ABASOC

AUTOMATED BIOINFORMATICS ANALYSIS SYSTEM ON A CHIP

Automated Bioinformatics Analysis System on Chip

ABASOC

version 1.1

Phillip Winston Miller, Priyam Patel, Daniel L. Johnson, PhD.

University of Tennessee Health Science Center
Office of Research Molecular Bioinformatics Core
71 S Manassas St. Room 110
Memphis, TN 38103
Office:(901) 448-3743 Email: djohn166@uthsc.edu

Contents

1.1 Introduction

Molecular tools are becoming more and more common practice in the field of health science and personal medicine. Petabytes of data are generated from RNA-Seq, microarrays, whole genome sequencing, whole exome sequencing, proteomics, and metabolomics. A "best approach" to the analysis workflow has typically been created for each individual sequencing platform. Publicly available workflows for NGS analysis, such as GTK and GALAXY, offer the bioinformatics tools; however, most investigators and physicians are not well-versed enough in data analysis to make decisions regarding the best methods for analyzing their data. There is also typically a lack of support in determining whether an error occurred due to the tool crashing, a formatting problem, or an issue inherent in the data when using these public platforms. Moreover, the cost of the systems and supporting a core administrator can constrain budgets of smaller institutions. Commercial tools are much better in terms of direct customer support; however, these tend to be extremely expensive and are often proprietary.

We introduced an automated workflow for RNA-Seq, microarray, and proteomics analysis using a package we developed in-house using R and Python. All three workflows provide visualization by generating heatmaps, Principal Component Analysis (PCA) plots, and Pearson's correlations coefficient plots. These outputs are all performed on a Raspberry Pi system that is very portable and cost effective to purchase and maintain.

For RNA-Seq analysis, the package accepts SAM files as input, performs counts per million (CPM) normalization across the experiment, calculates standard deviation, standard error of means and variance within each sample group. The package then calculates Welch's *t* test between groups, and provides false discovery rates using the Benjamini-Hochberg procedure.

The microarray analysis accepts a tab-delimited text file. This file contains the gene list as well as columns containing all of the raw expression values. The package uses a log transformation and scaling normalization to remove noise, calculates standard deviation, standard error of means and variance within each sample group. The package then calculates Welch's *t* test between groups, and provides false discovery rates using the Benjamini-Hochberg procedure.

The proteomics analysis also accepts a tab-delimited text file. This file contains the protein accession ids and the protein abundance values for each sample in columns. The package uses a cyclic loess normalization to remove any sample variance introduced by uneven abundance expression between intergroup samples, calculates standard deviation, standard error of means and variance within each sample group, and then calculates Welch's *t* test between groups, provides false discovery rates using the Benjamini-Hochberg procedure.

All three workflows provide visualization by generating heatmaps PCA plots, and Pearson's correlations coefficient plots. These outputs are all performed on a Raspberry Pi system that is very portable and is cost effective to purchase and to maintain.

This platform was developed to allow wet bench researchers and physicians to quickly extract primary results from their experiments, particularly when extensive bioinformatics support is not available locally. Unlike other platforms ABASOC is completely species independent.

1.2 What is ABABSOC?

Automated Bioinformatics Analysis System on Chip (ABASOC) is a collection of analysis and biostatistical workflows design to accept the basic information available for microarray, RNA-Seq or quantitative proteomics experiments. The package then applies standard analysis thresholds and methods to determine the differential expression between a control and experiment group of samples. This package provides basic publication quality images for visualization of molecular data.

1.3 What can it do for you?

This software package was designed for wet bench customers with little or no computational experience or resources to perform basic analysis of their molecular data. The interface was designed with the idea of minimizing the choices a biologist would need to make in order to obtain basic analysis results without the need of a bioinformatics expert.

1.4 Installation

If you purchased the Raspberry Pi 3 system from the UTHSC Molecular Bioinformatics Core, the unit shipped with everything loaded that you will need to get started. Please install a monitor, keyboard, and mouse to the raspberry pi as depicted in the manual. Please contact Dr. Daniel L. Johnson (djohn166@uthsc.edu) at the UTHSC Molecular Bioinformatics Core if you need assistance during setup. Once the system is prepared, unzip the analysis software to the directory you will use for file processing. To begin the software click the “run.sh” file included in the software directory.

2. RNA-Seq Analysis

2.1 Scope and Purpose

The RNA-Seq analysis module will convert SAM files generated by your alignment algorithm to normalized read counts and perform differential expression and visualization on your data. The module includes collecting the counts from SAM files, normalization, biostatistics, false discovery correction, and visualization.

2.2 Input Files

The input needed is the SAM format file for each sample to be analyzed, the list of samples, the list of conditions, and which visualizations that you require.

2.3 Output Files

The output of the module is the biostatistics for every sample in a tab-delimited text format, a differential expression list generated after the FDR adjustments in tab-delimited text format, the Pearson’s correlation coefficient plot, differential heatmap, and principle component analysis plot in PDF format if selected. The tab-delimited file format is compatible with Microsoft Excel, MATLAB, or R.

2.4 Normalization

The module provides counts per million (CPM) normalization in order to adjust the data to account for unequal sequencing depth. CPM normalization is a scalar method in which each individual gene count is divided by the total count for the current sample. This number is then divided by 1,000,000 to obtain a count average per million.

2.5 Biostatistics

The module calculates mean, median, variance, and standard deviation of each gene for each conditional group. The mean of each group is then used to calculate the fold change between the two conditions. The method then uses DeSeq2 to calculate the significance in change between the two conditions. The cutoffs of a p value ≤ 0.05 and a fold change ≥ 1.5 are used to gather the list of differentially expressed genes.

2.6 False Discovery Rate

The module then applies a Benjamini-Hochberg false discovery calculation to calculate the adjusted p values.

2.7 Determining Differential Genes

A first-pass filter based on cutoffs of a p value ≤ 0.05 and a fold change ≥ 1.5 are used to gather the list of potential differentially expressed genes. A second-pass filter based on a cutoff of an adjusted p value ≤ 0.05 is used to create the final differentially expressed gene list.

2.8 Visualization

The module provides three publication ready images, all of which are selected for at the beginning of the process, and then automatically generated.

1. The Pearson correlation plot shows how well complete sample expression profiles correlate with one another. Ideally, the researcher would see biological replicates correlating strongly with one another and weakly with those biological replicates from a different sample. However, this is not always the case.

2. The PCA Plot allows the researcher to determine and visualize genetic distance between biological replicates and across conditions.

3. The heatmap plots samples as columns and genes as rows. Allowing the researcher to visualize fold change across replicates and conditions.

2.9 How to Analyze Data

- Load your data into the same directory as the software.
- Click run.sh to activate the software.
- Click on RNA-Seq
- List your SAM file in the box labeled “SAM File Names”.
- Make sure SAM files in order with the first group containing the control group and the second group containing the experiment group.
- List your control condition in the first box under “Conditions”.
- List your experiment condition in the second box under “Conditions”.
- List the number of samples in the control group in the first box under “Samples”.
- List the number of samples in the experiment group in the second box under “Samples”.
- Select the visualization you wish to generate.
- Click Begin Analysis

3. Microarray Analysis

3.1 Scope and Purpose

The mircoarray module will accept raw expression data from your microarrays once combined into a tab-delimited file and return a list of differentially expressed genes as well as graphs for the data. The module includes normalization, biostatistics, false discovery correction, and visualization.

3.2 Input Files

The input needed is a tab-delimited file. The first column is the probe id, the second is the official gene symbol. The following entries are your sample data. The first row should be the header of the file with the column labels included. The column entries should be in order with the first grouping of sample being your controls and the second being your experimental condition.

3.3 Output Files

The output of the module is the biostatistics for every sample in a tab-delimited text format, a differential expression list generated after the FDR adjustments in tab-delimited text format, the Pearson's correlation coefficient plot, differential heatmap, and PCA plot in PDF format if selected. The tab-delimited file format is compatible with Microsoft Excel, MATLAB, or R.

3.4 Normalization

The module provides limma package in R to perform within array normalization to minimize the within condition variance.

3.5 Biostatistics

The module calculates mean, median, variance, and standard deviation of each gene for each conditional group. The mean of each group is then used to calculate the fold change between the two conditions. The method then uses Welch's *t* test to calculate the significance in change between the two conditions. The cutoffs of a p value ≤ 0.05 and a fold change ≥ 1.5 are used to gather the list of differentially expressed genes.

3.6 False Discovery Rate

The module then applies a Benjamini-Hochberg false discovery calculation to calculate the adjusted p values.

3.7 Determining Differential Genes

A first-pass filter based on cutoffs of a p value ≤ 0.05 and a fold change ≥ 1.5 are used to gather the list of potential differentially expressed genes. A second-pass filter based on a cutoff of an adjusted p value ≤ 0.05 is used to create the final differentially expressed gene list.

3.8 Visualization

The module provides three publication ready images, all of which are selected for at the beginning of the process, and then automatically generated.

1. The Pearson correlation plot shows how well complete sample expression profiles correlate with one another. Ideally, the researcher would see biological replicates correlating strongly with one another and weakly with those biological replicates from a different sample. However, this is not always the case.

2. The PCA Plot allows the researcher to determine and visualize genetic distance between biological replicates and across conditions.

3. The heatmap plots samples as columns and genes as rows. Allowing the researcher to visualize fold change across replicates and conditions.

3.9 How to Analyze Data

- Load your data into the same directory as the software.
- Click run.sh to activate the software.
- Click on Microarray
- List your file in the box labeled “File Name”.
- List your control condition in the first box under “Conditions”.
- List your experiment condition in the second box under “Conditions”.
- List the number of samples in the control group in the first box under “Samples”.
- List the number of samples in the experiment group in the second box under “Samples”.
- Select the visualization you wish to generate.
- Click Begin Analysis

4. Proteomics Analysis

4.1 Scope and Purpose

The Proteomics module will accept your quantitative proteomics data in a tab-delimited format and return a list of differentially abundant proteins between two conditions as well as images. The module provides normalization, biostatistics, false discovery rate adjustments, and visualization.

4.2 Input Files

The input needed is a tab-delimited file. The first column is the protein accession id. The following entries are your sample data. The first row should be the header of the file with the column labels included. The column entries should be in order with the first grouping of sample being your controls and the second being your experimental condition.

4.3 Output Files

The output of the module is the biostatistics for every sample in a tab-delimited text format, a differential abundance list generated after the FDR adjustments in tab-delimited text format, the Pearson’s correlation coefficient plot, differential heatmap, and PCA plot in PDF format if selected. The tab-delimited file format is compatible with Microsoft Excel, MATLAB, or R.

4.4 Normalization

The package uses a cyclic loess normalization to remove any sample variance introduced by uneven abundance expression between intergroup samples.

4.5 Biostatistics

The module calculates mean, median, variance, and standard deviation of each protein for each conditional group. The mean of each group is then used to calculate the fold change between the two conditions. The method then uses Welch’s *t* test to calculate the significance in change between the two conditions.

4.6 False Discovery Rate

The module then applies a Benjamini-Hochberg false discovery calculation to calculate the adjusted p values.

4.7 Determining Differential Proteins

A first-pass filter based on cutoffs of a p value ≤ 0.05 and a fold change ≥ 1.5 are used to gather the list of potential differentially abundant proteins. A second-pass filter based on a cutoff of an adjusted p value ≤ 0.05 is used to create the final differentially abundant protein list.

4.8 Visualization

The module provides three publication ready images, all of which are selected for at the beginning of the process, and then automatically generated.

1. The Pearson correlation plot shows how well complete sample expression profiles correlate with one another. Ideally, the researcher would see biological replicates correlating strongly with one another and weakly with those biological replicates from a different sample. However, this is not always the case.
2. The PCA Plot allows the researcher to determine and visualize genetic distance between biological replicates and across conditions.
3. The heatmap plots samples as columns and genes as rows. Allowing the researcher to visualize fold change across replicates and conditions.

4.9 How to Analyze Data

- Load your data into the same directory as the software.
- Click run.sh to activate the software.
- Click on Proteomics.
- List your file in the box labeled “File Name”.
- List your control condition in the first box under “Conditions”.
- List your experiment condition in the second box under “Conditions”.
- List the number of samples in the control group in the first box under “Samples”.
- List the number of samples in the experiment group in the second box under “Samples”.
- Select the visualization you wish to generate.
- Click Begin Analysis