STATISTICAL ASSESSMENT OF TIME AND MASS ALIGNMENT QUALITY IN LIQUID CHROMATOGRAPHY-MASS SPECTROMETRY

A Senior Scholars Thesis

by

ISAAC VELANDO

Submitted to the Office of Undergraduate Research
Texas A&M University
in partial fulfillment of the requirements for the designation as

UNDERGRADUATE RESEARCH SCHOLAR

April 2009

Major: Applied Mathematical Sciences

STATISTICAL ASSESSMENT OF TIME AND MASS ALIGNMENT QUALITY IN LIQUID CHROMATOGRAPHY-MASS SPECTROMETRY

A Senior Scholars Thesis

by

ISAAC VELANDO

Submitted to the Office of Undergraduate Research
Texas A&M University
in partial fulfillment of the requirements for the designation as

UNDERGRADUATE RESEARCH SCHOLAR

Approved by:

Research Advisor:	Alan Dabney
Associate Dean for Undergraduate Research:	Robert C. Webb

April 2009

Major: Applied Mathematical Sciences

ABSTRACT

Statistical Assessment of Time and Mass Alignment Quality in Liquid
Chromatography-Mass Spectrometry. (April 2009)

Isaac Velando
Department of Mathematics
Texas A&M University

Research Advisor: Dr. Alan Dabney
Department of Statistics

This research evaluated the efficacy of an alignment quality algorithm and follows its development. Proteomics research frequently involves liquid chromatography-mass spectrometry (LC-MS) methods for data collection. To correct for systematic errors, researchers often apply alignment algorithms to these data; the quality of these alignment procedures is often overlooked and needs to be assessed to offer confidence in results derived from LC-MS data. The data we worked with was aligned by a dynamic time warping (DTW) alignment algorithm. We developed an assessment algorithm based on a null hypothesis significance testing method applied to a generalized regression of particular LC-MS data. We found that the assessment algorithm alone was an insufficient indicator of the quality of alignment and could in some cases fail, but there is potential for it to be valuable as an aide with other information to make judgments on the alignment quality.

ACKNOWLEDGMENTS

This work would not be possible without my advisor Dr. Alan Dabney who introduced me to the world of academic research, proteomics, and applied statistics and who guided me during this undertaking. I want to thank the MacCoss Lab of Biological Mass Spectrometry in the Genome Sciences Department of the University of Washington for providing us with the data for testing the algorithm. I must acknowledge the work the Texas A&M Undergraduate Research program does to facilitate and promote the development of these theses. I'd also like to thank the Texas A&M Undergraduate Program in Biological and Mathematical Sciences for funding this research endeavor. Lastly I want to thank my family for their love and support and of special note I must thank my brother Alex for his thoughtful scrutiny of my decisions, which has bolstered my resolve when taking great strides toward my future.

NOMENCLATURE

LC	Liquid Chromatography
MS	Mass Spectrometry
Y_i	Estimated response variable
x_i	Predictor variable value
β_0	Regression model parameter giving y-intercept
β_1	Regression model parameter giving slope
ε_i	Error term
$N(\mu,\sigma^2)$	A normal distribution with mean μ and variance σ^2
T_{mn-2}	Student’s t distribution with mn-2 degrees of freedom
QC	Quality Control

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

LC-MS in proteomics

Proteomics is the study of the complete protein set in a population. While there are many technologies and tools available to proteomics researchers, protein MS has become the method of choice.[1] Protein MS allows for the identification and quantification of protein abundance levels in complex biological samples.[2]

A specific variety of protein MS is based on LC-MS in combination. The LC step is intended to separate peptides on the basis of something other than mass, allowing for higher resolution mass analysis.[3] Once peptides elute from the LC column, they are ionized and injected into a high-resolution MS instrument. The MS instrument then records accurate mass measurements and counts the number of ions for each unique mass feature, with the ion count roughly reflecting peptide abundance.

This thesis follows the style of *Journal of Proteome Research*.

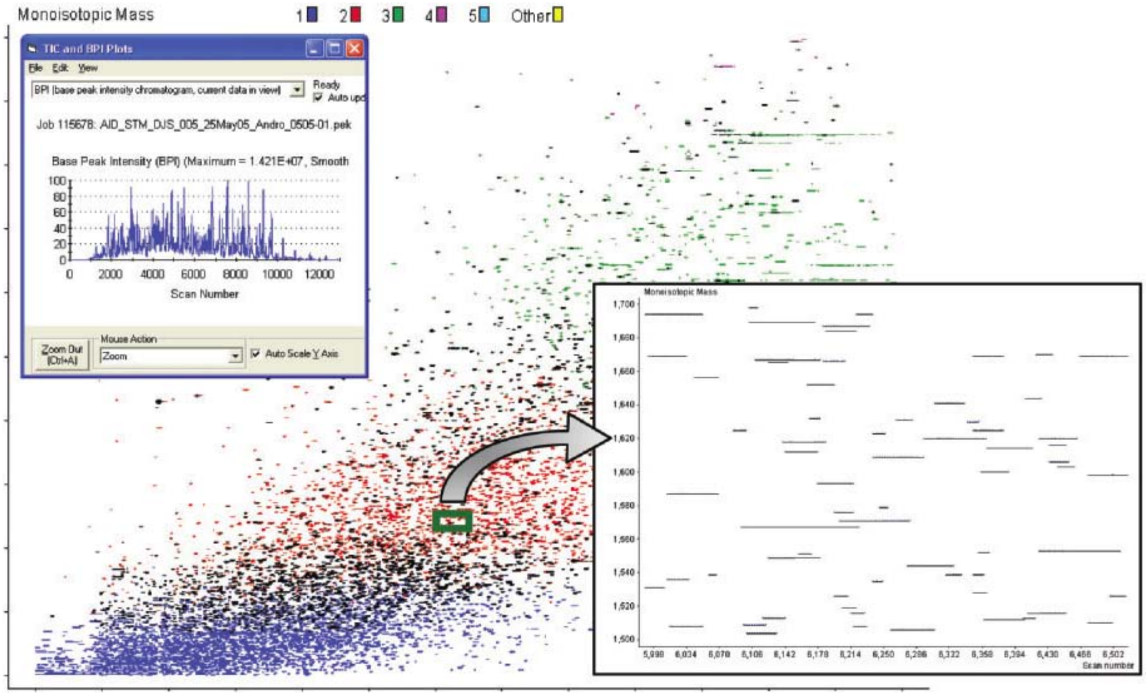

Figure 1.[4] The result of an LC-MS run. The x-axis shows the scan number for each spectrum, a function of the time required for a particular batch of peptide ions to elute from the LC column. The y-axis shows the mass of the corresponding peptide.

The result of an LC-MS run is a picture like that shown in Figure 1. Similar pictures are obtained for each of several replicate samples. A key question is then which spots or clusters of spots on different pictures correspond to the same peptide. A complication in making this determination is the fact that systematic errors inherent in LC-MS trials can result in images that are warped relative to one another. That is to say, where one peptide is located in one image, the identical peptide may be warped away from that spot in another as a result of these systematic errors. As a result, various algorithms have been proposed for aligning a set of LC-MS images and deriving a sensible set of common features.[4]

Dynamic time warping

The dynamic time warping (DTW) algorithm is an example of such an alignment algorithm and is used to align the data this research uses as a test case. Fundamentally, this algorithm computes an optimal alignment path between two time series (i.e. data sets where time is the independent variable) such that only the time axis is altered; this path is then applied to align a desired time series to another.[5] This has become a very common procedure in the alignment of LC-MS data and is often extended to align both the time axis and mass axis.[6] In the particular case of the data this research focuses on, the manner in which the DTW was applied involved aligning several replicate LC-MS samples to a selected template sample.

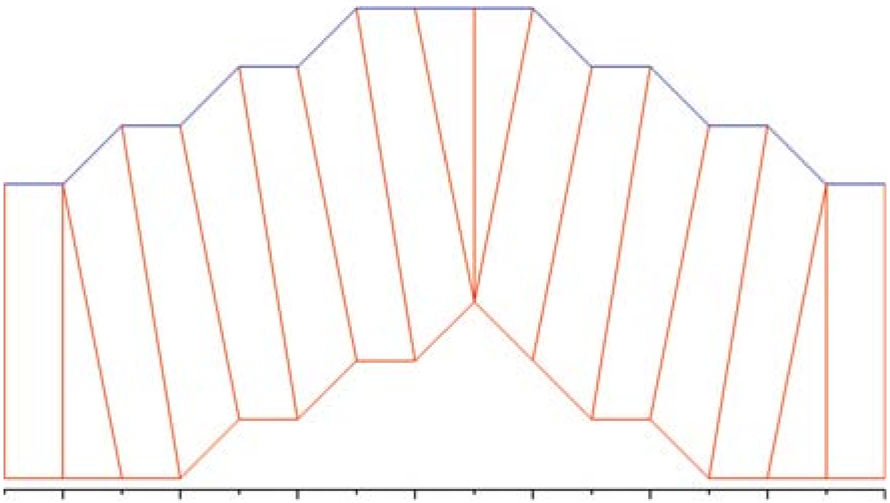

Figure 2. A view of the DTW alignment procedure where the lower (red) time series is aligned to the upper (blue) time series by shifts in the time axis.

Figure 2 offers a diagrammatic view of the DTW algorithm's procedure. The lines connecting the lower time series to the upper time series indicate how the time axis is mapped for particular values on the dependent axis. See Appendix A for a brief overview of the DTW alignment procedure.

Alignment quality

A key unsolved problem is the assessment of the quality of an alignment algorithm. Ideally, an alignment algorithm would return both the aligned images and a measure of the confidence in the alignment. Recent studies on the use of alignment algorithms in proteomics and metabolomics has shown a nascent trend towards standardized alignment procedures but a present lack of uniformity.[7] Because of this lack of uniformity in alignment procedures, an assessment algorithm should ideally be as general and robust as possible.

Regression and hypothesis testing

We utilize a statistical approach to the assessment of alignment quality. The first major component of the assessment approach involves regression analysis. We focus on a generalized linear regression involving components of the LC-MS data in order to characterize the alignment's effect.

Following this analysis we apply a null hypothesis significance test to a regression parameter of interest. The goal of this method is to offer a measure of statistical confidence in the particular results of some data; in this case we ask the question of how confident we are in the value of a particular regression parameter.

Results

The resulting algorithm first involves a generalized linear regression. For each time and mass value in an LC-MS data set there is a corresponding intensity value that can characterize the identity of a particular protein. Pairwise intensity-intensity values between an aligned data set and the template data set are compiled into a list upon which a general linear regression is performed. This model takes into account the fact that these data often exhibit non-constant variance. Once the regression is complete we perform a null hypothesis significance test on the slope parameter to determine whether the alignment resulted well or not.

We show that under limited conditions this algorithm offers a simplistic indication of the quality of an alignment, but it can fail to give the predicted results in cases of perturbed data. Possible extensions of this algorithm and the use of other potential indicators are considered for future work.

CHAPTER II

METHODS

A model for protein comparison in LC-MS data

The model-building process began with the consideration that the intensity values accompanying each time and mass pair effectively characterize the location and presence of a particular protein. We interpret a proper alignment between replicate LC-MS data sets to mean that a given protein should be found within a small time and mass region in each replicate set. Therefore, if we compare intensities between the template data set and an aligned data set in the same time and mass region, there should be a near one-to-one correspondence under a successful alignment. To explore this possibility we consider Figure 3.

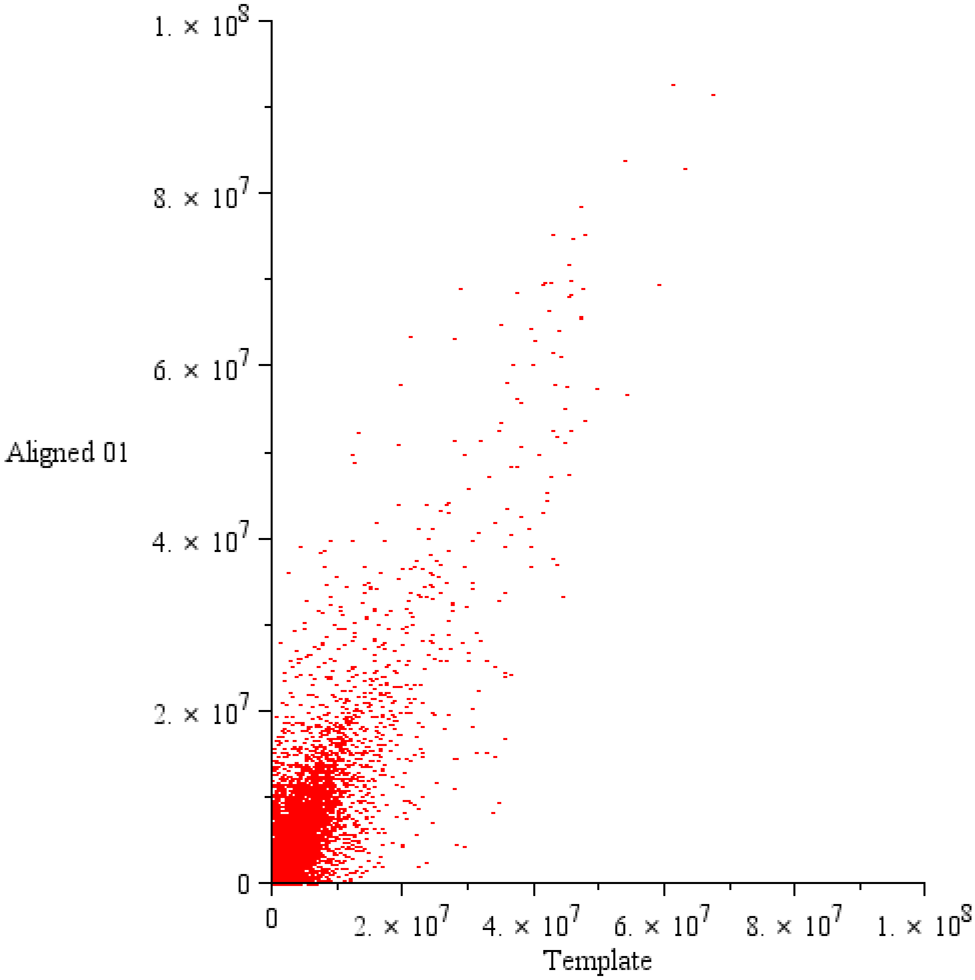

Figure 3. The intensity-intensity plot for the template vs. aligned 01 data set.

From Figure 3 it is evident that there is a linear relationship; a linear regression seems appropriate to characterize this data. A simple linear model follows:

$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ where $\varepsilon_i \sim N(0,\sigma^2)$ .... (1)

For this to hold we must assume the following:

1. The data are independent
2. There is a linear trend

3. All ε_i have constant variance
4. All ε_i are normally distributed

Assumption 1 is assumed to be true due to the nature of the LC-MS process not requiring repeated measurements of one subject. Assumption 2 is given by Figure 3. We use a Q-Q diagnostic plot to assess normality; we require a relatively linear relationship.

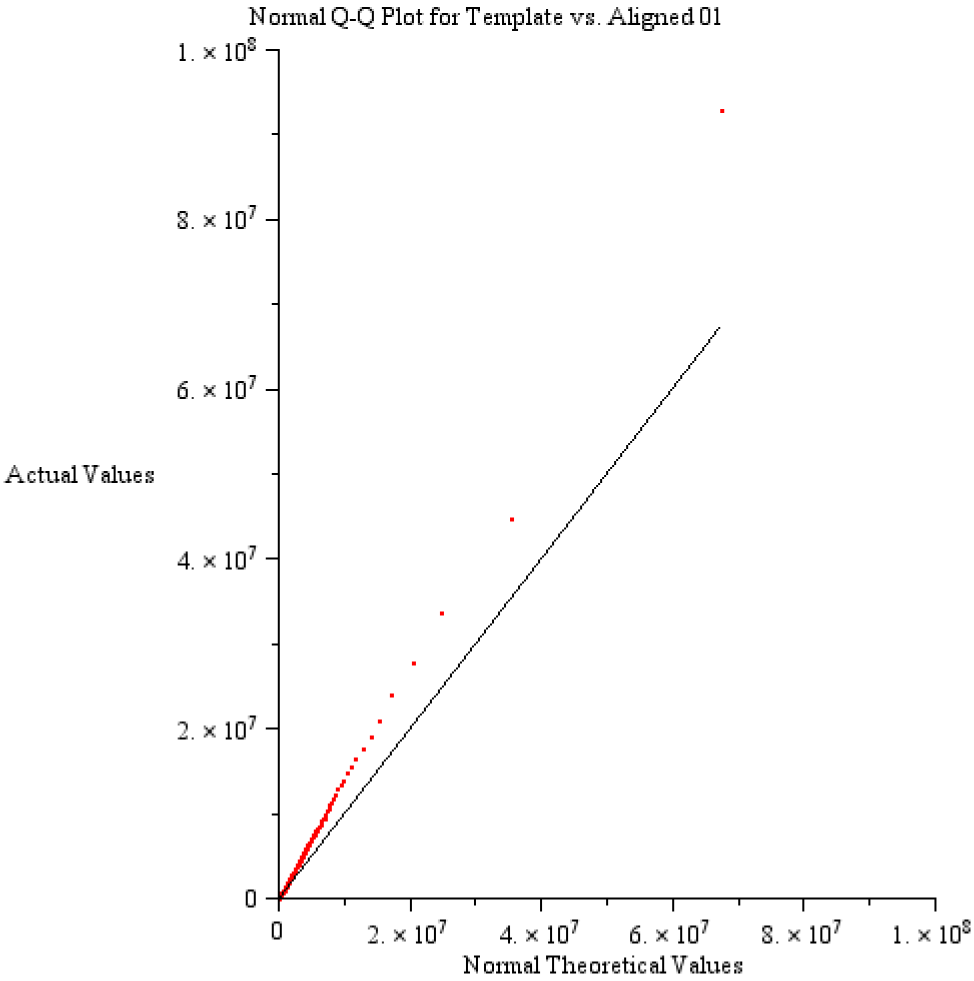

Figure 4. A normal Q-Q diagnostic plot for the template vs. aligned 01 data set.

Figure 4 shows that assumption 4 is satisfied; only assumption 3 is problematic. To address this we modify the error distribution condition to $\varepsilon_i \sim N(0,\sigma_i^2)$. This generalized linear model therefore must take into account non-constant variance. To do this we subject the data to a sampling procedure. The intensity-intensity data is ordered and then for every 100 consecutive data values we compute a sample variance which is associated with those 100 values. In preparing the intensity-intensity data, for simplicity we discard data pairs involving an intensity of 0. The primary motivation is that intensities of 0 may represent proteins that were present in too levels too to be detected or it may simply represent the absence of proteins; this could lead to confounding in the data interpretation.

Null hypothesis significance testing

The chief technique governing our approach to the alignment problem is null hypothesis significance testing. Null hypothesis significance testing seeks to make a statistical decision regarding a question posed about some data. This question is characterized by two competing hypotheses: the null hypothesis H_0 and the alternative hypothesis H_A. A way of interpreting the hypothesis test would be: assume that H_0 is true, then compute the probability of obtaining data at least as extreme as was observed.[8] This probability is known as a p-value, and that is the desired measure of alignment quality.

Here we ask the question: if the alignment is in fact perfect, what is the probability that we observed a particular alignment? Since we have established that under a perfect alignment there would be a one-to-one correspondence between corresponding intensity values, the slope parameter β_1 is exactly equal to one and the y-intercept parameter β_0 is exactly equal to zero; this leads to the following set of hypotheses:

H_0: $\beta_1 = 1$ vs. H_A: $\beta_1 \neq 1$

H_0: $\beta_0 = 0$ vs. H_A: $\beta_0 \neq 0$

With this established, we proceed to compute test statistics for the parameters β_0 and β_1. Using a significance level $\alpha = 0.05$, if the p-value falls beneath α then we would conclude that there is significant statistical evidence suggesting that the alignment was poor. Otherwise we would conclude that there is insufficient statistical evidence to suggest a poor alignment. See Appendix B for the details of the algorithm and Appendix C for an implementation in Maple software.

CHAPTER III

RESULTS

Figures 5 and 6 show the test cases with simulated random and well-aligned data respectively. Figures 7 through 18 are the six given aligned data sets' intensity-intensity plots along with the linear fit and a Q-Q diagnostic plot to assess normality.

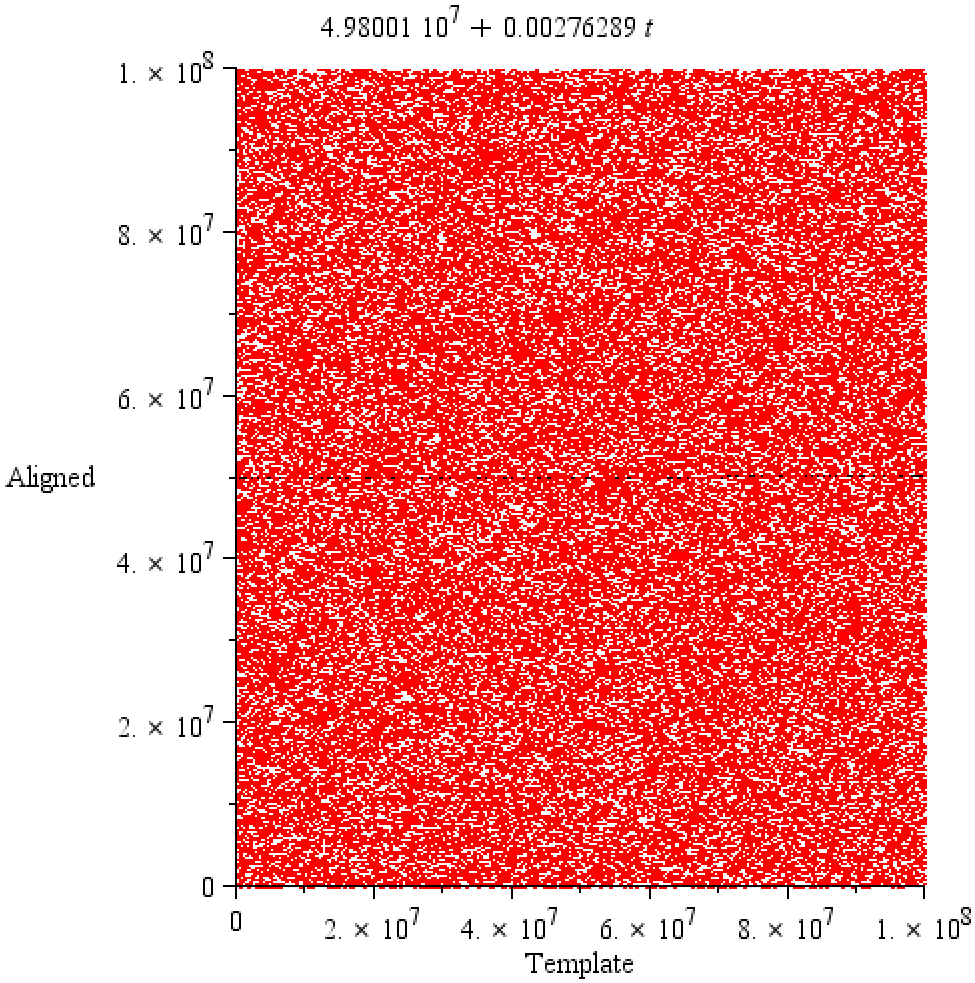

Figure 5. The intensity-intensity plot along with the linear fit for the randomly generated data simulating an extremely poor alignment of 100,000 points.

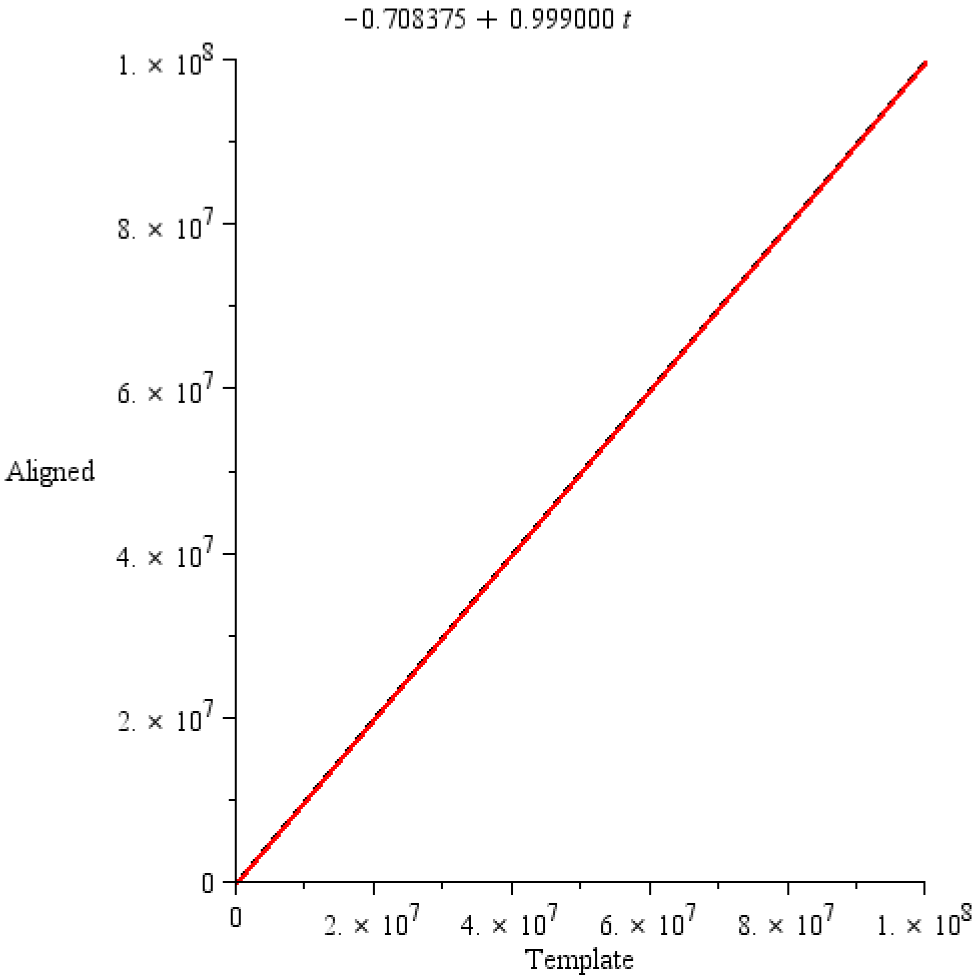

Figure 6. The intensity-intensity plot along with the linear fit for the simulated data for a near-perfect alignment of 100,000 points.

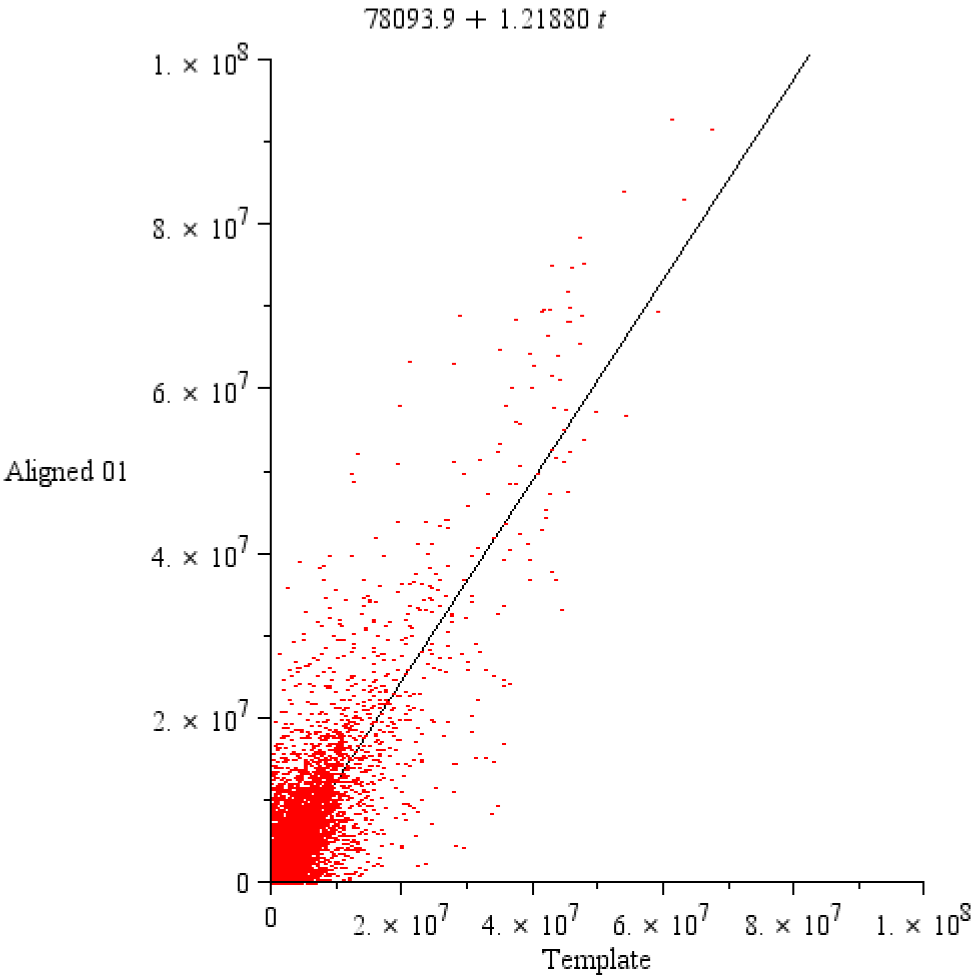

Figure 7. The intensity-intensity plot along with the linear fit for the template vs. aligned 01 data set.

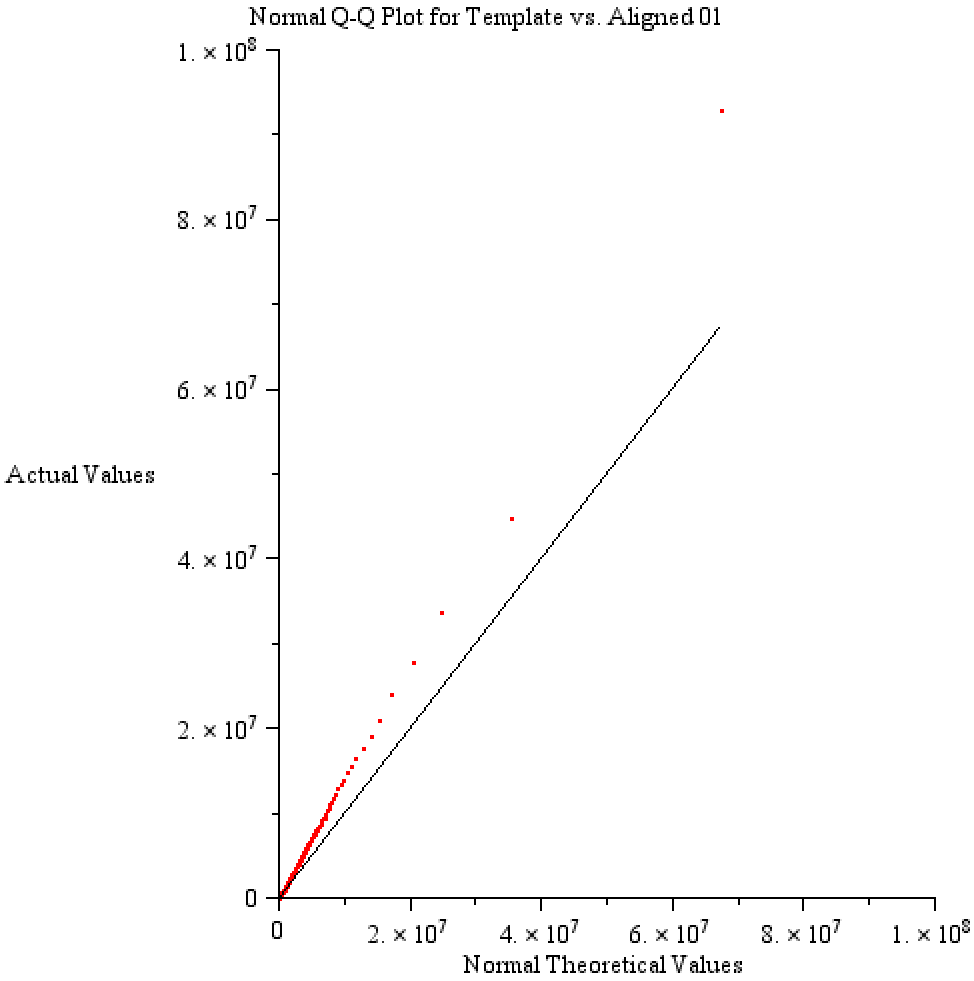

Figure 8. A Q-Q diagnostic plot for the template vs. aligned 01 data set.

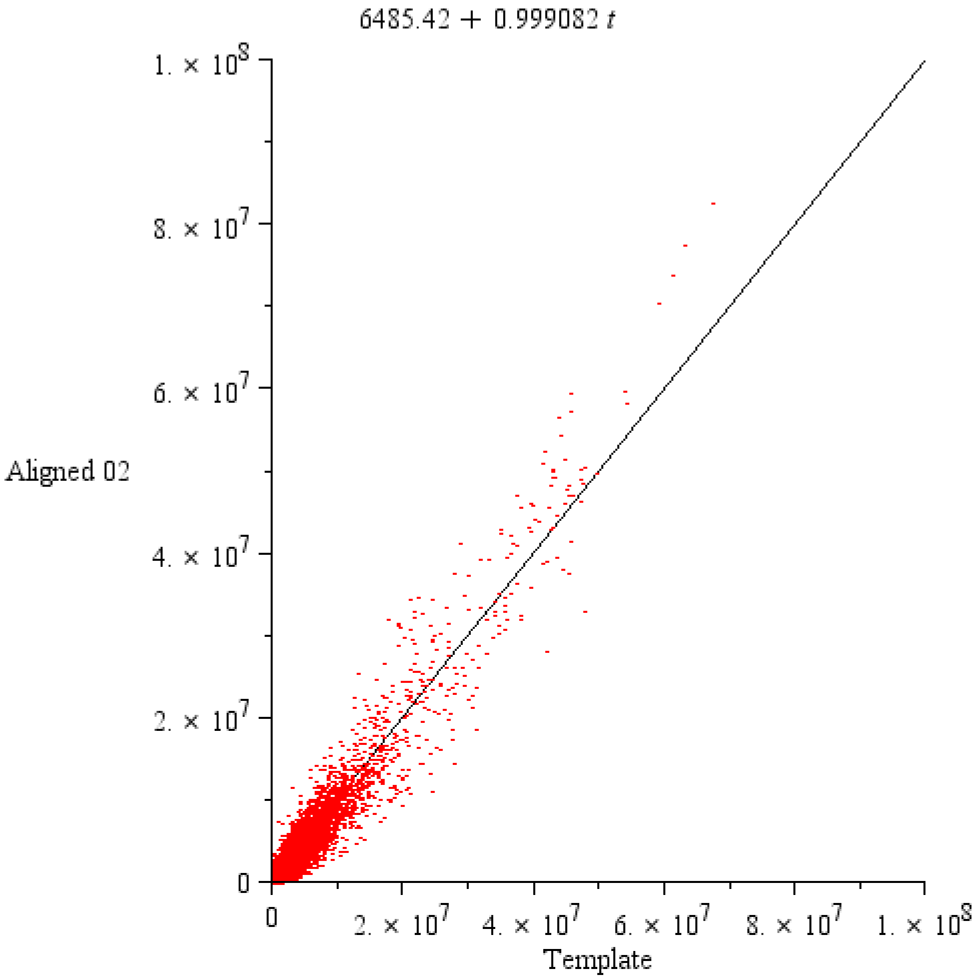

Figure 9. The intensity-intensity plot along with the linear fit for the template vs. aligned 02 data set.

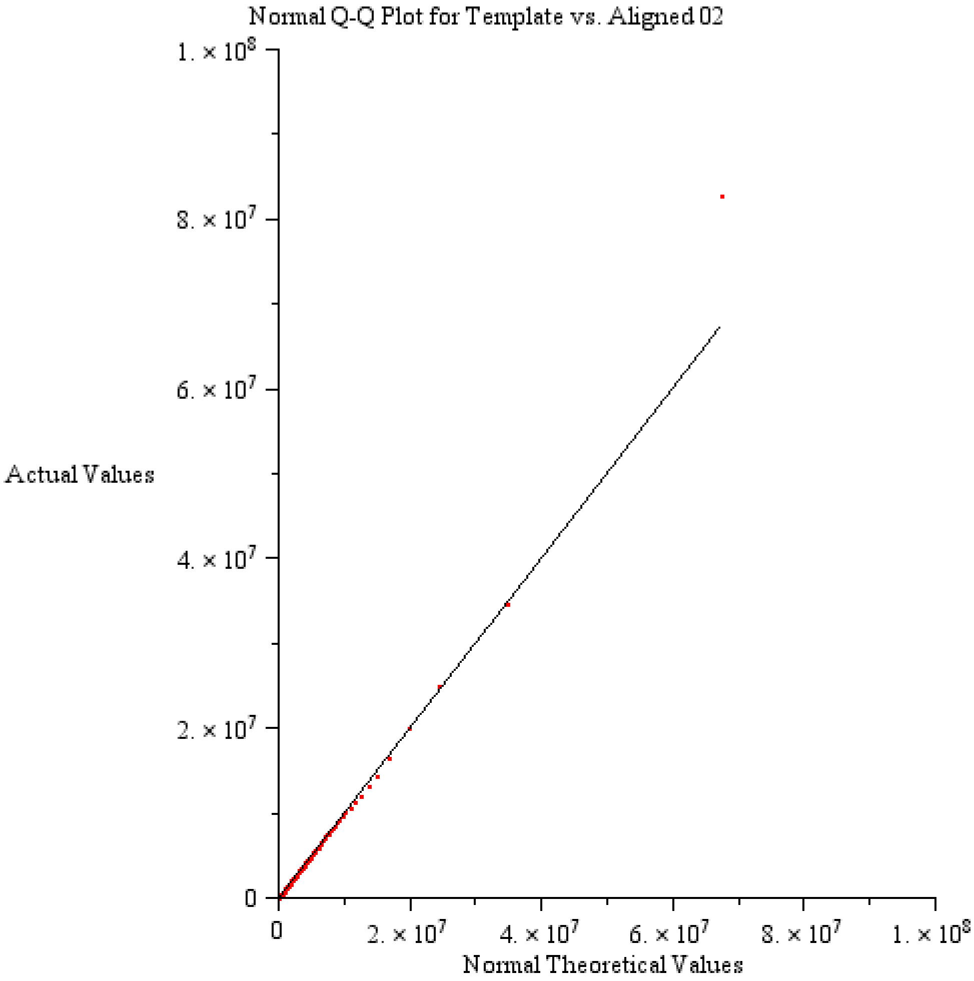

Figure 10. A Q-Q diagnostic plot for the template vs. aligned 02 data set.

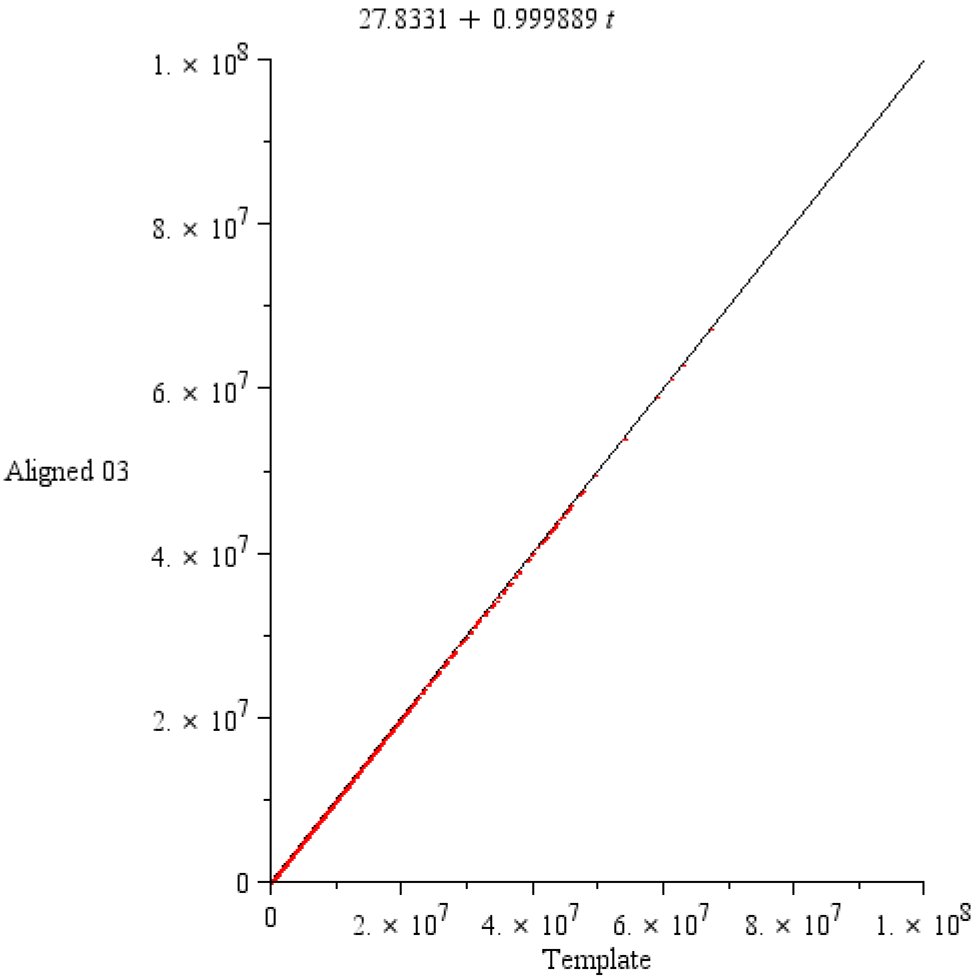

Figure 11. The intensity-intensity plot along with the linear fit for the template vs. aligned 03 data set. In this set of data, the aligned 03 set was in fact the template, so this is essentially the plot for template vs. template data set.

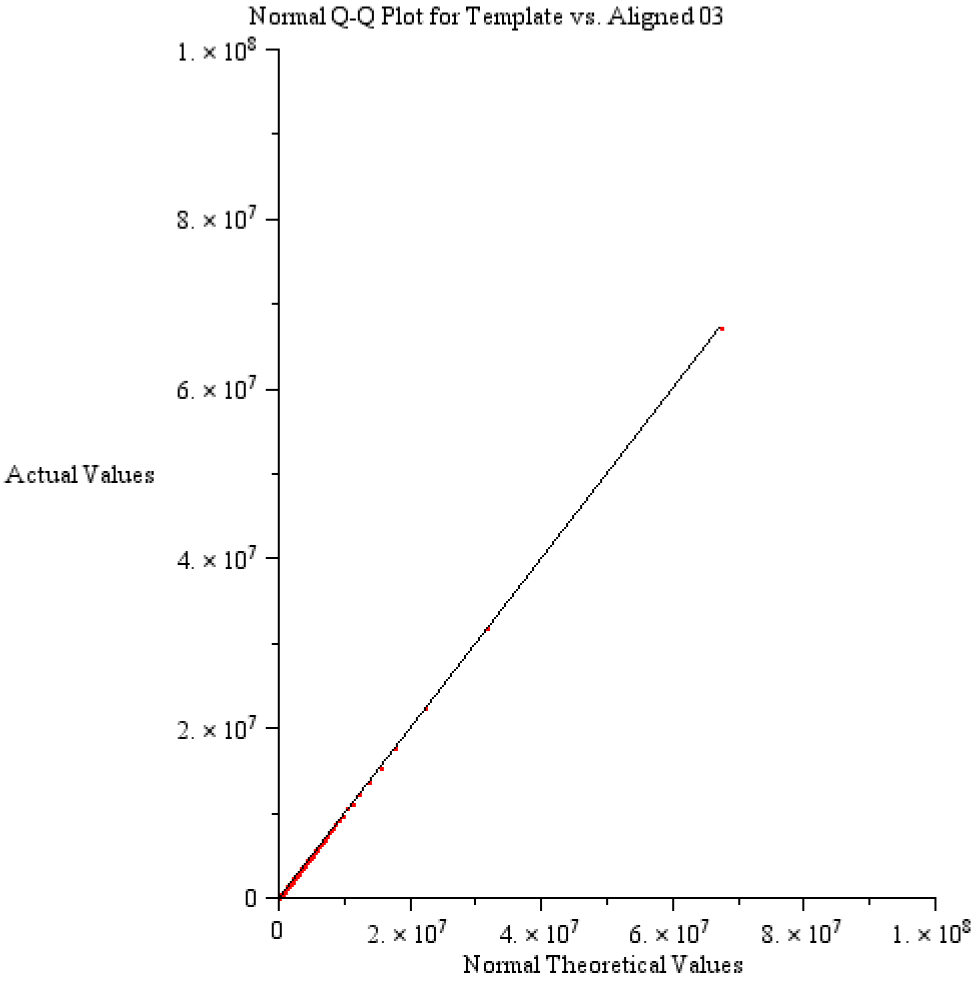

Figure 12. A Q-Q diagnostic plot for the template vs. aligned 03 data set.

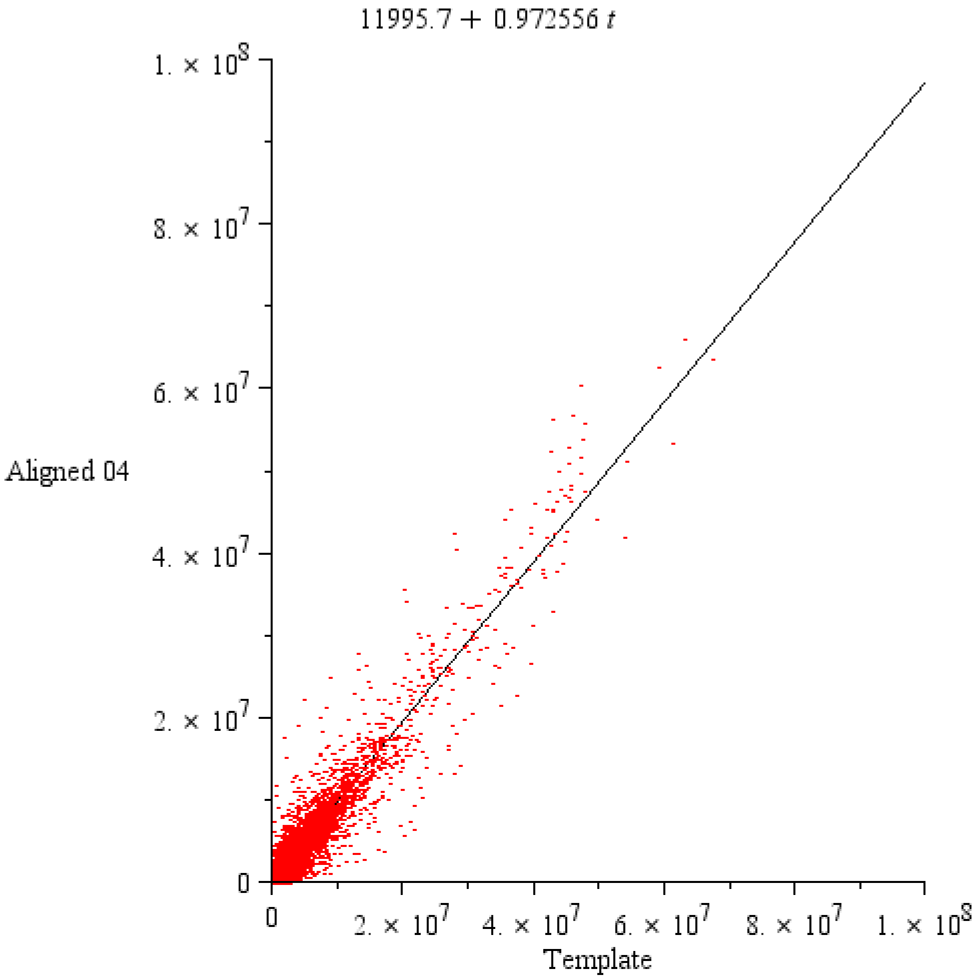

Figure 13. The intensity-intensity plot along with the linear fit for the template vs. aligned 04 data set.

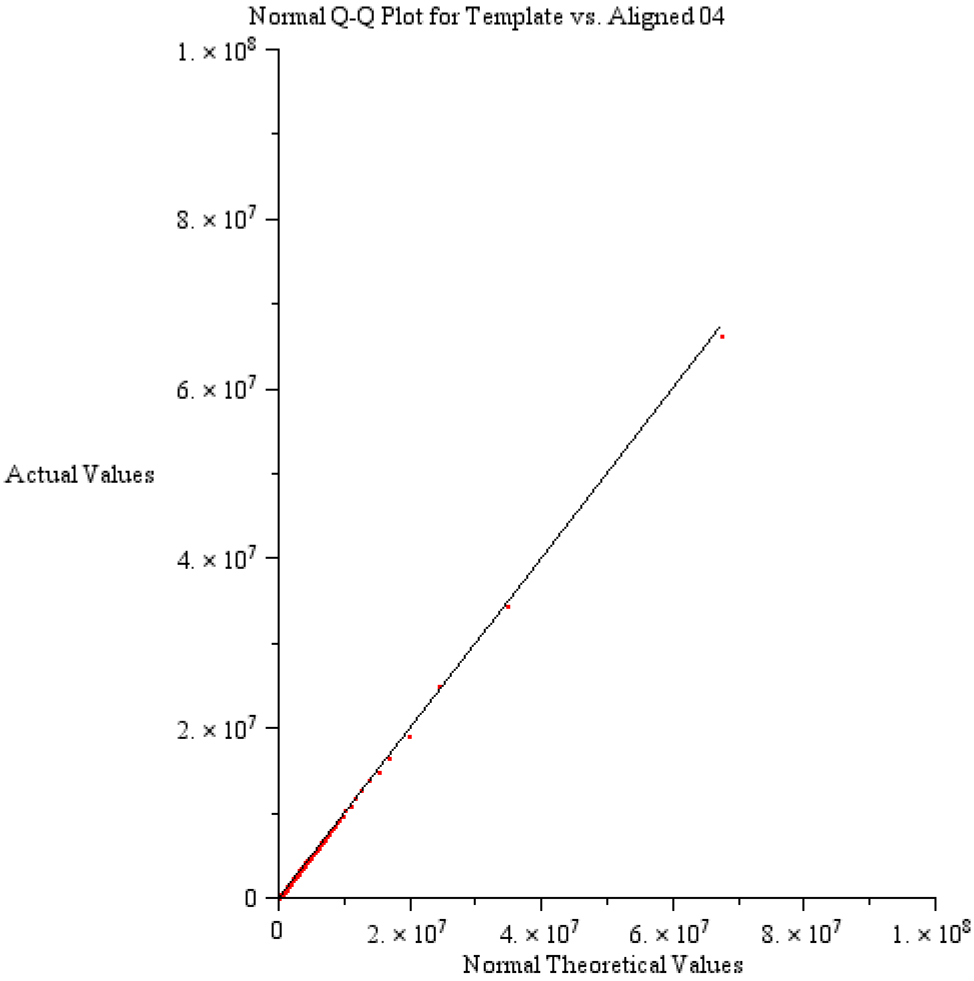

Figure 14. A Q-Q diagnostic plot for the template vs. aligned 04 data set.

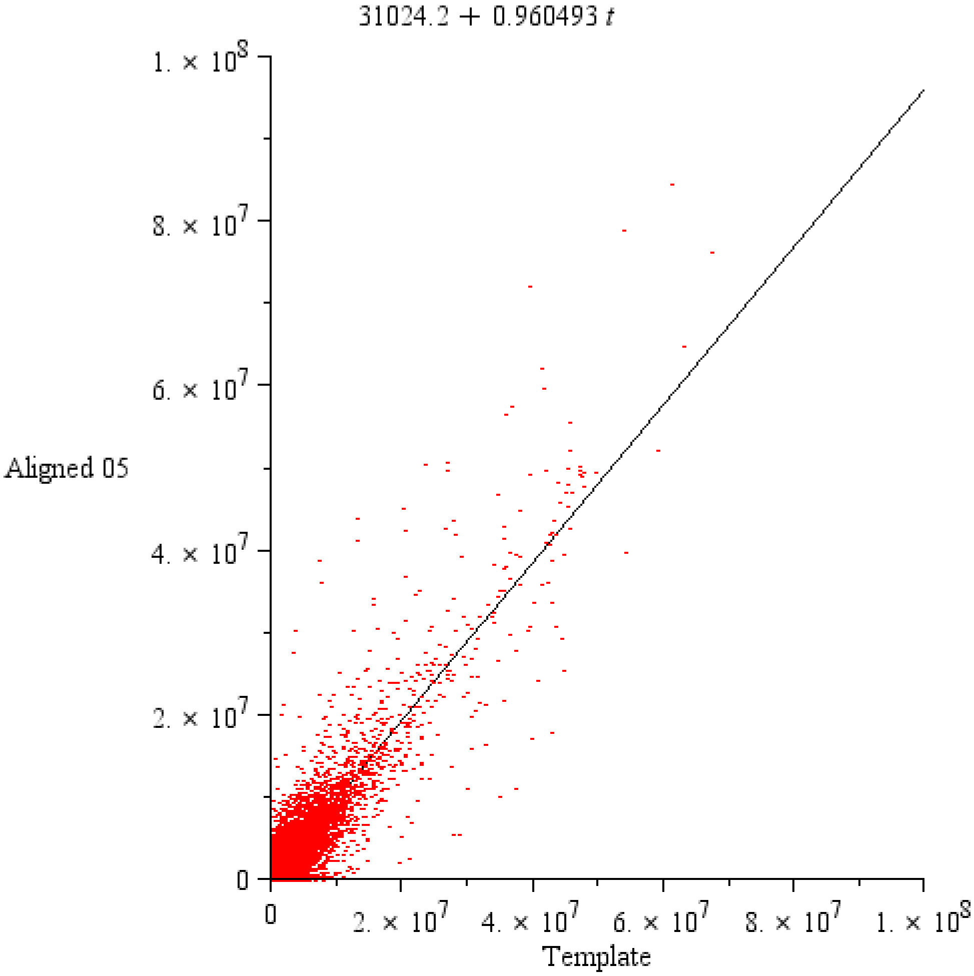

Figure 15. The intensity-intensity plot along with the linear fit for the template vs. aligned 05 data set.

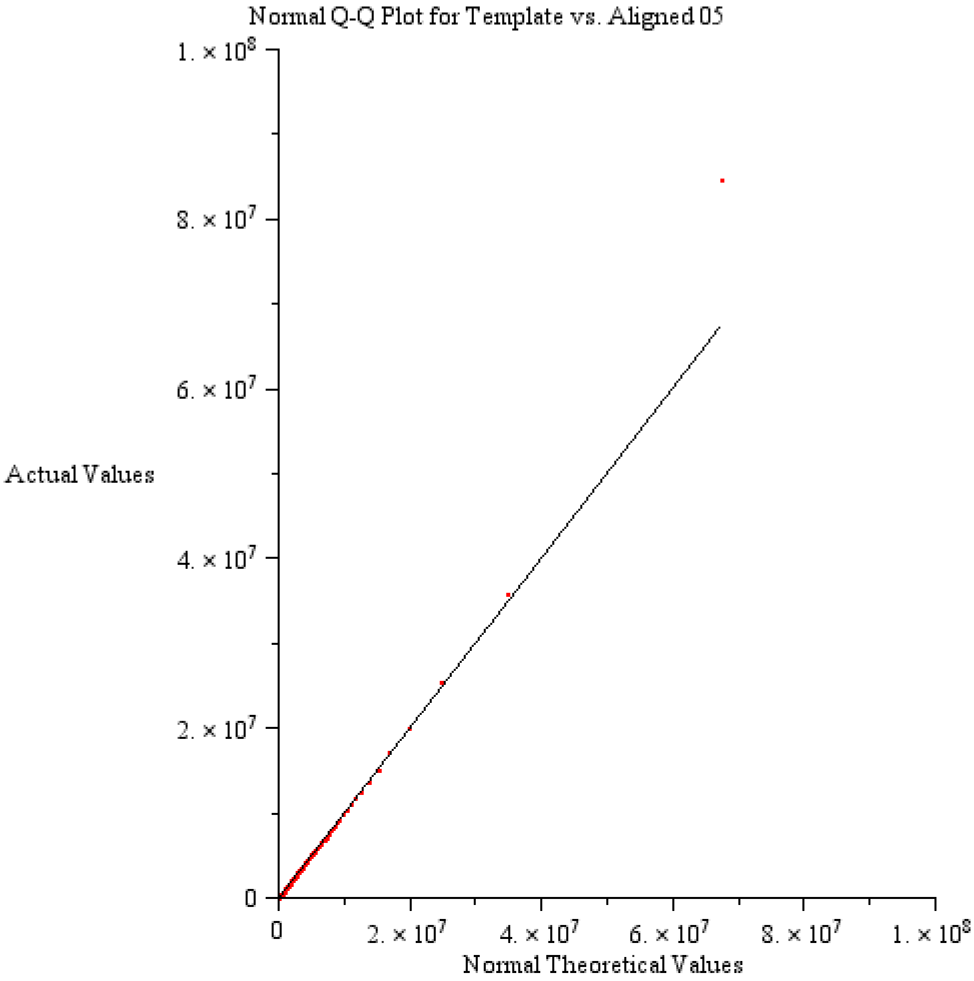

Figure 16. A Q-Q diagnostic plot for the template vs. aligned 05 data set.

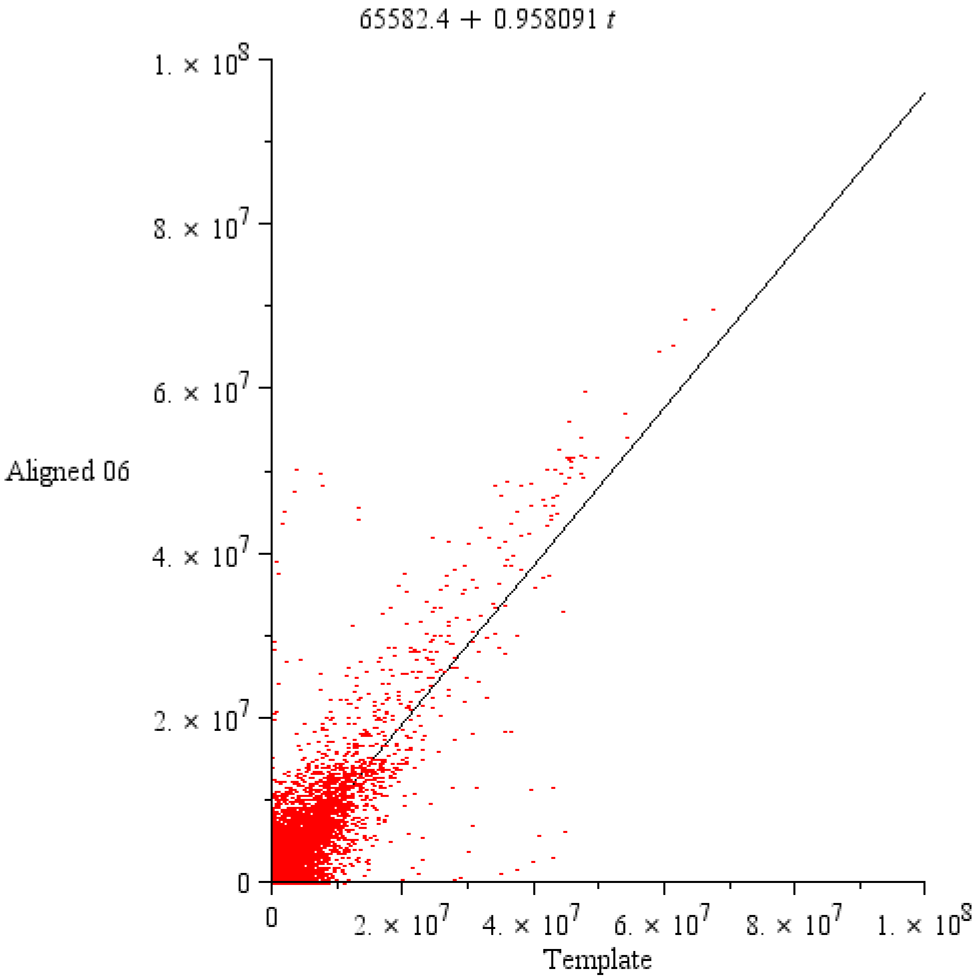

Figure 17. The intensity-intensity plot along with the linear fit for the template vs. aligned 06 data set.

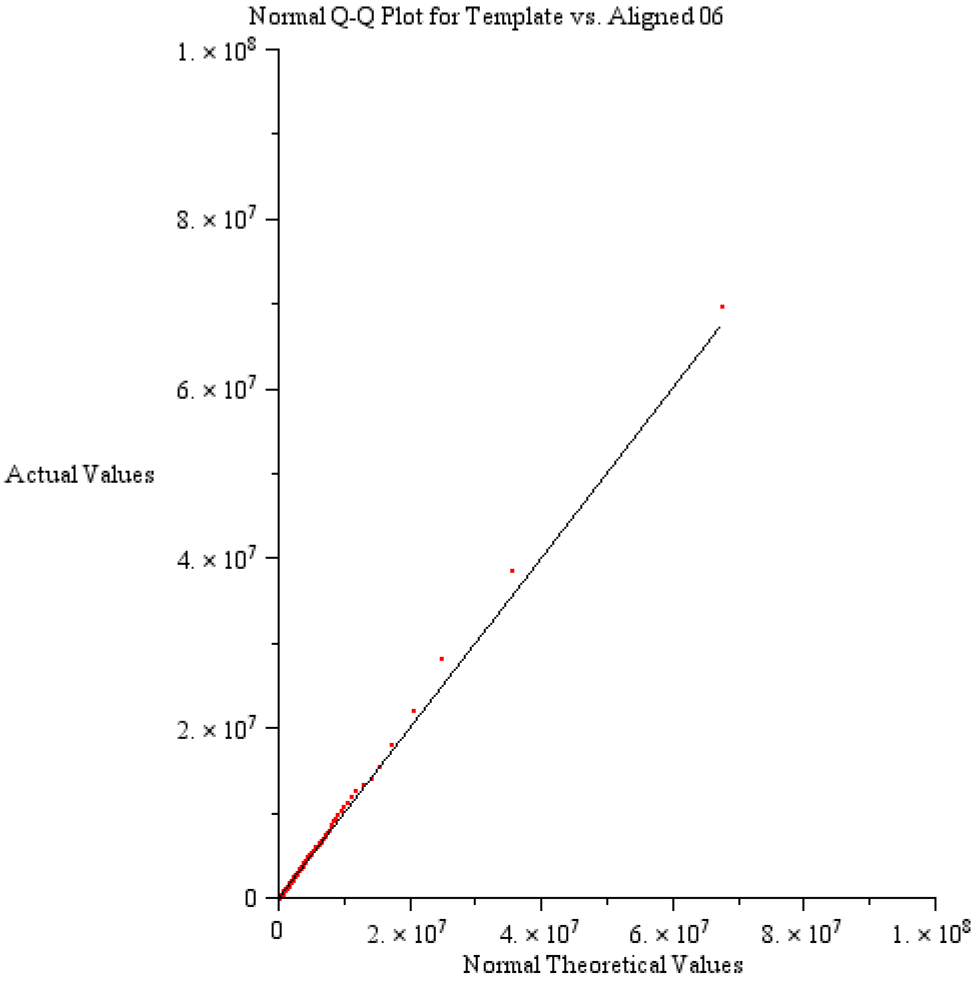

Figure 18. A Q-Q diagnostic plot for the template vs. aligned 06 data set.

Table 1 gives a summary of the linear regression parameters, test statistics, p-values, and r values for each of the six data sets and the simulated test data.

Table 1. Statistical values for each data set comparison.

aligned data set	β_0	β_1	β_0 Z test statistic	β_1 Z test statistic	β_0 p-value	β_1 p-value	Pearson's r
Random	4.98×10^{7}	2.76×10^{-3}	271	-313	0.000	0.000	-3.05×10^{-3}
Perfect	-0.708	0.999	-0.802	-62000	0.422	0.000	1.00
1	78100	1.22	11.8	12.8	0.000	0.000	0.874
2	6490	0.999	2.42	-0.130	0.0154	0.897	0.965
3	27.8	0.999	2.73	-3.51	< 0.01	< 0.001	1.00
4	12000	0.973	4.60	-4.00	$< 10^{-5}$	$< 10^{-4}$	0.960
5	31000	0.960	6.50	-3.22	0.000	< 0.01	0.918
6	65600	0.958	13.7	-3.48	0.000	$< 10^{-3}$	0.872

CHAPTER IV

SUMMARY AND CONCLUSIONS

Interpretation of results

Based on table 1, the simulated test cases alone show a weakness in the algorithm. We will refer to the situation where $\beta_1 \neq 1$ or $\beta_0 \neq 0$ as nonlinearity in that parameter. While the randomly generated poor alignment data did result in concluding nonlinearity and thus a bad alignment, the near-perfect example also concluded nonlinearity in the parameter β_1, whereas we would have expected a failure to suggest nonlinearity. The root of this problem lies in the number of points being tested, which is 100,000 in the case of the simulated data. Since the slope parameter β_1 is slightly perturbed from 1 and takes a value of 0.999, the large number of points drive the standard error increasingly lower. Essentially, the null hypothesis in this case is far too strict, resulting in a conclusion of nonlinearity.

In all but one case with the actual data the algorithm concluded that there was sufficient statistical evidence to suggest that there was not linearity for that parameter. The one exception was with the aligned data set 2 where there was insufficient statistical evidence to conclude nonlinearity for β_1. However, even then since nonlinearity was concluded for β_0 the overall alignment would be marked as poor.

Practical significance vs. statistical significance

While all but one hypothesis test resulted in a statistically significant rejection, the accompanying Pearson's r correlation values may suggest alternative conclusions. Speaking in terms of effect size given by r, conventions state that r = 0.1 is a small effect size, r = 0.3 is a medium effect size, and r = 0.5 is a large effect size for general purposes.[9] Using these guidelines, each of the r values has a large effect size, although some are certainly stronger than others. Beyond that, however, one must taken into account the context of the experiment to determine how to interpret the statistical results.[10] For example, according to Table 1 the aligned 03 data set, which should have a near-perfect alignment since it is essentially template vs. template, had tests for β_0 and β_1 that were both rejections indicating poor alignment. However the r value was 1.00 (to that precision) indicating a near-perfect positive linear correlation as expected. In this case, the value of r told us more about the alignment quality in terms of practical significance than the p-values did in terms of statistical significance.

This suggests that at the very least the effect size should be quantified either in terms of r or an alternative measure and reported along with significance levels. We conclude that in its current implementation, the hypothesis test algorithm is insufficient alone to give a sufficient indication of whether an alignment was good or bad, but if the reports of significance levels are accompanied by measures of effect size then a researcher can judge whether or not to accept an alignment. A fully automated procedure therefore has not been reached.

Usage of Z test statistics in relative comparisons

In addition to the r values for effect size, we can also use the specific Z test statistics to indicate that a particular alignment was superior to another, even if both result in a statistically significant rejection. Using this ranking system might write for y-intercepts 2 > 3 > 4 > 5 > 1 > 6 and for slopes 2 > 5 > 6 > 3 > 4 >1 since the smaller the magnitude of the Z test statistic the better the alignment.

Future work

The original goals of this project have not been fulfilled; we have not arrived at a fully-automated accurate method of judging alignment quality. Despite this, progress has been made and more work must be one. Based on the results of this research it seems that reliance on only an all-or-nothing significance test isn’t sufficient. Future work might focus more on measurements of effect size to assess the alignments performed or develop a categorical set of standard values for various parameters that determine levels of success of alignment.

Alternative data gathering methods

New methods of alignment assessment may arise from new methods of creating LC-MS data sets. One such example is manipulating the subject data by adding in quality-control (QC) proteins to act as internal controls. The idea is to use QC proteins such that we have as much information as possible prior to the LC-MS run to ensure their identification. Then we can analyze the warping of the QC proteins between aligned data sets to give a spatial assessment of the alignment quality.

REFERENCES

(1) Nesvizhskii, A. I.; Vitek, O.; Aebersold, R. Analysis and validation of proteomic data generated by tandem mass spectrometry. *Nature Methods* **2007**, *4* (10), 787-797.

(2) Aebersold, R.; Mann, M. Mass spectrometry-based proteomics. *Nature* **2003**, *422* (6928), 198-207.

(3) Pasa-Tolic, L.; Proteomic analyses using an accurate mass and time tag strategy. *Biotechniques* **2004**, *37* (4), 621.

(4) Monroe, M. E.; Tolić, N.; Jaitly, N.; Shaw, J. L.; Adkins, J. N. et al. VIPER: An advanced software package to support high-throughput LC-MS peptide identification. *Bioinformatics* **2007**, *23* (15), 2021-2023.

(5) Salvador, S.; Chan, P. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis* **2007**, *11* (5), 561-580.

(6) Prince, J.; Marcotte, E. Chromatographic alignment of ESI-LC-MS proteomics data sets by ordered bijective interpolated warping. *Analytical Chemistry* **2006**, *78* (17), 6140-52.

(7) Lange, E. Critical assessment of alignment procedures for LC-MS proteomics and metabolomics measurements. *Bioinformatics* **2008**, *9* (375).

(8) Fox, J. *Applied Regression Analysis, Linear Models, and Related Methods*.; Sage Publications, Inc: Thousand Oaks, California, **1997**.

(9) Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed.; Lawrence Earlbaum Associates: Hillsdale, New Jersey, **1988**.

(10) Alhija, F. N.; Levy, A. Effect size reporting practices in published articles. *Educational and Psychological Measurement* **2009**, *69* (2), 245-265.

(11) Sedgewick, R. *Algorithms in C++, Parts 1-4: Fundamentals, Data Structure, Sorting, Searching*, 3rd ed.; Addison-Wesley Publishing Company, Inc: Reading, MA, **1998**.

APPENDIX A

DTW ALGORITHM

Formulation

This implementation is adapted from Salvador and Chan.[5] Suppose we are given two time series X and Y with lengths |X| and |Y|:

$X = x_1, x_2, \ldots, x_i, \ldots, x_{|X|}$

$Y = y_1, y_2, \ldots, y_j, \ldots, y_{|Y|}$

We seek to construct a warp path W:

$W = w_1, w_2, \ldots, w_K$

Here we require $\max(|X|, |Y|) \leq K < |X| + |Y|$ and K is the length of W. Each $w_k = (i, j)$ and denotes a warping on the time axis from the ith index of X to the jth index of Y; every index of both time series X and Y must be used. We also take $w_1 = (1, 1)$ and $w_K = (|X|, |Y|)$. Lastly we seek to minimize the distance of the warp path:

$\mathrm{Dist}(W) = \Sigma_{\text{all } k}\, \mathrm{Dist}(w_{ki}, w_{kj})$

We may use different measures of distance but for the purposes of this implementation we use the squared Euclidian distance, so:

$\mathrm{Dist}(w_{ki}, w_{kj}) = (w_{kj}(x) - w_{ki}(x))^2 + (w_{kj}(y) - w_{ki}(y))^2$

Cost matrix creation

We use a cost matrix D to determine the minimum distance warp path. The cost function D(i, j) is defined recursively as follows:

$$D(i, j) = Dist(i, j) + \min[\ D(i - 1, j), D(i, j - 1), D(i - 1, j - 1)\ ]$$

To utilize the cost function in this recursive manner, the cost matrix must be filled from bottom to top, left to right, and one column at a time. Once the cost matrix is filled, the warp path is computed by a greedy search starting at D(|X|, |Y|) such that the next entry in the warp path is essentially min[D(i - 1, j), D(i, j - 1), D(i - 1, j - 1)]. The search is complete when D(1, 1) is added to the warp path. Once the warp path is computed each warping w_K is applied to the time series to be aligned, and the DTW alignment is complete.

APPENDIX B

ALIGNMENT ASSESSMENT ALGORITHM

Algorithm outline

Note that regression and hypothesis testing formulas presented here are adapted from Fox.[8] The quality assessment algorithm can be broken down into the following major steps when given a template data set T and an aligned data set A:

1. Intensity List Construction
2. Linear Regression
3. Sigma Matrix Computation
4. P-value Computation

Intensity list construction

The construction of the intensity data set must prepare the data to satisfy the model assumptions in the linear regression. We assume the following probability model:

$Y_k = \beta_0 + \beta_1 x_k + \varepsilon_k$ where $\varepsilon_k \sim N(0,\sigma_k^2)$ .... (2)

If we regard data sets T and A as m×n matrices where for each A_{ij} and T_{ij} there is a corresponding intensity value, we can construct a list M with *at most* mn elements where each element is given by $M_k = [A_{ij} , T_{ij}]$ where $A_{ij} \neq 0$ and $T_{ij} \neq 0$.

Linear regression

This step involves estimating the parameters β_0 and β_1 in the probability model given by equation 1. If we define the matrix X having row elements $[1\ M_{k1}]$ and the column vector **y** having elements $[M_{k2}]$ for all k from the given data, then we can compute the parameters by:

$\boldsymbol{\beta} = (X^T X)^{-1} X^T \mathbf{y}$

Here X^T denotes the transpose and all bolded quantities are vectors.

Sigma matrix computation

Here we must compute a mn×mn matrix which we denote by Σ. This step is used to take into account non-constant variance. Specifically, for every 100 intensity values in the intensity list we compute σ_k^2 as follows:

$\sigma_l^2 = 1 / 98\ \Sigma_{k=1\rightarrow 100}\ (y_k - Y)^2$

Here Y is the estimated value from the linear regression. Then assign $\sigma_k = \sigma_l$ for each 100 sampled k.

P-value computation

Once the matrix Σ is computed we can compute a variance-covariance matrix as follows:

$\mathrm{Var}(\boldsymbol{\beta}) = (X^T X)^{-1} X^T \Sigma X (X^T X)^{-1}$

This yields a 2×2 matrix where the entry $\mathrm{Var}(\boldsymbol{\beta})_{1,1} = \mathrm{Var}(\beta_0)$ and $\mathrm{Var}(\boldsymbol{\beta})_{2,2} = \mathrm{Var}(\beta_1)$. We will test the following hypotheses:

H_0: $\beta_1 = 1$ vs. H_A: $\beta_1 \neq 1$

H_0: $\beta_0 = 0$ vs. H_A: $\beta_0 \neq 0$

Here the test statistics are given by:

$T_{\beta 0} = \beta_0 / \mathrm{Var}(\beta_0)^{1/2}$

$T_{\beta 1} = (\beta_1 - 1) / \mathrm{Var}(\beta_1)^{1/2}$

These test statistics follow a T_{mn-2} distribution; with the extremely large number of data points used this has a normal N(0,1) distribution, so at this point it is just a matter of integration to compute the test statistics.

APPENDIX C

ALIGNMENT ASSESSMENT ALGORITHM IMPLEMENTATION

The following is the Maple code for the assessment algorithm. First we divide up the main code block into three sections and then describe particular procedures used that aren’t included in Maple packages. In particular, the DataManip and Proteomics packages are custom packages created for use in this research. The ArrayTools, Linear Algebra, and Statistics packages are default Maple packages.

Data import and preprocessing

```
with(Proteomics): with(LinearAlgebra): with(ArrayTools): with(DataManip):
with(Statistics):

M:=502: N:=1201:

for k from 1 to 6 do
temData:=Array(readdata("file_template.txt",float,M)):
aliData:=Array(readdata(cat("file_aligned_0",k,".txt"),float,M)):

for i from 2 to M do
    intensitiesVectorRaw[i]:=Alias(Array(Transpose(Matrix(temData))[i]),1,[1..N]):
    intensitiesVectorAli[i]:=Alias(Array(Transpose(Matrix(aliData))[i]),1,[1..N]):
```

```
od:

allTem:=Concatenate(2,seq(intensitiesVectorRaw[i],i=2..M)):
allAli:=Concatenate(2,seq(intensitiesVectorAli[i],i=2..M)):

i:=1: j:=1:
while i<=(M-1)*N do
    if allTem[i]<>0 and allAli[i]<>0 then
        pairedIntensities[j]:=[allTem[i],allAli[i]]:
        j:=j+1:
    fi:
    i:=i+1:
od:

writedata(cat("file_aligned_0",k,"_dezeroed.txt"),convert(Array([seq(pairedIntensities[i]
,i=1..j)]),array));

od:
```

Data sorting

```
for i from 1 to 6 do
```

```
allPairedIntensities:=Array(readdata(cat("file_aligned_0",i,"_dezeroed.txt"),float,999999
,2)):

f0:=QuickSorter(allPairedIntensities):
f00:=Array([seq([f0[i][1],f0[i][2]],i=1..nops(f0))]):

writedata(cat("file_aligned_0",i,"_dezeroed_sorted.txt"),convert(f00,array)):

od:
```

Regression analysis and hypothesis test

```
for i from 1 to 6 do
colPairedIntensities:=Array(readdata(cat("file_aligned_0",i,"_dezeroed_sorted.txt"),float
,999999,2)):

X:=Transpose(Matrix(colPairedIntensities))[1]:
Y:=Transpose(Matrix(colPairedIntensities))[2]:
nEles:=ArrayTools[NumElems](X):

LinFit:=LinearFit([1,t],X,Y,t):
B0:=coeffs(LinFit)[1]:
B1:=coeffs(LinFit)[2]:
```

```
Sig:=SigmaPartition(X,Y,100):


Mat1:=Vector[row]([seq(1,i=1..nEles)]):

XMat:=Matrix([Vector[column](Mat1),Vector[column](X)]):


P1:=(Transpose(XMat).XMat)^(-1).Transpose(XMat):

P1Row1:=Vector[row]([seq(P1[1,i]*Sig[i,i]^2,i=1..nEles)]):

P1Row2:=Vector[row]([seq(P1[2,i]*Sig[i,i]^2,i=1..nEles)]):

P2:=Matrix([[P1Row1],[P1Row2]]):

P3:=XMat.(Transpose(XMat).XMat)^(-1):


Final:=P2.P3:


TSB0:=B0/sqrt(Final[1,1]):

TSB1:=(B1-1)/sqrt(Final[2,2]):


writedata(cat("file_aligned_0",i,"_results txt"),convert([[TSB0,TSB1]],array)):


od:
```

Details of custom routines used

```
DataManip[SigmaPartition]:=proc(X::{Vector[row]},Y::{Vector[row]},n::posint)


local i, j, fit, nEles, sigmaDiag:


nEles:=LinearAlgebra[ColumnDimension](Matrix(X)):
fit:=Statistics[LinearFit]([1,t],X,Y,t):


#Compute all partitions except the last
for i from 1 to (nEles - (nEles mod n))/n do
    for j from n*(i-1)+1 to n*i do
        sigmaDiag[j]:=sqrt(evalf(1/(n-2)*add((Y[j]-subs(t=X[j],fit))^2,j=n*(i-
        1)+1..n*i))):
    od:
od:


if (nEles mod n)>0 then
    for j from (nEles - (nEles mod n)) + 1 to nEles do
        sigmaDiag[j]:=sigmaDiag[nEles - (nEles mod n)]:
    od:
fi:


LinearAlgebra[DiagonalMatrix](Vector[row]([seq(sigmaDiag[j],j=1..nEles)]));
```

```
end:
```

Note: The following quicksort algorithm is adapted from Sedgewick.[11]

```
DataManip[QuickSorter]:=proc(A::{list,Array,Matrix})

local top, left0, right0, left, right, i, neles;

neles:=ArrayTools[NumElems](A)/2:

top:=[seq(A[i], i=2..neles)];
left0:=select(x -> x[1] < A[1][1], top);
right0:=select(x -> x[1] >= A[1][1], top);
left:=Array([seq([left0[i][1],left0[i][2]],i=1..nops(left0))]);
right:=Array([seq([right0[i][1],right0[i][2]],i=1..nops(right0))]);

[op(DataManip[QuickSorter](left)), A[1], op(DataManip[QuickSorter](right))];

end:
```

CONTACT INFORMATION

Name: Isaac Velando

Professional Address: c/o Dr. Alan Dabney
Department of Statistics
MS 3143 TAMU
College Station, TX 77843-3143

Email Address: iwvelando@gmail.com

Education: B.S., Applied Mathematical Sciences, Texas A&M University, May 2010