5-1-2014

# Implementation and Application of the Curds and Whey Algorithm to Regression Problems

John Kidd
*Utah State University*

# IMPLEMENTATION AND APPLICATION OF THE CURDS AND WHEY ALGORITHM TO REGRESSION PROBLEMS

by

John Kidd

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Statistics

Approved:

| | |
|---|---|
| Dr. Richard Cutler<br>Major Professor | Dr. Adele Cutler<br>Committee Member |
| Dr. Christopher Corcoran<br>Committee Member | Dr. Mark McLellan<br>Vice President for Research<br>and Dean of the School of<br>Graduate Studies |

UTAH STATE UNIVERSITY
Logan, Utah

2014

# ABSTRACT

Implementation and Application of the Curds and Whey
Algorithm to Regression Problems

by

John C. Kidd, Master of Science
Utah State University, 2014

Major Professor: Dr. Richard Cutler
Department: Mathematics and Statistics

A common multivariate statistical problem is the prediction of two or more response variables using two or more predictor variables. The simplest model for this situation is the multivariate linear regression model. The standard least squares estimation for this model involves regressing each response variable separately on all the predictor variables. Breiman and Friedman found a way to take advantage of correlations among the response variables to increase the predictive accuracy for each of the response variables with an algorithm they called *Curds and Whey*. In this report, I describe an implementation of the Curds and Whey algorithm in the R language and environment for statistical computing, apply the algorithm to some simulated and real data sets, and discuss the R package I developed for Curds and Whey.

(62 pages)

# PUBLIC ABSTRACT

Implementation and Application of the Curds and Whey Algorithm to Regression Problems

John C. Kidd

A common statistical problem is trying to predict two or more variables using a set of predictor variables. The simplest model for this situation is called multivariate linear regression. This method uses each set of predictor variables to predict each of the response variables separately. This approach seems counter-intuitive as any possible relationship between the variables being predicted is ignored.

Breiman and Friedman found a way to take advantage of relationships among the response variables to increase the accuracy of the predictions for each of the predicted variables with an algorithm they called *Curds and Whey*. It uses other statistical techniques to extract additional information from the variables being predicted to improve predictions on those same variables.

In this report, I describe an implementation of the Curds and Whey algorithm in a statistical software package called R, apply the algorithm to some simulated and real data sets, and discuss the R software package I developed for the Curds and Whey algorithm.

# ACKNOWLEDGMENTS

I would like to thank Dr. Richard Cutler for all of his help and guidance while working through this algorithm and developing the code to use it effectively. I would also like to thank my other committee members, Dr. Adele Cutler and Dr. Chris Corcoran, for their input, guidance, and support along this journey as well.

I give special thanks to my family, colleagues, and friends for their support and encouragement along the way. I especially thank my wife, Cassie, for her love, support, and sacrifice that made this possible.

All analyses were performed using the statistical software package R version 3.0.3 (2014-03-06).

John Kidd

# CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# 1. Literature Review

Multiple Linear Regression has been a common statistical tool for a very long time. Karl Pearson (1914) wrote that the idea was introduced by his teacher, Sir Francis Galton. In his work with genetics and biostatistics, Galton is believed to have discovered the regression slope. From this original work, he generalized his work and introduced the concept of multiple regression.

The first believed instance of trying to describe the relationship between a set of multiple responses and multiple predictors was introduced by Harold Hotelling (1936). In a method he termed *Canonical Correlation*, he tried to find linear combinations of response and predictor variables such that the correlation between these was maximized.

The first real attempt to focus on multiple responses (thus, multivariate settings) in regards to performing actual tests was done by T.W. Anderson (1984) in *An Introduction to Multivariate Analysis*.

Though multivariate methods and data sets were becoming more common, no true application into multiple regression was found. Most people simply continued to treat multiple responses as separate responses, and then performed multiple regression upon each of them separately.

Breiman and Friedman (1997) attempted something very new with their *Curds and Whey* algorithm. In this algorithm, they combined the idea of canonical correlation with multiple regression in such a way as to create a truly multivariate form of multiple regression, such that information from each response variable would be included in the prediction of each individual response variable. In order to improve accuracy and the stability of this model, they also included an element of *shrinkage*.

One of the main shrinkage methods, developed by Hoerl and Kennard (1970), is called Ridge Regression. The idea in ridge regression (and all shrinkage applications) is to "shrink" the coefficients used in the regression model. In the model, this helps stabilize the predictions so that they will not change due to small variations, as well as limit the impact of highly correlated predictor variables. *Curds and Whey* desired to use this same concept to improve the predictive accuracy of the Curds and Whey algorithm.

More modern examples and work involving shrinkage include a procedure developed by Tibshirani (1996), called Lasso Regression, takes the same principle as ridge regression, but allows for different approaches to shrinkage such that instead of coefficients simply being minimized, they can be completely forced to zero, thus eliminating them from the model entirely. This allows for simple models, as well as increased accuracy and more stable models.

The current literature on Curds and Whey is quite small, but the concept is being considered and applied to further problems. D'Ambra and Lombardo (1999) extended Curds and Whey to additive spline functions, which is a first step towards developing a Curds and Whey algorithm to handle non-linear problems. Xu et al. (2004) developed a modified partial least squares algorithm that uses the same transfer to canonical coordinates and shrinkage estimates as Curds and Whey.

Curds and Whey has even been applied to some fields not directly related to statistics. Liu et al. (2010) developed a non-negative Curds and Whey algorithm for sparse, non-negative representations of computer images. Overall, we can see from these examples that there is research to be

done using Curds and Whey, and more knowledge of its statistical aility as well as easier implementation will aid in that pursuit.

# 2. Curds and Whey Algorithm

2.1. *Introduction and Background.*

2.1.1. *The Multiple Regression Model.* One of the most widely used statistical methods is *multiple linear regression*, in which a numerical response is modeled as a linear combination of values on two or more numerical *predictor* or *explanatory* variables. Examples include predicting oxygen uptake using fitness and anthropometric measurements on the subjects, insurance profits using industry and economic variables, human mortality rates using measurements of socioeconomic status and air pollution, and species abundances using ecological and climate measurements. The multiple linear regression model may be written as:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i \quad \text{for } i = 1, 2, \ldots, n \tag{1}$$

where $Y_i$ is the value of the response variable for the $i^{\text{th}}$ observation, $x_{i1}, x_{i2}, \ldots, x_{ip}$ are the values of the explanatory variables for the $i^{\text{th}}$ observation, $\varepsilon_i$ is a random error, and $\beta_0, \beta_1, \ldots, \beta_p$ are unknown parameters that must be estimated. Usually it is assumed that the $\varepsilon_i$ are statistically independent, with common mean 0 and variance $\sigma^2$, and are approximately normal in distribution. This model in matrix form may be written as:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} \tag{2}$$

where

$$(3)\quad \mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix} \qquad \mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} \qquad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}.$$

2.1.2. *Least Squares Estimation and Prediction.* Given a set of parameter estimates, $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \ldots \hat{\beta}_p)^T$, one can compute *fitted* or *predicted* values, $\hat{Y}_i$, by substitution into equation (1) and setting the random error term equal to zero. That is,

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2} + \cdots + \hat{\beta}_p x_{ip} \quad \text{for } i = 1, 2, \ldots, n.$$

In matrix form, $\hat{\mathbf{Y}} = \mathbf{X}\hat{\boldsymbol{\beta}}$.

To estimate $\boldsymbol{\beta}$, we may minimize the sum of squared deviations between the observed response variable value, $Y_i$, and the predicted values, $\hat{Y}_i$. That is, we minimize $\text{RSS}(\boldsymbol{\beta}) = \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$ with respect

to $\boldsymbol{\beta}$. Assuming the columns of the matrix $\mathbf{X}$, the predictor variables, are linearly independent, the least squares estimator of $\boldsymbol{\beta}$ has the elegant form

$$\hat{\boldsymbol{\beta}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}. \tag{4}$$

2.1.3. *Shrinkage Estimation.* For the multiple linear regression model when the predictor variables are correlated, a different estimation procedure called *ridge regression*, developed by Hoerl and Kennard (1970), yields more stable parameter estimates (the $\hat{\beta}_j$'s) and smaller prediction error. Following the notation of equation (3), the ridge regression estimate of $\boldsymbol{\beta}$ may be written

$$\hat{\boldsymbol{\beta}}^{(\tau)} = (\mathbf{X}^T\mathbf{X} + \tau\mathbf{I})^{-1}\mathbf{X}^T\mathbf{Y} \tag{5}$$

where $\mathbf{I}$ is the identity matrix with 1's down the diagonal and 0's in all the off-diagonal entries, and $\tau$ is a *shrinkage parameter*. Typically, $\tau$ is estimated by minimizing prediction error or by graphical means. Ridge regression is a *shrinkage estimation procedure* in the sense that as $\tau$ increases, the $\hat{\beta}_j^{(\tau)}$'s decrease in magnitude, sometimes changing sign in the process. As $\tau$ gets very large, all the $\hat{\beta}_j^{(\tau)}$'s will tend to zero.

Other forms of shrinkage have also been shown to provide more stable parameter estimates as well as smaller prediction errors (see James and Stein, 1961; Stone, 1974; Copas, 1983; Massy, 1965; Wold, 1975).

2.1.4. *Canonical Correlation.* Canonical Correlation analysis is a method for characterizing the linear associations among two sets of numerical variables. Let $\mathbf{X}$ be an $n \times p$ matrix with the columns being the measured values of one set of variables, and $\mathbf{Y}$ be an $n \times q$ matrix with the columns being the values on the other set of variables, and assume that $q \leq p$.

Let $V_1$ and $W_1$ be vectors such that $\mathbf{X}V_1$ and $\mathbf{Y}W_1$ maximize the correlation among all linear combinations of variables in $\mathbf{X}$ and $\mathbf{Y}$. This maximal correlation, $c_1$, is the *first canonical correlation.*

Next, $V_2$ and $W_2$ are found so that $\mathbf{X}V_2$ and $\mathbf{Y}W_2$ maximize the correlation among linear combinations of variables in $\mathbf{X}$ and $\mathbf{Y}$, *subject to the constraint that* $V_2 \perp V_1$ *and* $W_2 \perp W_1$ (where $\perp$ indicates they are orthogonal). The correlation, $c_2$, is the *second canonical correlation.* The process is continued, yielding $q$ canonical correlations and vectors $V_1, V_2, ...V_q$, and $W_1, W_2, ..., W_q$.

The vectors $W_1, W_2, ..., W_q$ may be stacked together to create a matrix $\mathbf{T}$ that may be used to transform the variables in $\mathbf{Y}$ into *canonical coordinates*, the coordinate system that yields the canonical correlations.

2.1.5. *The Multivariate Linear Regression Model.* The *multivariate* linear regression model extends the *multiple* linear regression model to predicting two or more response variables using the same suite of predictor variables. We may write the model as:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{B} + \mathbf{E}, \tag{6}$$

where $\mathbf{Y}$ is an $n \times q$ matrix, the columns of which are $q$ response variables. In this model $\mathbf{X}$ is an $n \times (p+1)$ matrix comprising, as columns, $p$ predictor variables and a column of 1's for the intercept term. $\mathbf{E}$ is an $n \times q$ matrix of residual or random error terms, and $\mathbf{B}$ is a $p + 1 \times q$ matrix of coefficients to be estimated. The $k^{\text{th}}$ column of $\mathbf{B}$ is the vector of coefficients for the predictor variables for the $k^{\text{th}}$ response variable.

The least squares estimate of $\mathbf{B}$ may be expressed as:

$$(7)\quad \hat{\mathbf{B}} = \left[\hat{\boldsymbol{\beta}}_1, \hat{\boldsymbol{\beta}}_2, \ldots, \hat{\boldsymbol{\beta}}_q\right] = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} = \left[(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}_1, (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}_2, \ldots, (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}_q,\right]$$

Thus, the $\hat{\beta}_j$'s corresponding to the $k^{\text{th}}$ response variable use only information from the $k^{\text{th}}$ response variable. In situations in which the response variables are highly correlated this result seems counter intuitive, and it is this observation which motivated Breiman and Friedman (1997) to develop an alternative approach. Their *Curds and Whey* algorithm employs elements of canonical correlation and shrinkage estimation to use the relationships among the response variables to enhance the accuracy of predictions for each of the response variables.

2.2. *The Curds and Whey Procedure.* The general idea behind the Curds and Whey algorithm is to take the least squares regressions, and then to modify the predicted values from those regressions by shrinking them using the canonical correlations between the response variables and the predictor variables. Thus, the equation can be thought of as:

$$(8)\qquad \tilde{\mathbf{Y}} = \mathbf{M}\hat{\mathbf{Y}}$$

where $\mathbf{M}$ is the matrix estimated such that $\mathbf{M} = \mathbf{T}^{-1}\mathbf{D}\mathbf{T}$ with $\mathbf{T}$ being a $q \times q$ matrix, where $q$ is the number of response variables one is trying to predict, whose rows are the canonical correlation coefficients of the response variables (2.1.4), and $\mathbf{D}$ is a diagonal matrix where each entry $d_k$ is a function of the canonical correlations and the ratio of predictors to the size of the

data set (2.2.2). In order to search for the best **M**, Breiman and Friedman (1997) suggest two formulas, one that is very simple, and one that produces a general cross-validation estimate for the shrinkage factors.

2.2.1. *Standardizing Data.* When dealing with multivariate data, often variables are measured on different scales. For many procedures, this can lead to one or more predictor variables having a much larger influence on the response than others simply because of its scale. This can make interpretation difficult, as well as cause extreme observations to influence results. Standardizing eliminates scale effects.

Also, due to the shrinkage nature of Curds and Whey (the predicted response values are what the shrinkage is applied to), standardizing the response variables causes the shrinkage to shrink our predicted values towards their mean, rather than 0.

For Curds and Whey, we standardize the response and predictor variables by subtracting the mean value and dividing by the standard deviation.

2.2.2. *Finding D.* There are two ways to estimate the optimal shrinking matrix, **D**. In the simplest case given $r = p/N$, with $N$ being equal to the total number of observations, define the $d_k$'s as:

$$d_k = \frac{c_k^2}{c_k^2 + r(1 - c_k^2)}, i = 1, 2, ..., q \tag{9}$$

with $c_k^2$ being the $i^{th}$ canonical correlation.

This gives improved predictions compared to ordinary least squares, but it does not provide enough shrinkage to be optimal as the correlations are over-

estimated, causing the shrinkage to be underestimated. A second approach, based upon generalized cross-validation sets the $d_k$'s as follows:

$$d_k = \frac{(1-r)(c_k^2 - r)}{(1-r)^2 c_k^2 + r^2(1-c_k^2)}, i = 1, 2, ..., q \tag{10}$$

In some instances, this will result in a $d_k$ that is less than 0. In this case, the $d_k$'s are set to 0.

# 3. The Procedure

The Curds and Whey algorithm follows these steps:

1. Standardize response and predictor variables to have mean 0 and variance 1.
2. Transform $\mathbf{Y}$ to the observed canonical coordinate system, $\mathbf{Y}^* = \mathbf{TY}$.
3. Perform a separate ordinary least squares regression of each of the $Y_k$*'s on all the predictor variables $\mathbf{X}$, obtaining a new variable, say $\hat{Y}_k$*.
4. Separately scale (shrink) each of the $\hat{Y}_k$*'s by the corresponding $d_k$ (10). Or, it can be thought of as $\hat{\mathbf{Y}}^*$. This gives a new set, called $\tilde{\mathbf{Y}}^*$.
5. Transform back to the original $\mathbf{Y}$ coordinate system, $\tilde{\mathbf{Y}} = \mathbf{T}^{-1}\ \tilde{\mathbf{Y}}^*$.

It should be noted that due to the use of canonical correlation, one faces the constraint that $q \leq p$. Thus, if one does find a situation where $q > p$, then one must use a subset of the response variables.

3.0.3. *Extracting Coefficients.* The main purpose of the Curds and Whey algorithm is to improve prediction. Breiman and Friedman (1997) suggest, however, that one can regress the predictor variables to obtain coefficient vectors $\tilde{\beta}_{\mathbf{k}}$ that may be used to facilitate interpretation of the Curds and Whey fit to the data.

3.0.4. *Measuring Improvement.* In order to assess the improvement over ordinary least squares regression achieved by Curds and Whey, 10-fold cross validation can be used to find the mean squared error using both methods. An easy way to compare the methods is to look at the percentage improve-

ment that is seen with Curds and Whey.

3.1. *Simulated Examples.* For each of the simulated examples, we followed the same basic procedures to generate the data. we randomly generated means between $-10$ and 10 for each of the predictor variables. Next, we used the `mvrnorm` function from the MASS package developed by Venables and Ripley (2002) to generate 100 independent realizations of each of the predictor variables (with the realizations having a mean equal to what we generated earlier). In other words, using a sample size of 100 observations. We then generated coefficients for each predictor variable. The realizations and the coefficients were used to generate response variables, with no noise.

At this point, the responses are all independent. To implement some correlation between the response variables, we randomly generated error terms (again using `mvrnorm`) with a mean of 0, and the first-order autoregressive correlation structure,

$$R = \begin{pmatrix} 1 & \rho & \rho^2 & \dots & \rho^{q-1} \\ \rho & 1 & \rho & \dots & \rho^{q-2} \\ \rho^2 & \rho & 1 & \dots & \rho^{q-3} \\ \dots & & & & \\ \rho^{q-1} & \rho^{q-2} & \rho^{q-3} & \dots & 1 \end{pmatrix}$$

where $\rho$ is some chosen correlation level. This structure allowed for correlation to exist within the response variables, but there was strong correlation only between a variable and a few of its "neighbors," or the responses generated near it in the matrix.

For each set of predictor and response variables, these error terms were

added to the response variables. This additional error, sometimes known as noise, is generally just error in our tests. By using the procedure we used to generate these errors, then we can conclude that the errors are directly related to the responses. In this sense, the errors contain information about the responses that is not available through the predictors.

To increase this amount of information, different multiples of this error matrix were added to the responses, generating multiple response matrices. Each of these responses is then used in the Curds and Whey algorithm, and the mean squared error for each response variable measured.

I repeated this process 100 times, generating new data for each repetition to get an average for each response variable's mean squared error with each multiple of the error terms, and then repeated this process 10 times changing the value of $\rho$ from 0.95 to 0.50.

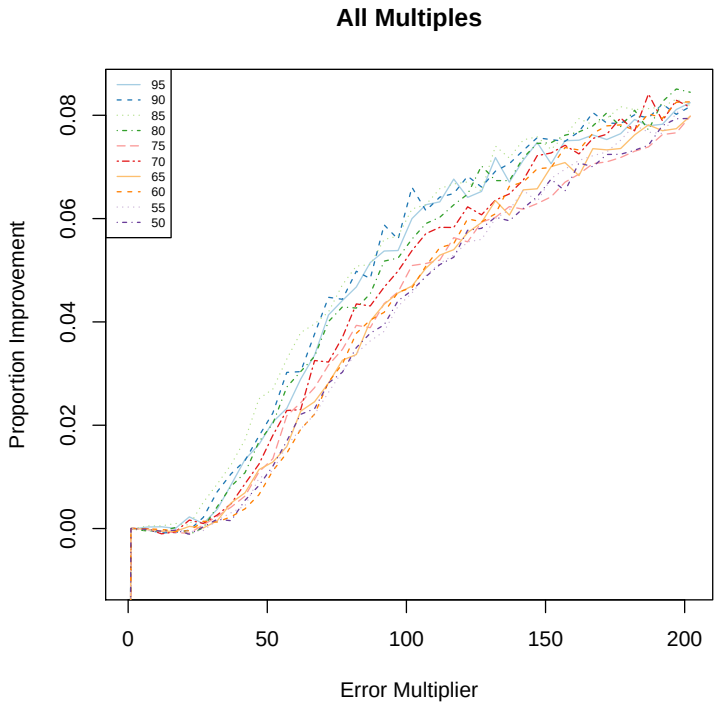
To see how Curds and Whey performs with different data sets, I set up two different simulations. In the first, every predictor variable was used to generate the response variables. In the second, a random sample of the predictor variables was selected to determine which were associated with the response variables, and which were not.

3.1.1. *All Variables Relevant.* With each predictor variable being associated with the response (each predictor variable being used to generate the response), I ran through several different values for $p$ and $q$ to see what in what situations we see the greatest improvement improvement from Curds and Whey. I used

| | | | | | |
|---|---|---|---|---|---|
| p | 10 | 10 | 25 | 50 | 50 |
| q | 5 | 10 | 10 | 25 | 50 |

The main goal here is to see how Curds and Whey performs with small up to larger data sets, and also to see how it does with different balances of predictor and response variables.

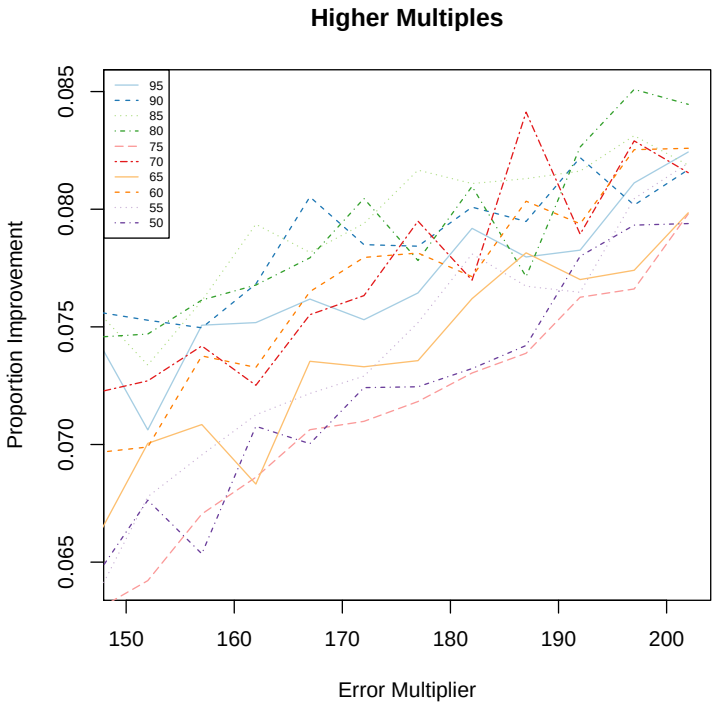
FIGURE 1. *Proportion improvement with $q = 5$ and $p = 10$ across all used correlation factors*




In Figure 1, we can see a general improvement over ordinary least squares as the multiples of the error terms is increased. This translates to the concept that as the information contained by the response variables, we are able to get better predictions with Curds and Whey. This holds steady across all correlations, but we see that generally the higher the correlation, the greater the improvement. There do seem to be some anomalies mixed in, but there

do not seem to be large enough differences among improvements to indicate a different level of improvement at lower correlations. Also, this could happen due to the nature of how we generate the error terms (that the base error terms are reused for each multiple).

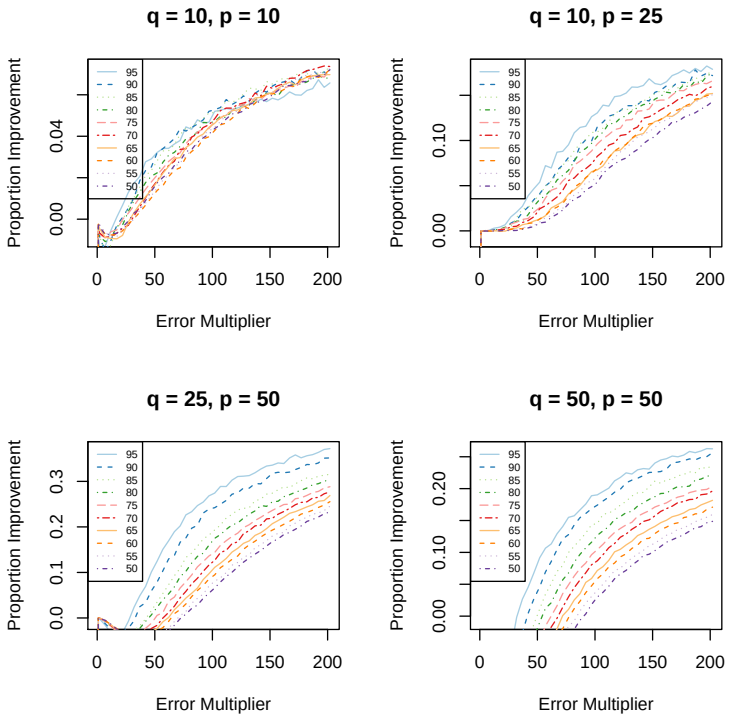
FIGURE 2. *Proportion improvement with $q = 5$ and $p = 10$ across all correlation factors*




Looking closer at the higher multiples of the error (Figure 2), we see a general spread, but still it appears that the higher correlations generally appear to see greater improvement.

Figure 3 shows the averages of the remaining combinations of $q$ and $p$. These each graph is individually printed in the appendix. Some interesting

FIGURE 3. *Proportion improvement across remaining levels*




trends emerge. Firstly, looking at the scales on the y-axis, it appears that the more predictor variables included in the model, the more improvement we see over ordinary least squares. Due to the measurement of error we have used (mean squared error from 10-fold cross validation), this may be indicative that Curds and Whey is less likely to over fit the data, and thus gives us a better fit.

Secondly, with more response variables, we typically see a lower level of improvement over ordinary least squares compared to when there are fewer responses. In Table 1, we see the maximum improvement seen on each level.

Third, with a higher number of predictors (and especially response vari-

TABLE 1
*Maximum improvement values from simulations*

| q | p | Max Improvement |
|---|---|---|
| 5 | 10 | 0.091 |
| 10 | 10 | 0.081 |
| 10 | 25 | 0.191 |
| 25 | 50 | 0.385 |
| 50 | 50 | 0.276 |

ables), even though Curds and Whey achieves a much higher level of improvement, there must be enough noise (the error term multiple in the simulations) for the improvement to be visible. If fact, for low levels of the additional signal or noise (low multiples of the added errors), Curds and Whey performs worse. The higher correlations do require less additional noise, but in both cases there must be some sort of extra information not contained within the predictor variables.

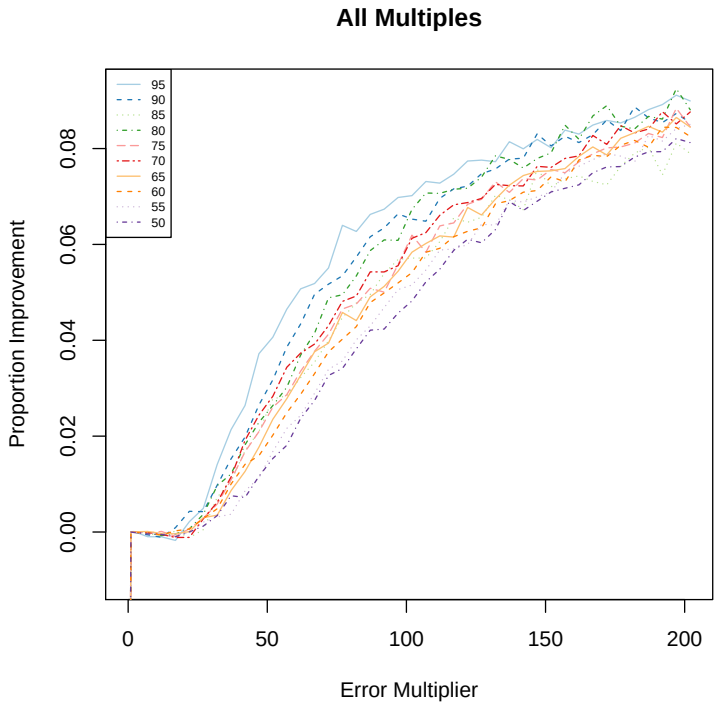
This last point seems very logical. As Curds and Whey is aimed at extracting extra information from the response variables, we will not see any improvement if all the information is contained in the predictor variables. Thus, there must be something not yet accounted for in the predictor variables that can be extracted from the different response variables to see improvement over ordinary least squares regression.

3.1.2. *Some Irrelevant Variables.* I ran another simulation where each predictor had an 80% chance of being used to generate the response variables. In other words, each predictor variable had a 20% chance of not being associated with the response variables, and thus it's associated coefficient is 0.

Doing this, we can look at the same levels used before to see how well

Curds and Whey does with irrelevant variables.

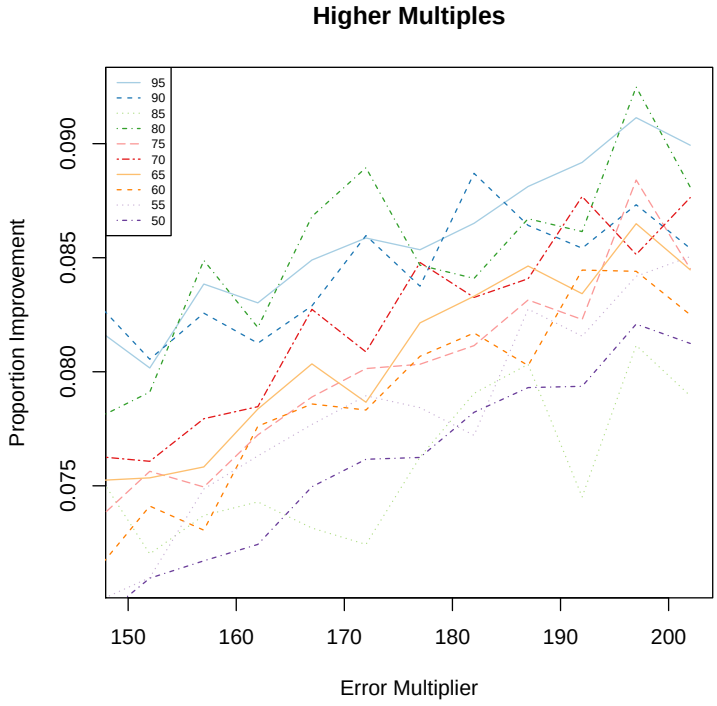
FIGURE 4. *Proportion improvement with $q = 5$ and $p = 10$ with irrelevant variables*




In Figure 4, we see almost the exact same situation that we viewed with all of the variables. We still see a general trend of improvement over ordinary least squares regression as the amount of error multiplier is increased.

Looking closer at the higher multiples (Figure 5), we see the improvement increase, and it still appears the higher correlations result in more improvement, but the improvements are too close to determine any specific pattern. This, again, could be due to the fact that the same base error model is used for each of the added error multiples.

FIGURE 5. *Proportion improvement with $q = 5$ and $p = 10$ with irrelevant variables*

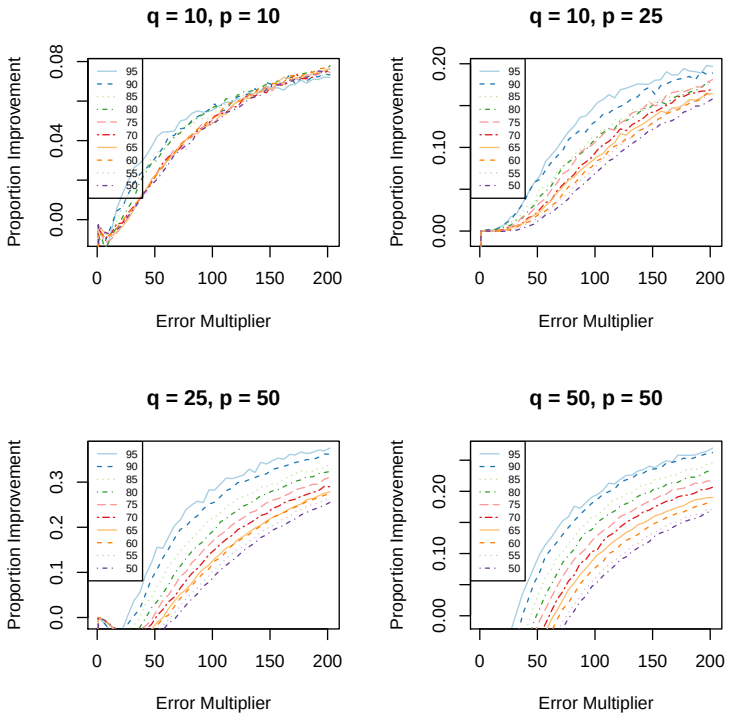



We look at the spread at the other values of $p$ and $q$ in Figure 6. These graphs are printed individually in the appendix as well. Again, the graphs seem extremely similar to those generated with all of the variables contributing to the response.

TABLE 2
*Max improvement values compared*

| q | p | Some Zeros | All Variables |
|---|---|---|---|
| 5 | 10 | 0.096 | 0.091 |
| 10 | 10 | 0.084 | 0.081 |
| 10 | 25 | 0.205 | 0.191 |
| 25 | 50 | 0.389 | 0.385 |
| 50 | 50 | 0.289 | 0.276 |

FIGURE 6. *Proportion improvement across remaining levels with irrelevant variables*




Looking at the highest improvement that is seen for cases with irrelevant variables, and no irrelevant variables (Table 2), there is next to no difference between the two, except that there appears to be a little less improvement when there are some unnecessary variables in the model. We would expect models to potentially be thrown off by unnecessary variables, but we still see some remarkable improvement.

We believe that the shrinkage effects of Curds and Whey will cause some variables' coefficients in the model to be forced to zero, thus allowing for the model to eliminate unnecessary variables.

3.2. *Real Data Examples.*

3.2.1. *Chemometrics.* An example given by Breiman and Friedman (1997) deals with chemometrics. In this data set, taken from Skagerberg, MacGregor and Kiparissides (1992), there are 56 observations ($N = 56$), each with 22 predictor variables ($p = 22$), and 6 responses ($q = 6$). The data were taken from a simulation of a low density tubular polyethylene reactor. The predictor variables are all temperatures measured at equal distances along the reactor together with the wall temperature of the reactor and feed rate. The responses are:

- $y_1$: number-average molecular weight
- $y_2$: weight-average molecular weight
- $y_3$: frequency of long chain branching
- $y_4$: frequency of short chain branching
- $y_5$: content of vinyl groups
- $y_6$: content of vinylidene groups

A log transformation was applied to all six response variables to correct for right-skewness. The average absolute correlation of response variables is .48. The correlations for the responses are as follows in Table 3.

TABLE 3
*Correlation between chemometrics responses*

| | lr1 | lr2 | lr3 | lr4 | lr5 | lr6 |
|---|---|---|---|---|---|---|
| lr1 | 1.0000 | 0.9567 | 0.0651 | 0.2543 | 0.2551 | 0.2592 |
| lr2 | 0.9567 | 1.0000 | -0.1284 | 0.2825 | 0.2656 | 0.2756 |
| lr3 | 0.0651 | -0.1284 | 1.0000 | -0.4997 | -0.4840 | -0.4787 |
| lr4 | 0.2543 | 0.2825 | -0.4997 | 1.0000 | 0.9744 | 0.9782 |
| lr5 | 0.2551 | 0.2656 | -0.4840 | 0.9744 | 1.0000 | 0.9760 |
| lr6 | 0.2592 | 0.2756 | -0.4787 | 0.9782 | 0.9760 | 1.0000 |

Note that response 1 appears to be very highly correlated with response 2, but barely correlated with the other responses. Responses 4, 5, and 6 are very highly correlated with each other. Thus, the response variables fall into three groups.

FIGURE 7. *Percentage improvement of chemometrics responses*

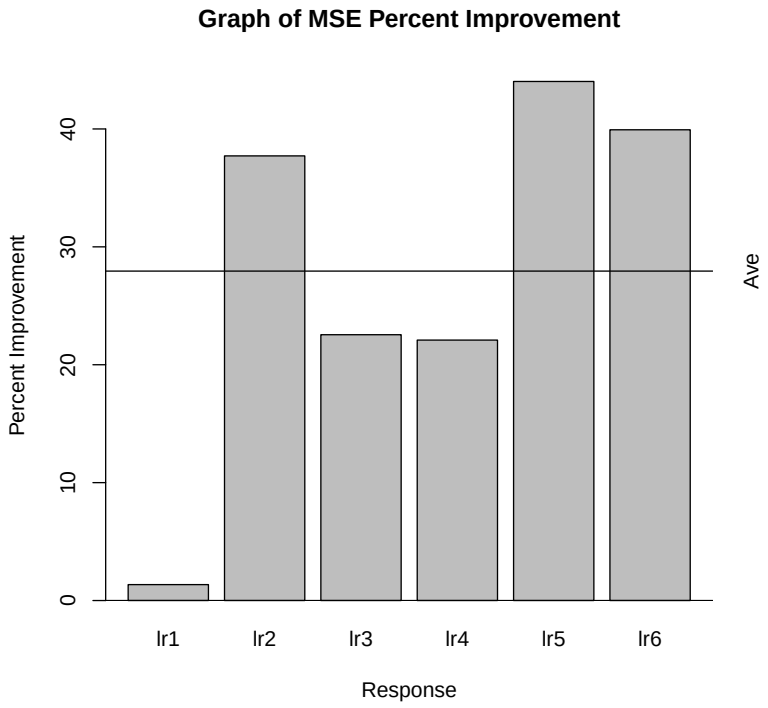



Figure 7 shows the percent improvement (of 10-fold cross validation mean squared error) achieved over ordinary least squares regression. To improve the accuracy of the reported error, the MSE was calculated using 200 different samples of the 10 fold cross-validation, and averaged over all of the repetitions.

Looking at the graph, we notice that response 1 (`lr1`) appears to not benefit from Curds and Whey with an improvement of essentially zero. We do see much higher improvements with responses 2, 5, and 6, with improvements of 37.72%, 44.03%, 39.93%, respectively, and an average improvement of 27.94%.

3.2.2. *Teen Crime Data.* This data set deals with violent crimes committed by teens in all 50 states and Washington D.C. The data was collected between the years of 1985 and 1993. It contains many possible predictor and response variables, which are as follows.

- $x_1$: Percentage of Seniors that graduate from High School.
- $x_2$: Standardized transformation of Scoring Method used in Survey.
- $x_3$: Number of 1- to 14-year-olds in 1985.
- $x_4$: Number of 1- to 14-year-olds that died in 1985.
- $x_5, x_6$: $x_3$ and $x_4$ repeated but for 1991.
- $x_7, x_8$: Percentage of Kids living in Poverty in 1985, 1991, respectively.
- $x_9$ to $x_{19}$: Percentage of Kids living in Single Parent Families from 1983 through 1993, respectively.
- $x_{20}$ to $x_{25}$: The Median income in 1987 through 1992, respectively.
- $y_1$ to $y_8$: Juvenile Violent Crimes per 100,000 people in 1985 to 1992.

A lot of research has been done on this data, and in a lot of circumstances, the Juvenile Violent Crime rates have been the natural response variables. With this data, I go through the data twice, once with all of the predictor variables to view the change in predictive accuracy, and then once with a more specialized set of predictors.

First, I ran the Curds and Whey procedure trying to predict the Juvenile Crime Rate using all of the other variables available.

FIGURE 8. *Percentage improvement teen crime data*

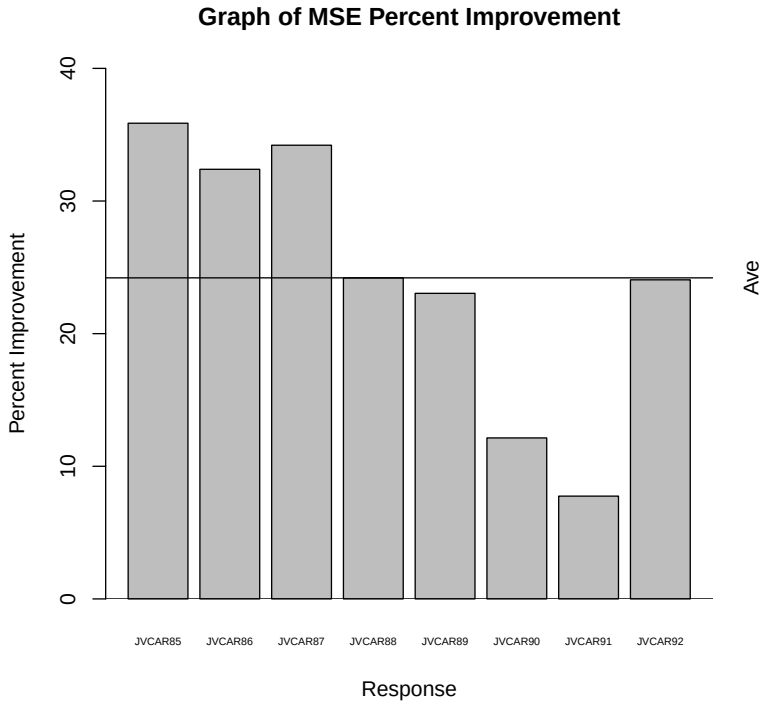
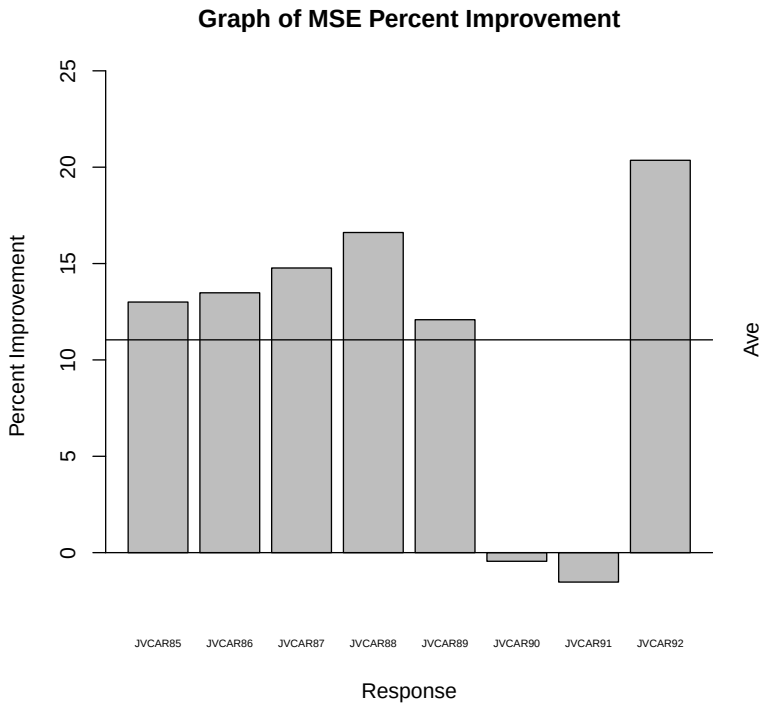



Figure 8 shows there is improvement in every category, and some substantial improvement on some years. It should be noted that 1990 and 1991 saw the least improvement. Yet, we still see what may be considered as a remarkable improvement on average.

The second set to be tested includes only the predictors involving the percentage of single parent households and median income.

Figure 9 shows the results of percentage of improvement for this smaller set. Again, 1990 and 1991 see less improvement than the other years. Fo-

FIGURE 9. *Teen crime data with single parent and median income*




cusing on 1991, we actually see a decrease in accuracy of 1.53%. In this case, there may be some information contained within the other predictor variables that allows for more information to be extracted from the response variables.

Table 4 shows the correlations between 1990-1991, and the other years. Though these seem very high, when compared to the correlations between other years (Table 5), we see that these correlations are generally lower than other years, showing again that we may not be able to pull out as much information for these years as we do for other years.

TABLE 4
*Correlation between 1990-1991, and other years*

| | JVCAR90 | JVCAR91 |
|---|---|---|
| JVCAR85 | 0.9049542 | 0.9126092 |
| JVCAR86 | 0.9506291 | 0.9490129 |
| JVCAR87 | 0.9390007 | 0.9284937 |
| JVCAR88 | 0.9472979 | 0.9355166 |
| JVCAR89 | 0.9425560 | 0.9390979 |
| JVCAR90 | 1.0000000 | 0.9762858 |
| JVCAR91 | 0.9762858 | 1.0000000 |
| JVCAR92 | 0.9426121 | 0.9680579 |

TABLE 5
*Correlation between 1990-1991, and other years*

| | JVCAR85 | JVCAR86 | JVCAR87 | JVCAR88 | JVCAR89 | JVCAR92 |
|---|---|---|---|---|---|---|
| JVCAR85 | 1.0000 | 0.9656 | 0.9511 | 0.9108 | 0.8981 | 0.9102 |
| JVCAR86 | 0.9656 | 1.0000 | 0.9763 | 0.9413 | 0.9417 | 0.9366 |
| JVCAR87 | 0.9511 | 0.9763 | 1.0000 | 0.9572 | 0.9524 | 0.9341 |
| JVCAR88 | 0.9108 | 0.9413 | 0.9572 | 1.0000 | 0.9581 | 0.9499 |
| JVCAR89 | 0.8981 | 0.9417 | 0.9524 | 0.9581 | 1.0000 | 0.9438 |
| JVCAR92 | 0.9102 | 0.9366 | 0.9341 | 0.9499 | 0.9438 | 1.0000 |

In the end, though, we see an average improvement of 11.04%. This is a sizable difference and improvement of this magnitude is always desirable. We would then be confident in saying that Curds and Whey has given better predictions in the end.

Notice that data from 1990 and 1991 are more highly correlated with each other than any other years. This may help explain why we fail to see much improvement. It may be that these two years are highly correlated with each other, but not the other years, so we are unable to extract much additional information to improve the predictive accuracy.

In the end, the focus largely is on how the average improvement was fairly substantial (over 10%), and conclude that overall, Curds and Whey did provide a good increase in prediction accuracy.

3.2.3. *Paper Data.* The next two data sets were used by Aldrin (1996). In these data, we find a more complex scenario in which there are more predictor variables than response variables. This is a problem for the Curds and Whey algorithm because the canonical correlation part of the algorithm yields only as many canonical correlations as the minimum of $p$ and $q$. One option, that is be used below, is to divide the response variables into different groups, and use Curds and Whey on each group separately. As Curds and Whey is able to improve predictive accuracy by extracting extra information from the other response variables, which grouping a response variable is in may have a large impact upon the accuracy and predictions returned by Curds and Whey. In the following analysis, we hope to show that the predictions found using Curds and Whey are somewhat similar across different groupings, but it should be stressed that such scenarios as this require much care and precision. The groupings should be heavily influenced by expert knowledge in the specified field.

*Variable Information.* The response variables are different measures of the quality of the paper. The predictor variables (9 in each set), are made up as follows:

TABLE 6
*Description of the format of the predictors in the Paper data sets*

| Predictor | Description |
|---|---|
| $p_1$ | A process systematically changed through the experiment. |
| $p_2$ | A process systematically changed through the experiment. |
| $p_3$ | A process systematically changed through the experiment. |
| $p_4$ | Predictor 1 squared ($p_1^2$) |
| $p_5$ | Predictor 2 squared ($p_2^2$) |
| $p_6$ | Predictor 3 squared ($p_3^2$) |
| $p_7$ | Interaction of Predictor 1 and Predictor 2 ($p_1 * p_2$). |
| $p_8$ | Interaction of Predictor 1 and Predictor 3.($p_1 * p_3$) |
| $p_9$ | Interaction of Predictor 2 and Predictor 3.($p_2 * p_3$) |

*Paper 1.* In this data set, one observation was removed from the data due to all of the response variable values being missing. There are 13 response variables. Within the response variables, $p_1 - p_3$ could take only the values -1, 0, or 1 (signifying a "low", "medium", or "high" setting).

To account for all of the response variables, and then to test possible differences from using different response variables, I ran through three different analyses. The first run through includes the first 7 response variables, the second includes the last 6 response variables (variables 8-13), and the final includes the $5^{th}$ through $10^{th}$.

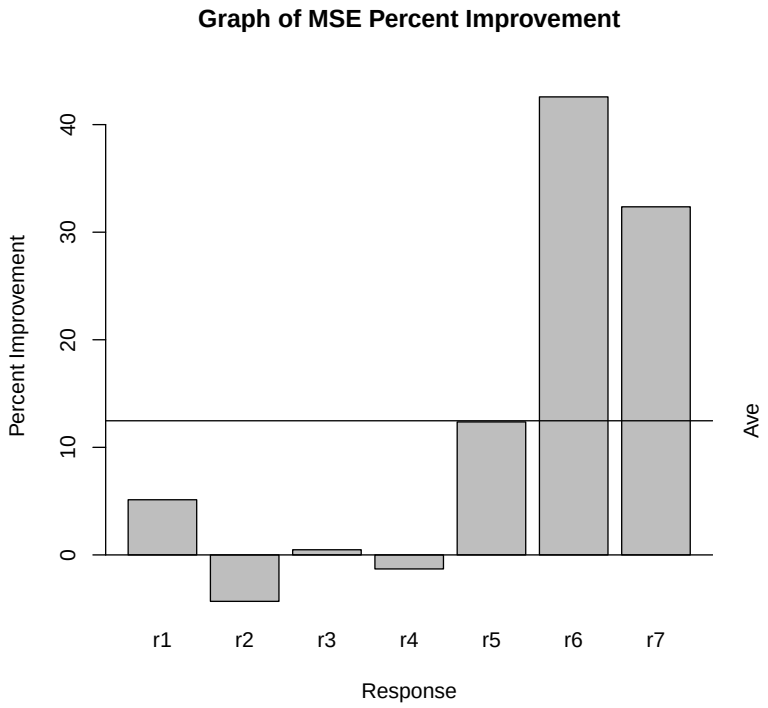
We can think of this final run through as being a "cross" between the first two sets. It contains some variables from each of the two sets, thus, it is a "cross" or intersection between them.

Figure 10 shows the first analysis with the first seven responses, Figure 11 shows the second analysis with the last six responses, and Figure 12 shows the final analysis with the cross of predictors.

Looking first at Figure 10, we see a huge range of "improvement" compared to ordinary least squares regression. We actually see a decrease in prediction accuracy of 4.32% in the second response. Then, we see an increase in accuracy of 42.58% in the sixth response. Overall, we see an average increase of 12.47%. So, though we do see some decreases in accuracy, they are close to zero.

Also, we may see some responses that do not see much improvement, but they may be contributing the increased accuracy of predicting the other response variables. In this sense, though they do not see much improvement, they are able to improve the prediction of the other responses.

FIGURE 10. *Percentage improvement of paper1 group 1*

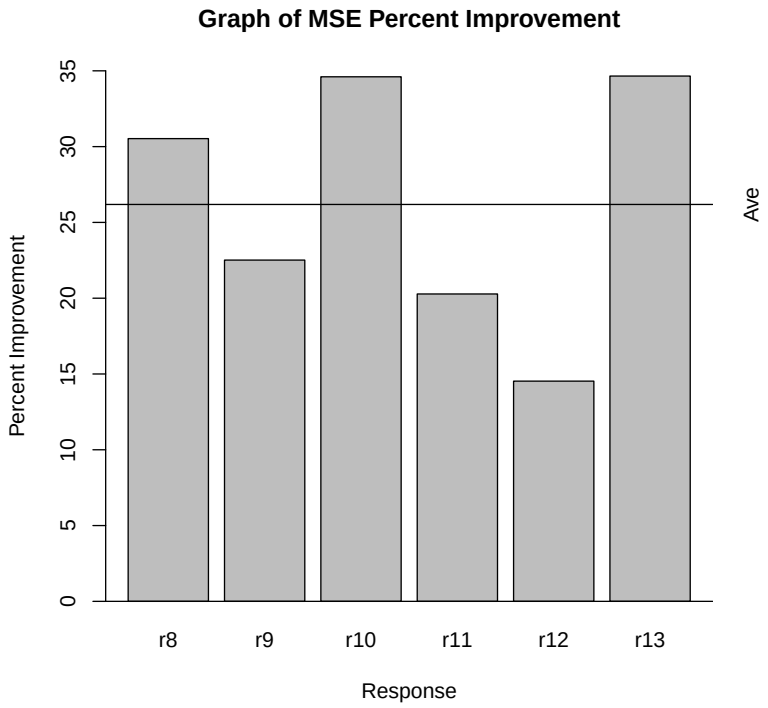



Next, Figure 11 shows substantial improvement across the board. Both responses 10 and 13 see very high improvement (20.28% and 34.65% respectively). The lowest improvement is 14.53% with response 12, and our average improvement is 26.19%.

Finally, referencing Figure 12, we look at the intersection between the two sets of variables. We observe that response 5 has the least improvement (8.53%), which had the least improvement of this set in the first tests. Response 6 does not see nearly as much improvement as was seen in the first test (42.58% vs 27.68%), but still improves by a good margin.

Thus, we see that information is still being extracted from the other re-

FIGURE 11. *Percentage improvement of paper1 group 2*




sponses, but that not all of the responses offer the same information. We can conclude from this that the set of responses used (when there are more responses than predictors) is very important, but yet we still see improvement.

Looking now at some information about the coefficients (Tables 7 and 8), we can see that almost all of the coefficients have the same sign and the differences in coefficients are fairly small. In detail, the biggest difference between any coefficient is 0.4112. When compared to the values of our predictors, this is relatively small.

We can also look specifically at the differences between values that have

FIGURE 12. *Percentage improvement of paper1 when fit in the groupings r5-r10 compared to separate groupings*

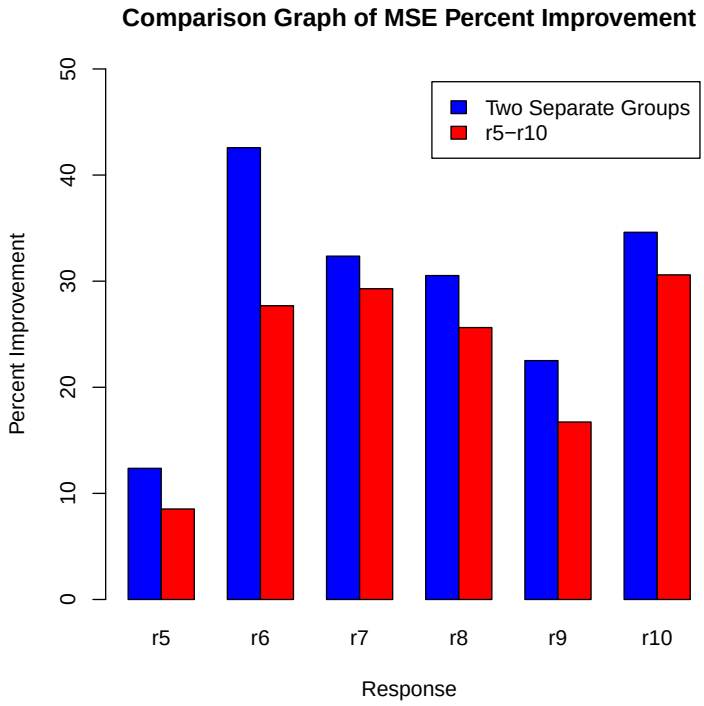



TABLE 7

*Agreement of the sign (+/-) for the coefficients of response variables r5-r10, when fit in the groupings r1-r7, r8-r13, and then r5-r10*

| | r5 | r6 | r7 | r8 | r9 | r10 |
|---|---|---|---|---|---|---|
| (Intercept) | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p1 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p2 | TRUE | FALSE | TRUE | FALSE | TRUE | TRUE |
| p3 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p4 | FALSE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p5 | TRUE | TRUE | FALSE | TRUE | FALSE | TRUE |
| p6 | TRUE | FALSE | TRUE | TRUE | TRUE | TRUE |
| p7 | TRUE | FALSE | FALSE | TRUE | TRUE | TRUE |
| p8 | TRUE | TRUE | TRUE | FALSE | TRUE | TRUE |
| p9 | TRUE | FALSE | TRUE | TRUE | TRUE | TRUE |

TABLE 8

*Difference between coefficients of the response variables r5-r10, when fit in the groupings r1-r7, r8-r13, and then r5-r10*

| | r5 | r6 | r7 | r8 | r9 | r10 |
|---|---|---|---|---|---|---|
| (Intercept) | 0.0390 | -0.0109 | 0.0369 | 0.1192 | -0.0031 | 0.1013 |
| p1 | 0.0047 | -0.1520 | 0.1230 | -0.0158 | -0.1241 | 0.0647 |
| p2 | -0.0209 | 0.1339 | -0.1301 | 0.1300 | 0.1978 | -0.0245 |
| p3 | 0.1039 | -0.0060 | 0.0465 | 0.0526 | 0.1623 | -0.0668 |
| p4 | 0.1771 | -0.2419 | 0.2980 | -0.1290 | -0.1555 | -0.0039 |
| p5 | -0.0979 | 0.1708 | -0.1966 | 0.0458 | 0.1858 | -0.0890 |
| p6 | -0.1387 | 0.1073 | -0.1785 | -0.1134 | 0.0142 | -0.1005 |
| p7 | 0.1064 | -0.1772 | 0.1939 | 0.0700 | 0.0337 | 0.0369 |
| p8 | 0.1519 | 0.4112 | -0.2521 | 0.0457 | 0.1189 | -0.0412 |
| p9 | 0.3332 | -0.1423 | 0.2960 | 0.1964 | 0.0616 | 0.1223 |

TABLE 9

*Difference between coefficients r5-r10, when fit in the groupings r1-r7, r8-r13, and then r5-r10, which had a sign disagreement*

| | r5 | r6 | r7 | r8 | r9 | r10 |
|---|---|---|---|---|---|---|
| (Intercept) | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |
| p1 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |
| p2 | 0.0000 | 0.1339 | 0.0000 | 0.1300 | 0.0000 | 0 |
| p3 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |
| p4 | 0.1771 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |
| p5 | 0.0000 | 0.0000 | 0.1966 | 0.0000 | 0.1858 | 0 |
| p6 | 0.0000 | 0.1073 | 0.0000 | 0.0000 | 0.0000 | 0 |
| p7 | 0.0000 | 0.1772 | 0.1939 | 0.0000 | 0.0000 | 0 |
| p8 | 0.0000 | 0.0000 | 0.0000 | 0.0457 | 0.0000 | 0 |
| p9 | 0.0000 | 0.1423 | 0.0000 | 0.0000 | 0.0000 | 0 |

different signs in Table 9. Here, we again see that there are not any major differences between the coefficient values.

We may also want to look at the difference between the predictions using these two different methods. Table 10 shows the absolute difference between the predicted responses when fit in groupings r1-r7, r8-r13, and then r5-r10. Here, it can be seen that the largest difference is 0.8108, and the average difference is 0.1793.

To get a better feel for how this relates to the response variables in the

TABLE 10

*Difference between predicted responses r5-r10, when fit in the groupings r1-r7, r8-r13, and then r5-r10*

| | r5 | r6 | r7 | r8 | r9 | r10 |
|---|---|---|---|---|---|---|
| 1 | 0.1737 | 0.7027 | 0.4850 | 0.2374 | 0.5497 | 0.1696 |
| 2 | 0.1737 | 0.7027 | 0.4850 | 0.2374 | 0.5497 | 0.1696 |
| 3 | 0.6556 | 0.5494 | 0.8108 | 0.0674 | 0.1707 | 0.1682 |
| 4 | 0.6556 | 0.5494 | 0.8108 | 0.0674 | 0.1707 | 0.1682 |
| 5 | 0.0390 | 0.0109 | 0.0369 | 0.1192 | 0.0031 | 0.1013 |
| 6 | 0.0390 | 0.0109 | 0.0369 | 0.1192 | 0.0031 | 0.1013 |
| 7 | 0.0247 | 0.4106 | 0.3321 | 0.1006 | 0.0231 | 0.0934 |
| 8 | 0.0247 | 0.4106 | 0.3321 | 0.1006 | 0.0231 | 0.0934 |
| 9 | 0.2408 | 0.2410 | 0.3393 | 0.0082 | 0.0128 | 0.0050 |
| 10 | 0.2408 | 0.2410 | 0.3393 | 0.0082 | 0.0128 | 0.0050 |
| 11 | 0.4060 | 0.2696 | 0.4575 | 0.2222 | 0.0998 | 0.2528 |
| 12 | 0.4060 | 0.2696 | 0.4575 | 0.2222 | 0.0998 | 0.2528 |
| 13 | 0.0390 | 0.0109 | 0.0369 | 0.1192 | 0.0031 | 0.1013 |
| 14 | 0.0390 | 0.0109 | 0.0369 | 0.1192 | 0.0031 | 0.1013 |
| 15 | 0.0374 | 0.1907 | 0.1976 | 0.1797 | 0.3284 | 0.0606 |
| 16 | 0.0374 | 0.1907 | 0.1976 | 0.1797 | 0.3284 | 0.0606 |
| 17 | 0.3380 | 0.1076 | 0.0739 | 0.0407 | 0.0128 | 0.0466 |
| 18 | 0.3380 | 0.1076 | 0.0739 | 0.0407 | 0.0128 | 0.0466 |
| 19 | 0.0527 | 0.0030 | 0.0414 | 0.0654 | 0.1017 | 0.1253 |
| 20 | 0.0527 | 0.0030 | 0.0414 | 0.0654 | 0.1017 | 0.1253 |
| 21 | 0.0390 | 0.0109 | 0.0369 | 0.1192 | 0.0031 | 0.1013 |
| 22 | 0.0390 | 0.0109 | 0.0369 | 0.1192 | 0.0031 | 0.1013 |
| 23 | 0.0138 | 0.3812 | 0.3087 | 0.1119 | 0.3154 | 0.1177 |
| 24 | 0.0138 | 0.3812 | 0.3087 | 0.1119 | 0.3154 | 0.1177 |
| 25 | 0.1207 | 0.4237 | 0.2652 | 0.1143 | 0.0637 | 0.0423 |
| 26 | 0.1207 | 0.4237 | 0.2652 | 0.1143 | 0.0637 | 0.0423 |
| 27 | 0.2187 | 0.2528 | 0.1258 | 0.4306 | 0.6186 | 0.0573 |
| 28 | 0.2084 | 0.2772 | 0.3251 | 0.2202 | 0.1346 | 0.0854 |
| 29 | 0.2084 | 0.2772 | 0.3251 | 0.2202 | 0.1346 | 0.0854 |

data set, divide each of these differences by the corresponding true value for the response to the percentage the difference is in relation to that true value. This can be seen in Table 11. Here, the largest proportional difference is 0.0822, and the average proportional difference is 0.0132

So, overall, it would be recommended that in the case where $q > p$ that the responses used are selected carefully, but it does appear that even without the optimal set we will get somewhat similar results for this data set.

TABLE 11

*Proportional difference between predicted responses r5-r10, when fit in the groupings r1-r7, r8-r13, and then r5-r10*

| | r5 | r6 | r7 | r8 | r9 | r10 |
|---|---|---|---|---|---|---|
| 1 | 0.0114 | 0.0633 | 0.0421 | 0.0128 | 0.0241 | 0.0096 |
| 2 | 0.0114 | 0.0576 | 0.0455 | 0.0117 | 0.0218 | 0.0098 |
| 3 | 0.0431 | 0.0424 | 0.0822 | 0.0040 | 0.0071 | 0.0112 |
| 4 | 0.0422 | 0.0479 | 0.0713 | 0.0036 | 0.0070 | 0.0100 |
| 5 | 0.0026 | 0.0009 | 0.0034 | 0.0062 | 0.0001 | 0.0057 |
| 6 | 0.0028 | 0.0010 | 0.0036 | 0.0062 | 0.0001 | 0.0054 |
| 7 | 0.0019 | 0.0292 | 0.0431 | 0.0052 | 0.0011 | 0.0047 |
| 8 | 0.0019 | 0.0336 | 0.0360 | 0.0056 | 0.0011 | 0.0053 |
| 9 | 0.0173 | 0.0186 | 0.0371 | 0.0004 | 0.0006 | 0.0003 |
| 10 | 0.0177 | 0.0186 | 0.0383 | 0.0004 | 0.0006 | 0.0003 |
| 11 | 0.0313 | 0.0221 | 0.0517 | 0.0130 | 0.0045 | 0.0150 |
| 12 | 0.0313 | 0.0235 | 0.0471 | 0.0119 | 0.0044 | 0.0143 |
| 13 | 0.0026 | 0.0009 | 0.0034 | 0.0063 | 0.0001 | 0.0056 |
| 14 | 0.0026 | 0.0009 | 0.0036 | 0.0069 | 0.0001 | 0.0060 |
| 15 | 0.0026 | 0.0172 | 0.0181 | 0.0105 | 0.0153 | 0.0035 |
| 16 | 0.0026 | 0.0172 | 0.0182 | 0.0114 | 0.0156 | 0.0037 |
| 17 | 0.0205 | 0.0088 | 0.0066 | 0.0024 | 0.0006 | 0.0028 |
| 18 | 0.0205 | 0.0081 | 0.0071 | 0.0024 | 0.0005 | 0.0029 |
| 19 | 0.0035 | 0.0002 | 0.0044 | 0.0036 | 0.0043 | 0.0074 |
| 20 | 0.0031 | 0.0002 | 0.0037 | 0.0035 | 0.0045 | 0.0069 |
| 21 | 0.0027 | 0.0008 | 0.0043 | 0.0068 | 0.0001 | 0.0059 |
| 22 | 0.0027 | 0.0008 | 0.0040 | 0.0063 | 0.0001 | 0.0056 |
| 23 | 0.0010 | 0.0278 | 0.0357 | 0.0061 | 0.0141 | 0.0066 |
| 24 | 0.0010 | 0.0286 | 0.0338 | 0.0056 | 0.0139 | 0.0062 |
| 25 | 0.0083 | 0.0301 | 0.0297 | 0.0063 | 0.0027 | 0.0025 |
| 26 | 0.0078 | 0.0310 | 0.0277 | 0.0058 | 0.0027 | 0.0023 |
| 27 | 0.0153 | 0.0236 | 0.0115 | 0.0237 | 0.0266 | 0.0034 |
| 28 | 0.0143 | 0.0234 | 0.0309 | 0.0122 | 0.0061 | 0.0048 |
| 29 | 0.0150 | 0.0220 | 0.0355 | 0.0124 | 0.0058 | 0.0051 |

*Data Set 2.* In this data set, we are able to use every observation. Also, the response and predictor variables are all continuous variables. There are 41 response variables, and the 3 base predictor variables ($p_1 - p_3$) were not restricted. Due to the small number of predictors, I again split the response variables into different groups. This time, there are 4 groups, with each group containing 8 response variables. I then ran one set that used response variables from two of the groups so that we can see the differences in results

for those variables.

FIGURE 13. *Percentage improvement of on paper 2 groupings*

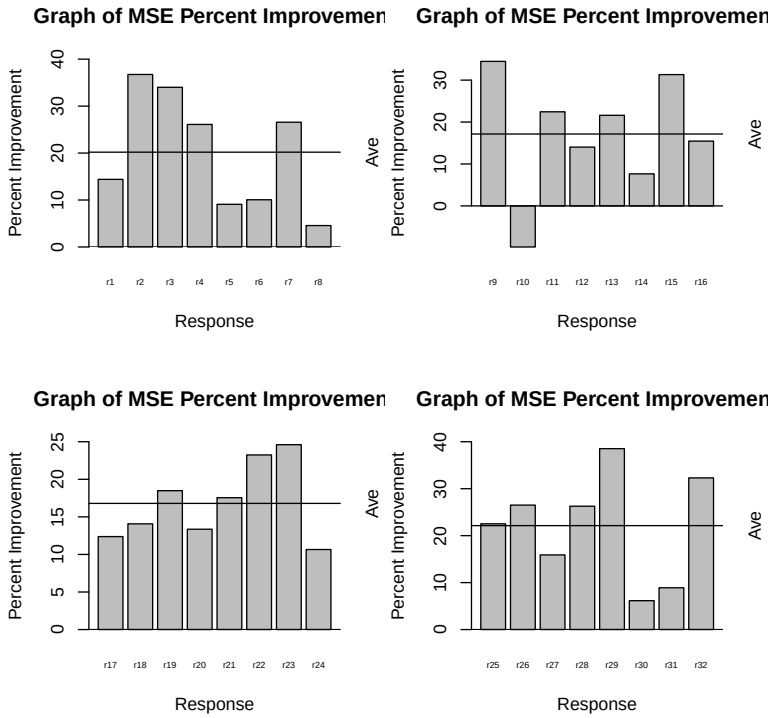
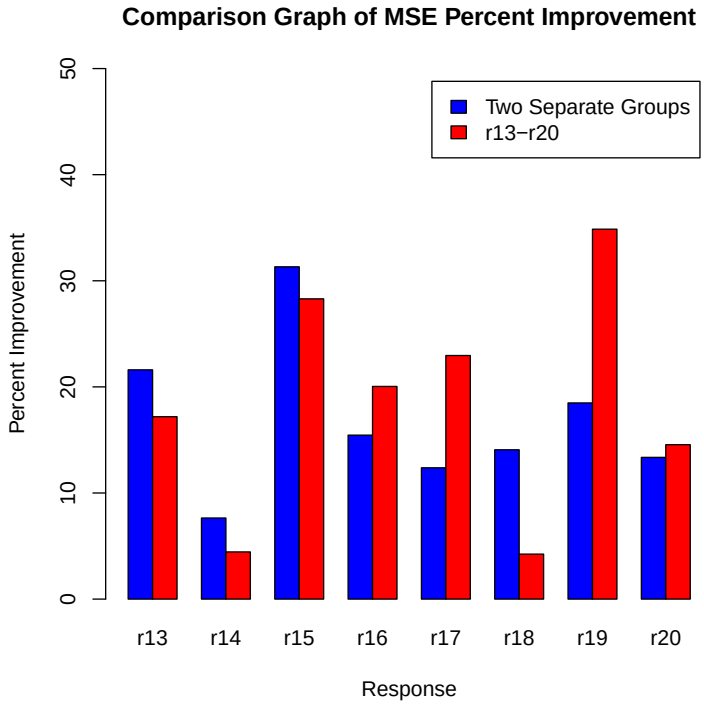



Figure 13 shows the improvement for each of the response variables in the four different groups. With the exception of r10, every response saw an increased prediction accuracy (or a decrease in mean squared error). In some cases (responses 4 and 29), we see improvement of almost 40%.

Looking then at the variables chosen from two groups, Figure 14 shows the improvement found when using responses 13-20. Again, we see improvement for all of the response variables when they are used together, or in two separate groups. Looking at the results closely, we see that most of the responses achieve similar levels of improvement in predictive accuracy, re-

gardless of the group they were in. The exception is r18, which shows a much lower level of improvement in the group with variables r13-r20 than it did with response variables r17-r24. This demonstrates the fact that gains in predictive accuracy depend on which other variables are used, and the need for expert knowledge in choosing the groupings. It also suggests the need for an algorithm to combine or optimize gains in predictive accuracy when the number of response variables exceeds the number of predictor variables.

FIGURE 14. *Percentage improvement of paper2 when fit in the groupings r13-r20 compared to earlier groupings*




Looking more closely at the coefficients extracted during each method, we look at tables similar to the ones we had for the first paper data set.

Table 14 shows the coefficients that had the same sign in both groups (a total of 4). Table 15 shows the differences between all the coefficients (where the biggest difference is 0.1494), and Table 16 shows the difference between coefficients where the signs were different (the biggest difference here being 0.1066).

TABLE 12

*Difference between predicted responses r13-r20, when fit in the groupings r9-r16, r17-r24, and then r13-r20*

| | r13 | r14 | r15 | r16 | r17 | r18 | r19 | r20 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.2532 | 0.0969 | 0.1459 | 0.1302 | 0.2764 | 0.0631 | 0.2835 | 0.1151 |
| 2 | 0.4707 | 0.2254 | 0.2721 | 0.2568 | 0.5175 | 0.0463 | 0.4454 | 0.3143 |
| 3 | 0.0342 | 0.0621 | 0.0163 | 0.0299 | 0.0823 | 0.0239 | 0.0144 | 0.0946 |
| 4 | 0.0155 | 0.0645 | 0.0013 | 0.0157 | 0.1325 | 0.0378 | 0.0571 | 0.1236 |
| 5 | 0.1835 | 0.4178 | 0.1788 | 0.1543 | 0.1892 | 0.0934 | 0.2844 | 0.0218 |
| 6 | 0.2406 | 0.2746 | 0.1973 | 0.1706 | 0.2379 | 0.1092 | 0.3123 | 0.1043 |
| 7 | 0.0124 | 0.1403 | 0.0082 | 0.0121 | 0.0338 | 0.0823 | 0.0471 | 0.0496 |
| 8 | 0.0405 | 0.0957 | 0.0181 | 0.0201 | 0.0527 | 0.0483 | 0.0225 | 0.0787 |
| 9 | 0.2394 | 0.0394 | 0.0297 | 0.0665 | 0.2715 | 0.1424 | 0.1233 | 0.4003 |
| 10 | 0.0999 | 0.0659 | 0.0492 | 0.0093 | 0.1561 | 0.1105 | 0.0950 | 0.1665 |
| 11 | 0.1419 | 0.1718 | 0.0779 | 0.0684 | 0.1603 | 0.0019 | 0.1756 | 0.0732 |
| 12 | 0.0475 | 0.1349 | 0.0953 | 0.0239 | 0.1514 | 0.1644 | 0.0968 | 0.0291 |
| 13 | 0.1286 | 0.1016 | 0.0777 | 0.1251 | 0.0622 | 0.0407 | 0.0801 | 0.0240 |
| 14 | 0.1312 | 0.1031 | 0.0703 | 0.1502 | 0.0378 | 0.0035 | 0.0636 | 0.0327 |
| 15 | 0.2394 | 0.3376 | 0.2925 | 0.1604 | 0.4511 | 0.0851 | 0.4651 | 0.0917 |
| 16 | 0.1140 | 0.0577 | 0.0788 | 0.0967 | 0.3299 | 0.0429 | 0.3172 | 0.1707 |
| 17 | 0.0223 | 0.1131 | 0.0726 | 0.0464 | 0.0833 | 0.0076 | 0.0101 | 0.1962 |
| 18 | 0.0631 | 0.0489 | 0.0809 | 0.0073 | 0.0151 | 0.0270 | 0.0518 | 0.1267 |
| 19 | 0.2691 | 0.4440 | 0.2317 | 0.0620 | 0.2591 | 0.2519 | 0.3268 | 0.1989 |
| 20 | 0.0663 | 0.2004 | 0.0229 | 0.0736 | 0.0102 | 0.1020 | 0.0687 | 0.0301 |
| 21 | 0.1165 | 0.0973 | 0.2040 | 0.0656 | 0.4210 | 0.1537 | 0.3789 | 0.2745 |
| 22 | 0.1477 | 0.0163 | 0.1837 | 0.1113 | 0.4062 | 0.2016 | 0.3890 | 0.2678 |
| 23 | 0.0137 | 0.0420 | 0.0030 | 0.0188 | 0.1663 | 0.0315 | 0.0837 | 0.1486 |
| 24 | 0.0155 | 0.0569 | 0.0032 | 0.0165 | 0.1450 | 0.0383 | 0.0638 | 0.1344 |
| 25 | 0.3567 | 0.2248 | 0.2850 | 0.1605 | 0.2520 | 0.0439 | 0.2018 | 0.2468 |
| 26 | 0.1341 | 0.1732 | 0.1663 | 0.0454 | 0.1826 | 0.0495 | 0.0921 | 0.2723 |
| 27 | 0.1420 | 0.1959 | 0.1643 | 0.0129 | 0.0264 | 0.0890 | 0.0263 | 0.0570 |
| 28 | 0.0480 | 0.1657 | 0.0598 | 0.1132 | 0.0319 | 0.0915 | 0.1828 | 0.0669 |
| 29 | 0.1104 | 0.2294 | 0.1376 | 0.0748 | 0.1372 | 0.0257 | 0.1672 | 0.1491 |
| 30 | 0.1376 | 0.3088 | 0.1563 | 0.0515 | 0.1025 | 0.0492 | 0.1435 | 0.0900 |

Now looking at the difference in predicted responses, we can look at Table 12 which shows the absolute difference between the predicted responses

when fit in groupings r9-r16, r17-r24, and then r13-r20. Here, it can be seen that the largest difference is 0.5175, and the average difference is 0.1305.

TABLE 13
*Proportional difference between predicted responses r13-r20, when fit in the groupings r9-r16, r17-r24, and then r13-r20*

| | r5 | r6 | r7 | r8 | r9 | r10 |
|---|---|---|---|---|---|---|
| 1 | 0.0114 | 0.0633 | 0.0421 | 0.0128 | 0.0241 | 0.0096 |
| 2 | 0.0114 | 0.0576 | 0.0455 | 0.0117 | 0.0218 | 0.0098 |
| 3 | 0.0431 | 0.0424 | 0.0822 | 0.0040 | 0.0071 | 0.0112 |
| 4 | 0.0422 | 0.0479 | 0.0713 | 0.0036 | 0.0070 | 0.0100 |
| 5 | 0.0026 | 0.0009 | 0.0034 | 0.0062 | 0.0001 | 0.0057 |
| 6 | 0.0028 | 0.0010 | 0.0036 | 0.0062 | 0.0001 | 0.0054 |
| 7 | 0.0019 | 0.0292 | 0.0431 | 0.0052 | 0.0011 | 0.0047 |
| 8 | 0.0019 | 0.0336 | 0.0360 | 0.0056 | 0.0011 | 0.0053 |
| 9 | 0.0173 | 0.0186 | 0.0371 | 0.0004 | 0.0006 | 0.0003 |
| 10 | 0.0177 | 0.0186 | 0.0383 | 0.0004 | 0.0006 | 0.0003 |
| 11 | 0.0313 | 0.0221 | 0.0517 | 0.0130 | 0.0045 | 0.0150 |
| 12 | 0.0313 | 0.0235 | 0.0471 | 0.0119 | 0.0044 | 0.0143 |
| 13 | 0.0026 | 0.0009 | 0.0034 | 0.0063 | 0.0001 | 0.0056 |
| 14 | 0.0026 | 0.0009 | 0.0036 | 0.0069 | 0.0001 | 0.0060 |
| 15 | 0.0026 | 0.0172 | 0.0181 | 0.0105 | 0.0153 | 0.0035 |
| 16 | 0.0026 | 0.0172 | 0.0182 | 0.0114 | 0.0156 | 0.0037 |
| 17 | 0.0205 | 0.0088 | 0.0066 | 0.0024 | 0.0006 | 0.0028 |
| 18 | 0.0205 | 0.0081 | 0.0071 | 0.0024 | 0.0005 | 0.0029 |
| 19 | 0.0035 | 0.0002 | 0.0044 | 0.0036 | 0.0043 | 0.0074 |
| 20 | 0.0031 | 0.0002 | 0.0037 | 0.0035 | 0.0045 | 0.0069 |
| 21 | 0.0027 | 0.0008 | 0.0043 | 0.0068 | 0.0001 | 0.0059 |
| 22 | 0.0027 | 0.0008 | 0.0040 | 0.0063 | 0.0001 | 0.0056 |
| 23 | 0.0010 | 0.0278 | 0.0357 | 0.0061 | 0.0141 | 0.0066 |
| 24 | 0.0010 | 0.0286 | 0.0338 | 0.0056 | 0.0139 | 0.0062 |
| 25 | 0.0083 | 0.0301 | 0.0297 | 0.0063 | 0.0027 | 0.0025 |
| 26 | 0.0078 | 0.0310 | 0.0277 | 0.0058 | 0.0027 | 0.0023 |
| 27 | 0.0153 | 0.0236 | 0.0115 | 0.0237 | 0.0266 | 0.0034 |
| 28 | 0.0143 | 0.0234 | 0.0309 | 0.0122 | 0.0061 | 0.0048 |
| 29 | 0.0150 | 0.0220 | 0.0355 | 0.0124 | 0.0058 | 0.0051 |

Again, to get a better feel for how this relates to the response variables in the data set, we find the proportional difference. This can be seen in Table 13. Here, the largest proportional difference is 0.0351, and the average proportional difference is 0.0079

Here, we again see that even using different groups of response variables

gives us a similar set of coefficients for the responses, as well as similar responses. Once more, the sets of response variables were completely arbitrary in this example, and should be decided upon by experts (or many trials) to get the best predictions in real analysis. That being said, Curds and Whey does give similar predictions with different sets of response variables in situations where there are more response variables than predictor variables.

TABLE 14

*Agreement of the sign (+/-) for the coefficients of response variables r13-r20, when fit in the groupings r9-r16, r17-r24, and then r13-r20*

| | r13 | r14 | r15 | r16 | r17 | r18 | r19 | r20 |
|---|---|---|---|---|---|---|---|---|
| (Intercept) | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p1 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p2 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p3 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p4 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p5 | TRUE | TRUE | TRUE | TRUE | TRUE | FALSE | FALSE | TRUE |
| p6 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p7 | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |
| p8 | TRUE | TRUE | TRUE | TRUE | TRUE | FALSE | TRUE | TRUE |
| p9 | TRUE | FALSE | TRUE | TRUE | TRUE | TRUE | TRUE | TRUE |

TABLE 15

*Difference between coefficients of response variables r13-r20, when fit in the groupings r9-r16, r17-r24, and then r13-r20*

| | r13 | r14 | r15 | r16 | r17 | r18 | r19 | r20 |
|---|---|---|---|---|---|---|---|---|
| (Intercept) | 0.0140 | 0.0623 | 0.0011 | 0.0141 | -0.1355 | 0.0409 | -0.0557 | -0.1273 |
| p1 | -0.0390 | 0.0530 | -0.0404 | -0.0488 | 0.1200 | 0.0457 | 0.1161 | 0.0833 |
| p2 | 0.0441 | 0.0426 | 0.0335 | 0.0248 | 0.0542 | -0.0120 | 0.0392 | 0.0332 |
| p3 | 0.0363 | 0.0192 | 0.0362 | 0.0015 | 0.1104 | 0.0283 | 0.1494 | 0.0311 |
| p4 | -0.0089 | 0.0908 | 0.0347 | -0.0205 | 0.0099 | 0.0075 | -0.0044 | 0.0496 |
| p5 | -0.0507 | -0.1149 | -0.0568 | -0.0067 | 0.1018 | -0.0665 | 0.0573 | 0.0753 |
| p6 | 0.0243 | -0.0515 | 0.0074 | 0.0004 | 0.0283 | 0.0211 | 0.0088 | 0.0072 |
| p7 | 0.0955 | 0.0074 | 0.0525 | 0.0478 | -0.0764 | -0.0236 | -0.0577 | -0.0980 |
| p8 | 0.0635 | 0.0617 | 0.0436 | 0.0441 | -0.0037 | 0.0458 | 0.0092 | 0.0268 |
| p9 | 0.0893 | 0.1066 | 0.0646 | 0.0622 | 0.0334 | 0.0018 | 0.0061 | 0.0542 |

TABLE 16

*Difference between coefficients of response variables r13-r20, when fit in the groupings r9-r16, r17-r24, and then r13-r20 that had a sign disagreement*

| | r13 | r14 | r15 | r16 | r17 | r18 | r19 | r20 |
|---|---|---|---|---|---|---|---|---|
| (Intercept) | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p1 | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p2 | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p3 | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p4 | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p5 | 0 | 0.0000 | 0 | 0 | 0 | 0.0665 | 0.0573 | 0 |
| p6 | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p7 | 0 | 0.0000 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |
| p8 | 0 | 0.0000 | 0 | 0 | 0 | 0.0458 | 0.0000 | 0 |
| p9 | 0 | 0.1066 | 0 | 0 | 0 | 0.0000 | 0.0000 | 0 |

# 4. R Package

4.1. *Introduction and Motivation.* The Curds and Whey algorithm provides a new approach to multivariate linear regression that could open many new paths for statistical analysis in the Multivariate fields. To make the algorithm more accessible, I have developed functions that will be compiled into a package in the R language (R Core Team, 2013).

With this package, I hope that Curds and Whey will able to be utilized by many to get increased accuracy for predictions and models. In addition, I believe that there are many more areas that Curds and Whey could be implemented such as classification problems and logistic regression. I hope that this R package will allow others to build upon what has already been done and use this information to find further applications that will increase predictive accuracy across many other areas of study within statistics.

4.2. *Functions.* Below are descriptions of and steps to use the functions that have been developed for this R package.

4.2.1. *curds Function.* This is the main function that performs the Curds and Whey algorithm on a data set. It is called using the command `curds(predictors, responses, x10fold)`. `predictors` is a matrix of the predictor variables to be used for the analysis. `responses` is a matrix of the response variables to be used in the analysis. `x10fold` is set to a default value of 1, and is an optional variable to indicate the number of times that 10-fold-cross validation should be performed to find the average percent improvement over ordinary least squares regression.

The `curds` function makes use of a smaller function called `smallCurds`

that performs the majority of the algorithm. `smallCurds` should not be called directly, but rather used simply as part of the bigger `curds` function.

`curds` returns an S4 object named `Curds`. The return values contained in the Curds object can be divided into two main parts. Due to the standardizing done as part of the algorithm, much of the information is initially standardized in the algorithm. To try to increase usability, all of the standardized results are returned as well as their unstandardized counterparts. The returned values are:

- Predicted values

  These can be accessed by `yhat` for the standardized values and `standYhat` for the standardized predicted values.

- Coefficients

  These are the coefficients for the predictor variables. They can be accessed by `coef` for the unstandardized values, and `standCoef` for the standardized values.

- Cross Validated Mean Square Error

  This is the value of the mean square error from predicting onto a test data set from a training data set. This is determined using 10 fold cross validation in the algorithm. This value will be an average of different mean squared errors found by doing `x10fold` number of different repetitions of the cross validation. There are four different mean squared errors returned by the `curds` function. Firstly, we have the mean squared error for the Curds and Whey algorithm. This is accessed by `cvMSE`. For the mean squared error of the standardized data, type `cvMSESTD`. To access the ordinary least squares mean squared

errors, type `olsMSE` for the unstandardized and `olsMSESTD` for the standardized errors.

- Shrinkage Factors

  The shrinkage factors ($d_k$s) from the Curds and Whey algorithm are returned and can be accessed via `shrink`.

- Standardized Data

  As Curds and Whey requires the data to be standardized, the standardized data must be calculated and is returned for other possible uses. It can be accessed by `standard`.

4.2.2. *Plot Method.* Included with this package is a plot function that can be accessed simply by calling `plot(curds)`. `curds` is any Curds object. Plot produces bar graphs showing the improvement gained by using Curds and Whey over ordinary least squares regression. The improvement is measured as the percent improvement in mean squared error. This percentage is found by $100\% * (1 - MSE_{curds}/MSE_{ols})$

4.2.3. *Predict Method.* This method returns predicted values given a new set of predictor variables. It is called by `predict(object, newdata)` where `object` is a Curds object, and newdata is a matrix of new predictor variables of the same dimension as the original predictor variables used to generate the Curds object.

4.2.4. *Summary Method.* To allow for easy viewing of some of the more interesting portions of the output from the `curds` function, the `summary` method can be called. It is called by `summary(object)`, where object is any `Curds` object returned by the `curds` function. The summary function prints

out the percent improvement in mean squared error for Curds and Whey over ordinary least squares regression for each response variable, as well as the average improvement seen over all of the variables. The shrinkage factors for the function are then displayed, along with a note of how to access the coefficients.

4.2.5. *Print Method.* Due to the large amount of data contained within a `curds` object, the `print` method is implemented to reduce frustration at seeing far more information than is desired. It displays the same information as the summary function, and is called simply by typing the name of the `Curds` object you are interested in.

4.2.6. *roundCurds 1 and 2 functions.* The roundCurds functions included in this package are included for possible further application and development in classification.

`roundCurds1` is called by `roundCurds1(yhat, y)` where `yhat` is a matrix of the predicted values from the `curds` function, and `y` is a matrix of actual values of the response variables. The function returns a list with four structures. Three of the four are matrices of responses with different rounding rules applied to them. `mean` contains a matrix of responses where the upper half of each response variable's predicted observations are assigned to presences (1), and the lower half are assigned to absences (0). `prop` contains a matrix of responses determined by assigning the same number of presences and absences as were found in the actual response variables. `arb` contains a matrix of response variables assigned to presence or absences simply dependent on whether their predicted value lies closer to 1 or to 0.

Finally, `propSplit` returns the values used to split the proportion (`prop`) predictions. This can then be passed into the second rounding function, which is called by `roundCurds2(yhat, breakpoints)`. The `yhat` argument is the predicted values from the `curds` function, and `breakpoints` are the points returned by the first rounding function. This function is designed for test data sets where the `breakpoints` argument was obtained from a training data set.

The motivation for this concept is to assume that the test data set will follow a similar distribution as the training set, therefore there should be approximately the same number of presences and absences. Thus, we follow the same break points as we used for the first data set.

4.2.7. *confuse function.* The `confuse` function displays a confusion matrix from the predicted values. It is called by `confuse(actual, pred)` where `actual` is a vector of the values of the actual response variables, and `pred` is a vector of the predicted values. As a note, `confuse` can only handle one response variable at a time. It also displays the percent correctly classified, type I error rate, and type II error rate.

4.3. *Further Work.* Being able to improve upon linear regression with multiple response variables opens up possibilities for future work in other related areas in statistics. One possible field is classification with linear or near-linear classifiers. As most linear classifiers perform at a similar level, any amount of improvement may be a significant contribution to the subject. With regards to what has been found thus far, more work can be done to increase the utility of this procedure to include things such as hypothesis

tests, a fitting criterion (AIC, etc.), and algorithms to assist with variable groupings when the number of response variables exceeds the number of predictor variables.

# References


Aldrin, M. (1996). Moderate projection pursuit regression for multivariate response data. *Computational Statistics & Data Analysis* **21** 501–531.

Anderson, T. W. (1984). *An Introduction To Multivariate Statistical Analysis*, Second ed. Wiley, New York, NY.

Breiman, L. and Friedman, J. H. (1997). Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **59** 3–54.

Copas, J. B. (1983). Regression, prediction and shrinkage. *Journal of the Royal Statistical Society. Series B. Methodological* **45** 311–354.

D'Ambra, L. and Lombardo, R. (1999). Predicting multivariate responses in non-linear regression. *Bull. Int'l Statistical Inst.*

Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: biased estimation for nonorthogonal problems. *Technometrics* **12** 55–67.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika* **28** 321–377.

James, W. and Stein, C. (1961). Estimation with quadratic loss. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability* **1** 361–379.

Liu, Y., Wu, F., Zhang, Z., Zhuang, Y. and Yan, S. (2010). Sparse representation using nonnegative curds and whey. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* 3578–3585.

Massy, W. F. (1965). Principal components regression in exploratory statistical research. *Journal of the American Statistical Association* **60** 234–256.

Pearson, K. (1914). *The life, letters and labours of Francis Galton*. Cambridge, UK: University Press.

Skagerberg, B., MacGregor, J. F. and Kiparissides, C. (1992). Multivariate data analysis applied to low-density polyethylene reactors. *Chemometrics and Intelligent Laboratory Systems* **14** 341–356.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)* **36** 111–147.

R Core Team (2013). R: A Language and Environment for Statistical Computing R Foundation for Statistical Computing, Vienna, Austria ISBN 3-900051-07-0.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)* **58** 267–288.

Venables, W. N. and Ripley, B. D. (2002). *Modern applied statistics with S*, Fourth ed. New York: Springer.

Wold, H. (1975). Soft modelling by latent variables: the non-linear iterative partial least squares (NIPALS) approach. *Perspectives in Probability and Statistics, in Honor of MS Bartlett* **1** 117–144.

Xu, Q., De Jong, S., Lewi, P. and Massart, D. (2004). Partial least squares regression with Curds and Whey. *Chemometrics and Intelligent Laboratory Systems* **71** 21–31.

# APPENDIX

FIGURE 15. *Proportion improvement with 10 predictors and 10 responses*

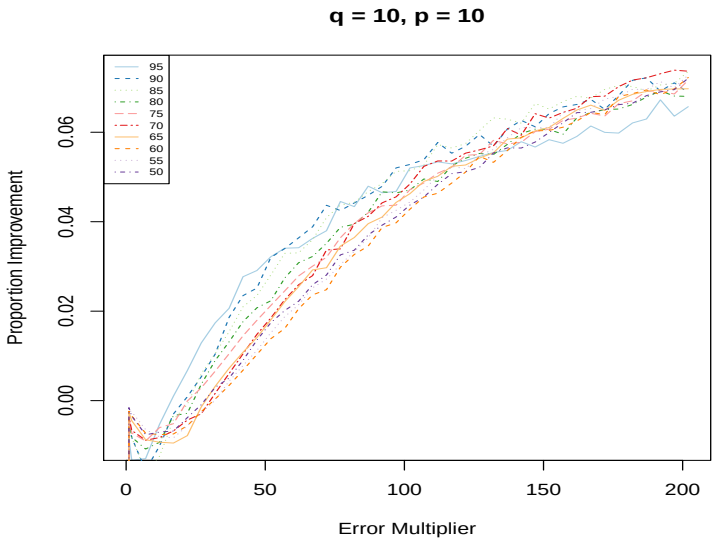



FIGURE 16. *Proportion improvement with 25 predictors and 10 responses*

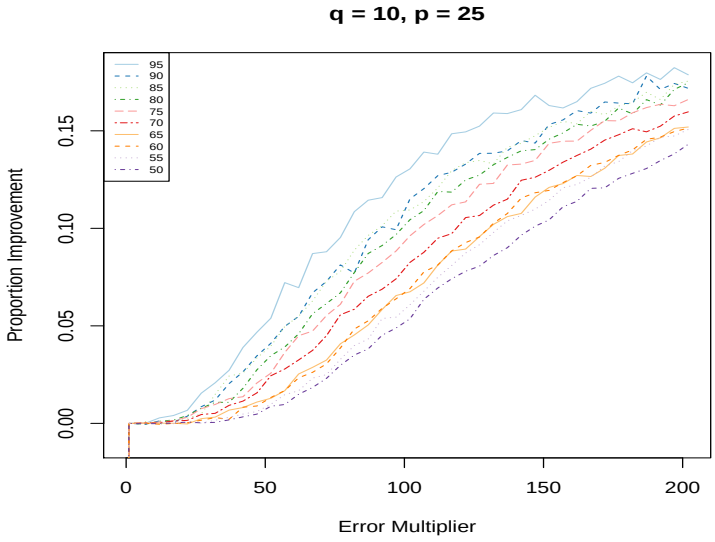


FIGURE 17. *Proportion improvement with 50 predictors and 25 responses*

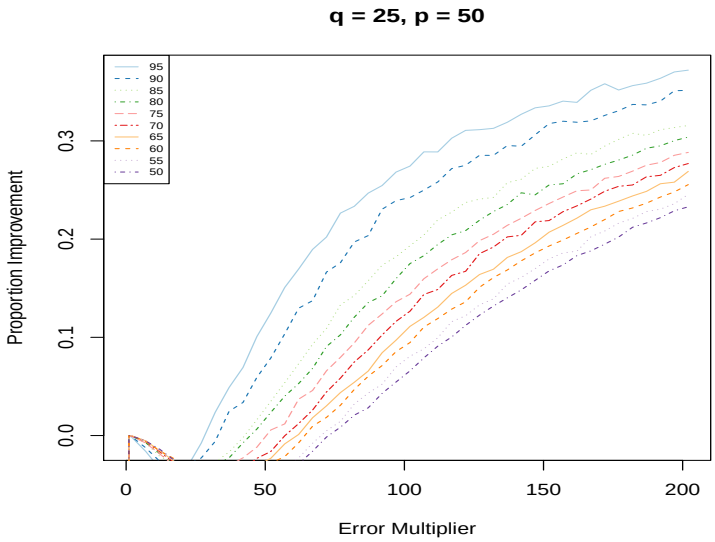



FIGURE 18. *Proportion improvement with 50 predictors and 50 responses*

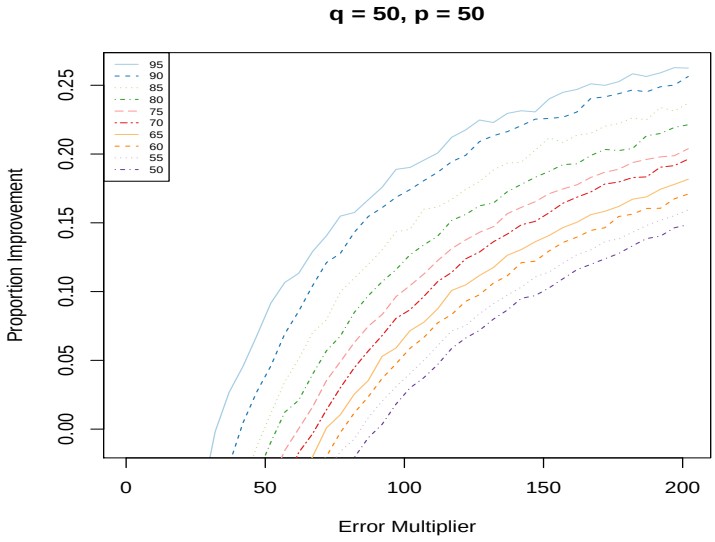


FIGURE 19. *Proportion improvement with irrelevant variables, with 10 predictors and 10 responses*

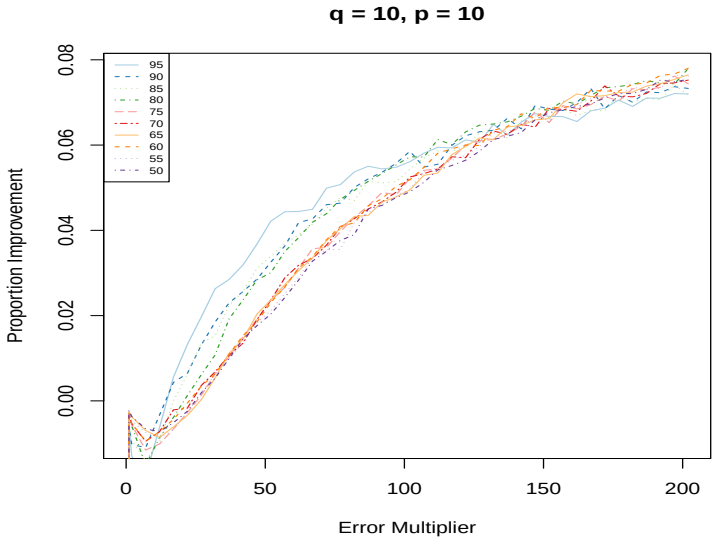



FIGURE 20. *Proportion improvement with irrelevant variables, with 25 predictors and 10 responses*

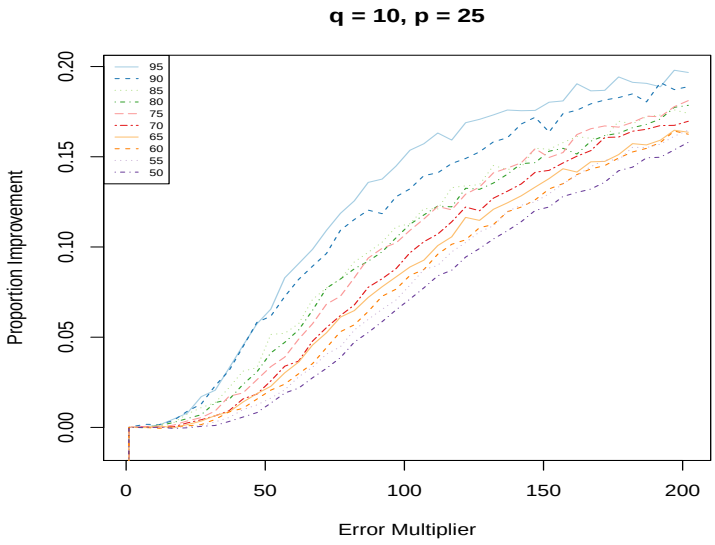


FIGURE 21. *Proportion improvement with irrelevant variables, with 50 predictors and 25 responses*

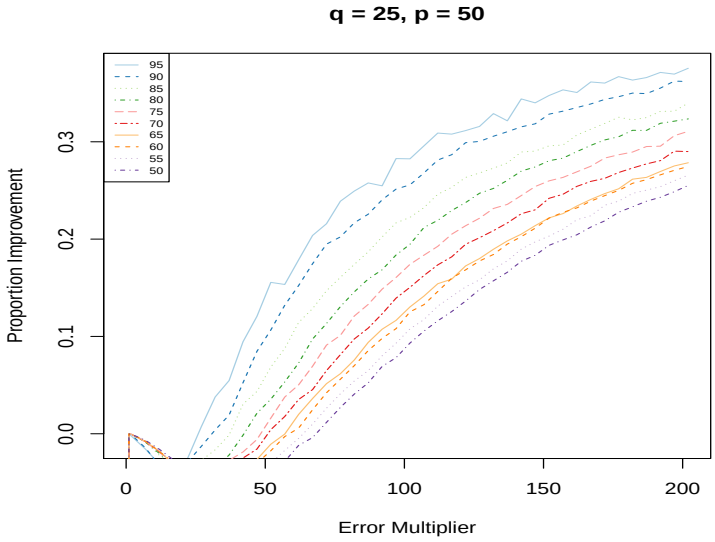



FIGURE 22. *Proportion improvement with irrelevant variables, with 50 predictors and 50 responses*