Bi-Directionally Imputing Missing Data in Gene Microarrays

Mortaza Jamshidian, Ph.D. * Amol Kumar, M.Sc.†

Abstract

We obtained the gene microarray data from Alizadeh and Yoshimoto to compare modified imputation techniques. 10%, 20%, 25% and 30% missing data was introduced randomly into the complete portions of the data sets and after imputing we computed a normalized Frobenius norm and the correlation between the imputed data set and the complete data set. K-nearest neighbors, principal components analysis and normal distribution based imputation were considered. We sought improvements by modifying current techniques; in particular we found that imputing a microarray *and* its transpose and taking the average of the results may yield improvements; we call this method bi-directional imputation. For methods which require a covariance matrix when there are more variables than observations we used a shrinkage estimator.

Key Words: imputation, missing data, gene microarray, genes, tissues

1. Introduction

Microarrays have become increasingly prominent and useful in modern medicine due to technological advances. One type of microarray is a DNA microarray, it is made by creating thousands of tiny wells in a glass or nylon slide (McLachlan 2004) and placing small amounts of single stranded DNA, called probes, in these wells (Liew, 2011). These types of slides can contain about 10,000 different probes per square centimeter (Glick 1994) and the selection of probes is fundamental to the design of experiment.

For analysis, two tissues are generally needed: a test tissue to be analyzed, and a healthy tissue for baseline comparison. mRNA is extracted from both tissues and reverse transcribed to cDNA which is then labeled with fluorescent dyes–for example, the healthy tissue may be labeled with green dye (Cy3) and the test tissue sample may be labeled with a red dye (Cy5). The labeled cDNA samples are mixed together and washed over the microarray probes where cDNA fragments will bind to their *complementary* probes, see figures 1 and 2.

A laser scanner can then read the color intensity at each well and determine the relative expression of cDNA strands which yields information regarding gene expression. For example, a certain probe may appear yellow which indicates that equal amounts of red and green are present indicating no difference in gene expression. On the other hand, another probe may appear green which indicates that the test tissue does not express the gene related to this cDNA.

A microarray may be visualized as an n $\times$ P array A, where n is the number of genes and P is the number of samples whose gene expression is being observed. For instance, row n may be a gene related to a BRCA1 breast cancer tissue and column p may be cDNA derived from a patient's breast cancer tissue; if data shows high red intensity at location (n,p) it may indicate that the patient produces breast cancer type 1 susceptibility protein.

Like any experiment, microarray procedures may contain portions of unusable data. This data is typically introduced during the laser scanning portion of the microarray process depicted in figure 3. Missing data may be a result of dust particles, scratches on the

*California State University, Fullerton
†California State University, Fullerton

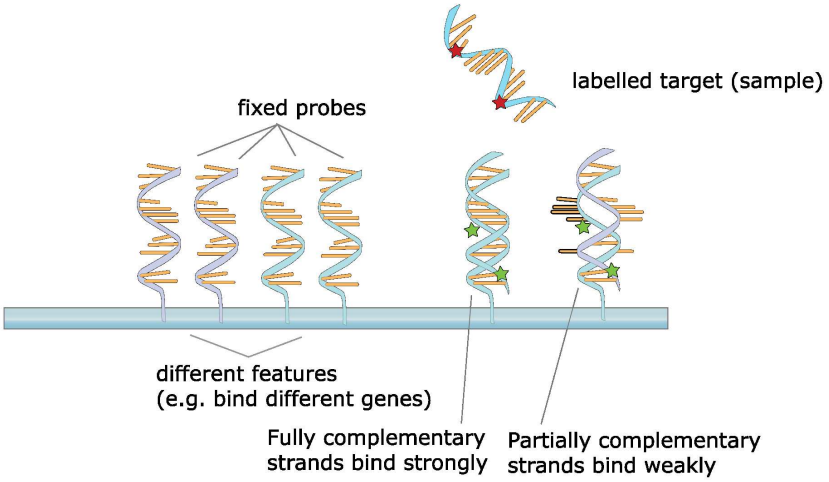

Figure 1: Hybridization of single stranded DNA on a microarray chip. (Wikipedia 2013)

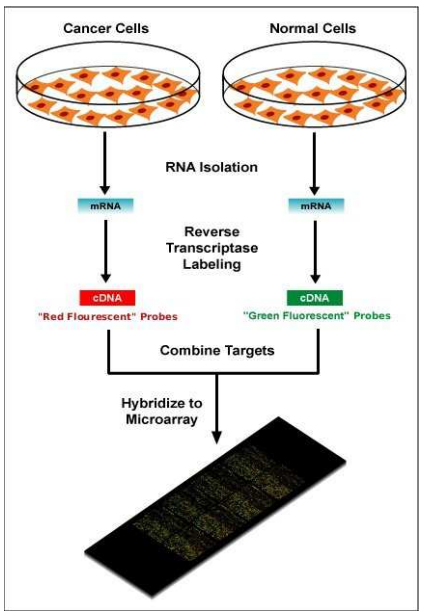

Figure 2: Schema illustrating creation of a microarray for analyzing cancer tissue. (Wikipedia 2013)

slides, machine error, poor image resolution or image corruption (Wang 2006). In particular, element (n,p) of the microarray may be missing or an outlier–the question becomes: how to impute this value? There are several approaches considered in the literature, among the most popular are K-nearest neighbors, principal components analysis (PCA) and normal distribution based imputation. The K-nearest neighbor algorithm is the most common imputation technique and estimates a missing value (n,p) by finding the K-nearest genes to gene n and averaging their p^{th} components together as an estimation for (n,p). Nearness is usually determined by the Euclidean metric, but other distances like the Manhattan distance may be used. Additionally, a weighted average may be employed to give more weight to closer neighbors and less to distant neighbors. The K-nearest neighbors approach is easy to implement but can be negatively affected by outliers, does not utilize additional informa-

tion in the data like the covariance structure and obtaining the optimal number of neighbors is not trivial.

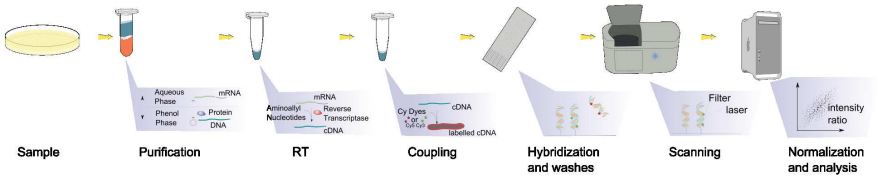

Figure 3: Scheme for obtaining a gene microarray from a tissue sample. The scanning stage is where most error is introduced, and consequently unusable data is created (Wikipedia 2013).

Principal components imputation is an iterative technique that utilizes the covariance structure of the microarray. The following procedure outlines the principal components based imputation approach:

1. Mean imputation is used to impute missing values
2. Sample covariance, S is calculated
3. Form a matrix V that consists of columns that correspond to the largest K eigenvalues of S
4. Obtain the predicted value of a row, say gene x, by $VV^T x^T$
5. If convergence criterion is not met, return to step 2 and repeat

Unlike kNN the PCA based approach utilizes the covariance structure of the data, but it can be computationally expensive because it is iterative, and determining the number of principal components is not trivial.

Another imputation technique is a distribution based approach. Most commonly, data are assumed to be distributed according to the multivariate normal distribution; the following procedure is used to impute missing values:

1. Obtain the maximum likelihood estimates for the mean and covariance matrix,
2. For $x = (x_{obs}, x_{miss})$ impute x_{miss} by a draw from the conditional distribution of x_{miss} given x_{obs} with the current mean and covariance parameter estimates,

where x_{obs} is the observed part of row (gene) x and x_{miss} is the missing part of row x. This method works well if the distribution is known and can be easily drawn from, but outliers can have adverse effects in the performance of the method.

To handle outliers Branden and Verboven proposed the following technique which they call "Robust Imputation" (Branden 2009).

1. Start with completely observed data and remove a percentage of outliers using the method of Stahel and Donoho (Stahel 1981), (Donoho 1982), (Hubert 2005)
2. Compute the mean and covariance based on the data obtained in step 1,
3. Use the multivariate normal imputation to impute a set of data with the least amount of missing data

4. Re-evaluate and remove outliers and update the mean and covariance

5. Fill in another set of incomplete cases and return to step 4.

The robust imputation technique works well for normally distributed data and is robust to outliers but like principal components analysis it is computationally expensive, and ad hoc parameters like the percentage of outliers need to be determined. Further, genes that have outlying expression values are simply thrown out of the data analysis.

Throughout the literature the variables in a microarray experiment are considered to be the tissues. This is reasonable because there are many more genes than tissues in an experiment. As such, imputation methods like K-nearest neighbors find the K nearest *genes*, not the K nearest tissues for imputation. However, it is reasonable to view genes as variables, albeit with relatively few observations, and for a method like K-nearest neighbors impute values by finding the K-nearest tissues. On the other hand, for methods like principal components analysis that require a covariance matrix, we are faced with the $N << p$ problem. Raychaudhuri et. al. write:

> "A PCA analysis of DNA microarray data can consider the genes as variables or the experiments as variables or both. When genes are variables, the analysis creates a set of principal gene components that indicate the features of genes that best explain the experimental responses they produce. When experiments are the variables, the analysis creates a set of principal experiment components that indicate the features of the experimental conditions that best explain the gene behaviors they elicit. When both experiments and genes are analyzed together, there is a combination of these affects, the utility of which remains to be explored" (Raychaudhuri 2013).

In this paper we outline an approach to utilizing both genes and tissues as variables for imputation purposes–we call this method *bidirectional imputation*. The methodology is as follows:

1. Impute the data using genes as rows and tissues as columns,

2. Impute the data using tissues as rows and genes as columns,

3. Impute missing values using a (weighted) average of (1) and (2).

We note that the implementation of bi-directional kNN is straightforward but to implement methods that require a covariance matrix we encounter the $N << p$ problem (many more variables than observations) where sample covariances are non-positive definite. To overcome the $N << p$ issue we propose using a shrinkage estimator for the covariance (Schaefer 2005.

2. Methods

We obtained the Alizadeh (2317 genes by 65 tissues) (Alizadeh 2000) and Yoshimoto (4380 genes by 24 tissues) (Yoshimoto 2002) data sets considered by Branden and Verboven (Branden 2009). These data sets were imported into the R statistical environment and incomplete rows were removed. For each dataset, missing values were introduced randomly and the values were imputed using K-nearest neighbors, bi-directional K-nearest neighbors, principle components, bi-directional principle components, the robust imputation technique considered by Branden and Verboven (Branden 2009) and the bi-directional version of

robust imputation. The outliers determined in robust imputation were removed across the whole spectrum of data techniques since the methods of Stahel and Donoho do not require the imputation technique be normal distribution based. In order for the robust imputation technique to function we kept 5 percent of the columns and 5 percent of the rows entirely complete.

The amount of missing data introduced was 10%, 20%, 25% and 30% of the total data size. The number of nearest neighbors was chosen with cross validation from the R "imputation" package. To minimize computational time the number of nearest neighbors was only determined once and the result was used for all iterations, the number of principal components was chosen to be 3 and for the robust imputation technique $\alpha = .9$ (see Branden 2009). The scheme was repeated 25 times for a given missing percentage, introducing the missing data randomly at each iteration.

Methods were compared in several ways. We utilized a modified Frobenius norm given in equation 1, computed the correlation between imputed and actual values and we plotted the imputed data points against the actual data points. If we let A represent the original data set, and I represent the imputed data set, then the computed relative Frobenius error is given by equation 1 and the averaged correlations are given by equation 2:

$$\text{Relative Frobenius Error} = \frac{\sum_{j=1}^{25} (\|A - I\|_F / \|A\|_F)_j}{25} \tag{1}$$

$$\text{Correlation Metric} = \frac{\sum_{j=1}^{25} cor(A, I)_j}{25} \tag{2}$$

Where j is the simulation number and $\|A\|_F = \sqrt{\sum_{i,j} a_{ij}^2}$. These metrics are essentially the same ones used by Branden and Verboven.

The code used to perform bi-directional kNN is presented in figure 4, and the code used to perform bi-directional PCA is presented in figure 5. In lines 6 and 7 of the code in figure 4, rows without missing data are identified and used to retain the complete portion of the dataset. In lines 8 and 9 the dimensions of the data are obtained. In lines 12 through 15 missing data is introduced at some percentage (here it is shown at 5%). In lines 18 and 19 the number of nearest genes and tissues are chosen respectively. In line 22 the data is imputed with tissues as variables, while in line 25 the data is imputed with genes as variables. Finally, in line 28 the imputation results are averaged together. We note that the methodology is simple to employ, which makes it particularly nice. If a technique requires a covariance, then when performing the imputation of the transpose we first find the shrinkage estimate of the covariance matrix and use this in the technique.

Figure 5 shows how to implement a technique that requires a covariance matrix. The code used before line 17 in 5 is the same as the code in figure 4. At line 17 the number of principal components is set for both the microarray and its transpose. Then, in line 22 mean imputation is used to impute missing values so that a covariance matrix can be calculated in line 23. At this point, principal components analysis is used to impute the microarray (line 26). Lines 29 and 30 show the use of the cov.shrink() function found in R's 'corpcor' package to get an estimate of the transposed data's covariance matrix. Next, in line 30 principal components analysis is used to impute the transposed microarray with the shrinkage estimate obtained before. Finally, the imputations are averaged in line 36.

```
#load dataset: yoshimoto data
rawData = yoshimoto;

#Remove missing data from raw_data
k = which(!is.na(rowSums(rawData)))
completeData = rawData[k,];
completeData.size = dim(completeData)
completeData.rows = completeData.size[1]; completeData.cols = completeData.size[2]

#Introduce missing data into completeData at 5%
n = completeData.rows; p = completeData.cols;
missing = matrix(runif(n*p), nrow = n) < .05;
missingData = completeData;
missingData[missing] = NA;

#set the number of neighbors to use in nearest neighbors imputation
kStandard = 10;
kTranspose = 2;

#impute dataset using bioconductor's impute.knn function
imputedDataNormal = impute.knn(missingData, kStandard)

#impute transpose of data
imputedDataTranspose = impute.knn(t(missingData), kTranspose)

#average the results of both imputations together
imputedData = (imputedDataNormal$data + t(imputedDataTranspose$data))/2
```

Figure 4: Sample bi-directional imputation with kNN.

```
#set the number of principal components to use in nearest neighbors imputation
npcs = 4;
npcs.transpose = 3;

# perform mean imputation to obtain an estimate of the missing data's covariance matrix
mean.imputed.data = mean.imputation(missingData)
cov = cov(mean.imputed.ata)

#impute dataset using bioconductor's impute.knn function
imputedDataNormal = pca.impute(missingData, k = npcs, cov = cov)

# use a shrinkage estimate on the transpose to obtain an estimate of the covariance matrix
data.transpose = t(mean.imputed.data)
shrinkage.estimate = cov.shrink(data.transpose) # shrinkage estimate

#impute transpose of data
imputedDataTranspose = pca.impute(t(missingData), k = npcs.transpose, cov = shrinkage.estimate)

#average the results of both imputations together
imputedData = (imputedDataNormal$data + t(imputedDataTranspose$data))/2
```

Figure 5: Sample bi-directional imputation with principal components analysis.

3. Results and Calculations

Imputed data from each technique was compared to the complete data using the methods outlined above. The results for the Alizadeh and Yoshimoto datasets are presented below. In figure 6 a comparison of kNN using the standard imputation, transpose imputation and bi-directional imputation are shown according to the comparisons outlined in the methods section. Similarly, figures 7 and 8 compare the PCA and robust imputation methods respectively. For Yoshimoto's data only the kNN and PCA methods were compared and these results are shown in figures 9 and 10 respectively. Finally, we summarize the comparison between all bi-directional methods on Alizadeh's data in figure 11.

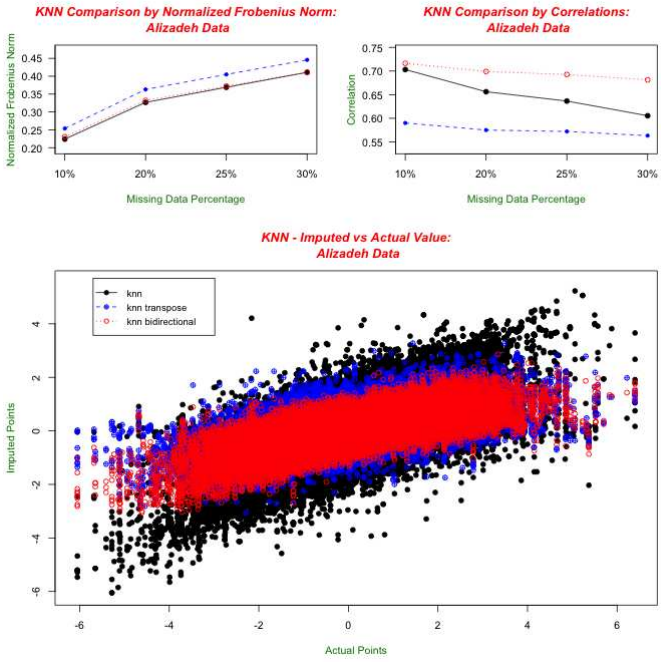

Figure 6: Results from imputing Alizadeh data with kNN.

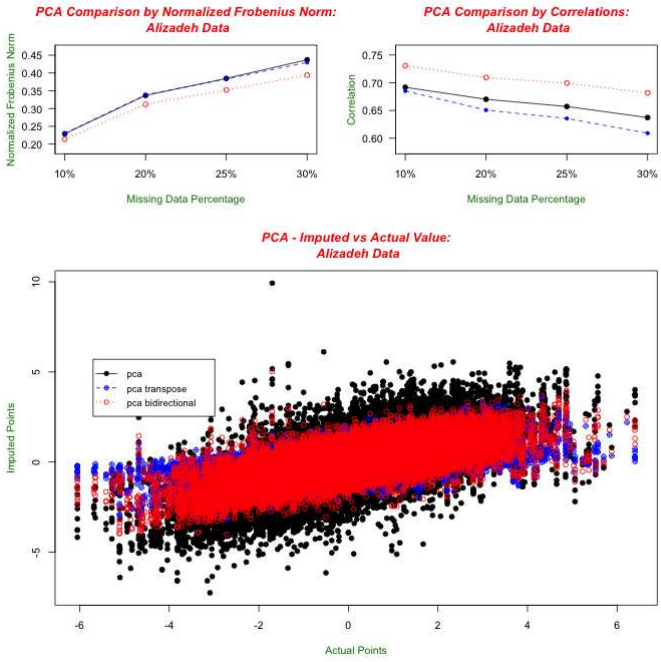

Figure 7: Results from imputing Alizadeh data with PCA.

4. Discussion and Conclusions

The results show us that bi-directional imputation may yield an improvement to the standard imputation methods. To begin, the modified Frobenius norm for the bi-directional methods is almost as small as the standard imputation, if not smaller, in all cases and tends to be smaller with more missing data. Thus, if more missing data is observed, bi-directional imputation may be preferred over standard imputation. We note that for Alizadeh's data,

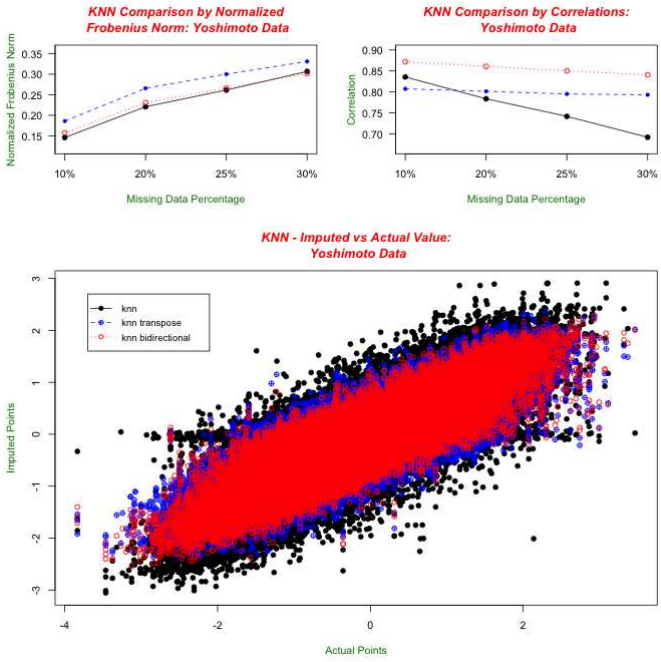

Figure 8: Results from imputing Alizadeh data with Robust Imputation.

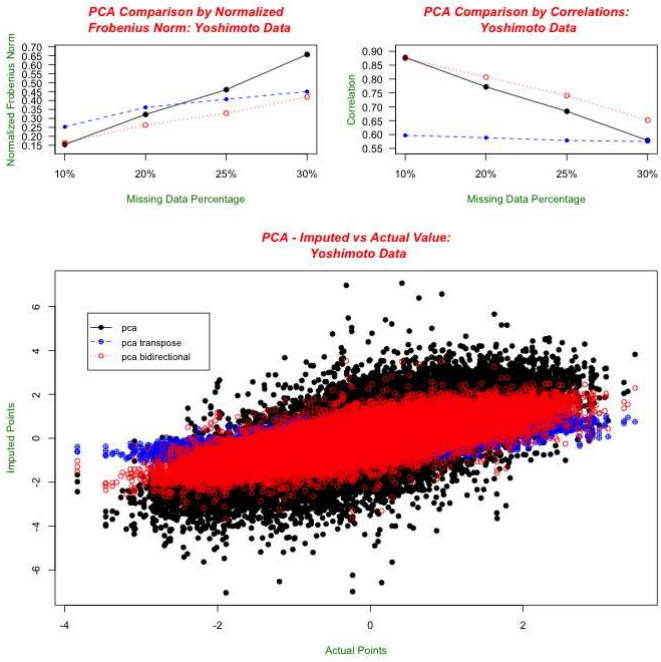

Figure 9: Results from imputing Yoshimoto data with kNN.

removing outliers and imputing bidirectionally using the multivariate normal distribution (that is, using bidirectional robust imputation) is optimal according to the Frobenius norm and correlation metrics.

Additionally, the correlation between actual and imputed data seems to be improved in all cases when imputing bi-directionally. This can be also be seen in the plots of imputed versus actual data and show that variance is reduced in all methodologies when bi-

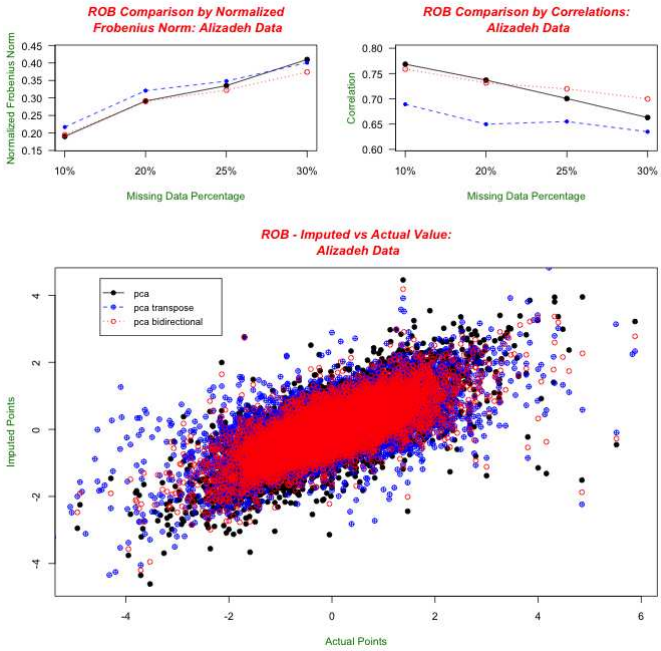

Figure 10: Results from imputing Yoshimoto data with PCA.

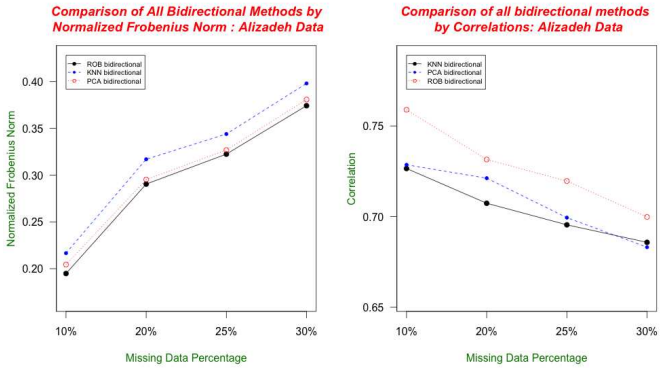

Figure 11: Comparison of all bi-directional methods on Alizadeh's data.

directionally imputing. A decrease in variance is not terribly surprising because the bi-directional technique imputes twice (once with tissues as variables and once with genes as variables), nevertheless it is interesting to see that imputing with genes as variables yields useful information–verifying the view of Raychaudhuri et. al. (Raychaudhuri 2013).

Though the variance is reduced and the results appear to be more accurate than standard imputation they are also more skewed and biased. Indeed, the plots of imputed data versus actual data all appear to be tilting horizontally. An appropriate bias correction could yield a significant improvement though it is unclear exactly how to fix the bias.

Modifications may be used in the bi-directional methodology. For example, one may compute a weighted average instead of a standard average. A possible way to select weights may be to retain only the complete portion of the data, add missing values, impute, and compute the correlation between actual values and imputed values. If we let $\text{cor}_{tissues}$ and cor_{genes} denote the correlation when imputing with tissues and genes as variables respectively then the weights for the standard imputation and the transpose imputation may be chosen as:

$$w_{tissues} = \frac{cor_{tissues}}{cor_{tissues} + cor_{genes}},$$

$$w_{genes} = \frac{cor_{genes}}{cor_{tissues} + cor_{genes}}.$$

Where $w_{tissues}$ denotes the weight for the imputation when tissues are variables, and w_{genes} denotes the weight for the imputation when genes are variables.

Future work might evaluate different choices of weights for averaging the imputation results or how to successfully correct bias. Additionally, one may investigate whether the bi-directional methodology improves downstream clustering of genes and tissues as was done for Hedenfalk's data (Hedenfalk 2001) by Branden and Verboven (Branden 2009). Finally, different methods may be implemented bi-directionally especially since the methodology is easy to code and the $p >> N$ problem can be overcome with a shrinkage estimate of the covariance matrix.

REFERENCES

Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson Jr., J., Lu, L., Lewis, D.B., Tibshirani, R., Sherlock, G., Chan, W.C., Greiner, T.C., Weisenburger, D.D., Armitage, J.O.,Warnke, R., Levy, R., Wilson,W., Grever, M.R., Byrd, J.C., Botstein, D., Brown, P.O., Staudt, L.M., 2000. "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling." *Nature* 403 (3), 503511.

Branden, K., and S. Verboven. "Robust Data Imputation." *Computational Biology and Chemistry* 33.1 (2009): 7-13. Print.

"DNA Microarray." Wikipedia. Wikimedia Foundation, 06 Mar. 2013. Web. 11 June 2013.

Donoho, D.L., 1982. Breakdown properties of multivariate location estimators. Qualifying paper. Harvard University, Boston.

Glick, Bernard R., and Jack J. Pasternak. "Molecular Biotechnology: Principles and Applications of Recombinant DNA." 4th ed. Washington, D.C.: ASM, 1994. 155. Print.

Hedenfalk, I., Duggan, D.D., Chen, Y., Radmacher, M., Bittner,M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Kallioniemi, O.-P., Wilfond, B., Borg, ., Trent, J., 2001. "Gene-expression profiles in hereditary breast cancer." *The New England Journal of Medicine* 344, 539548.

Hubert, M., Rousseeuw, P.J., Vanden Branden, K., 2005. *"ROBPCA: a new approach to robust principal components analysis"*. Technometrics 47, 6479.

Liew, A. W.-C., N.-F. Law, and H. Yan. "Missing Value Imputation for Gene Expression Data: Computational Techniques to Recover Missing Data from Available Information." *Briefings in Bioinformatics* 12.5 (2011): 498-513. Print.

McLachlan, Geoffrey J., Kim-Anh Do, and Christophe Ambroise. "Microarrays in Gene Expression Studies." *Analyzing Microarray Gene Expression Data*. Hoboken, NJ: Wiley-Interscience, 2004. 7. Print.

Raychaudhuri, Soumya, Joshua M. Stuart, and Russ B. Altman. "*Principal Components Analysis to Summarize Microarray Experiments: Application to Sporulation Time Series*." National Center for Biotechnology Information. U.S. National Library of Medicine, n.d. Web. 19 Sept. 2013.

Schafer, Juliane, and Korbinian Strimmer. "*A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics*." Statistical Applications in Genetics and Molecular Biology 4.1 (2005): Web.

Stahel,W.A., 1981.Robuste Schätzungen: Infinitesimale Optimalitästundschätzungen von Kovarianzmatrizen. Ph.D. Thesis. ETH, Zrich.

Wang, Xian, Ao Li, Zhaohui Jiang, and Huanqing Feng. "*Missing Value Estimation for DNA Microarray Gene Expressoin Data by Support Vector Regression Imputation and Orthogonal Coding Scheme*." BMC Bioinformatics 7.32 (2006)

Yoshimoto, H., Saltsman, K., Gasch, A.P., Li, H.X., Ogawa, N.D.B., Brown, P.O., Cyert, M.S., 2002. "Genome-wide analysis of gene expression regulated by the Calcineurin/Crzlp signaling pathway in Saccharomyces cerevisiae." *The Journal of Biological Chemistry* 277 (34), 3107931088.