Fall Technical Conference
Statistics & Quality: Riding the Big Data Wave
October 3 - 5, 2018 | West Palm Beach, FL

Model Validation Strategies for Designed Experiments

Dr. Phil Ramsey
University of New Hampshire
Durham, NH
philip.ramsey@unh.edu

Dr. Chris Gotwalt
JMP Division, SAS Institute
Cary, NC

INTRODUCTION

Designed experiments (DOEs) are a best practice for product and process improvement.

There is ever increasing interest in the use of “Big Data” methods to build predictive models.

A inherent feature of “Big Data” methods is that they take advantage of the data’s “Bigness.”

Basic “Big Data” modeling strategy fits or “trains” the model one subset of the data and uses a second subset to evaluate and select that model.

To what extent can we apply methods from “Big Data” to smaller datasets typical of a designed experiments

INTRODUCTION

Designed Experiments and “Big Data” modeling exercises share some common problems

In both cases we want:

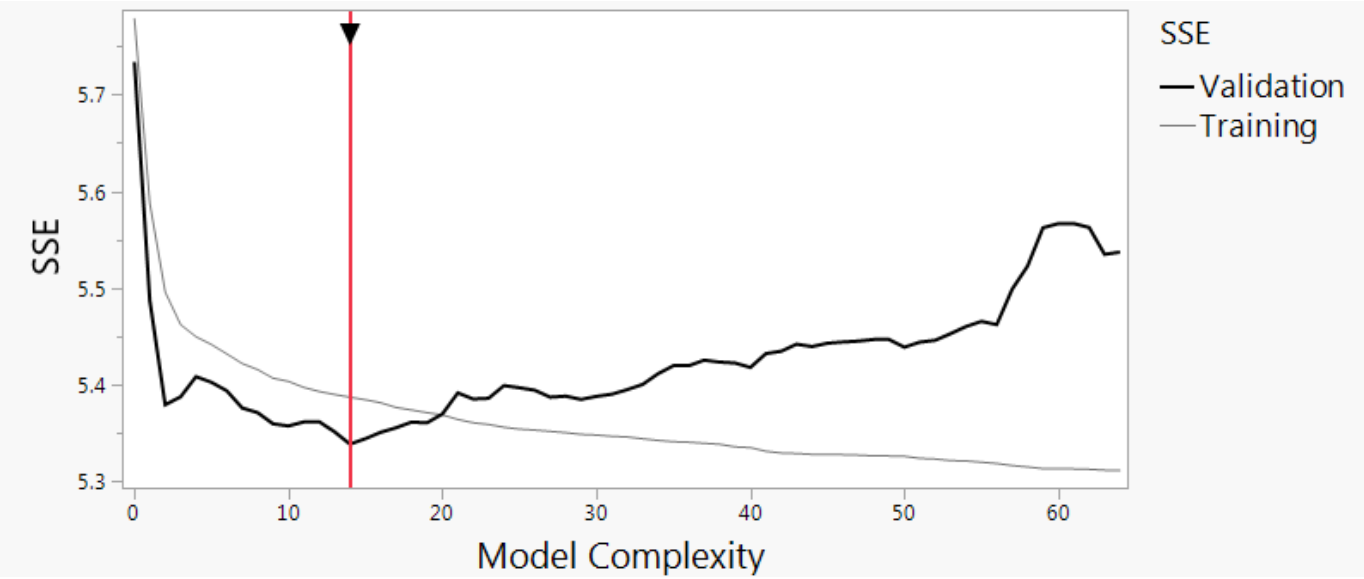
- To find a model that accurately predicts how a response variable will change as a function of input variables
- To separate the variables that are highly predictive of the response from the ones that are less relevant (variable or feature selection)
- The actual methods for deciding on a model are often quite different, which is largely a reflection of the size of the datasets.

"BIG DATA" MODEL SELECTION

Partition data into non-overlapping "Training" and "Validation" subsets. ***Rows are in one or the other set, but not both*** because we want an independent assessment of model fit.

We fit models using the training data using a variety of methods and choose the one that has the best performance on the Validation set.

Appealing, direct approach to finding models that will predict well.



"BIG DATA" MODEL SELECTION

Building predictive models is a bias-variance tradeoff exercise.

Smaller models may have high bias, but low prediction variance.

Larger models have little or no bias, but high prediction variance.

The goal is to find the models that provide the best tradeoff in bias and prediction variance; i.e., low Mean Square Prediction Error.

Under fitting can often be detected with the use of various residual plots and lack of fit tests.

Over fitting is problematic given the models fit the training data very well (e.g., high R^2), but may have large prediction errors on validation data that was not used in fitting those models.

Without a validation set available to evaluate fitted models, it is very difficult to know with certainty that a model is over fit.

DOE MODEL SELECTION

A tremendous amount of research into variable selection for DOEs has continued over many decades. Some ideas:

- Fit a “large” model to the data, remove the terms with large p-values;
- Bayes and Half-Normal plots to visually screen predictors;
- Penalized Regression such as Lasso and Dantzig Selector using AICc/BIC or similar objective functions;
- Stepwise/Best Subset Selection with AICc/BIC;
- Novel methods that exploit the special structure of specific designs (Jones and Nachtsheim 2017).

DOE MODEL SELECTION

Why don't we use holdback crossvalidation for DOE model selection?

The simple reason is that DOEs are carefully constructed datasets that are designed to extract as much information as possible in the smallest number of observations (e.g., Definitive Screening Designs)

- Removing rows fundamentally alters the structure of DOE data;
- Introduces singularities, making the set of identifiable models smaller;
- Induced aliasing fundamentally changes the meaning of the parameters;

As a result, we use all the data to fit the models and use methods like AICc, BIC, and p-values to select models because we cannot afford to hold observations back for validation.

DOE MODEL SELECTION

Why not just use the same data for Training and Validation?

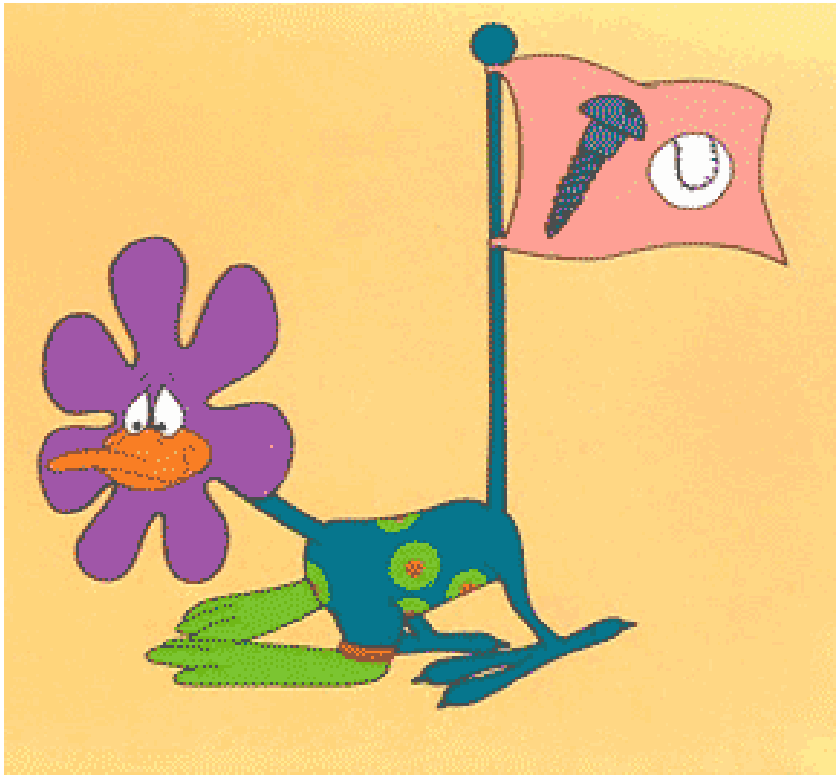
	X1	X2	X3	Y
1	-1	0	1	11.17
2	1	0	0	11.90
3	-1	1	-1	14.45
4	0	0	0	9.00
5	-1	-1	0	9.43
6	1	-1	1	9.81
7	0	-1	-1	10.33
8	0	1	1	13.27

	X1	X2	X3	Y	Validation
1	-1	0	1	11.17	Training
2	-1	0	1	11.17	Validation
3	1	0	0	11.90	Training
4	1	0	0	11.90	Validation
5	-1	1	-1	14.45	Training
6	-1	1	-1	14.45	Validation
7	0	0	0	9.00	Training
8	0	0	0	9.00	Validation
9	-1	-1	0	9.43	Training
10	-1	-1	0	9.43	Validation
11	1	-1	1	9.81	Training
12	1	-1	1	9.81	Validation
13	0	-1	-1	10.33	Training
14	0	-1	-1	10.33	Validation
15	0	1	1	13.27	Training
16	0	1	1	13.27	Validation

DOE MODEL SELECTION

Why not just use the same data for Training and Validation?

Because that's just crazy!



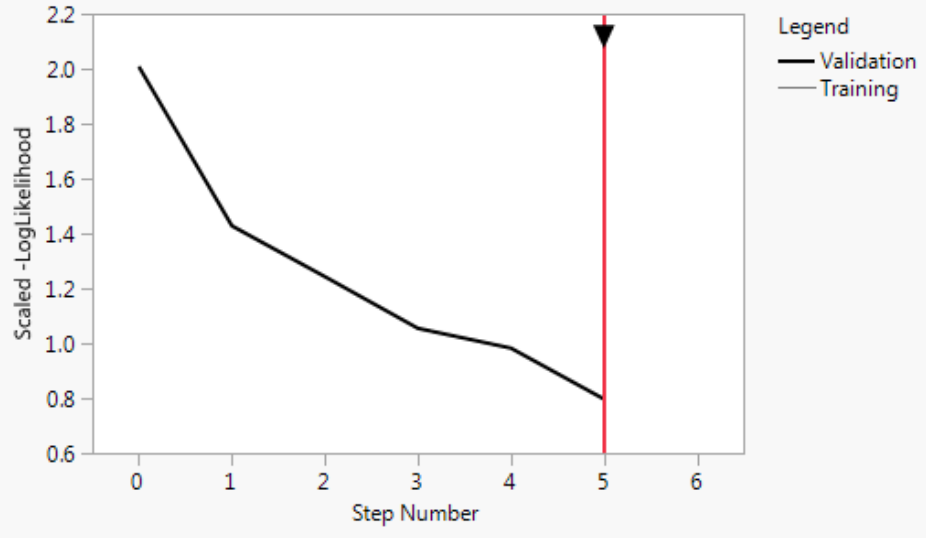
DOE MODEL SELECTION

Why not just use the same data for Training and Validation?

Because that's just crazy!

No independent assessment of goodness of fit since training and validation are the same: *it's the exact opposite of what we want*!

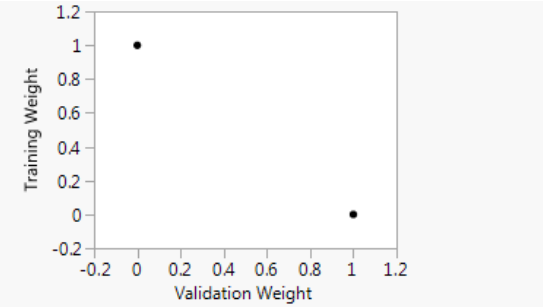
Selection algorithms would always choose the most complicated model possible that will utterly over-fit the data.



CROSS-VALIDATION AS A WEIGHTING SCHEME

Cross-validation is the same as having two sets of weights.

	X1	X2	X3	Y	Training Weight	Validation Weight	Validation
1	1	-1	-1	13.1	1	0	Training
2	1	1	-1	10.1	1	0	Training
3	1	1	1	9.95	1	0	Training
4	-1	1	-1	6.01	1	0	Training
5	-1	-1	-1	10.2	1	0	Training
6	-1	-1	1	8.87	1	0	Training
7	-1	1	1	5.57	1	0	Training
8	1	-1	1	11.9	1	0	Training
9	1	-1	-1	13.3	0	1	Validation
10	1	1	-1	9.29	0	1	Validation
11	1	1	1	9.53	0	1	Validation
12	-1	1	-1	4.29	0	1	Validation
13	-1	-1	-1	9.99	0	1	Validation
14	-1	-1	1	10.1	0	1	Validation
15	-1	1	1	7.47	0	1	Validation
16	1	-1	1	13.5	0	1	Validation



Summary Statistics

	Value	Lower 95%	Upper 95%	Signif. Prob
Correlation	-1	-1	-1	<.0001*

$SSE^T(\beta) = \sum_i w_i^T (y_i - f(X_i, \beta))^2$ is minimized by estimating β.

$SSE^V(\beta) = \sum_i w_i^V (y_i - f(X_i, \beta))^2$ is used to evaluate models.

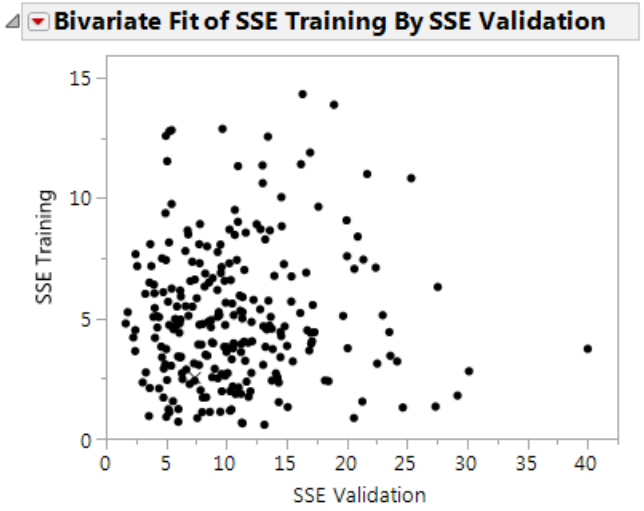
w_i^T (training weight) is 1.0 whenever w_i^V (validation weight) is 0.0 and vice versa, giving us an independent assessment of goodness of fit.

CROSSVALIDATION AS A WEIGHTING SCHEME

To illustrate, fit 250 models (single models, no variable selection) and compare the SSEs on Training (8 obs) and Validation (8 obs).

SSEs are uncorrelated (as expected).

Validation SSEs bigger on average then SSE Training (as expected).



Summary Statistics

	Value	Lower 95%	Upper 95%	Signif. Prob
Correlation	0.033888	-0.09056	0.157294	0.5938

Variable	Mean	Std Dev
SSE Validation	10.69826	5.83895
SSE Training	5.052329	2.84495

CROSSVALIDATION AS A WEIGHTING SCHEME

If we assume β is known, errors are iid normal, and both sets of weights have the same marginal distribution with mean and variance equal to 1.0:

$$Cov(SSE^T(\beta), SSE^V(\beta)) \propto E(w_i^T w_i^V)$$
$$\propto Corr(w_i^T, w_i^V) + 1$$

This suggests that the more anticorrelated w_i^T and w_i^V are the more nearly independent $SSE^T(\beta)$ and $SSE^V(\beta)$ will be!

CROSSVALIDATION AS A WEIGHTING SCHEME

What if we used the same data for Training and Validation, but used anti-correlated non-integer weights?

Observations that contribute more to the Training SSE will contribute less to the Validation SSE, and vice versa.

No observations are removed completely from Training and Validation and the essential structure of the design is entirely intact!

FRACTIONALLY WEIGHTED BOOTSTRAP

The Fractionally Weighted Bootstrap (FWB) suggests an approach.

In the standard FWB we generate weights that are Gamma distributed

Create exponentially distributed weights by the *probability integral transform:*

$u_i \sim$ Uniform(0,1)

$w_i^T = Exponential\ Quantile(\ u_i\ , 1)$

$w_i^{Training}$ has a Exponential distribution with mean and variance 1.0.

$1 - u_i$ is also Uniform(0,1) and perfectly anticorrelated with u_i, therefore,

$w_i^V = Exponential\ Quantile(\ 1 - u_i\ , 1)$

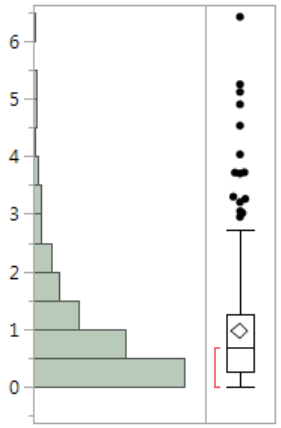
w_i^V will be highly anticorrelated with w_i^T and yet has the exact same distribution!

FRACTIONALLY WEIGHTED BOOTSTRAP PAIRS

250 FWB Pairs using an Exponential(1)

The pairs are identically distributed, and negatively correlated

Training Weight



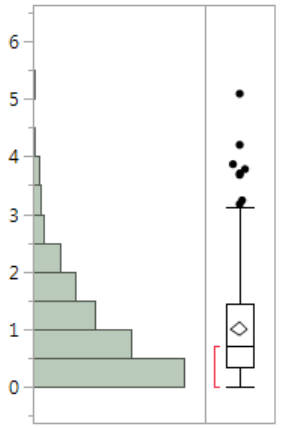
Quantiles

100%	maximum	6.4215486218
75%	quartile	1.2613890694
50%	median	0.6797202993
25%	quartile	0.2663855736
0%	minimum	0.0061940315

Summary Statistics

Mean	0.9687426
Std Dev	1.0295733

Validation Weight



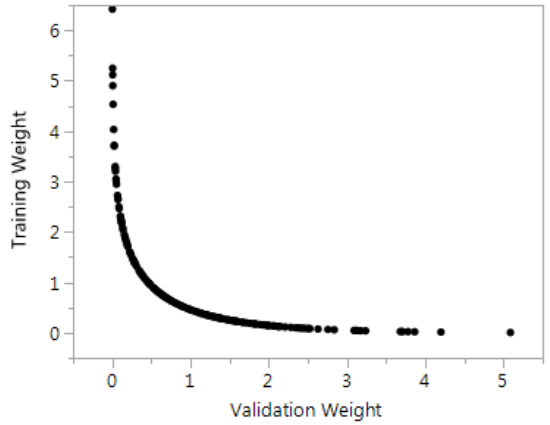
Quantiles

100%	maximum	5.0872645349
75%	quartile	1.4531939938
50%	median	0.7068156829
25%	quartile	0.3333945184
0%	minimum	0.0016274596

Summary Statistics

Mean	1.0064644
Std Dev	0.9014337

Bivariate Fit of Training Weight By Validation Weight



Summary Statistics

	Value	Lower 95%	Upper 95%	Signif. Prob
Correlation	-0.66743	-0.73097	-0.59242	<.0001*

Variable	Mean	Std Dev
Validation Weight	1.006464	0.901434
Training Weight	0.968743	1.029573

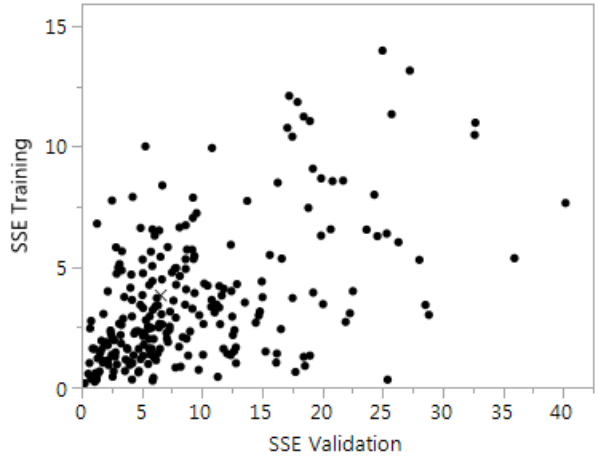
AUTO-VALIDATION

Fit 250 models and calculate SSEs using same data for Training and Validation, but using anti-correlated FWB Pairs as weights.

SSEs are somewhat correlated (as expected).

Validation SSEs bigger on average then SSE Training (as expected).

Bivariate Fit of SSE Training By SSE Validation



Summary Statistics

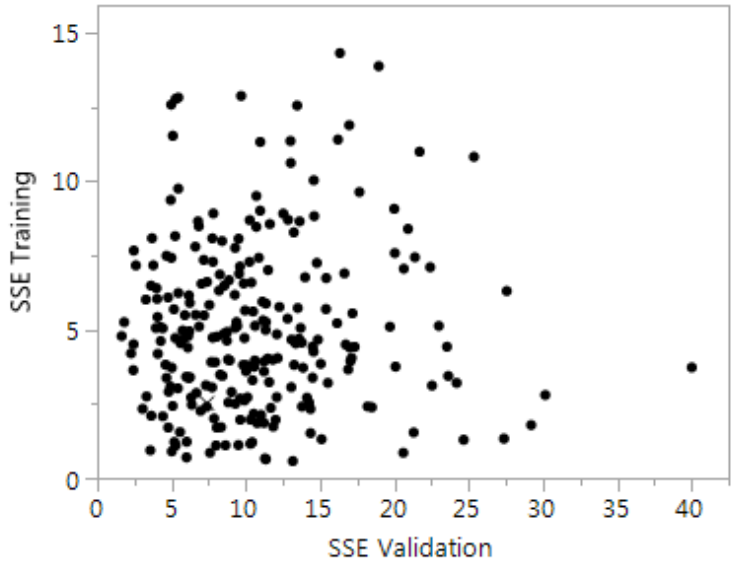
	Value	Lower 95%	Upper 95%	Signif. Prob
Correlation	0.524757	0.427758	0.609807	<.0001*

Variable	Mean	Std Dev
SSE Validation	9.659373	8.334366
SSE Training	3.758965	3.072835

AUTO-VALIDATION

Comparison of results with ‘true’ holdback validation, and autovalidation.

Bivariate Fit of SSE Training By SSE Validation

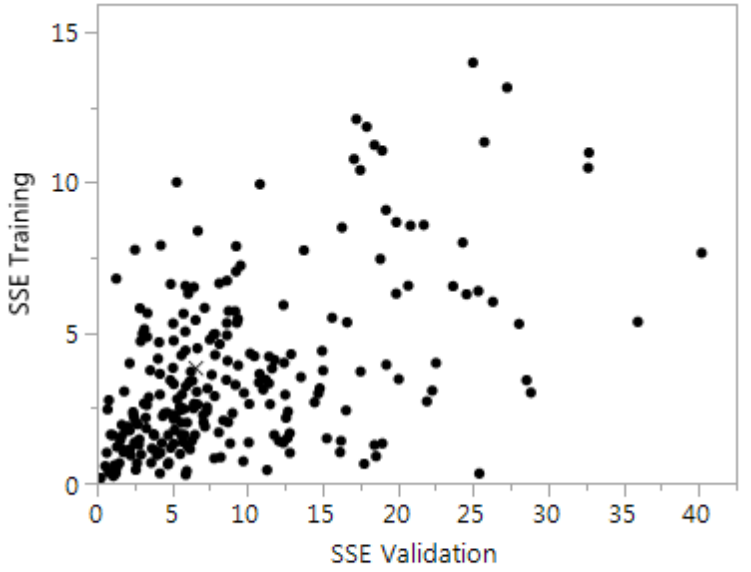


Summary Statistics

	Value	Lower 95%	Upper 95%	Signif. Prob
Correlation	0.033888	-0.09056	0.157294	0.5938

Variable	Mean	Std Dev
SSE Validation	10.69826	5.83895
SSE Training	5.052329	2.84495

Bivariate Fit of SSE Training By SSE Validation



Summary Statistics

	Value	Lower 95%	Upper 95%	Signif. Prob
Correlation	0.524757	0.427758	0.609807	<.0001*

Variable	Mean	Std Dev
SSE Validation	9.659373	8.334366
SSE Training	3.758965	3.072835

AUTO-VALIDATION

To use the same data for both training and validation set up a column of weights, as described, that are anti-correlated FWB pairs.

Then use a variable selection technique such as Forward Selection with the weights applied to the response column.

Next, perform a large number of iterations of the technique with new weights generated on each iteration and keep track of the number of times an effect is selected.

Effects not selected on each iteration are assigned a 0 value, else the estimated coefficient value is recorded.

	Y	Proportion Nonzero
1	X1	0.971
2	X2	0.744
3	X3	0.202
4	X1*X2	0.387
5	X1*X3	0.056
6	X2*X3	0.105

This begs the question – What proportion of the time should an effect be in the model to be considered important?

NULL FACTORS

We can calibrate the proportion of times an effect needs to be in the model to be considered important using what we call a *Null Factor.*

Same basic idea as Y. Wu, Boos, and Stefanski (2007)

A Null Factor is a randomly distributed column that is independent of the response Y.

We know that there is no relationship between Y and the Null Factor.

	X1	X2	X3	Y	Paired Fractionally Weighted Bootstrap Weight	Validation	Null Factor
1	1	-1	-1	15	0.44	Training	0.48
2	1	1	-1	8.92	0.5637	Training	0.23
3	1	1	1	10.9	2.061	Training	-0.6
4	-1	1	-1	6.55	1.5338	Training	0.46
5	-1	-1	-1	7.89	0.1215	Training	-1
6	-1	-1	1	8.16	1.0174	Training	-0.1
7	-1	1	1	7.98	2.0052	Training	-0.2
8	1	-1	1	16.3	2.9025	Training	0.04
9	1	-1	-1	15	0.3135	Validation	0.48
10	1	1	-1	8.92	0.1861	Validation	0.23
11	1	1	1	10.9	2.6602	Validation	-0.6
12	-1	1	-1	6.55	1.1736	Validation	0.46
13	-1	-1	-1	7.89	0.1151	Validation	-1
14	-1	-1	1	8.16	0.0072	Validation	-0.1
15	-1	1	1	7.98	0.4128	Validation	-0.2
16	1	-1	1	16.3	0.3455	Validation	0.04

NULL FACTORS

Include the Null Factor in the model and the Null Factor is reset to new random values during each simulation iteration.

We know that the Null Factor is independent of Y, so any effect that appears as or less often than the Null Factor also has as weak a relationship to Y.

This means that any effect that enters the model as or less often then the Null Factor is a candidate be dropped from a final model.

In this example X2*X3 and X1*X3 are candidates to be removed from the final model.

Model: X1, X2, X3, Null Factor, X1*X2, X1*X3, X2*X3

	Y	Proportion Nonzero
1	X2	0.912
2	X1	0.908
3	X1*X2	0.722
4	X3	0.282
5	Null Factor	0.246
6	X2*X3	0.133
7	X1*X3	0.093

Applying Autovalidation or FWB to Actual Experiments

We present three case studies to demonstrate the technique of autovalidation and fractionally weighted bootstrapping.

The JMP Pro® 14 software is used in the following analyses.

Case Study 1: A six factor DSD is used to develop a formylation process to prevent inter and intra crosslinking of therapeutic proteins produced in a bio-process.

Case Study 2: A five factor DSD was run concurrently with a CCD to optimize the performance of an HPAE analytic method to glycoprofile therapeutic proteins produced in a bio-process. The CCD forms a natural validation set.

Case Study 3: A D-optimal mixture-amount experiment was performed to assess the possibility of using fly ash effluent from coal fired power plants as an ingredient in concrete – possible way to safely dispose of effluents.

Case Study 1: The formylation Process DSD

Cross-linking is a problem in the production of therapeutic proteins such as monoclonal antibodies. The cross-linking makes the affected proteins non-therapeutic and may cause undesirable immune system responses in the host.

The case study was from an experiment in 2011 and the first published case study applying a DSD to develop a formylation process to prevent cross-linking in therapeutic proteins.

Erler, DeMas, Ramsey, and Henderson (2011). *Efficient biological process characterization by definitive-screening designs: the formaldehyde treatment of a therapeutic protein as a case study.* Biotechnology Letters.

Case Study 1: The formylation Process DSD

The experimental factors and response are:

- **Cprotein** = initial concentration of the target protein;
- **Clysine** = lysine concentration in the solution;
- **Duration** = time duration of the reaction.
- **Temperature** = temperature of the solution.
- **pH** = solution pH.
- **HCHO:Protei**n = ratio of formaldehyde to target protein in the solution.
- The response is the **extent of polymerization** and smaller values are desirable (there is a problem with intra and inter cross-linking of the proteins during extraction).

Case Study 1: The formylation Process DSD

A 13 run DSD was generated and 4 center runs were added for a total of 17 runs in the experiment.

The design is supersaturated for the full quadratic model, which has 27 potential effects plus the intercept.

A null factor (standard normal variate) is added to the model for the autovalidation.

We use autovalidation and **Two-Stage Forward Selection** to search for the important factors to predict extent of polymerization.

Based upon 2500 autovalidation runs the number of times an effect entered the model compared to the null factor was expressed as a ratio and a Pareto Plot generated; other approaches can be used.

The **Nzero** (or **Prop Nzero**) function in JMP returns the number of times an effect was not included (or included) in the model.

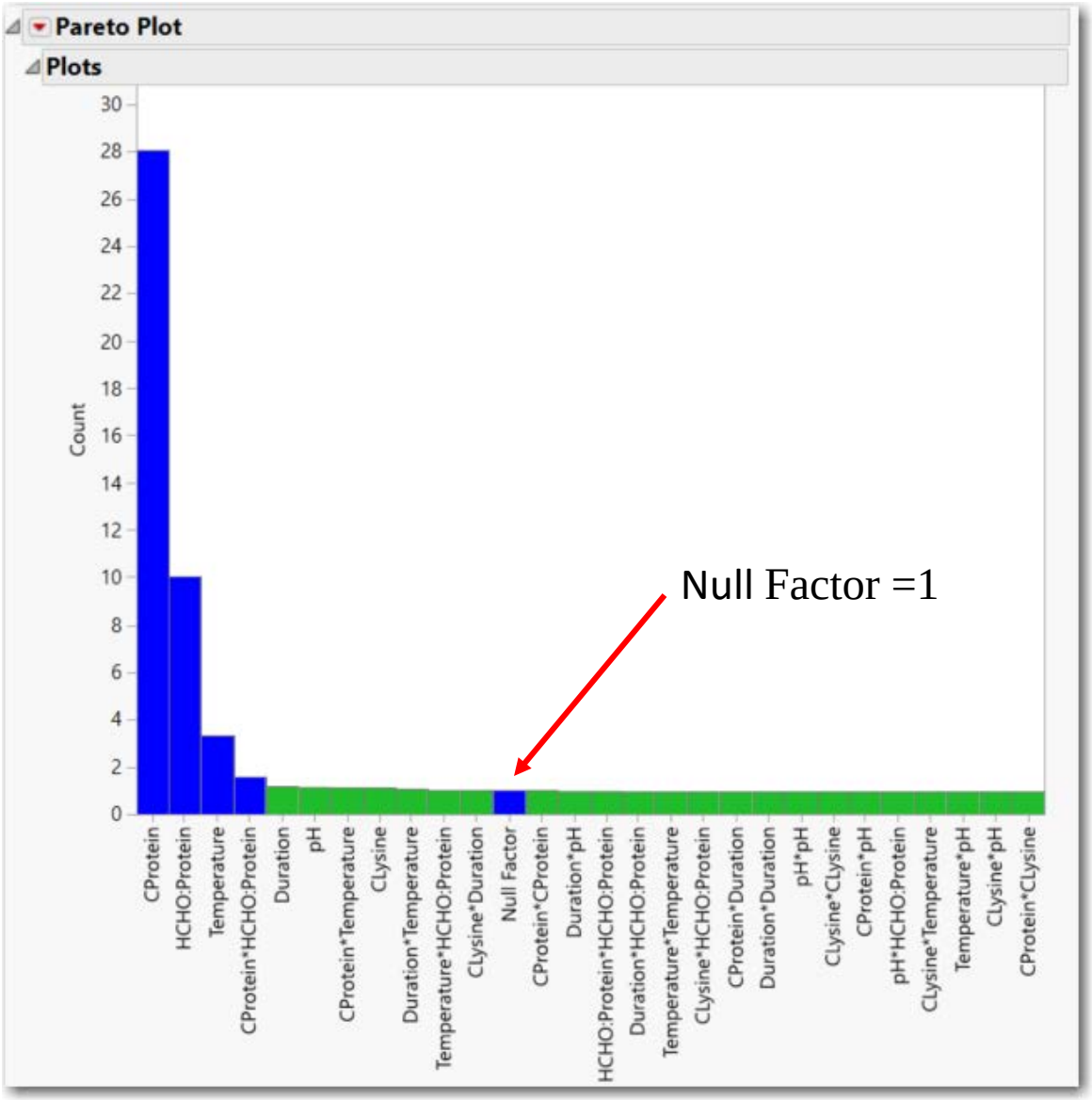
Case Study 1: The formylation Process DSD

Below is a weighted Pareto Plot of the results, notice that four important factors are detected.

$$Wt_i = \frac{NZero\left(Null\ Factor\right)}{NZero\left(Effect_i\right)}$$

At this point a model can be fit to these four effects.

Hopefully at some point confirmation trials can be performed.



Case Study 1: The formylation Process DSD

Since the DSD with K = 6 main effects resolved to only K = 3 main effects it is possible to estimate the full quadratic model. Below is the final model estimated using Two Stage Forward Selection.

Parameter Estimates for Original Predictors

Term	Estimate	Std Error	Prob > ChiSquare
Intercept	14.835294	0.4941512	<.0001*
CProtein	10.62	0.6442943	<.0001*
Temperature	3.9	0.6442943	<.0001*
HCHO:Protein	5.46	0.6442943	<.0001*
CProtein*HCHO:Protein	1.875	0.720343	0.0092*
Scale	2.0374376	0.4674202	<.0001*

Correlation of Estimates

Corr

	Intercept	CProtein	Temperature	HCHO:Protein	CProtein*HCHO:Protein
Intercept	1.0000	0.0000	0.0000	0.0000	0.0000
CProtein	0.0000	1.0000	-0.0000	-0.0000	0.0000
Temperature	0.0000	-0.0000	1.0000	0.0000	-0.0000
HCHO:Protein	0.0000	-0.0000	0.0000	1.0000	-0.0000
CProtein*HCHO:Protein	0.0000	0.0000	-0.0000	-0.0000	1.0000
Scale	-0.0000	-0.0000	0.0000	0.0000	-0.0000

Case Study 2: Analytic Method Development DSD

Acknowledgement: This experimental work was performed by **Eliza Yeung, Ph.D.**, **Cytovance Biologics, Oklahoma City, OK, USA**. Much thanks to Eliza for the great deal of experimental work and scientific explanations and to Cytovance for sharing the data.

Glycosylation of therapeutic proteins is an important step in a bio-process and the amount and structure of the glycoforms is important to the therapeutic effect.

A five factor DSD was used to try and optimize an HPAE – PAD method to glycoprofile proteins post transcription and processing.

For validation of the DSD, **a five factor CCD was run in parallel** with the DSD giving us the **opportunity to compare autovalidation with actual validation**.

Case Study 2: Analytic Method Development DSD

The experimental factors and levels are:

Factor (level)	-1	0	1
Initial %NaOAc (% A)	0	10	20
Initial %NaOH (% B)	30	40	50
Gradient_01 (mM NaOAc /min)	0.415	1.25	2.085
Gradient_02 (mM NaOAc /min)	1.25	2.085	2.915
Gradient_03 (mM NaOAc /min)	4.72	5.555	6.39

Case Study 2: Analytic Method Development DSD

There are five glycoforms that are most important resulting in seven responses. The key response is retention time for glycoform 3.

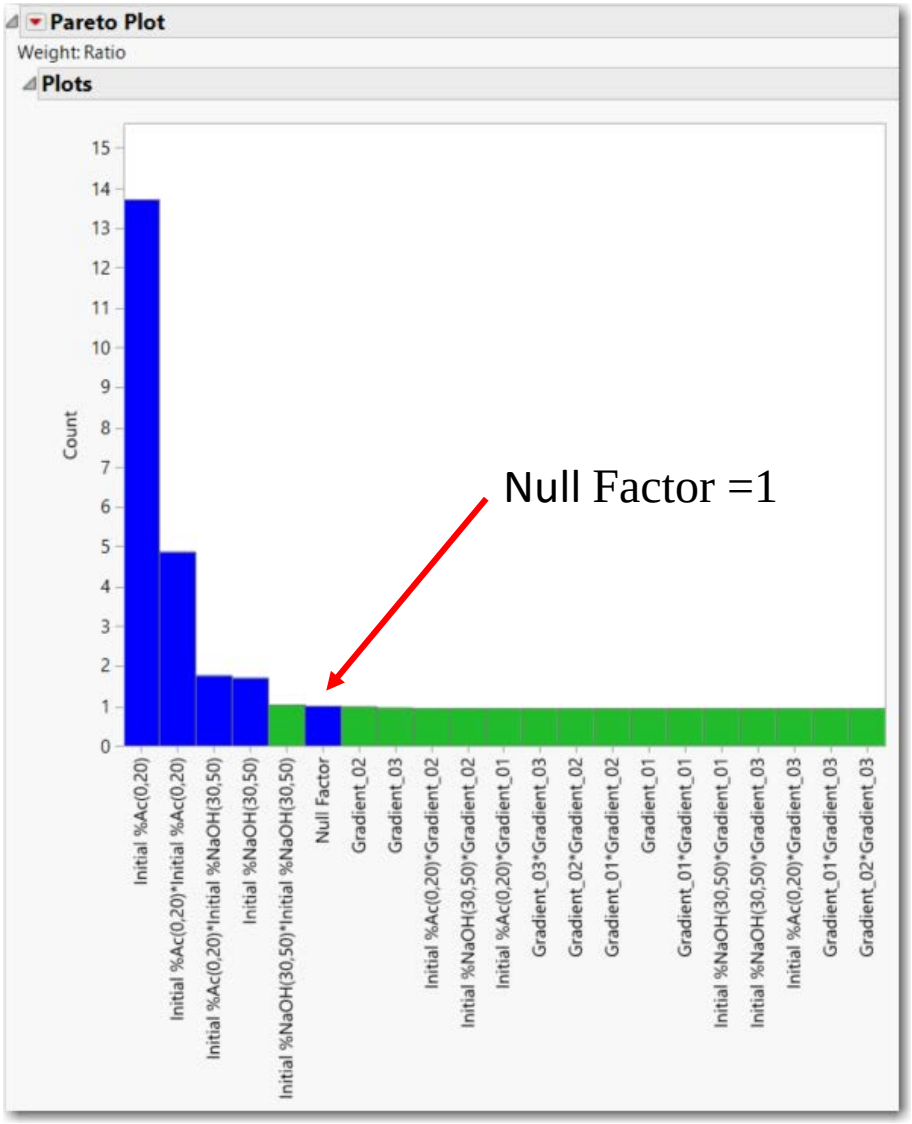
Response	Description	Optimization
RT_G03	Retention Time	Target ~ 8.5 min
Resol_G03	Resolution G03-G04	Maximize
Resol_G04	Resolution G04-G05	Maximize
Resol_G05	Resolution G05-G06	Maximize
Resol_G09	Resolution G09-G10	Maximize
Resol_G10	Resolution G10-G11	Maximize
USP Tailing	USP Tailing G04	Monitor (0.8-1.2)

Case Study 2: Analytic Method Development DSD

Using autovalidation on the DSD data four important effects are identified.

All of the effects involving the Gradient factors are rejected.

This is important because glycoform 3 elutes before the gradient dilutions are applied and therefore those effects cannot impact the retention time.



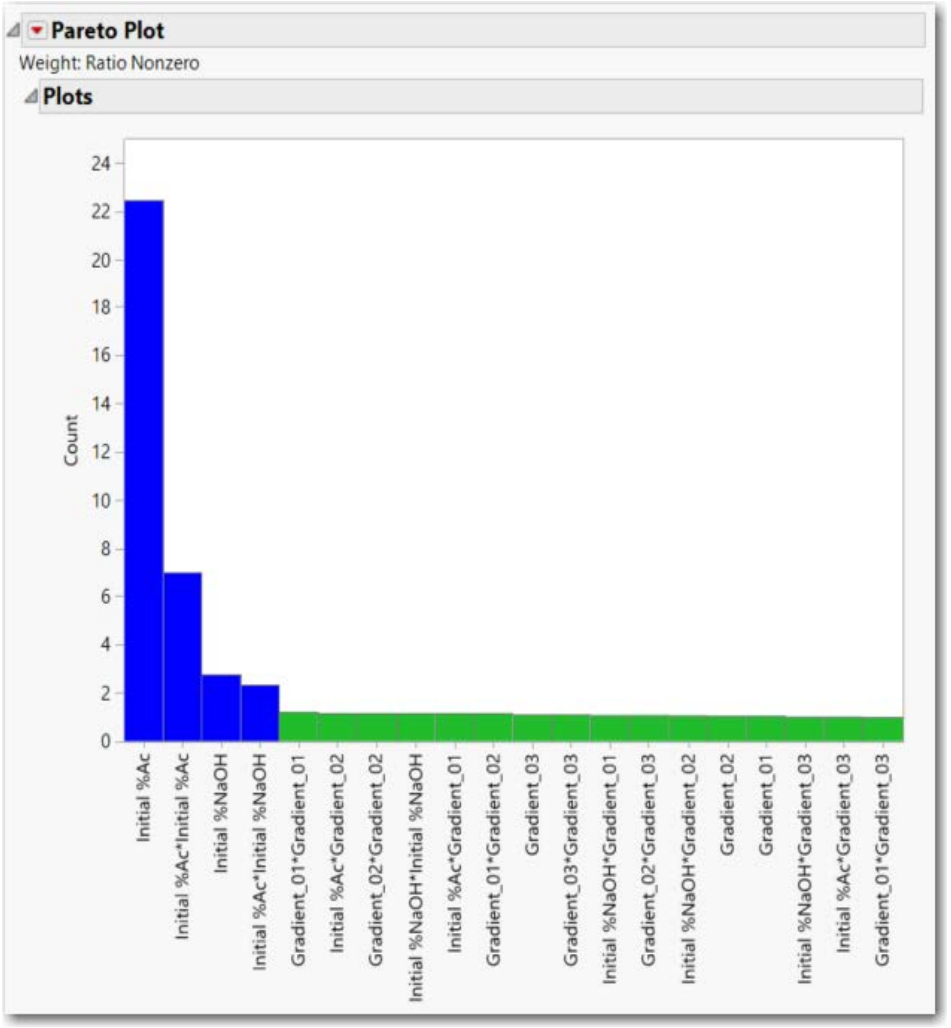
Case Study 2: Analytic Method Development DSD

Next we apply FWB with DSD as the training set and the CCD as the validation set.

Even if a validation set exists, the FWB method may still yield excellent results.

The results, with Two Stage Forward, are identical to the autovalidation on the DSD alone.

Autovalidation and actual validation have led to the same conclusion.

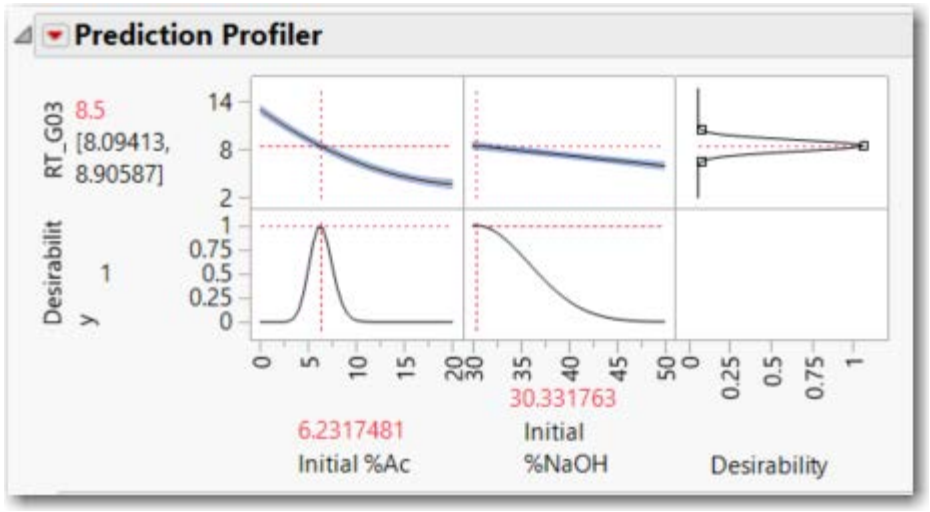


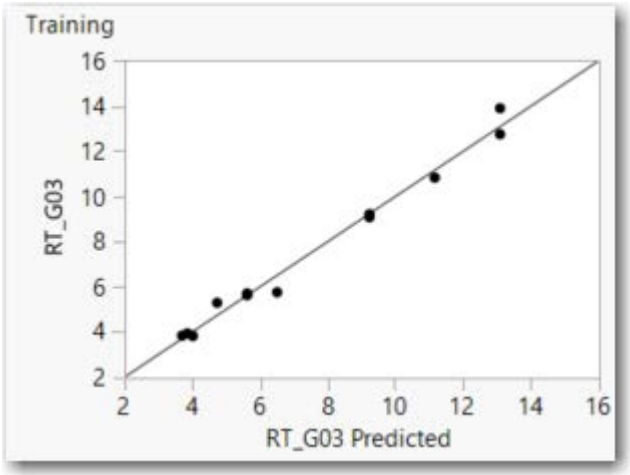
Case Study 2: Analytic Method Development DSD

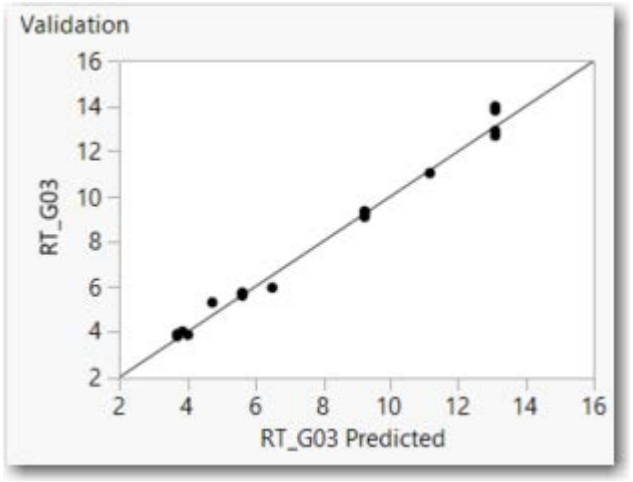
The final fitted model using only the DSD training data, Actual by Predicted Plots and optimized settings to achieve a target RT_03 = 8.5 minutes; note how well the model fits the CCD validation data.

Parameter Estimates for Original Predictors

Term	Estimate	Std Error	Wald ChiSquare	Prob > ChiSquare
Intercept	12.8036	0.5670175	509.88387	<.0001*
Initial %Ac	-0.3645	0.0131239	771.38096	<.0001*
Initial %NaOH	-0.08834	0.0131239	45.309434	<.0001*
(Initial %Ac-10)*(Initial %NaOH-40)	0.0104163	0.0014673	50.394905	<.0001*
(Initial %Ac-10)*(Initial %Ac-10)	0.018866	0.0021431	77.493411	<.0001*







Case Study 3: Mixture Amount Experiment – Fly Ash

A mixture – amount experiment was performed to determine if fly ash from coal fired plants could be used as an ingredient in concrete; potential way to safely dispose of an effluent.

The mixture factors are

- **Limestone**
- **Gypsum**
- **Fly Ash**

The **Amount** factor is the total amount of the mixture added to a standard cement paste.

The response is the peak **temperature** reached in an exothermic reaction in the concrete mixture.

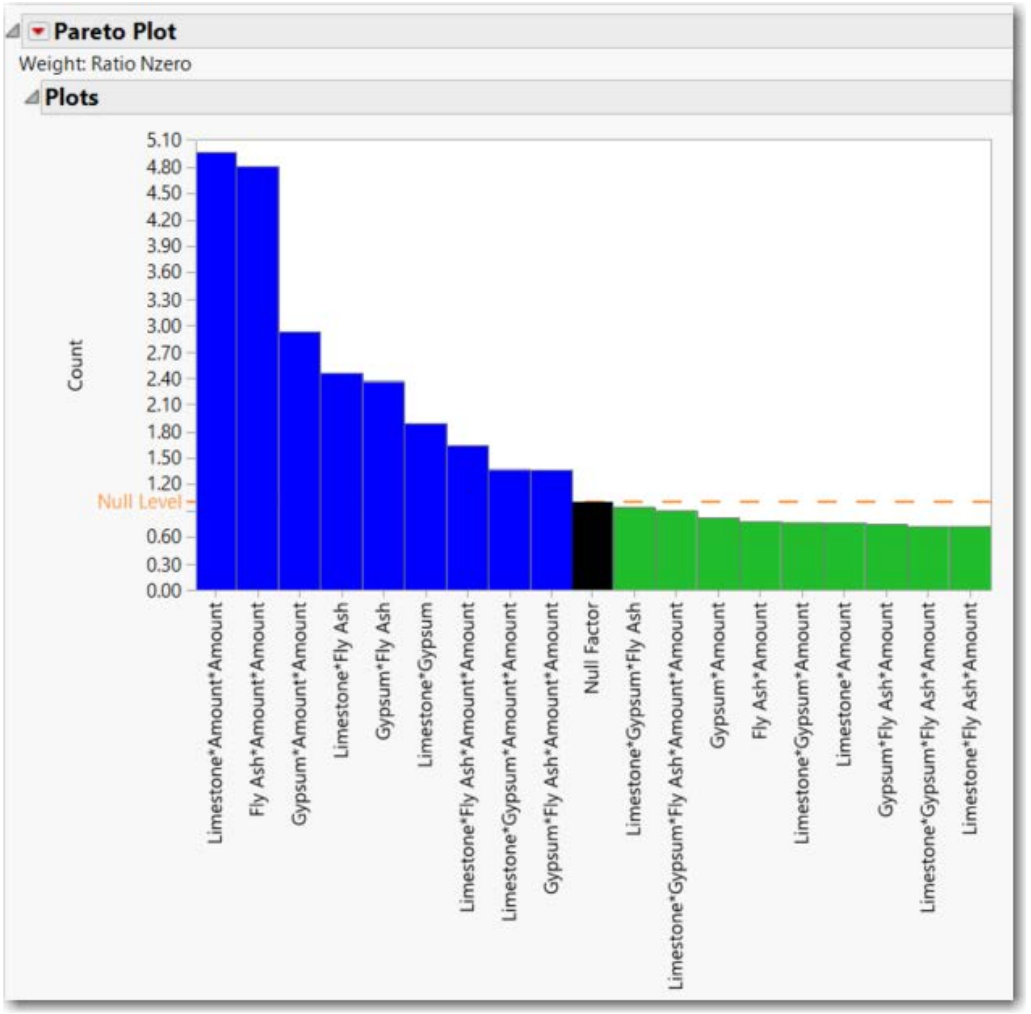
Temperature is a surrogate for tensile strength after 28 days.

Case Study 3: Mixture Amount Experiment – Fly Ash

Historically selection of mixture and mixture process factor models is difficult due to the high degree of correlation in the effect estimates and the need to retain the pure mixture components in the models.

Autovalidation and **Pruned Forward** Selection were used to select models.

To the right is a weighted Pareto plot of the autovalidation results. Again, the three mixture factors are forced into all models.

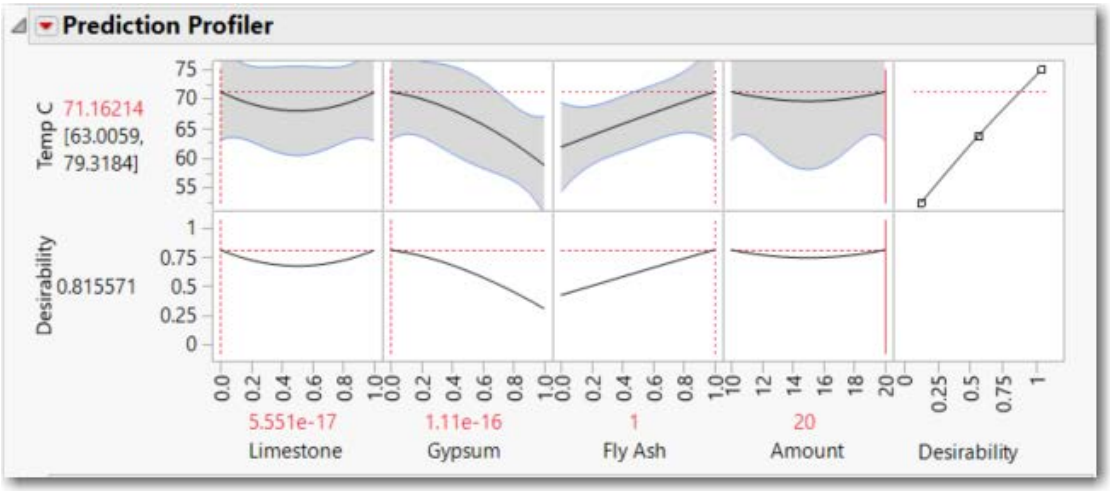


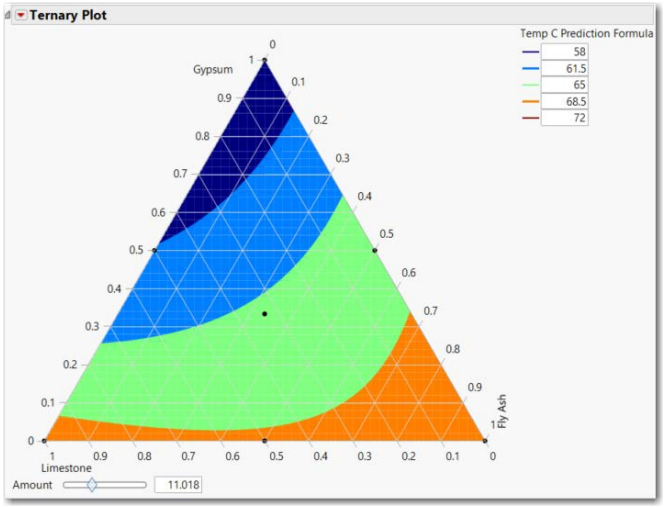
Case Study 3: Mixture Amount Experiment – Fly Ash

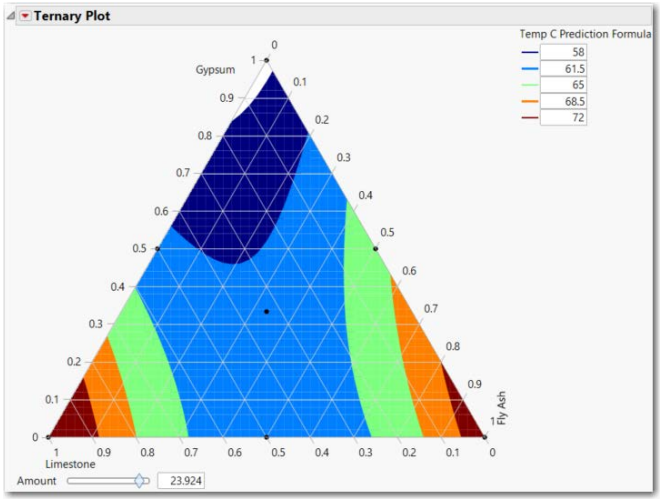
The optimized results indicate max amount of Fly Ash and max amount total. ***This is highly desirable.***

Parameter Estimates for Original Predictors

Term	Estimate	Std Error	Wald ChiSquare	Prob > ChiSquare
Limestone	67.890984	5.880998	133.26685	<.0001*
Gypsum	59.430984	5.880998	102.12305	<.0001*
Fly Ash	69.590984	5.880998	140.02445	<.0001*
Amount*Amount*Limestone	3.1661593	7.2044274	0.1931375	0.6603
Amount*Amount*Fly Ash	1.5711593	7.2044274	0.04756	0.8274
Amount*Amount*Gypsum	-0.648841	7.2044274	0.0081111	0.9282
Limestone*Fly Ash	6.5403279	26.19871	0.0623217	0.8029
Gypsum*Fly Ash	8.9003279	26.19871	0.1154123	0.7341
Limestone*Gypsum	-8.899672	26.19871	0.1153953	0.7341
Amount*Amount*Limestone*Fly Ash	-19.17319	32.239506	0.353681	0.5520
Limestone*Gypsum*Amount*Amount	-3.473185	32.239506	0.0116059	0.9142
Gypsum*Fly Ash*Amount*Amount	-1.063185	32.239506	0.0010875	0.9737







Summary and Conclusions

Traditionally building predictive models from DOE data has been limited by the lack of validation trials to control over fitting.

Typically a DOE budget and time does not make it feasible to perform a separate set of validation trials.

Autovalidation and Fractionally Weighted Bootstrapping are two new viable techniques that enable predictive modeling from DOE data without running a set of validation trials.

The techniques use the original training data and bootstrapping concepts to form validation sets.

The technique was demonstrated on three different case studies and performed very well in all three.

Autovalidation and FWB should be considered a part of DOE analysis where the goal is to build predictive models.

COMMENTS

This is a completely empirical approach that makes very few assumptions.

The Null Factor provides a way to decide which terms to keep in the model in a way that automatically takes into account the model selection approach.

It frees us from using p-values, or the AICc, BIC which all have issues

- p-values are problematic once the model changes from the initial one.
- The AICc can cause underfit in small datasets, overfit in large ones.
- BIC can overfit in small samples and underfit in large ones.

Auto-Validation uses simulation rather than asymptotic approximations.

COMMENTS

Auto-Validation and FWB are very general and can be applied to many families of models, including:

- Partial Least Squares (PLS)
- Neural Networks
- Generalized Linear Models
- Penalized Regression – Better used on observational data than DOEs.

The Auto-Validation may also be useful for very wide data where the number of potential effects p is much larger than the number of observations n.

Even k-fold Cross-validation can be problematic with very wide data.

Wide data is common in many disciplines such as genomics.

BIBLIOGRAPHY

Rubin, D. (1981), “The Bayesian Bootstrap”, *Annals of Statistics*.

Wu, Y., Boos, D., and Stefanski, L. (2007), “Controlling Variable Selection by the Addition of Pseudovariables”, *Journal of the American Statistical Association.*

Jones B., and Nachtsheim, C. (2017), “Effective Design-Based Model Selection for Definitive Screening Designs”, *Technometrics*.