA fragments have been spotted in an organized grid. Each spot is called a feature.
nucleotide	the components of DNA are the nucleotides deoxyadenosine monophosphate, deoxycytidine monophosphate, deoxyguanosine monophosphate, deoxythymidine monophosphate, and the chemical bonds which join them into long chains. Genetic information is encoded in the order in which the nucleotides occur in the DNA chain.
oligonucleotide (oligo)	a fragment of DNA (or RNA) chemically synthesized, and often representing some section of genetic material from which the sequence is already known. Oligos may also be "random" in sequence, such that the oligo sequence is not intentionally derived from known DNA sequences
species	the grouping of microbes according to major genetic differences (e.g. the ability to grow (or not) in an oxygen-free environment)
strain	a microbe which differs from other members of the same species by minor or additional genetic characters (e.g. resistance or sensitivity to penicillin).

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for who the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in Section 8.)

Defence R&D Canada – Suffield
PO Box 4000, Station Main
Medicine Hat, AB T1A 8K6

2. SECURITY CLASSIFICATION
(overall security classification of the document, including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title).

ChromaBlast – a Data Visualization Tool (U)

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

Ford, Barry N., Shei, Y., Bjarnason, S., Richardson, C.

5. DATE OF PUBLICATION (month and year of publication of document)

June 2006

6a. NO. OF PAGES (total containing information, include Annexes, Appendices, etc) 30

6b. NO. OF REFS (total cited in document) 6

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Technical Memorandum

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

DRDC Suffield/Chemical and Biological Defence Section

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)

9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

DRDC Suffield TM 2006-049

10b. OTHER DOCUMENT NOs. (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(x) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally corresponded to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected).

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C) or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

Experimental systems in biology now have the capability to produce massive amounts of numerical data. Large scale analysis of such data is facilitated by costly integrated software packages. Often however, such packages have limited or no data reduction or manipulation tools, such as basic spreadsheet functionality. Thus the researcher is compelled to utilize a competent spreadsheet package, such as Microsoft Excel, then export the data set to analysis software. Reformatting data or reducing it for analysis must then be redone in the spreadsheet after each analysis run, until the final dataset is appropriately formatted for the analysis package. In order to facilitate the use of the capabilities of the spreadsheet software and perform data reduction or basic analysis without having to switch back and forth between software units, we are building a set of analysis tools in Excel. One of the most useful of the tool set is ChromaBlast, which normalizes columnar data, sorts the data into user-selectable range-driven bins, develops a colour heat map from the data, and outputs the heat map and bin assortment for review. Using intrinsic tools in the spreadsheet software, output data can be filtered and sorted to emphasize data patterns, and facilitate rapid data review.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifies, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

microarray
genomic fingerprint
data analysis
Excel