

Table of Contents

Introduction.....	2
Background.....	2
Goals and Findings	3
Include Examples of Graphs: Data, Data w/ curves, Data w/ curve and CI. Non-linear Regression and Estimation.....	5
Non-linear Regression and Estimation	6
Methods of Parameter Estimation.....	6
Positive Definite.....	8
Confidence Intervals	10
Control Charts.....	11
Numerical Methods.....	14
Methods in S-PLUS	14
Methods in MATLAB.....	14
Conclusion	15
Problem Section	15
Appendix A: Program User's Guide.....	16
Overview.....	16
Expected Directory Structure.....	16
Preparing a Data Set.....	17
Obtaining Parameter Estimates.....	19
Interpreting the Program's Output	19
Appendix B: Program Theory of Operation	20
Overview.....	20
Description of the Problem to be Solved	20
Method for Finding the Solution.....	20
Algorithm for Finding the Solution	22
Appendix C: Out-of-Control Signals for x-bar Charts of Data Sets.....	23
Test A: One point beyond the three standard deviation level.....	23
Test B: Run of nine points on one side of the residual mean	23
Test C: Two out of three points beyond two standard deviations.....	23
Use of control chart program	29
Appendix D: Maximum Likelihood Estimation	30
Appendix D: Maximum Likelihood Estimation	30
Appendix E: Positive Definite Matrices	31

Table of Figures

Figure 1: Estimated Values	4
Figure 2. Errors	6
Figure 3. Error Histogram.....	7
Figure 4. Confidence Interval Display.....	10
Figure 5. Quality Control using different averaging terms.....	12
Table C1. Distribution of Out-of-control Signals in Data Sets of Various Sizes	25
Table C2. Distribution of points beyond the three standard deviation level	26
Table C3. Distribution of a run of nine points on one side of the centerline.....	27
Table C4. Distribution of two out of three points beyond 2 standard deviations	28

Introduction

Background

Historically, the study of “heterogeneous” reactions in chemistry have been met with difficult or disappointing ends. Heterogeneous reactions in the context of atmospheric chemistry are defined as those between gases and either solids or liquids (Danckwerts). These types of reactions have a key role in understanding atmospheric conditions coupled with the study of ozone breakdown, particularly over Antarctica. Along these lines of study there have been few definitive studies done to determine exactly how the chemistry acts. The chemical makeup of the reactants are understood, and the understanding of the reactions at a molecular level is generally well understood. In contrast, though, what the molecular makeup of the surface of a condensed-phase gas looks like, little is known. Dr. Laura Iraci of NASA/Ames Research Center has been conducting experiments along these lines to determine the constants of the reactions in a controlled environment. She has established that the equation that her experimental data should follow has the form noted in equation (1).

$$\frac{1}{\gamma} = \frac{1}{\Gamma} + \frac{1}{\alpha} + \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right) \quad (1)$$

This equation sets a function γ , the net gas uptake, in terms of Γ , the gas-phase diffusion to the surface, and α , the mass accommodation across the interface, added to a function of the products of the solubility and the reaction in the bulk aqueous phase. The solubility term is comprised of \bar{c} , the average molecular speed in the gas phase, R , the gas constant, T , the absolute temperature, H Henry’s law constant, and D , the diffusion constant in the liquid phase. The reaction in the bulk aqueous phase is comprised of t , the seconds since the beginning of the reaction, and k , the rate of an irreversible, first-order reaction. This equations is only an approximation to the true differential equation associated with these types of reactions. Furthermore, this approximation is only valid after 36 seconds and when $(kt)^{1/2} < 1$ (Finlayson-Pitts).

The parameters and variables in equation (1) are defined as follows: γ is measured directly in the experiments, and t is the time (measured in seconds) associated with the γ measurement. R is the known gas constant (0.08206), and T is the absolute temperature (298K) fixed at the time of the experiment. D and \bar{c} are both dependent on the gases being studied, but are known at the time of the experiments. Therefore H , k , α , and Γ are unknown parameters to be determined after the experiment through statistical means. Also, all of these parameters must be non-negative due to chemical and physical properties.

Goals and Findings

After conducting her experiments, Dr. Iraci gave our CAMCOS team the data from her experiment to examine. Each set of data included her measured values of γ , the net gas uptake, and t , time in seconds, as well as the known values for D , \bar{c} which vary with each data set. Our goal was to estimate, through statistical investigation, the remaining unknown variables in Equation 1. Along with these estimates we wanted to create code in MATLAB so Dr. Iraci would be able to estimate parameters in a similar fashion for future data sets which she may obtain through further experimentation. Since our investigation will produce estimates for the unknown values in Equation 1, we also want to provide an interval around each estimate in which we are highly confident the true value of the parameter resides. Lastly, we also desire to produce control charts from the data so she may be able to establish if her experiments are continuing to produce stable information, and detect problems, if any, with the processes.

The two terms $1/\Gamma$ and $1/\alpha$ are not independently identifiable. Due to this complication, we may only estimate one intercept term for equation (1). Thus we need to combine the α and Γ terms into one term, the intercept term, as follows,

$$\beta_0 = \frac{1}{\Gamma} + \frac{1}{\alpha} \quad (2)$$

For computational ease, we also occasionally make use of the following substitution

$$\beta_1 = \frac{\bar{c}\sqrt{\pi}}{4RT\sqrt{D}} \left(\frac{1}{H} \right) \quad (3)$$

This substitution allows us to find β_1 without the overhead of the constants, and then we are able to solve for H to find our estimate. This substitution reduces the non-linear equation (1) to the following equation, for which we will find estimates of the parameters β_0 , H and k .

$$\frac{1}{\gamma} = \beta_0 + \beta_1 \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right) \quad (4)$$

Also, it is known that k may equal zero for some of the data sets. When this occurs equation (4) simplifies to equation (5) which is a simple linear equation in the parameters to be estimated (Montgomery, 1992).

$$\frac{1}{\gamma} = \beta_0 + \beta_1 \sqrt{t} \quad (5)$$

Using equations (4) and (5) we are able to produce estimates of the intercept, H and k . The methods that enabled use to produce these estimates will be thoroughly discussed in the following section, Non-linear Regression and Estimation. Table 1 contains the estimated values of the intercept, H and k for each data set that was provided to us.

Figure 1: Estimated Values

Parameter Estimates			
10 May 2002			
	Parameter		
Data Set	beta0	H	k
b08run6	3.44E+01	8.61E+04	9.02E-02
b11run3	1.60E-03	2.46E+04	1.81E-02
b11run4	2.64E-03	2.08E+04	1.89E-03
b11run6	0.00E+00	3.12E+04	0.00E+00
b11run7	0.00E+00	2.99E+03	0.00E+00
c17run8	6.73E+00	2.89E+05	2.30E-03
c17run9	2.75E+01	5.91E+06	0.00E+00
e09r3val	1.08E+01	5.53E+07	0.00E+00
e11r8only	2.39E+01	6.07E+05	0.00E+00
m24run12	3.55E-04	2.49E+05	6.38E-03
m24run13	4.39E+01	8.89E-03	3.47E-04
m25run2	3.44E-04	5.94E+05	1.23E-02
m25run3	7.64E+00	5.65E+05	1.60E-02
m25run5	3.75E+01	1.87E+06	2.12E-04
u07run296	3.28E+01	2.05E+05	0.00E+00
u07run298	2.56E+01	1.84E+05	2.73E-06
u08run296	1.37E+01	1.82E+05	6.45E-05
u08run298	5.37E+00	1.70E+05	8.99E-05
u08run396	2.87E+01	1.94E+05	3.53E-05
u08run398	2.54E+01	1.98E+05	2.88E-05
u08run496	2.65E+01	1.43E+05	2.12E-04
u08run498	2.43E+01	1.54E+05	1.60E-04
u08run596	9.83E-04	1.04E+04	3.43E-02
u08run598	1.22E+02	3.44E+04	7.48E-03
x22run11	1.54E+01	1.16E+05	0.00E+00
x22run12	1.18E+02	1.05E+05	0.00E+00

Table 1. (cont)

x23run2	1.93E-03	3.37E+05	9.47E-04
x23run4	1.65E-03	1.69E+05	2.40E-03
x23run9	8.63E+01	1.27E+05	0.00E+00
x23run10	4.59E+01	2.21E+05	5.46E-05
x28run196	1.55E+01	2.63E+06	0.00E+00
x28run198	1.38E+01	2.49E+06	0.00E+00
x28run2	1.06E+01	2.48E+06	0.00E+00
x28run3	1.25E+01	1.05E+07	0.00E+00
x28run496	1.07E+01	1.17E+07	0.00E+00
x28run498	9.72E+00	8.11E+06	1.22E-05
x28run5	1.73E+01	1.30E+07	0.00E+00

Include Examples of Graphs: Data, Data w/ curves, Data w/ curve and CI.

Non-linear Regression and Estimation

Methods of Parameter Estimation

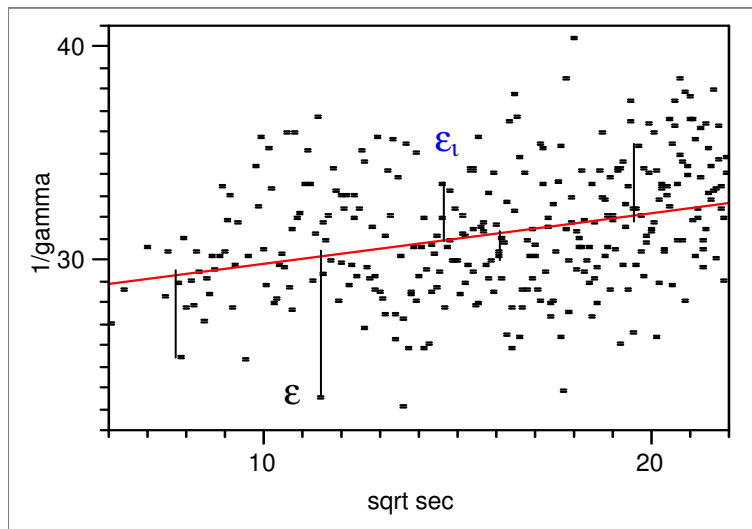
The chemical equation (4) has a form that is not linear, therefore it must be analyzed using nonlinear multivariate methods. It is our desire to find parameter estimates of the intercept, β_1 and k which will fit equation (4) for each set of data. These estimates should be unbiased and have the smallest possible variance that can possibly found (unbiased meaning that the estimates are on average correctly estimating the parameter, with variance being the amount of variation or noise in the estimate). To accomplish this the method of Maximum Likelihood Estimation (MLE) is used, for MLE produces parameter estimates that are asymptotically unbiased and have minimum variance (Lehmann). MLE parameter estimates also have other useful properties that make them desirable if they may be found.

After plotting the data, a curve can be drawn that best fits the data which has the form $1/\gamma = f(t)$,

$$\text{where } f(t) = \beta_0 + \beta_1 \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right).$$

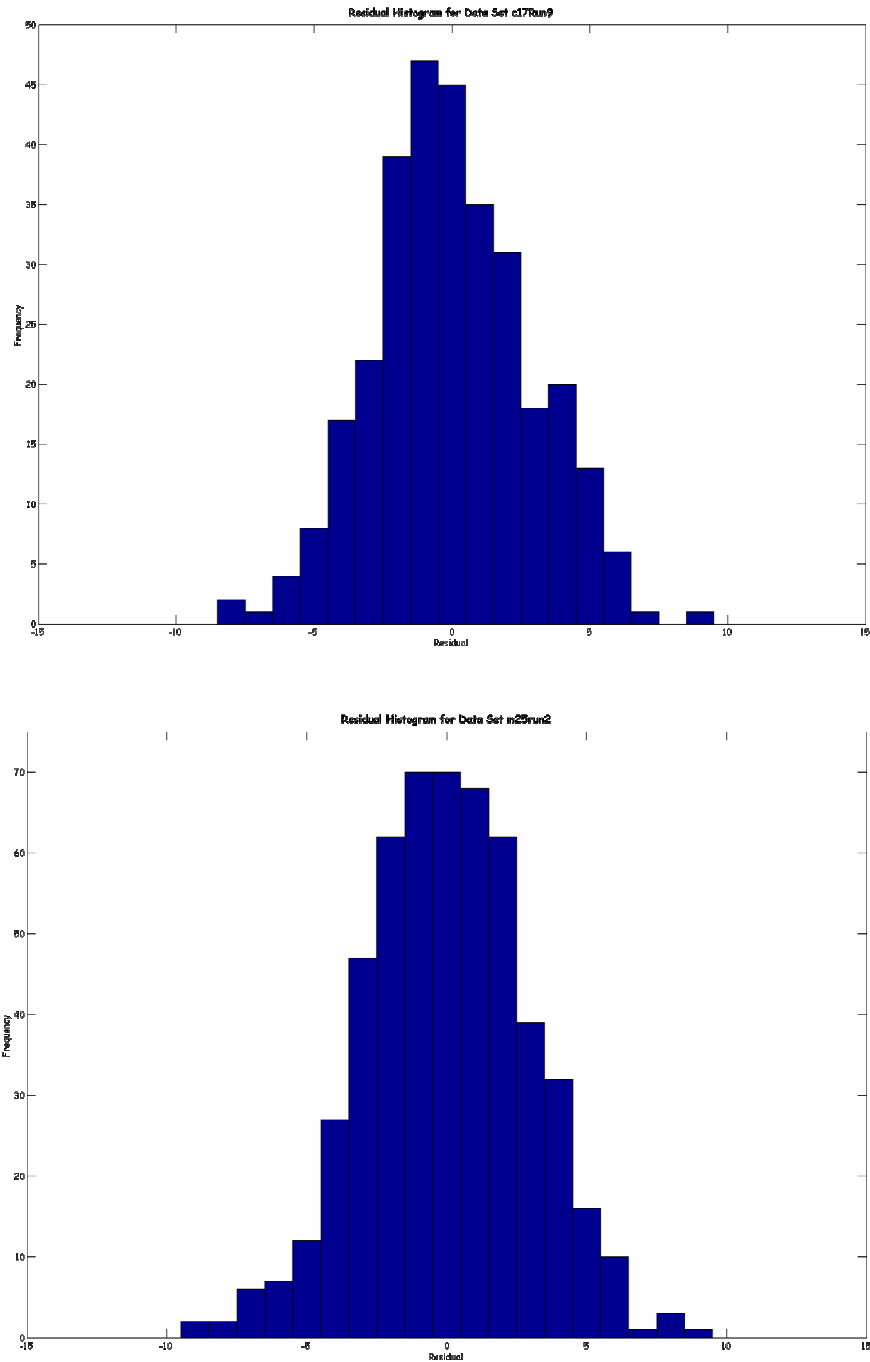
The fit of this curve can be measured by how far each point is from the curve; this is the error of the 'i'th point or ϵ_i , represented by the length of the lines in Figure 1. Each measurement of the net gas uptake, with its associated time in seconds can then be written as $1/\gamma_i = f(t_i) + \epsilon_i$, where the subscript 'i' represents the 'i'th tuple.

Figure 2. Errors



We assume these errors follow a Normal distribution and are independently and identically distributed (iid), ie. random noise, stated as $\epsilon_i \sim N(0, \sigma^2)$. This is a standard assumption and may be verified afterwards by examination of the residuals (the two graphs in Figure 3 verify this assumption of Normality, as can be seen in the near bell shaped curves.)

Figure 3. Error Histograms



When the errors are Normally distributed the MLE estimates are identical to the estimates which would be found by minimizing $\Sigma \varepsilon_i^2$ (see Appendix D). This is appropriately described as the sum of the squared errors, or SSE. Since $1/\gamma_i = f(t_i) + \varepsilon_i$. then $\varepsilon_i = 1/\gamma_i - f(t_i)$. Thus in order to find the MLE parameter estimates, we must find the values of the parameters that minimize $\Sigma \varepsilon_i^2 = \Sigma (1/\gamma_i - f(t_i))^2$.

Analytically, we would like to minimize the function $\Sigma \varepsilon_i^2 = \Sigma (1/\gamma_i - f(t_i))^2$, which is $L(\beta_0, H, k) = \text{SSE}$. Taking derivatives of L with respect to β_0 , H, and k; then setting these equal to zero.

$$\frac{\partial L}{\partial \beta_0} = -2 \sum_{i=1}^n \left[\frac{1}{\gamma_i} - \beta_0 - \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{1}{t_i^{-1/2} + \sqrt{\pi k}} \right) \right] \stackrel{\text{set}}{=} 0$$

$$\frac{\partial L}{\partial H} = 2 \sum_{i=1}^n \left[\frac{1}{\gamma_i} - \beta_0 - \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{1}{t_i^{-1/2} + \sqrt{\pi k}} \right) \right] \left[\left(\frac{1}{2H} \right) \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{1}{t_i^{-1/2} + \sqrt{\pi k}} \right) \right] \stackrel{\text{set}}{=} 0$$

$$\frac{\partial L}{\partial k} = \sum_{i=1}^n \left[\frac{1}{\gamma_i} - \beta_0 - \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{1}{t_i^{-1/2} + \sqrt{\pi k}} \right) \right] \left[\left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{t_i \sqrt{\pi}}{\sqrt{k} (t_i^{-1/2} + \sqrt{\pi k})^2} \right) \right] \stackrel{\text{set}}{=} 0$$

We then would solve for the parameters β_0 , H, and k. This method does not work, however, because the equations are not analytically tractable. Therefore, we must use numerical methods. Since this function is a sum of squares, it must be non-negative, therefore a minimum exists. We wish to ensure that when our numerical methods arrive at a minimum of L, that this point is a global minimum rather than a local minimum. One way of proving this is true is to show that the function has a unique minimum. Proving this would guarantee that we have found a global minimum. However, disproving this leads to no conclusion about finding a global minimum. One method to prove that there is a unique minimum is to construct a Hessian matrix of second derivatives of the function $L = \text{SSE}$ and check to see that it is positive definite.

Positive Definite

This method for showing that a function has a minimum is to construct a matrix containing the second order partial derivatives of L, the Hessian matrix A (Harville).

$$A = \begin{bmatrix} \frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_0 \partial \beta_1} & \frac{\partial^2 L}{\partial \beta_0 \partial k} \\ \frac{\partial^2 L}{\partial \beta_1 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_1 \partial \beta_1} & \frac{\partial^2 L}{\partial \beta_1 \partial k} \\ \frac{\partial^2 L}{\partial k \partial \beta_0} & \frac{\partial^2 L}{\partial k \partial \beta_1} & \frac{\partial^2 L}{\partial k \partial k} \end{bmatrix}$$

If this matrix is positive definite then the function has a unique minimum. There are many methods to determine whether a matrix is positive definite; we chose to check all sub-matrices for the property that their determinants are positive. With a positive definite 3x3 matrix, only three specific sub-matrices need to have positive determinants for the whole matrix to be considered positive definite. The principal minor determinants are

$$A_1 = \left[\frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} \right], \quad A_2 = \begin{bmatrix} \frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_0 \partial \beta_1} \\ \frac{\partial^2 L}{\partial \beta_1 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_1 \partial \beta_1} \end{bmatrix}, \quad A_3 = \begin{bmatrix} \frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_0 \partial \beta_1} & \frac{\partial^2 L}{\partial \beta_0 \partial k} \\ \frac{\partial^2 L}{\partial \beta_1 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_1 \partial \beta_1} & \frac{\partial^2 L}{\partial \beta_1 \partial k} \\ \frac{\partial^2 L}{\partial k \partial \beta_0} & \frac{\partial^2 L}{\partial k \partial \beta_1} & \frac{\partial^2 L}{\partial k \partial k} \end{bmatrix}.$$

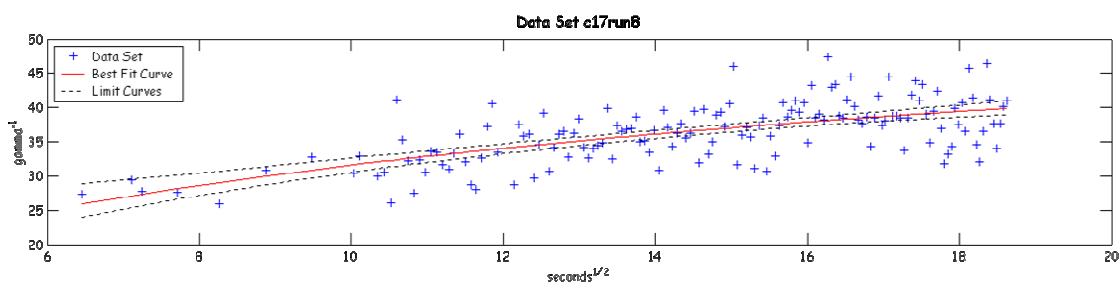
The first two matrices have positive determinants (see Appendix D). As of this writing we have yet to show that the determinant of the third matrix is positive.

Confidence Intervals

Through our numerical methods we were able to find the best estimates for the β_0 , H , and k . We then wanted to establish confidence intervals to take into account any variation or noise in the chemical system and give an interval for the possible true values of the parameters. We used a 95% confidence level for the intervals, so we can be 95% confident that the true values for β_0 , H , and k are inside the given intervals. Since all three parameters must be positive or in some cases zero, we cannot assume these parameters follow a Normal distribution. Since we cannot make this assumption, the usual methods for computing confidence intervals cannot be used. Again, a numerical method is employed.

The method of “bootstrapping” is chosen to establish the confidence intervals (Chernick and Davidson). In this method, random noise equivalent to that found in original data is used to simulate new sets of data, which leads to new parameters estimates. Then new curves can be drawn over the original data set. Then the area around that original curve is bounded by these new curves to establish an interval in which it is reasonable to assume that true curve lies (Huet and Bates). A 95% confidence interval is desired, so to 2.5% of the curves should be excluded, as well as the bottom 2.5%. In practice, a 1000 more data sets were randomly generated with a similar amount of noise as found in the original data set, resulting in 1000 new curves. At each point of time, the 25th curve from the bottom (representing the lower 2.5% bound), and the 25th curve from the top (representing the upper 2.5% bound), is selected. A line is then drawn through these points. These upper and lower curves then form a graphical 95% confidence interval for the true curve $1/\gamma = f(t)$. Similarly, estimates of β_0 , H , k are formed from these 1000 simulated data sets. After sorting them from smallest to largest, the 25th and 975th values then form a 95% confidence interval for the true value of the parameter in question.

Figure 4. Confidence Interval Display



Parameter Values			
	-- Estimate --	-- 95% Confidence Interval --	
beta0:	6.7258e+000	8.5990e-005	2.2898e+001
H:	2.8897e+005	1.5875e+005	1.4535e+006
k:	2.2982e-003	0.0000e+000	8.1330e-003

Control Charts

Control charts can be used to monitor the quality of a process in question. A control chart plots the data in a time ordered sequence where patterns may be seen. To consolidate the data, the averages of groups of points may be plotted, which will show trends in the processes being monitored (Montgomery, 2000). We use these charts as a means to test whether there may be something happening in the process that is not explained by our expected chemical equation: an out of control process. Tests can be applied to the charts to identify any problems with the process. Three such tests are applied to the residuals (errors) of data set after the parameters were estimated to determine if there are any special circumstances or problems in her data. (It should be noted that when referring to control chart points, they may represent single residual points from a data set, or an average of a number of points, creating a x-bar control chart.)

The first test seeks to find if there exists a points beyond three standard deviations(which is a function of the spread of the original data) away from the mean of the residuals (which is always zero). The probability of such a point, when the errors are from a Normal distributed is 0.3%, or about 3 times in one thousand, therefore it is useful to know when these points happen and how often.

The second test that is implemented is a check to see if there is a run of nine points on only one side of the mean. Such a run may be a signal of unsteadiness in the process. The probability of such an occurrence is again very small at 0.4%.

The last test implemented by this team is a test seeking two out of three points beyond two standard deviations on the same side of the mean. This may be an indication that more than random noise is affecting the process. The probability of this happening is about 0.5%, or 5 times out of one thousand, again prompting a desire to know when these points occur.

Since the data sets that we are working with may be quite large, we desired to know about how often these three errors may happen together when the errors are Normally distributed. Therefore we created random data of varying sizes to count the number of errors that occur from a strictly random set of residuals. This allows us to build guidelines in determining if a data set has too many errors in it to explain as random, thus prompting further investigation into that process or data set. Appendix C contains tables summarizes the number of errors that occurred in our random data sets of varying sizes. Each size was run 1000 times in the creation of this table.

Figure 5. Quality Control using different averaging terms

Test A <i>Data point beyond the 3 standard deviation level</i> Test B <i>9 consecutive points lying on the same side of the centerline</i> Test C <i>2 out of 3 points beyond the 2 standard deviation level</i>						
b08run6	88	1	1	1	0	2
b11run3	71	1	0	1	0	1
b11run4	97	1	1	0	0	1
b11run6	72	1	0	0	0	0
b11run7	65	1	3	2	0	5
c17run8	151	1	0	1	0	1
c17run9	310	1	1	0	0	1
e09r3val	1035	1	6	7	4	17
	(207)	5	4	3	4	11
e11r8only	551	1	1	1	2	4
m24run12	558	1	2	3	0	5
m24run13	873	1	5	3	4	12
	(218)	4	3	2	0	4
m25run2	537	1	2	2	0	4
m25run3	579	1	2	1	2	5
m25run5	634	1	2	1	2	5
u07run296	433	1	2	4	3	9
	(216)	2	2	3	1	6
u07run298	433	1	1	2	2	5
u08run296	641	1	2	0	2	4
u08run298	641	1	2	1	2	5
u08run396	453	1	2	1	1	4
u08run398	453	1	1	0	3	4
u08run496	483	1	2	0	1	3
u08run498	483	1	4	0	0	4
u08run596	406	1	3	1	1	5
u08run598	406	1	2	1	0	3
x22run11	179	1	3	0	2	5
	(89)	2	0	0	0	0
x22run12	133	1	3	1	1	5
	(66)	2	1	1	2	4
x23run10	279	1	2	0	0	2
x23run2	194	1	1	2	0	3
x23run4	242	1	2	0	1	3
x23run9	121	1	2	0	0	2
x28run196	348	1	1	0	0	1

<i>x28run198</i>	348	1	1	0	0	1
<i>x28run2</i>	436	1	2	2	2	6
<i>x28run3</i>	237	1	2	1	0	3
<i>x28run496</i>	216	1	0	0	0	0
<i>x28run498</i>	216	1	1	0	0	1
<i>x28run5</i>	155	1	0	1	0	1

*parenthesized values in the size columns represent the number of data points after averaging

Shaded cells indicate data sets having more out-of-control signals than expected. For a further explanation of control charts see Montgomery, Douglas C. An explanation of how they are specifically implemented in our MATLAB program, consult Appendix C.

Numerical Methods

Methods in S-PLUS

In seeking the best parameter estimates for the chemical equation, the statistical software package S-PLUS was used. This powerful professional package contains many functions that we used for our computations. Since an analytical solution does not exist, a minimization function that is built into the program is used. This function is specifically prepared to deal with the minimization of non-linear equations. To use this function, the equation to be minimized, the starting points for the parameters and the data must be entered. Limits to the search range for each parameter may also be entered which is useful as we need to force the parameters to be non-negative, as required by the chemical and physical properties of the equation. S-PLUS also was used to generate plots of the data with the curve of the parameter estimates, as well as a plot of the errors (a plot of the distance of from the curve to each data point, also called a residual plot.) Parameter estimates were found for the first two thirds of the data set. However, we sought to implement a program that could be used by Dr. Iraci in the future, so a program in S-PLUS would not be feasible to her. MatLab and Maple were available to her, so a program finding the parameter estimates, confidence intervals, and control charts were coded into MatLab.

Methods in MatLab

Conclusion

Though the problem of finding parameter estimates for Dr. Iraci's data was more difficult than first suspected, an implementation that suits her needs was created. For each set of data, estimates for the parameters of her equation were found using a minimization function in MatLab. Using the 'bootstrapping' method, confidence intervals were also created in which we are 95% confident the true values of the parameters lie. Another program in MatLab was also created to display control charts, with any out-of-control points that may be present in the data.

One complication that must be remembered is that the equation that we were minimizing is only an estimation to the differential equation that exactly represents the reaction that is taking place. Due to this fact, the estimating equation is only valid after 36 seconds and when the estimated value of k must keep the following inequality, $(kt)^{1/2} < 1$.

Add a conclusion here

Problem Section

In some special cases we found the best estimate of k to be zero. In these cases the non-linear equation becomes a linear equation with an analytic solution to the confidence interval. A 95% confidence interval can be constructed with

$$\bar{f}(t) \pm z_{\alpha/2} \sqrt{\text{MSE} \left(\frac{1}{n} + \frac{(x - \bar{x})^2}{\text{SXX}} \right)} \quad \text{where } \text{SXX} = \sum_{i=1}^n (x_i - \bar{x})^2 \text{ and } x = \sqrt{t}$$

Instead of using the $t_{\alpha/2}$, we use $Z_{\alpha/2}$ since n is large. In the case of estimating 95% confidence intervals $Z_{\alpha/2}$ is approximated to 1.96. An estimated confidence interval is made for only β_0 and H; it is not possible to make a confidence interval for k in this special case. So, we also opted to use the bootstrapping method for estimating in the cases for $k = 0$; with this method we obtain confidence intervals for all three parameters, β_0 , H, and k.

Appendix A: Program User's Guide

Overview

The *fit* program runs in the MatLab command window environment. It takes a prepared data set as input and provides estimates and confidence intervals for the parameters β_0 (intercept), H , and k . The program creates output graphic files which show the resulting fitted curve against the data set and provides other statistical information about the fit.

Expected Directory Structure

Several program and data files are used or created during a *fit* program run. The *fit* program and this document assume the following directory structure will be used to contain the various files:

```
Root
Root\RawData
Root\PrepData
Root\Plots
Root\PsFiles
```

The name “Root” is used here as a placeholder for the top level or root of the directory structure. Any convenient name may be used in its place. The Root directory should contain only the MatLab function (.M) files:

```
Fit.m
SAMEst.m
ConfIn95.m
ShowFit.m
```

This document assumes that the raw data sets are stored in the Root\RawData directory. These files typically have names like b12run11, where the first letter represents the month, the following digits represent the day-of-month, and the ending digits represent the run number on that day.

Prepared data sets (see Preparing a Data Set, below) should be stored in the Root\PrepData directory. Prepared data sets must have the MatLab workspace file extension (.mat). It is suggested that the prepared data files use the same name as their corresponding raw data files with the .mat extension (e.g., b12run11.mat).

The *fit* program creates two output files for each data set. The first output file contains the graphical output in MatLab figure (.fig) file format. The program will create this file in the Root\Plots directory, using the input file name with the .fig extension

(e.g., `Plots\b12run11.fig`). This file is not portable outside the MatLab environment, but may be opened from the MatLab work environment for later viewing.

The second output file contains the graphical output in Postscript Level 2 Color Print file format. The program will create this file in the `Root\PsfFiles` directory, using the input file name with the `.ps` extension (e.g., `PsfFiles\b12run11.ps`). This output file format was chosen because it is widely used and thus fairly portable.

Preparing a Data Set

Before a raw data set can be analyzed, it must be stored in a MatLab workspace (`.mat`) file. The MatLab program has an “import wizard” that makes it easy to convert the raw data into the required format. The following paragraphs describe the conversion process. For purposes of illustration, the name `b12run11` will continue to be used for the raw data set.

The raw data set includes two parameters \bar{c} (the mean thermal velocity of the gas) and D (the diffusion coefficient of the liquid), and a number of tuples consisting of a sample time and a corresponding gas uptake measurement γ . Typically, the tuples are stored in an ASCII text file, with a pair of column headers and each tuple on a separate line:

seconds	gamma
21.86	7.864E-03
31.52	7.641E-03
41.41	6.526E-03
53.38	6.068E-03
65.52	5.727E-03
.

The first step in the conversion process is to import the tuples into a MatLab workspace file. Start the MatLab program. Use the text selection box labeled “Current Directory” on the top tool bar to make the `Root` directory (see “Expected Directory Structure”, above) the current working directory. Next, open the “File” pull-down menu and select the “Import Data...” option. This will open a new dialog box which allows an input file to be chosen. Select the file `RawData\b12run11`. When the file is selected, the “Import Wizard” dialog box will open.

The Import Wizard’s initial box shows what the wizard has determined about the data set. Typically, it will show the data loaded correctly in a spreadsheet-like form, i.e., the first datum is in row 1, column 1, and the remaining data are in their correct respective positions. In rare instances it may be necessary to give the wizard a hint about what sort of delimiting character is used (tabs, commas, etc.) or the number of rows used for

column headers. Once the data is shown in its expected form, the “Next > “ button should be used to proceed to the next step.

The second “Import Wizard” dialog box presents choices for how the data will be imported. Choose the button marked “Create vectors from each column using column names”. This will cause two variables to appear in the variable list box, one called “seconds” and one called “gamma”. The variables will be vectors with the same size (e.g., 73 x 1). Click on the “Finish” button to complete the tuple conversion process.

At this point, the “Import Wizard” box will close, and the MatLab command window will again be the active window. The command window will contain this notification:

Import Wizard created variables in the current workspace.

From the >> command prompt, type these commands:

```
>> t = seconds;  
>> clear seconds;  
>> g = gamma;  
>> clear gamma;
```

These commands rename the “seconds” vector to “t” and the “gamma” vector to “g”. These are the names that the *fit* program expects to find in the input workspace file. (This step can be avoided by using the column headings “t” and “g” rather than “seconds” and “gamma” in the raw data text file.)

Next, the values for \bar{c} (the mean thermal velocity of the gas) and D (the diffusion coefficient of the liquid) need to be given. The *fit* program expects the variable “c” to represent \bar{c} and “D” to represent D . The values are simply entered at the command prompt. Exponential format may be used if desired:

```
>> c = 25400;  
>> D = 8.31e-8;
```

Now the workspace is complete. This command saves the workspace to the appropriate place on disk:

```
>> save( 'PrepData\b12run11' );
```

The prepared data set (consisting of tuple-vectors t and g, and constants c and D) is now saved in the file `Root\PrepData\b12run11.mat`. Once this process has been

completed for a given raw data set, it should not need to be repeated. The prepared data set is used for all subsequent MatLab operations.

Obtaining Parameter Estimates

Parameter estimates may be obtained for a prepared data file by using this command:

```
>> fit( 'PrepData\b12run11', 36 );
```

The first parameter is the name of the prepared data file to use for input. The second parameter is an optional lower cutoff limit for the data. If the parameter is N , all the tuples in the data set with a sample time less than N will be ignored.

Estimating the parameters for a data set can take a lot of computation. A crude run-time estimate for a PC type system with a 1.3GHz Pentium processor is one second of processing time for every five tuples in the data set. The parameter k determines the nonlinearity of the estimate, which has a large effect on the computation requirements. When $k = 0$ the estimates tend to complete much faster. The measured run times in Table 1 were obtained on a Dell Inspiron 8200 laptop system with a 1.3GHz Pentium III processor and 256MB of system memory.

Data Set	Number of Points Used for Estimate	Parameter k	Run Time (Seconds)
b11run3	73	1.81e-002	15
b11run4	97	1.89e-003	13
c17run8	151	2.30e-003	33
c17run9	310	0.00	14
e11r8only	1035	0.00	23
x28run2	436	0.00	28

Table 1
Some Typical Program Execution Times

Interpreting the Program's Output

The fit program creates two graphic output pages. The first page depicts the data set, the fitted curve, the 95% confidence limit curves, the residual pattern, and the actual fitted parameters. The second page shows which (if any) points in the data set are suspect for various statistical reasons.

Appendix B: Program Theory of Operation

Overview

The *fit* program runs in the MatLab command window environment. It takes a prepared data set as input and provides estimates and confidence intervals for the parameters β_0 (intercept), H , and k . The program creates output graphic files which show the resulting fitted curve against the data set and provides other statistical information about the fit.

Description of the Problem to be Solved

An experimental run creates a raw data set containing of a number of tuples. Each tuple consists of a sample time t and an associated gas uptake ratio measurement γ . The tuples are expected to fit the equation:

$$\frac{1}{\gamma} = \frac{1}{\Gamma_g} + \frac{1}{\alpha} + \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right) \quad (1)$$

The program attempts to find the best curve in the form of equation (1) for a given data set. The measurements in the data set contain a fair amount of noise, which is assumed is normally distributed around the “true” data. Based on this assumption, the best curve is the one that has the smallest sum of squared errors (SSE), where “error” means the difference between the measured value and the fitted curve. It turns out that finding the parameters for the best curve is equivalent to finding the most likely values of the parameters for the data set. Finding the most likely values of the parameters for the data set is the actual problem to be solved.

Not all of the parameters in equation (1) need to be found. The parameters in the term $\frac{1}{\Gamma_g} + \frac{1}{\alpha}$ can not be individually discovered from the data, so they are lumped together into an intercept term, β_0 . Parameters t and γ are measured. Parameters R , T , \bar{c} , and D are presumed to be known and constant throughout the experiment. This leaves three unknowns in equation (1): β_0 (the combined intercept term), H , and k .

Method for Finding the Solution

Equation (1) is nonlinear. However, if parameter k is zero, then the equation takes on the form:

$$\frac{1}{\gamma} = \frac{1}{\Gamma} + \frac{1}{\alpha} + \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right) \sqrt{t} \quad (2)$$

If the substitutions $y = \frac{1}{\gamma}$, $\beta_0 = \left(\frac{1}{\Gamma} + \frac{1}{\alpha} \right)$, $\beta_1 = \left(\frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \right)$, and $T = \sqrt{t}$ are made, equation (2) becomes linear:

$$y = \beta_0 + \beta_1 T \quad (3)$$

The “least-squares” method provides an analytic solution to the problem of finding the best fit for equation (3) against a data set. Since k is typically small, and is sometimes zero, assuming $k=0$ and fitting equation (3) to the data provides a reasonable and easily-obtained starting estimate for parameters β_0 and β_1 .

If parameter k is zero, the problem is solved. However, it is necessary to determine whether k is in fact zero before concluding that the solution is in hand.

If k is nonzero, equation (3) must be modified slightly, to

$$y = \beta_0 + \beta_1 U(t, k) \quad (4)$$

where

$$U(t, k) = \frac{1}{t^{-1/2} + \sqrt{\pi k}}$$

At this point it is useful to introduce the error function

$$SSE(\beta_0, \beta_1, k) = \sum_{i=1}^n \left(\frac{1}{\gamma_i} - \beta_0 - \beta_1 U(t_i, k) \right)^2 \quad (5)$$

This function provides the sum of the squared errors (SSE) for a given curve (parameterized by β_0 , β_1 , and k) and a given data set (parameterized by the set of n tuples (γ_i, t_i)). The problem of finding the best curve through the data is equivalent to the problem of finding the parameter set $\{ \beta_0, \beta_1, k \}$ that results in the minimum value for the error function.

The error function is known to have only one minimum. It is easy to find the value for $SSE(\beta_0, \beta_1, 0)$, using the values β_0 and β_1 found in the linear estimate. The problem is to find a set $\{ \beta'_0, \beta'_1, k' \}$ such that $SSE(\beta'_0, \beta'_1, k')$ is the minimum value of the error

function and is less than $SSE(\beta_0, \beta_1, 0)$. If such a set can be found, then k is shown to be nonzero and the set $\{\beta'_0, \beta'_1, k'\}$ parameterizes the best curve through the data. If no set $\{\beta'_0, \beta'_1, k'\}$ can be found such that $SSE(\beta'_0, \beta'_1, k')$ is less than $SSE(\beta_0, \beta_1, 0)$, then it is presumed that k is zero, and the set $\{\beta_0, \beta_1, 0\}$ parameterizes the best line through the data..

Algorithm for Finding the Solution

Appendix C: Out-of-Control Signals for x-bar Charts of Data Sets

The purpose of a control chart is to monitor the quality of a process of interest. By means of examining a chart of an averaged groups of points, the probability of having false indications of irregularity in the process is reduced. The following tests are used in our control chart program to identify special cause variations which detect for signals of instability that are possibly beyond the random noise of the experiment data sets. The standard deviation (σ) of the data set is the calculated Mean Square of the Errors(MSE), which is computed by dividing the SSE by the number of data points minus 3. If the residuals are grouped and then averaged to make an x-bar control chart, the standard deviation is σ/\sqrt{n} .

Test A: One point beyond the three standard deviation level.

A point outside $\pm 3\sigma/\sqrt{n}$ is unlikely and may be a signal of special cause variation in the process. The probability of such a point being a random noise in the data set is only less than 0.003 (by the empirical rule—99.7% lie within 3 standard deviations). Thus, these points may be evidences of instability in the experiment within that time interval.

Test B: Run of nine points on one side of the residual mean

A run of nine points having positive residuals or a run of nine points having negative residuals can be a signal of unsteadiness in the experiment. In a random distribution, the probability of nine consecutive points above the centerline is $(1/2)^9 = 0.002$ (by the multiplication rule) and nine consecutive points below is also 0.002. Hence, the probability of a run of either type being a false signal is only about 0.004, or <1%.

Test C: Two out of three points beyond two standard deviations

At least two out of three consecutive points falling either above or below two the standard deviation from the centerline may not only be random noise. The chance of producing an x-bar larger than $+2\sigma/\sqrt{n}$ is about 0.025 (by the empirical rule—that's half of the 0.05 left over from the 95%) in a process that is in control. The probability of the one of the next two samples being beyond this level, bringing a false signal, can be calculated by the binomial formula $P(X \geq 2)$ which yields to approximately 0.002. The probability of the next three points giving a false signal on the low side is the same. Thus, the overall probability of a false signal is about 0.004, or 0.4%

Seeing one or two signals on a control chart may not signify any abnormality in the process. Since a Normal distribution has a 0.3% chance of having type A signals (3 in 1000) and 0.4% chance of type B and type C signals (an odds of 4 in 1000 each). We would expect to see fewer signals in smaller data sets and more in larger data sets. To determine the statistical distribution of out-of-control signals, we generated 1000 random data sets of various size (50 to 1500) and tallied the number of indications of each test (distTable.xls sheet 1 is the cumulative count, sheet 2,3,4 are of test A,B,C signals

respectively). The distribution table shows the number of data sets of size corresponding to the row label having the number of out-of-control signals corresponding to the column label. The shaded portion of each row marks the $\leq 1\%$ tail of 1000 data sets generated in our simulation having the number of signals corresponding to the column label. In other words, shaded area of the table implies that the process is out-of-control.

It must be noted that these tables should be checked in order. This is needed because the algorithm counting the out-of-control signals are checked in order and double counting is not possible. Therefore, the checks should proceed from Test A, to B and then to C.

The following tables include the random Normal distribution of the possible out-of-control points from runs of various sizes. 1000 runs were used to make these tables. The grayed out portion of the four following charts represents when the data is likely out-of-control (set at the 99% confidence level). If a data set of a certain size contains a number of out of that is in the gray area, then that data set should be scrutinized further to determine if the data should be discarded, truncated or otherwise modified to remove the out-of-control data. Table C1 contains the distribution of the sum of all of the different Test, while Table C2, Table C3, and Table C4 contain the distributions of the different Tests, A, B and C respectively.

Table C1. Distribution of Out-of-control Signals in Data Sets of Various Sizes

1000 random data sets of different sizes are generated to find the statistical distribution of out-of-control signals.

Total number of Out-of-Control Signals

Size	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
50	706	246	45	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	545	313	111	27	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	392	341	184	71	10	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
200	287	338	234	91	36	12	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
250	204	303	250	155	59	22	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	143	278	271	174	81	43	7	1	2	0	0	0	0	0	0	0	0	0	0	0	0
350	111	227	266	200	110	54	19	10	3	0	0	0	0	0	0	0	0	0	0	0	0
400	75	181	257	212	158	68	31	12	5	1	0	0	0	0	0	0	0	0	0	0	0
450	58	145	199	242	190	86	45	25	5	5	0	0	0	0	0	0	0	0	0	0	0
500	43	131	205	233	181	97	64	33	10	2	1	0	0	0	0	0	0	0	0	0	0
550	28	120	178	232	166	144	68	40	15	5	3	0	1	0	0	0	0	0	0	0	0
600	20	75	163	206	191	157	111	42	23	9	3	0	0	0	0	0	0	0	0	0	0
650	13	60	123	189	218	152	116	73	36	14	3	3	0	0	0	0	0	0	0	0	0
700	10	62	107	166	176	167	127	84	57	25	9	5	3	2	0	0	0	0	0	0	0
750	6	42	105	148	160	177	135	105	64	34	11	7	4	0	2	0	0	0	0	0	0
800	5	26	84	133	154	185	149	114	68	44	20	11	4	3	0	0	0	0	0	0	0
850	4	20	70	125	134	186	143	122	82	53	36	18	6	1	0	0	0	0	0	0	0
900	2	11	51	91	135	176	152	144	105	62	32	26	5	4	4	0	0	0	0	0	0
950	2	3	30	81	129	154	176	158	119	64	36	24	17	4	2	1	0	0	0	0	0
1000	1	9	32	65	114	149	153	145	107	89	63	39	18	8	5	3	0	0	0	0	0
1100	0	4	22	47	86	121	160	157	133	97	60	49	31	18	9	2	0	0	0	0	0
1200	0	4	15	31	56	107	132	133	128	122	101	74	41	21	19	9	6	1	0	0	0
1300	0	2	7	22	57	60	119	123	150	136	99	83	60	44	22	9	4	1	1	1	0
1400	0	0	3	15	36	70	107	105	109	119	117	95	84	48	40	30	13	5	2	1	1
1500	0	1	5	9	24	36	81	90	110	150	108	96	91	73	57	25	18	13	3	6	4

First Column: The size of a data set (the number of data points)
First Row: The number of out-of-control signals in one data set

The value in each cell represents the number of data sets of the size corresponding to the row label having the number of out-of-control signals corresponding to the column label.

Table C2. Distribution of points beyond the three standard deviation level

Total number of Out-of-Control Signals																					
Size	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
50	850	145	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	760	207	31	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	662	257	70	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
200	582	320	84	13	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
250	496	336	127	36	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	428	372	158	30	11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
350	369	373	188	57	11	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
400	339	378	162	86	28	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
450	289	362	212	105	25	5	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	268	349	234	112	28	6	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0
550	232	357	233	111	48	14	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0
600	202	334	256	137	52	11	6	2	0	0	0	0	0	0	0	0	0	0	0	0	0
650	166	314	279	142	71	25	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
700	142	287	283	175	77	28	3	2	3	0	0	0	0	0	0	0	0	0	0	0	0
750	131	272	284	182	82	33	9	6	0	0	0	0	1	0	0	0	0	0	0	0	0
800	102	246	263	209	111	47	15	2	5	0	0	0	0	0	0	0	0	0	0	0	0
850	102	229	244	195	137	60	23	9	0	1	0	0	0	0	0	0	0	0	0	0	0
900	85	216	248	210	152	62	21	5	1	0	0	0	0	0	0	0	0	0	0	0	0
950	69	202	248	219	163	60	29	8	2	0	0	0	0	0	0	0	0	0	0	0	0
1000	63	187	222	232	143	90	43	10	7	3	0	0	0	0	0	0	0	0	0	0	0
1100	39	165	231	223	177	86	43	21	10	4	1	0	0	0	0	0	0	0	0	0	0
1200	39	111	216	212	197	111	58	37	13	6	3	0	0	0	0	0	0	0	0	0	0
1300	36	102	176	203	191	145	81	45	17	0	4	0	0	0	0	0	0	0	0	0	0
1400	24	91	177	205	184	131	95	46	28	14	4	0	1	0	0	0	0	0	0	0	0
1500	21	69	140	186	200	151	101	62	45	19	4	2	0	0	0	0	0	0	0	0	0

Table C3. Distribution of a run of nine points on one side of the centerline

Size	Number of Test B Signals																				
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
50	916	83	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	838	151	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	742	223	29	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
200	661	282	51	5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
250	616	300	68	14	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	547	343	94	14	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
350	489	374	107	25	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
400	463	344	154	32	6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
450	397	370	167	50	14	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	367	361	189	63	18	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
550	350	347	201	85	16	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
600	298	378	220	84	16	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
650	264	374	235	93	25	7	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
700	234	357	235	110	49	14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	221	331	247	134	43	20	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0
800	213	325	268	134	43	16	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
850	202	308	272	145	50	21	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
900	147	318	269	162	66	31	6	1	0	0	0	0	0	0	0	0	0	0	0	0	0
950	134	299	269	160	90	36	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1000	130	292	258	191	79	33	12	5	0	0	0	0	0	0	0	0	0	0	0	0	0
1100	105	282	245	191	114	48	8	6	1	0	0	0	0	0	0	0	0	0	0	0	0
1200	102	237	246	213	118	50	25	9	0	0	0	0	0	0	0	0	0	0	0	0	0
1300	72	215	271	202	127	69	33	11	0	0	0	0	0	0	0	0	0	0	0	0	0
1400	51	178	243	217	156	83	45	21	5	1	0	0	0	0	0	0	0	0	0	0	0
1500	42	135	229	251	173	99	43	16	9	2	1	0	0	0	0	0	0	0	0	0	0

Table C4. Distribution of two out of three points beyond 2 standard deviations

		Number of Test C Signals																				
Size		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
50	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
100	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
200	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
250	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
350	999	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
400	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
450	999	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
550	999	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
600	999	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
650	999	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
700	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	999	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
800	999	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
850	999	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
900	999	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
950	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1000	999	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1100	1000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1200	999	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1300	999	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1400	999	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1500	999	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Use of control chart program

The control chart program draws a plot using the time and residual vectors (vector `t` and `u` from Joe's program) with special markers to indicate out-of-control signals. Type A signals are denoted by a red circle. The ninth point of Type B detection is marked by a purple triangle. The second point that is beyond the two standard deviation of each group of Type C signal is indicated by a green asterisk.

The program outputs the time at which signals of each type are shown on the control chart to the MATLAB workspace. The user needs to specify the number of points to be averaging over (If a plot of the original residuals is desired, set `averageSize=1`).

For example, to obtain a plot averaging every 5 points, type the following commands:

```
>> averageSize = 5;  
>> controlChart
```

Appendix D: Maximum Likelihood Estimation

To find the Maximum Likelihood Estimation (MLE) parameter estimates we must maximize the MLE function using the error function that was found.

Recall that $1/\gamma_i = f(t_i) + \varepsilon_i$

$$\text{where } f(t) = \beta_0 + \beta_1 \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right)$$

Therefore, $\varepsilon_i = 1/\gamma_i - f(t_i)$

Also recall that the errors are from a Normal distribution, $\varepsilon_i \sim N(0, \sigma^2)$.

In this case then, the MLE function, G, has the following form.

$$\begin{aligned} G &= \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma}} \exp\left(\frac{-1}{2\sigma^2} (\varepsilon_i)^2\right) \\ &= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(\frac{-1}{2\sigma^2} \sum_{i=1}^n \varepsilon_i^2\right) \\ &= \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(\frac{-1}{2\sigma^2} \sum (1/\gamma_i - f(t_i))^2\right) \end{aligned}$$

As it can be seen, to maximize this function the exponential term must be minimized because it is negative. This is the same as minimizing $L = \sum (1/\gamma_i - f(t_i))^2$, which is the sum of the squares of the errors (SSE).

$$\text{SSE} = \sum (1/\gamma_i - f(t_i))^2 = \sum_{i=1}^n \left(\frac{1}{\gamma_i} - \beta_0 - \frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right) \right)^2$$

Therefore, when the errors are Normally distributed, the MLE parameter estimates can be found by minimizing the sum of squared errors (SSE), which is the method that we implemented.

Appendix E: Positive Definite Matrices

In minimizing the error function associated with our chemical equation, it is desired to find the global minimum, as opposed to a local minima. It can be proved that only a global minimum exists if the Hessian matrix of second derivatives of the error function is positive definite (Strang). One method to prove that a matrix is positive definite is to show that the determinants of its principle minor matrices are all positive. Therefore the following determinates must all be positive for our error function to have a unique global minimum.

$$A_1 = \left[\frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} \right], \quad A_2 = \begin{bmatrix} \frac{\partial^2 L}{\partial B_0 \partial B_0} & \frac{\partial^2 L}{\partial B_0 \partial H} \\ \frac{\partial^2 L}{\partial H \partial B_0} & \frac{\partial^2 L}{\partial H \partial H} \end{bmatrix}, \quad A_3 = \begin{bmatrix} \frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_0 \partial H} & \frac{\partial^2 L}{\partial \beta_0 \partial k} \\ \frac{\partial^2 L}{\partial H \partial \beta_0} & \frac{\partial^2 L}{\partial H \partial H} & \frac{\partial^2 L}{\partial H \partial k} \\ \frac{\partial^2 L}{\partial k \partial \beta_0} & \frac{\partial^2 L}{\partial k \partial H} & \frac{\partial^2 L}{\partial k \partial k} \end{bmatrix}.$$

$$\text{Recall that } L = \sum_{i=1}^n \left(\frac{1}{\gamma_i} - B_0 - \frac{\bar{c}\sqrt{\pi}}{4RTH\sqrt{D}} \left(\frac{1}{t^{-1/2} + \sqrt{\pi k}} \right) \right)^2$$

First, we worked on the 1x1 matrix to determine that it is positive.

$$\text{Det} \left[\frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} \right] = 2 \sum_{i=1}^n 1 = 2n$$

2n is always positive since n is the number of data points collected, which can never be negative. So, this determinant is always positive.

Second, we tried to find the determinant for the 2x2 matrix. We found it easier to start with a substitution. The derivates of L were taken with respect to β_1 instead of H.

$$\text{Det} \begin{bmatrix} \frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_0 \partial \beta_1} \\ \frac{\partial^2 L}{\partial \beta_1 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_1 \partial \beta_1} \end{bmatrix} = (2n) \left(2 \sum_{i=1}^n \frac{t_i}{(1 + \sqrt{\pi k t_i})^2} \right) - \left(2 \sum_{i=1}^n \left(\frac{\sqrt{t_i}}{1 + \sqrt{\pi k t_i}} \right) \right)^2$$

$$\begin{aligned}
&= 4n \sum_{i=1}^n A_i^2 - 4 \left(\sum_{i=1}^n A_i \right)^2, \text{ where } A_i = \frac{\sqrt{t_i}}{1 + \sqrt{\pi k t_i}} \\
&= 4 \left[n \sum A_i^2 - \left(\sum A_i \right)^2 \right] \\
&= 4 \left[\sum A_i^2 - \frac{1}{n} \left(\sum A_i \right)^2 \right] \\
&= \frac{4}{n-1} \left[\sum A_i^2 - \frac{1}{n} \left(\sum A_i \right)^2 \right] \\
&= \frac{4 \sum (A_i - \bar{A})^2}{n-1} = 4s^2 \geq 0 \text{ by the Cauchy-Shwarz inequality,}
\end{aligned}$$

with the equality only when $A_i = \bar{A} \forall i$, which can never be true in our data (Rudin). Therefore, our determinant of the second matrix is also positive.

The third and last matrix was much more complex and we were not able to finish with the analysis of this last determinant equation.

$$\text{Det} \begin{bmatrix} \frac{\partial^2 L}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 L}{\partial \beta_0 \partial H} & \frac{\partial^2 L}{\partial \beta_0 \partial k} \\ \frac{\partial^2 L}{\partial H \partial \beta_0} & \frac{\partial^2 L}{\partial H \partial H} & \frac{\partial^2 L}{\partial H \partial k} \\ \frac{\partial^2 L}{\partial k \partial \beta_0} & \frac{\partial^2 L}{\partial k \partial H} & \frac{\partial^2 L}{\partial k \partial k} \end{bmatrix}$$

Though we have not yet finished the analysis of this following determinant equation, we suspect it to be positive. Any input would be appreciated.

$$\begin{aligned}
&= - \frac{2 Z^2 \pi \left(\sum_{i=1}^n A_i^2 \right)^3}{k} + 2n \left(\sum_{i=1}^n A_i^2 \right) Z \pi \left(\sum_{i=1}^n \left(\right. \right. \\
&\quad \left. \left. -2 \frac{A_i^3 (\pi k)^{(3/2)}}{g_i} - \frac{A_i^2 \pi k}{g_i} + 2 A_i^3 B (\pi k)^{(3/2)} + A_i^2 B \pi k + \frac{3 Z (\pi k)^{(3/2)}}{y_i^4} + Z A_i^3 \pi k \right) \right)
\end{aligned}$$

$$\left. \right) / (k(\pi k)^{(3/2)}) - \frac{2n\pi \left(\sum_{i=1}^n \left(-\frac{A_i^2}{g_i} + A_i^2 B + 2ZA_i^3 \right) \right)^2}{k} - 2 \left(\sum_{i=1}^n A_i \right)^2 Z \pi \left(\sum_{i=1}^n \left(\right. \right.$$

$$\left. \left. -2 \frac{A_i^3 (\pi k)^{(3/2)}}{g_i} - \frac{A_i^2 \pi k}{g_i} + 2A_i^3 B (\pi k)^{(3/2)} + A_i^2 B \pi k + \frac{3Z (\pi k)^{(3/2)}}{4} + Z A_i^3 \pi k \right) \right)$$

$$\left. \right) / (k(\pi k)^{(3/2)}) + \frac{4 \left(\sum_{i=1}^n A_i \right) Z \pi \left(\sum_{i=1}^n A_i^2 \right) \left(\sum_{i=1}^n \left(-\frac{A_i}{g_i} + A_i^2 B + 2ZA_i^3 \right) \right)}{k}$$

and

$$A_i = \frac{\sqrt{t_i}}{1 + \sqrt{\pi k t_i}}$$

where $B = \beta_0$, $Z = \beta_1$,

Bibliography

Text Resources

Bates, Douglas M. and Donald G. Watts. Nonlinear Regression Analysis and Its Applications. Wiley, 1988.

Chernick, Michael R. Bootstrap Methods: A Practitioner's Guide. Wiley, September 1999.

Danckwerts, P. V. Gas-liquid Reactions. New York: McGraw-Hill Book Co., 1970.

Davison, A.C. and D.V. Hinkley. Bootstrap Methods and Their Application. Cambridge University Press, 1997.

Finlayson-Pitts, B.J. and J.N. Pitts Jr. Chemistry of the Upper and Lower Atmosphere. Academic Press, 2000, pgs.156 – 167.

Harville, David A. Matrix Algebra from a Statistician's Perspective. New York: Springer, 1997.

Huet, S., A. Bouvier, M.-A. Gruet, and E. Joliet. Statistical Tools for Nonlinear Regression: A Practical Guide with S-PLUS Examples. New York: Springer-Verlag, 1996.

Lehmann, E.L. Theory of Point Estimation. New York: Chapman & Hall, 1991.

Montgomery, Douglas C. and Elizabeth A. Peck Introduction to Linear Regression Analysis. New York: John Wiley & Sons, Inc., 1992.

Montgomery, Douglas C. Introduction to Statistical Quality Control. New York: John Wiley & Sons, Inc., 2000.

Rudin, Walter. Principles of Mathematical Analysis. New York: McGraw-Hill, Inc., 1976.

Strang, Gilbert. Linear Algebra and its Applications. San Diego: Harcourt Brace Jovanovich, Inc., 1988.

Software

S-Plus 6: Insightful Corporation of Seattle, Washington, July 2001.

Matlab:

JmpIn 4: SAS Institute Inc. of the USA, 2001.