Contractor Disclosure Form 712A – Deadline: 2 June 2014

MORS Symposium

16-19 June 2014, Hilton Mark Center, Alexandria, VA

Fax completed form to 703-933-9066 or email to liz@mors.org

Abstract 606

PART I **Author Request** - The following author(s) request authority to disclose the following presentation at the MORS Symposium with subsequent publication in the MORS Final Report, and posting on the MORS website, if applicable.

Principal Author: Thomas A Donnelly	Other Author(s): N/A
Principal Author's **Organization** and complete mailing address: SAS Institute Inc. 27 Farmingdale Ln Newark, DE 19711	Principal Author's Signature: X Thomas A. Donnelly Date: 28 May 2014 Phone: 302-489-9291 FAX: 919-677-4444 Email: tom.donnelly@jmp.com

Title of Presentation:

Surrogate Modeling of Stochastic Computer Simulation Data - Identifying Insurgents from a Helicopter Flying Surveillance

This presentation is: ☐ SECRET ☐ SECRET//REL TO FVEY ☐ CONFIDENTIAL ☐ CONFIDENTIAL//REL TO FVEY ■ UNCLASSIFIED ☐ Other ________ and will be presented in:

☐ Tutorial ■ List all WG(s) #: WG-28

This work was performed in connection with a government contract.	☐ YES (Complete Parts I, II, & III)
This presentation is based on material developed by the author as part of company-approved research e.g. IR&D and was NOT done under a government contract.	☐ YES (Complete Parts I, II & III)
This presentation was NOT done under a government contract, contains no government information, is my own work and is approved for public release.	■ YES (Complete Part I only)

Statistical Discovery.™ From SAS.

THE POWER TO KNOW.

Surrogate Modeling of Stochastic Computer Simulation Data – Identifying Insurgents from a Helicopter Flying Surveillance

82nd MORS Symposium
Alexandria, VA
June 19th, 2014

Tom Donnelly, PhD
Systems Engineer & Co-insurrectionist

Statistical Discovery.™ From SAS.

THE POWER TO KNOW.

Outline

- Background and Goals
- Visualize Results
- Modeling Approaches
- Comparing Models
- Summary

Abstract

Data for identifying insurgents from a stochastic computer simulation of a helicopter flying surveillance for a convoy are modeled using several different methods. The six factors affecting Proportion Insurgents Identified (the response) are Helicopter Height, Helicopter Speed (relative to convoy), Helicopter Distance (from convoy), Convoy Speed, Number of Insurgents with AK47s, and Insurgent Camouflage level. Models employed include several types of decision tree, neural net, and regression (Generalized Linear Model). Relative strengths, weaknesses and prediction accuracy of models are compared. Discussion of the insights the different types of models offer is also presented.

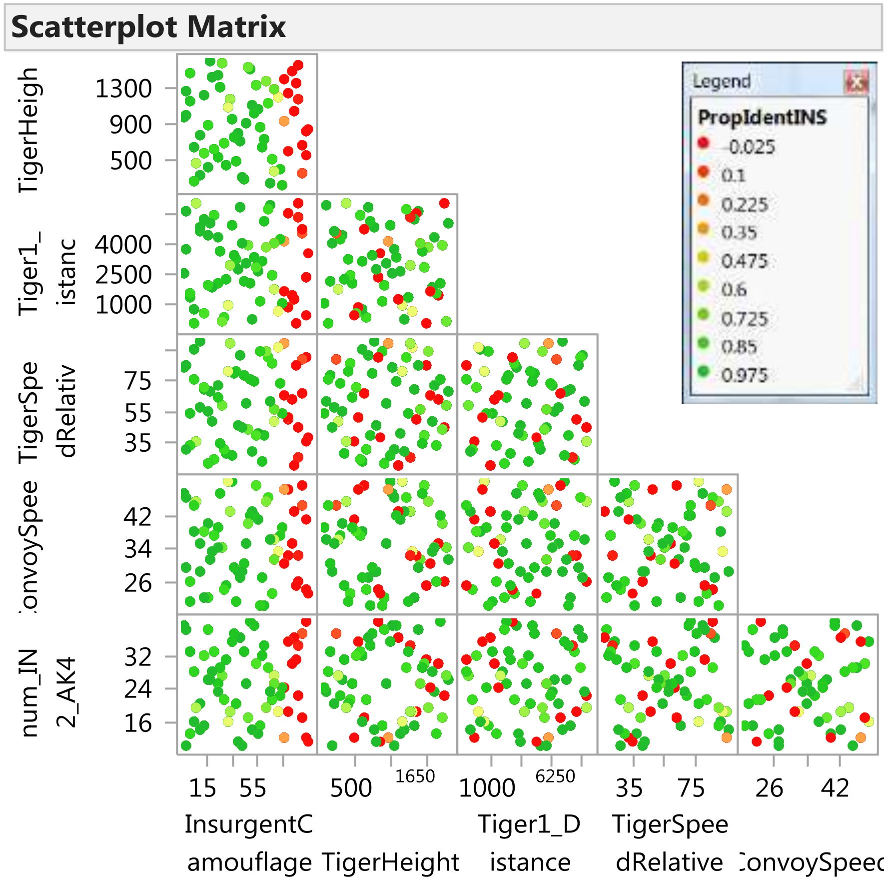
Surrogate Modeling of a Computer Simulation - Helicopter Surveillance – Identifying Insurgents

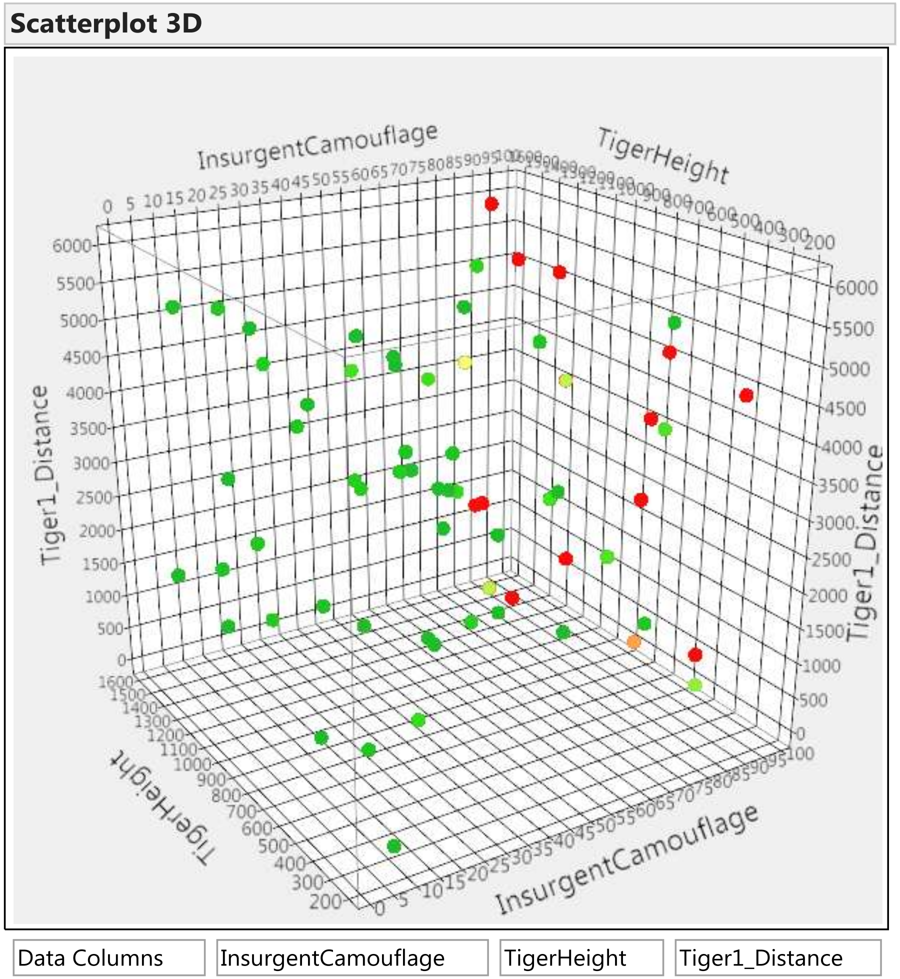
- 2009 International Data Farming Workshop - IDFW21, Lisbon, Portugal
- Largely German team (6 of 8) – their simulation
- 6500 simulations run overnight on cluster in Frankfurt
  - 65 unique combinations of 6 factors (each factor at 65 levels)
  - each case had 97 to 100 replications (lost a few)
- Response = Proportion of Insurgents Identified = *PropIdentINS* Data bounded between 0 and 1
- Explore data visually first
- Fit many different models – “Train, Validate (Tune), Test” 60/20/20 subsets
- Compare Actual vs. Predicted for Test Set

Goals

- Build a variety of surrogate models
- Evaluate and compare to choose best predictor
- Gain insight into simulation model
- Learn about different approaches to data mining

Preview End Result – Space-Filling DOE





Low detection associated with high levels of camouflage.

Why Use Design of Experiments Methods with Simulation Experiments?

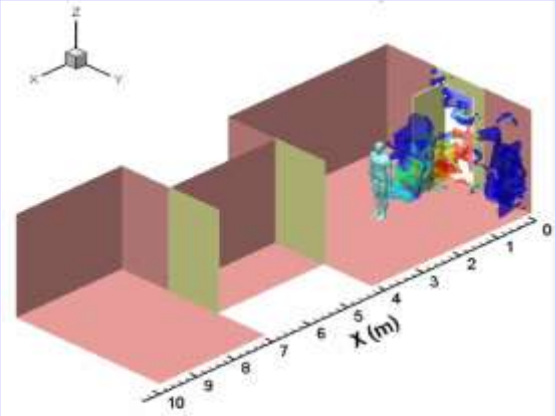
Quicker answers, lower costs, solve bigger problems

- Obtain a fast surrogate model of the simulation
  - Individual simulations can run for hours, days, weeks
    - » Computational Fluid Dynamics (CFD)
    - » Simulation runs in real-time
  - Numbers of factors can be very large (40+)
  - Numbers of simulations needed can be large (thousands in many cases)
  - Simulations can be stochastic requiring many replications
- Surrogate model yields a fast approximation of the simulation
  - more rapidly answer "what if?" questions
  - do sensitivity analysis of the control factors
  - optimize multiple responses and make trade-offs
- By running efficient subsets of all possible combinations, one can – for the same resources and constraints – *solve bigger problems*
- By running sequences of designs one can be as *cost effective as possible* & run *no more trials than are needed* to get a useful answer

Long Running Physics-Based Simulation

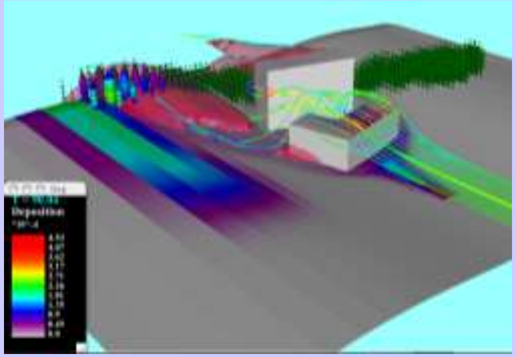
Detailed Physics Models can require a great deal of runtime to generate a short period of simulation time.

Computational Fluid Dynamics (CFD) Models



Developed for Interior
Moving Man in Simulation
8M cells
10 Seconds of Simulation
64 CPUs – 4K slower
12 Hours of Runtime

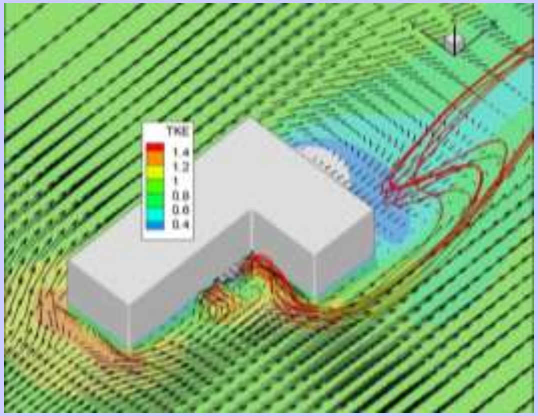
Detailed Ingress/Egress, Internal Airflow and Convection



Developed for Exterior
Stationary Grids
1.5M Cells
30 Seconds of Simulation
Single CPU – 20K slower
7 Days of Runtime

External CW Deposition/ Evaporation, Vegetation, Solar Heating

Lagrangian-Particle

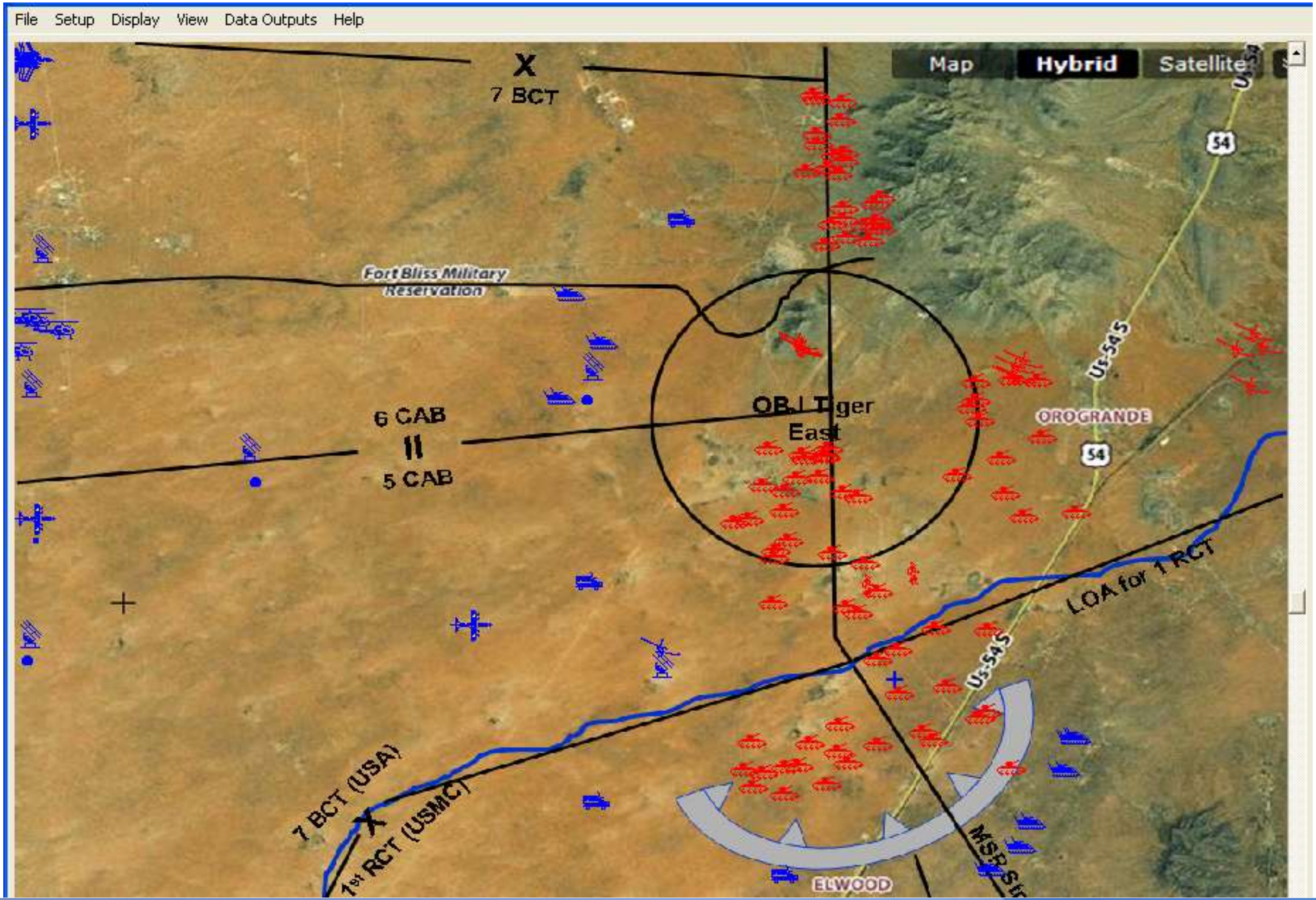


Developed for Exterior
Stationary Grids
TBD Cells
Min-Hours of Simulation
Single CPU
Minutes-Days of Runtime

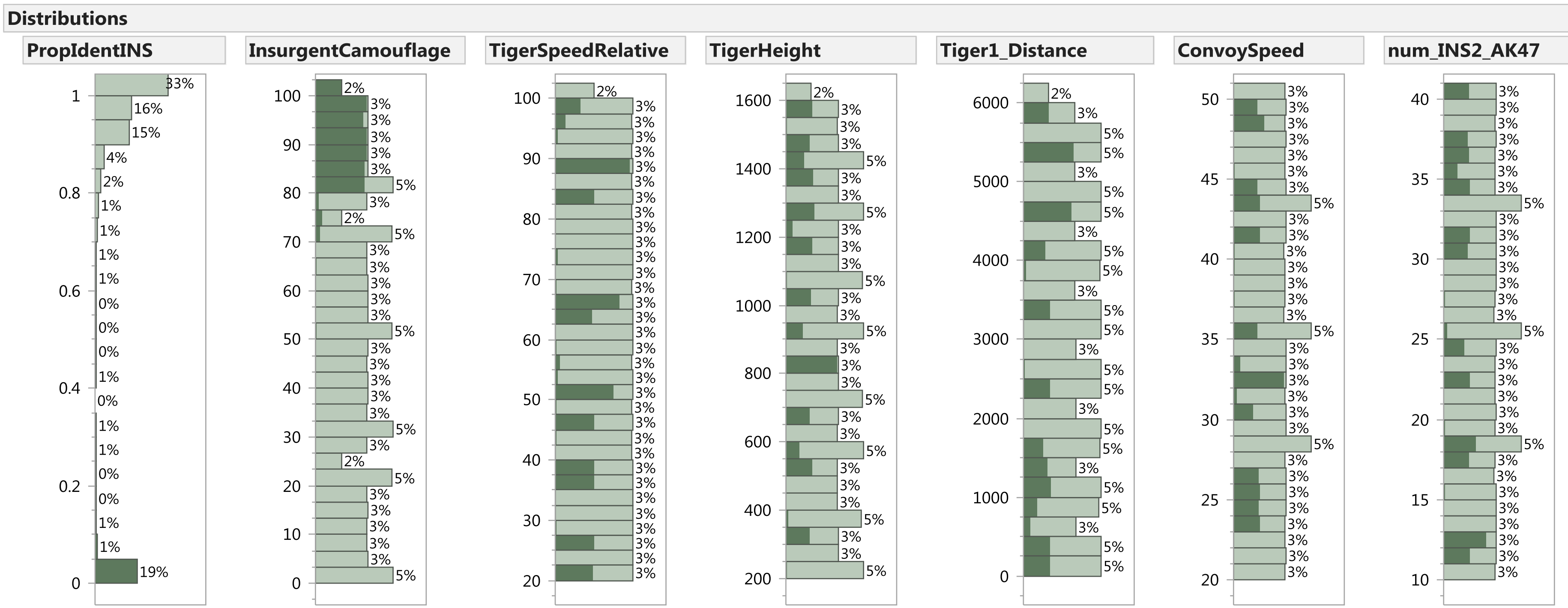
Speed, Flexibility, More User Friendly, V&V

Stochastic Simulations with Many Replicates

Agent-Based Simulations

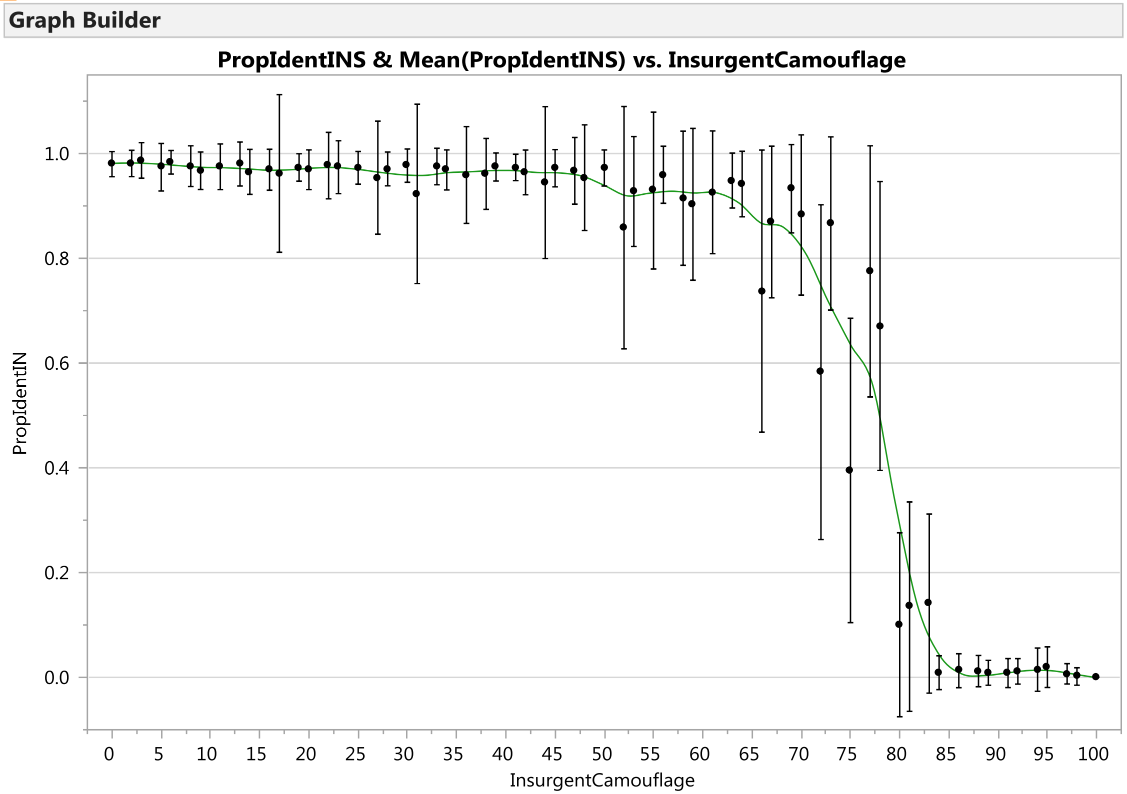


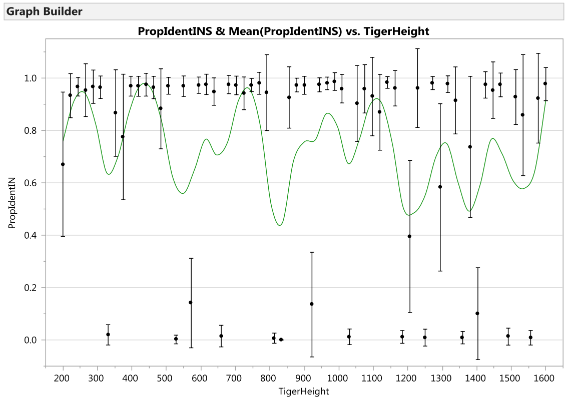
Distributions of Response and 6 Factors

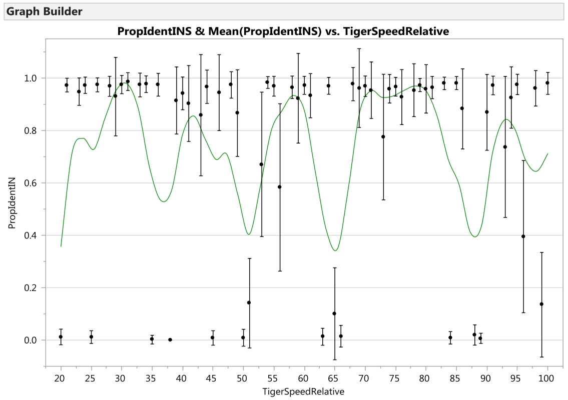


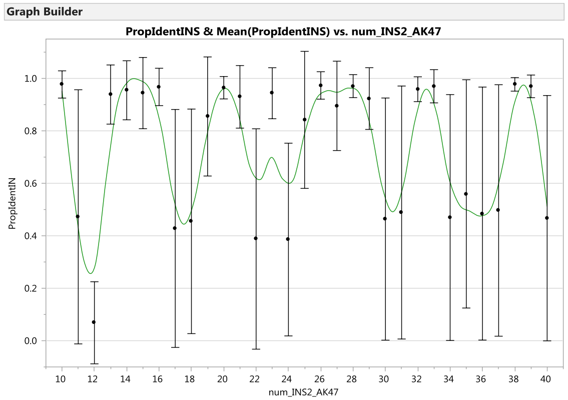
Before modeling look for correlations between good or poor levels of *PropIdentINS* and the factors.
Strong correlation between poor *PropIdentINS* and high levels of *InsurgentCamouflage*.
No other factor shows very much correlation with the response.

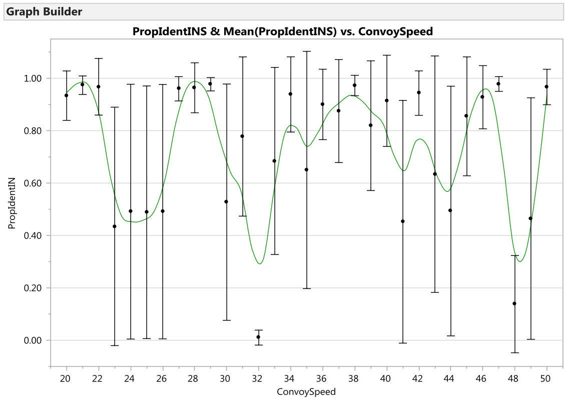
PropIdentINS vs. X for 6 Factors

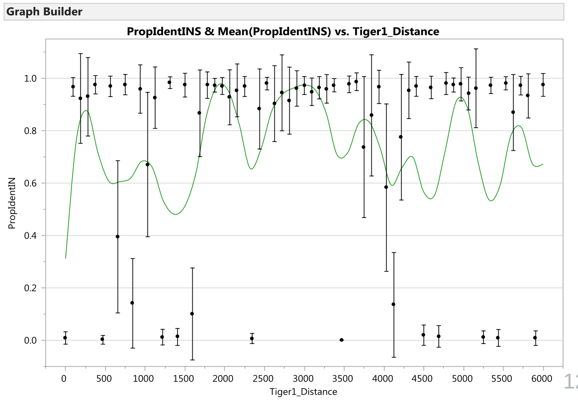




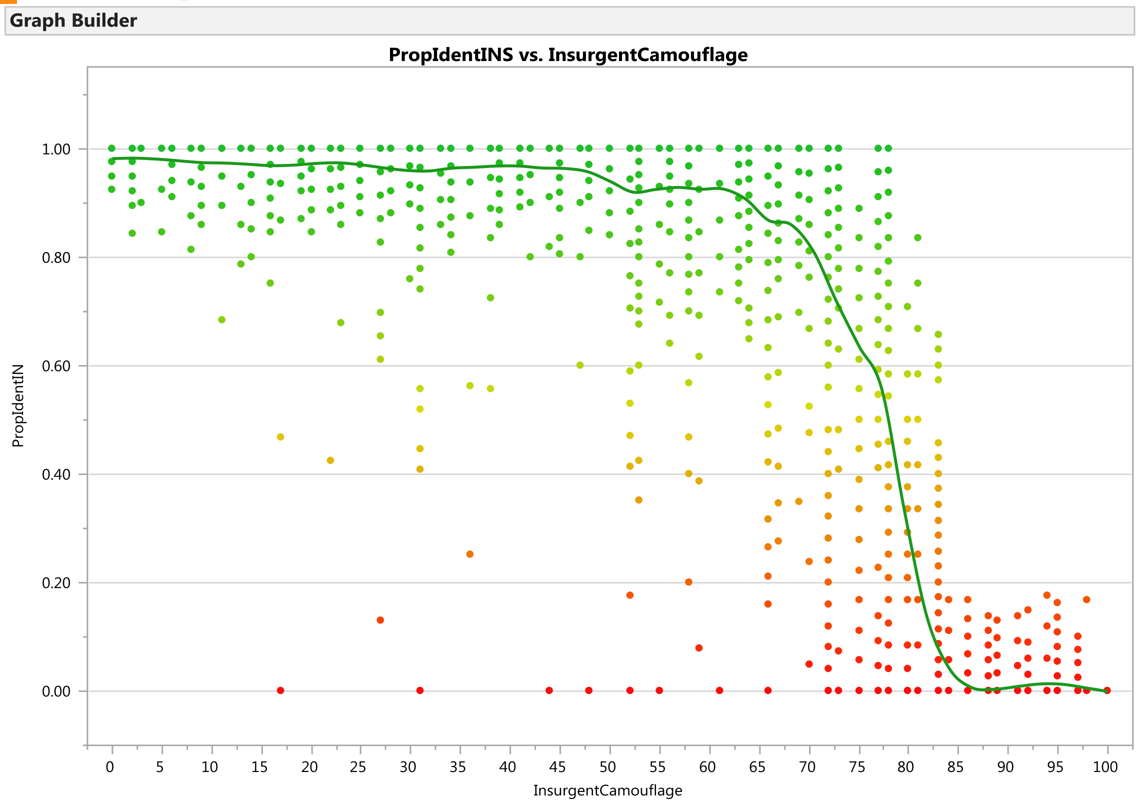


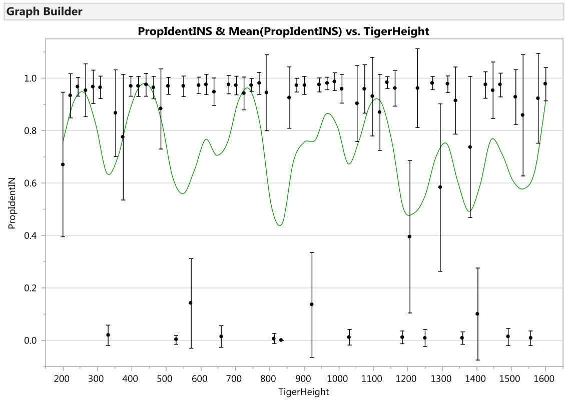


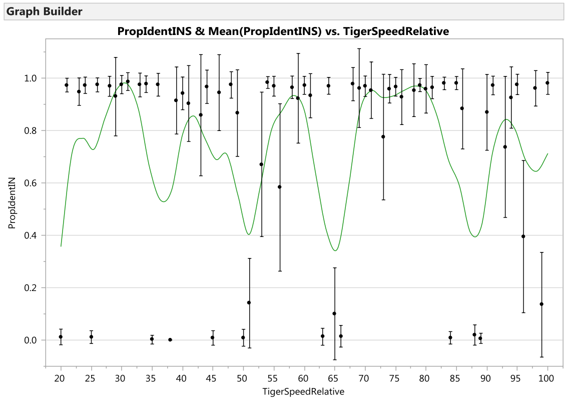


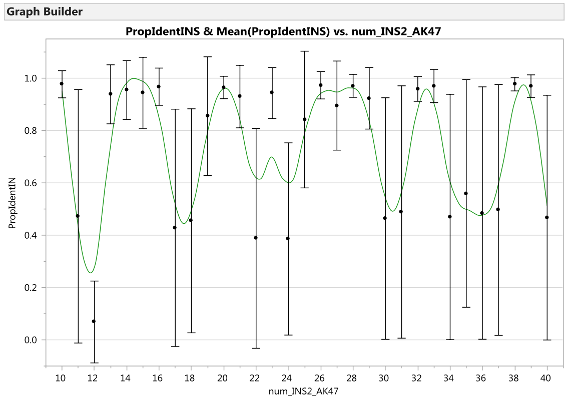


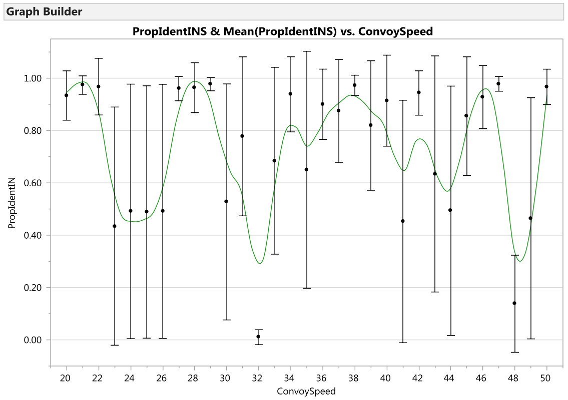
PropIdentINS vs. X for 6 Factors

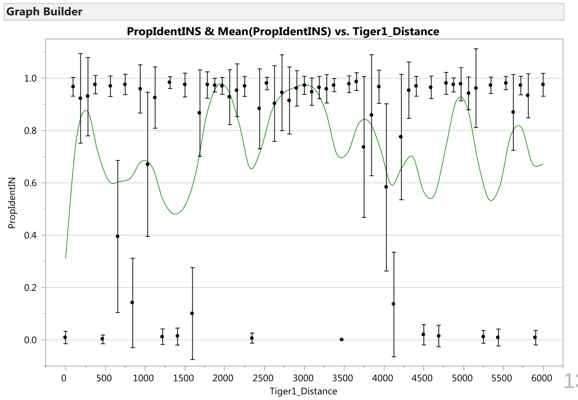




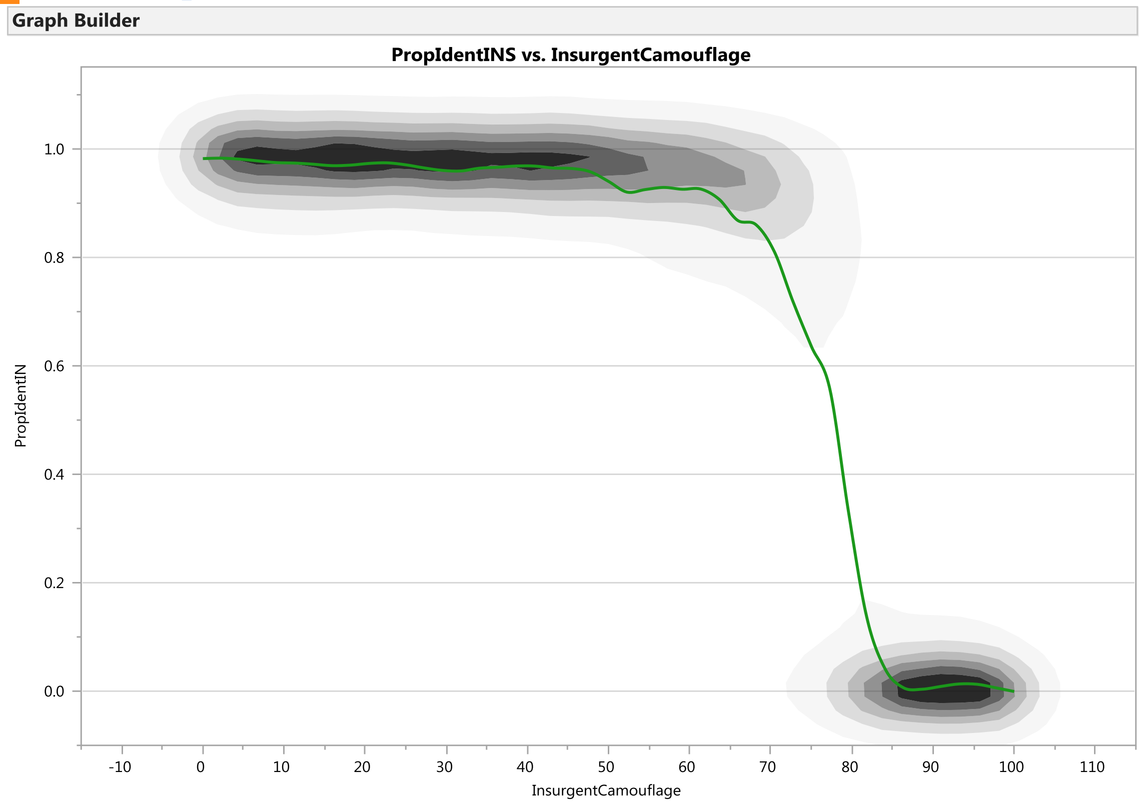


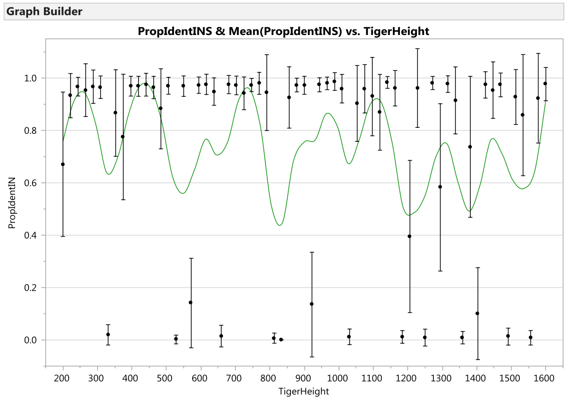


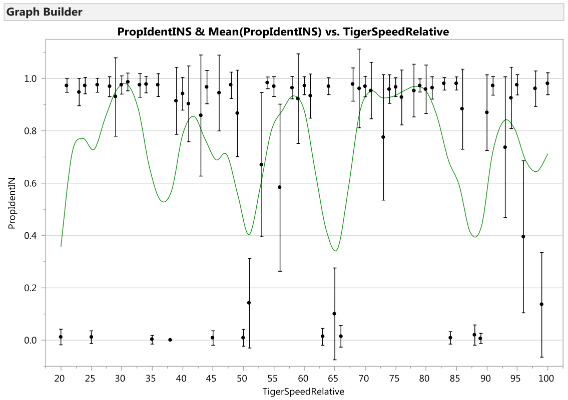


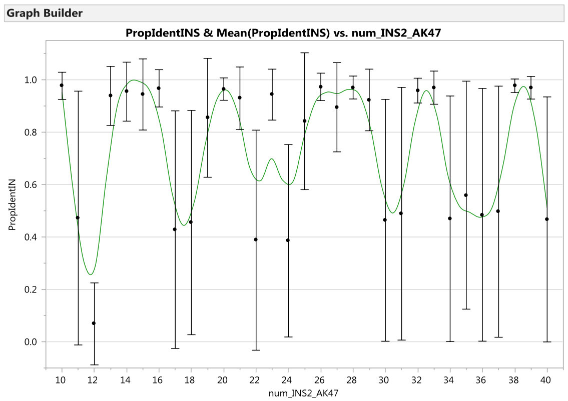


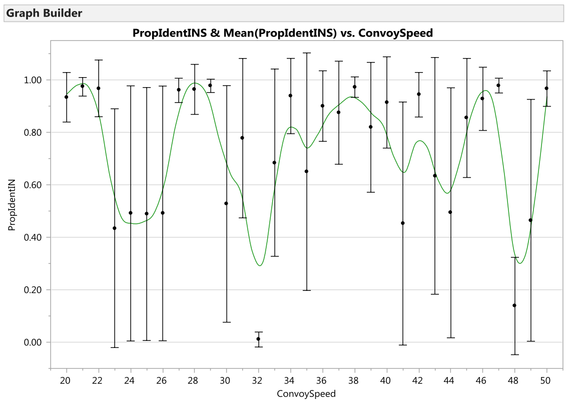
PropIdentINS vs. X for 6 Factors

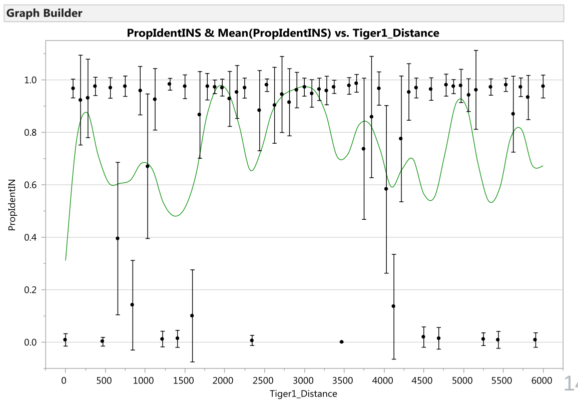




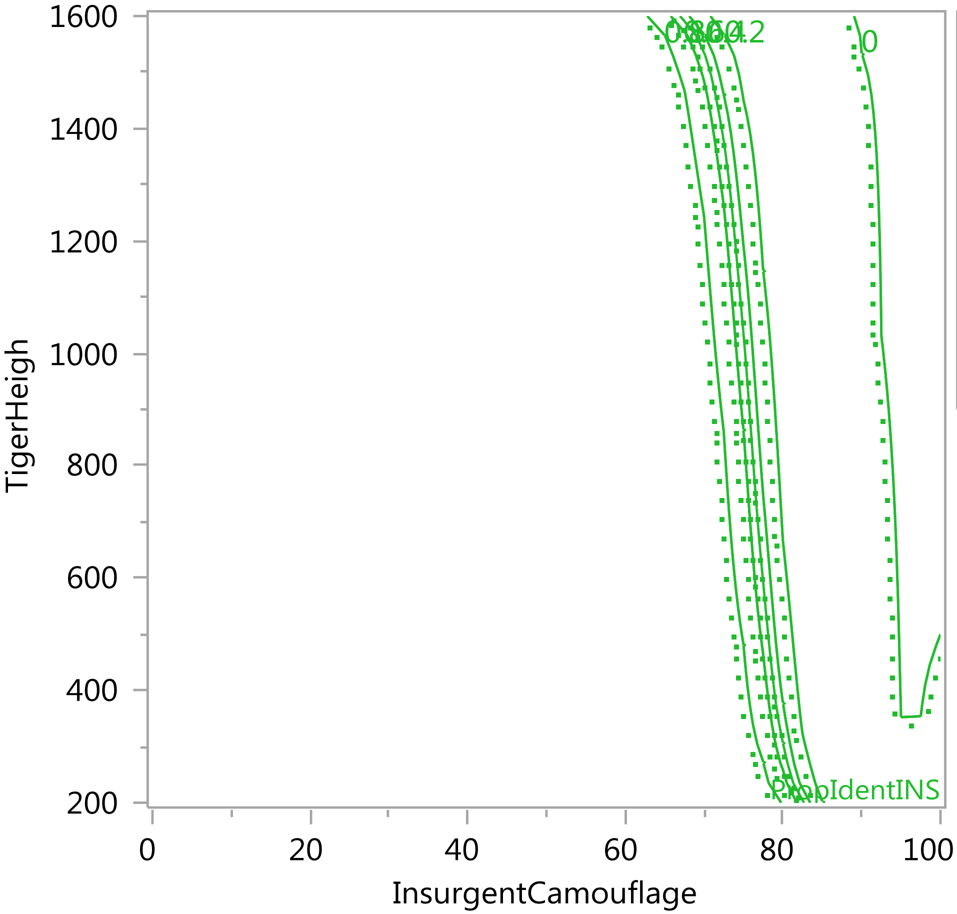


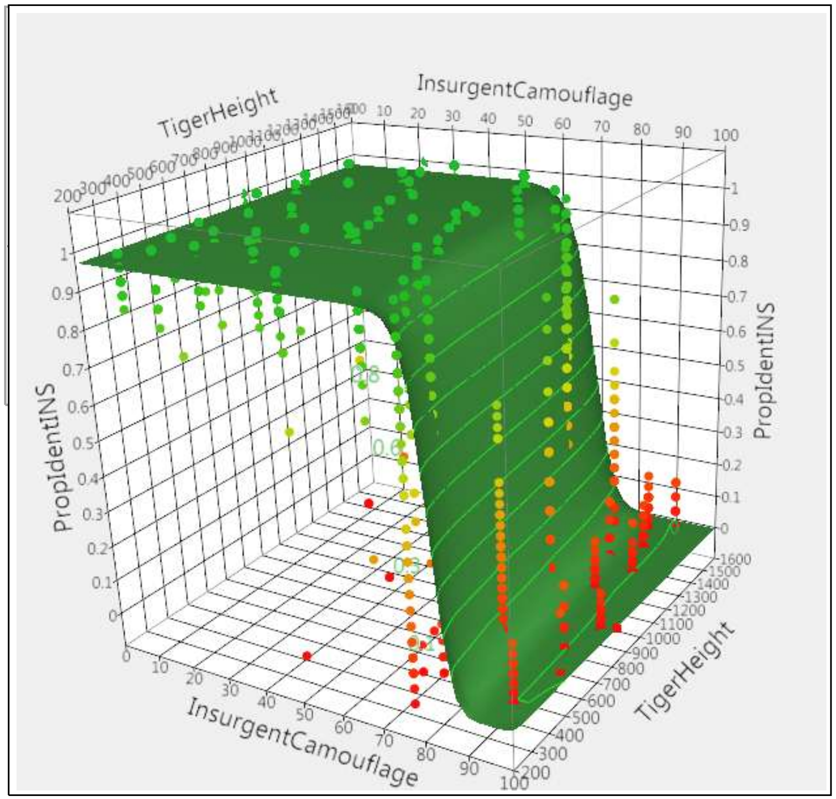




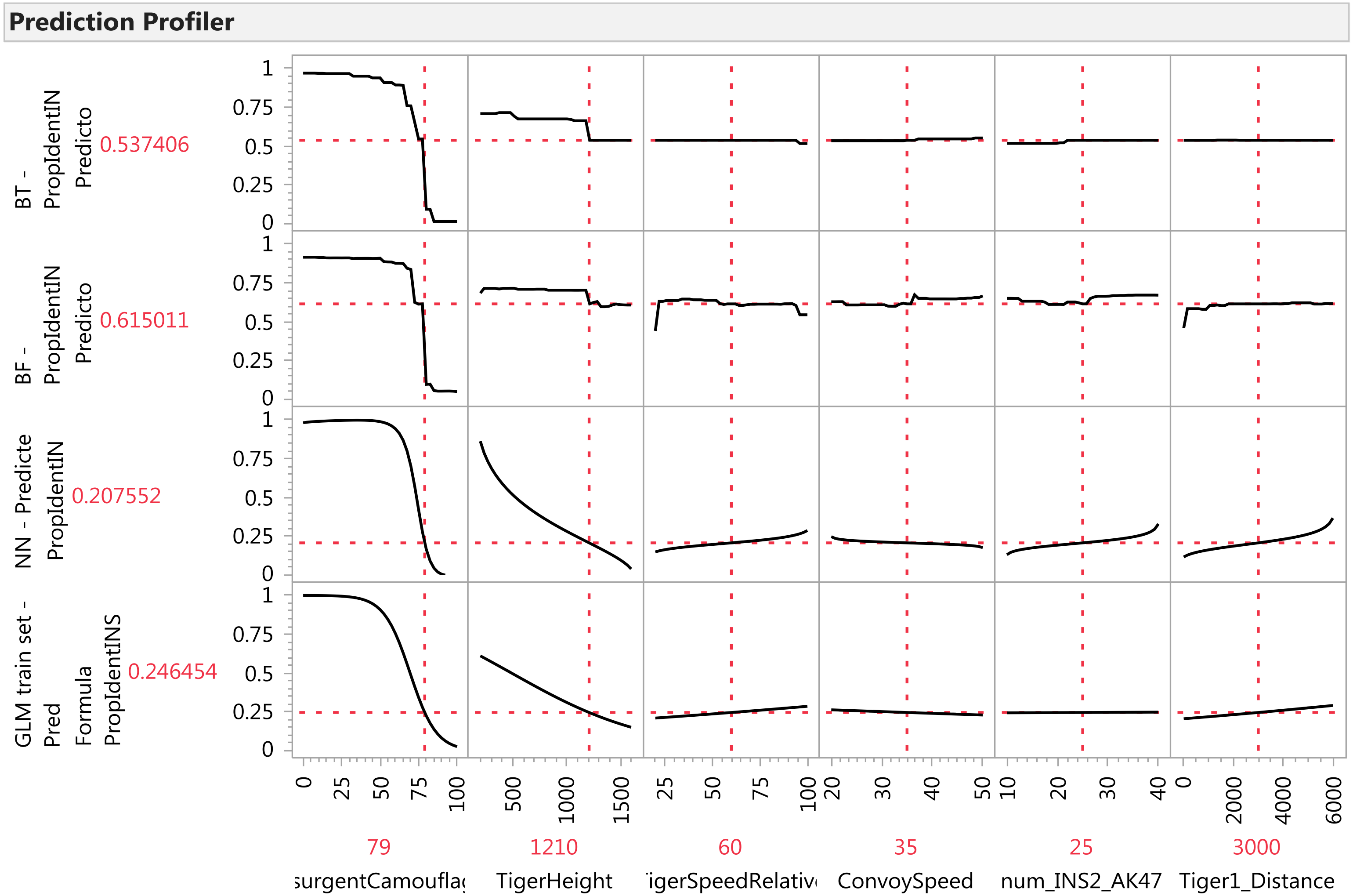


2-D Contour Plot and 3-D Response Surface PropIdentINS vs. Camouflage & Height

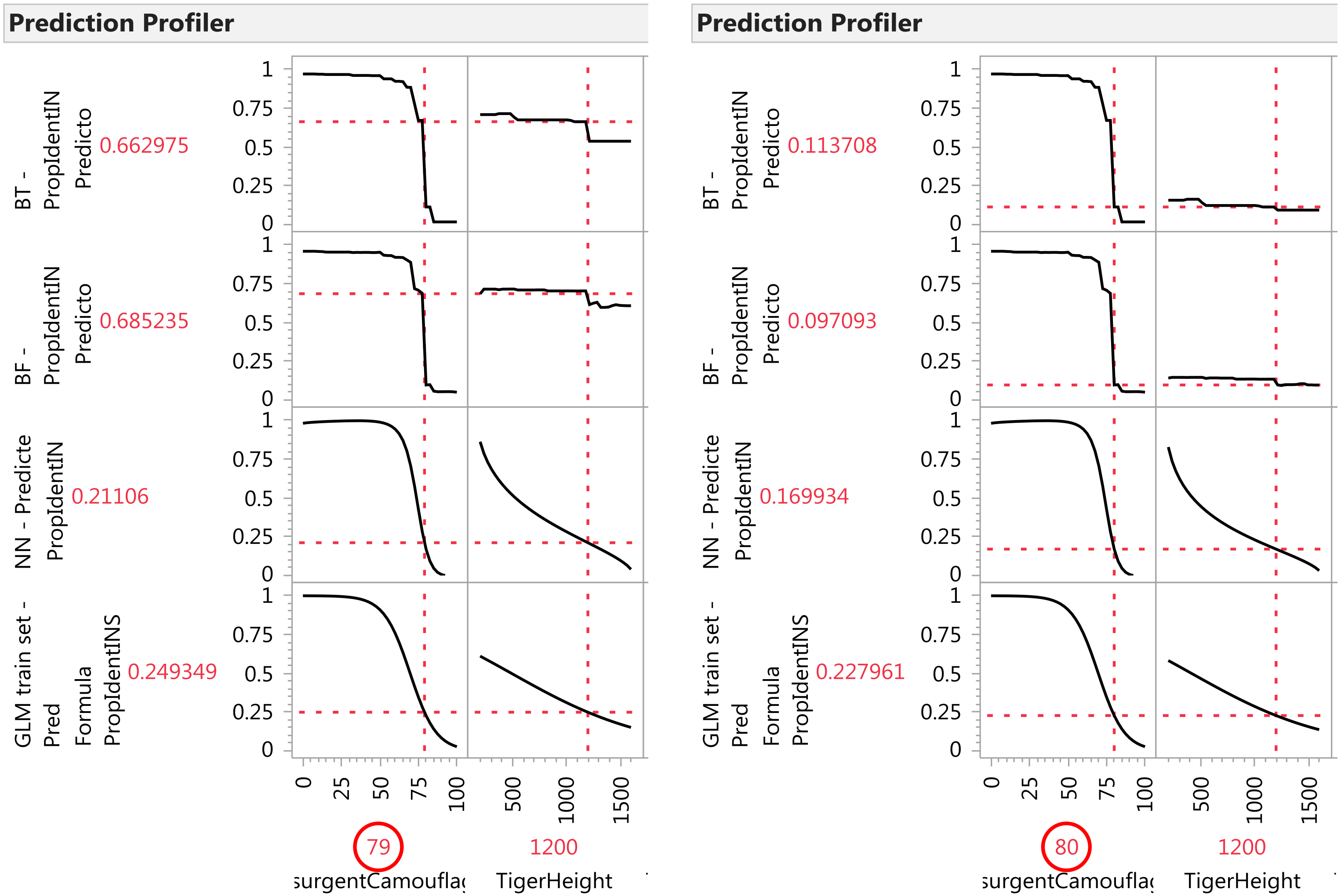




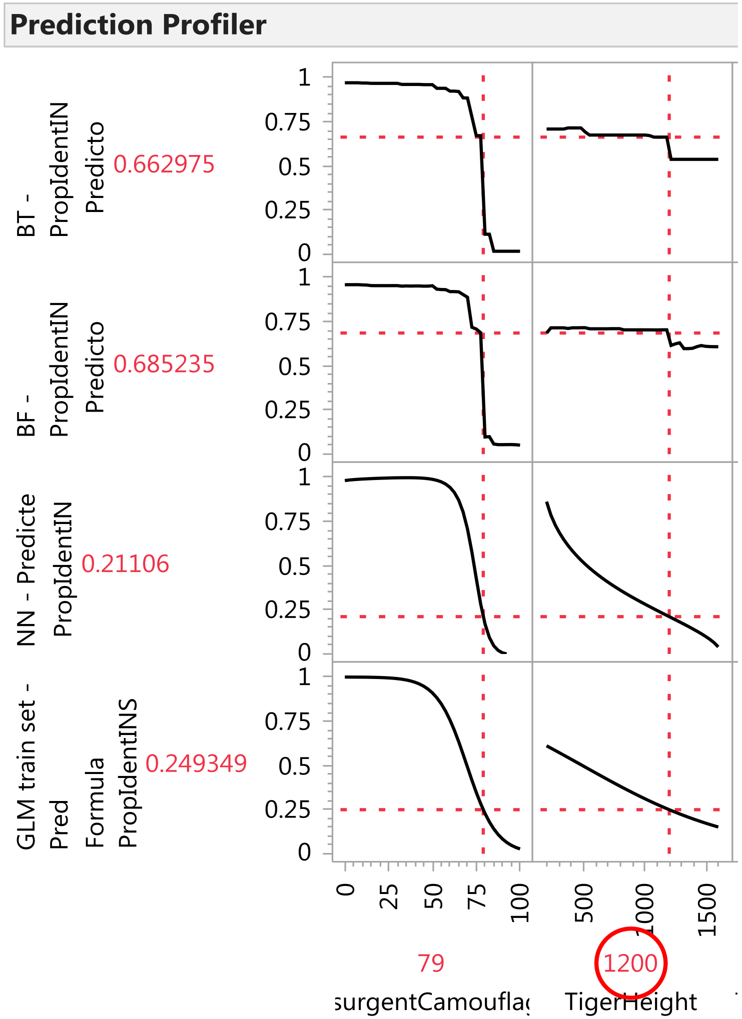
Compare Several Models – top 2 are decision tree variants bottom two are “smoother” models - Neural Net and GLM

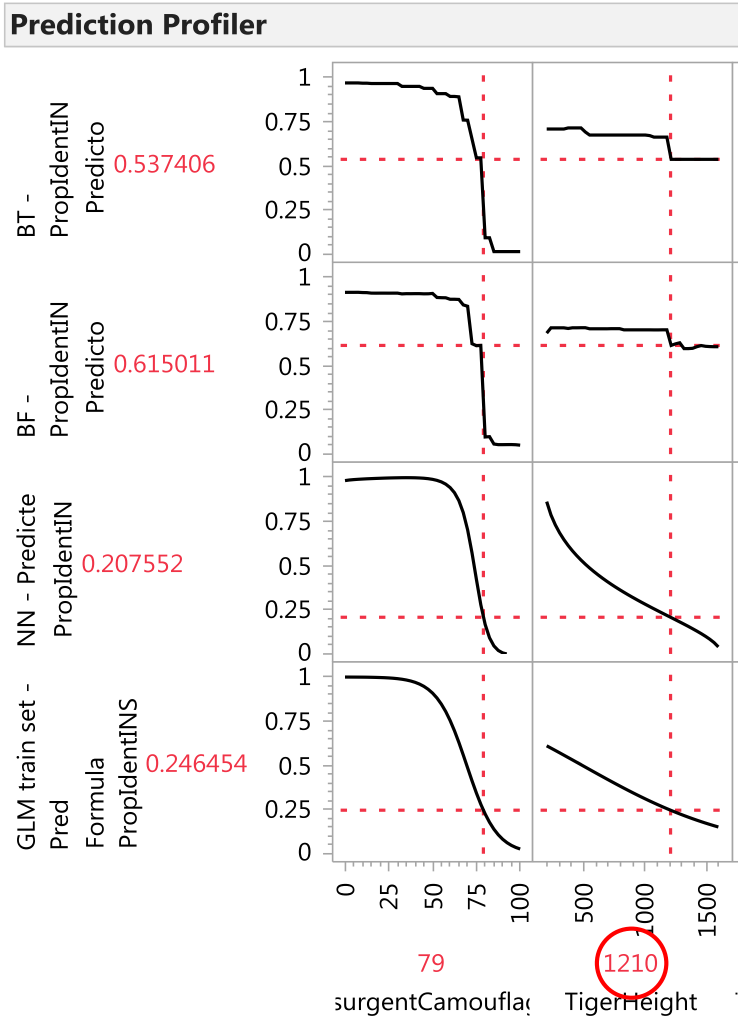


Change Camouflage from 79 to 80 and Decision Tree Predictions Drop by 6X – Talk to Developer?



Change Tiger Height from 1200 to 1210 and Decision Tree Predictions Drop by 10% to 20%! – Plausible?





Model Quotes

- "No *good* model ever accounted for all the facts, since some data was bound to be misleading, if not wrong."
  - – James Dewey Watson (1988)
- "Essentially, *all* models are wrong, but some are useful."
  - – George Box (1987)
- "The purpose of models is *not* to fit the data but to sharpen the questions."
  - – Samuel Karlin (1983)
- "The *best* material model of a cat is another, or preferably the *same*, cat."
  - – A. Rosenbleuth (1945)

What is a statistical model?

- An empirical model that relates a set of inputs (predictors, $\mathbf{X}$) to one or more outcomes (responses, $\mathbf{Y}$)
- Separates the response variation into signal and noise

$$\mathbf{Y} = f(\mathbf{X}) + \mathbf{E}$$

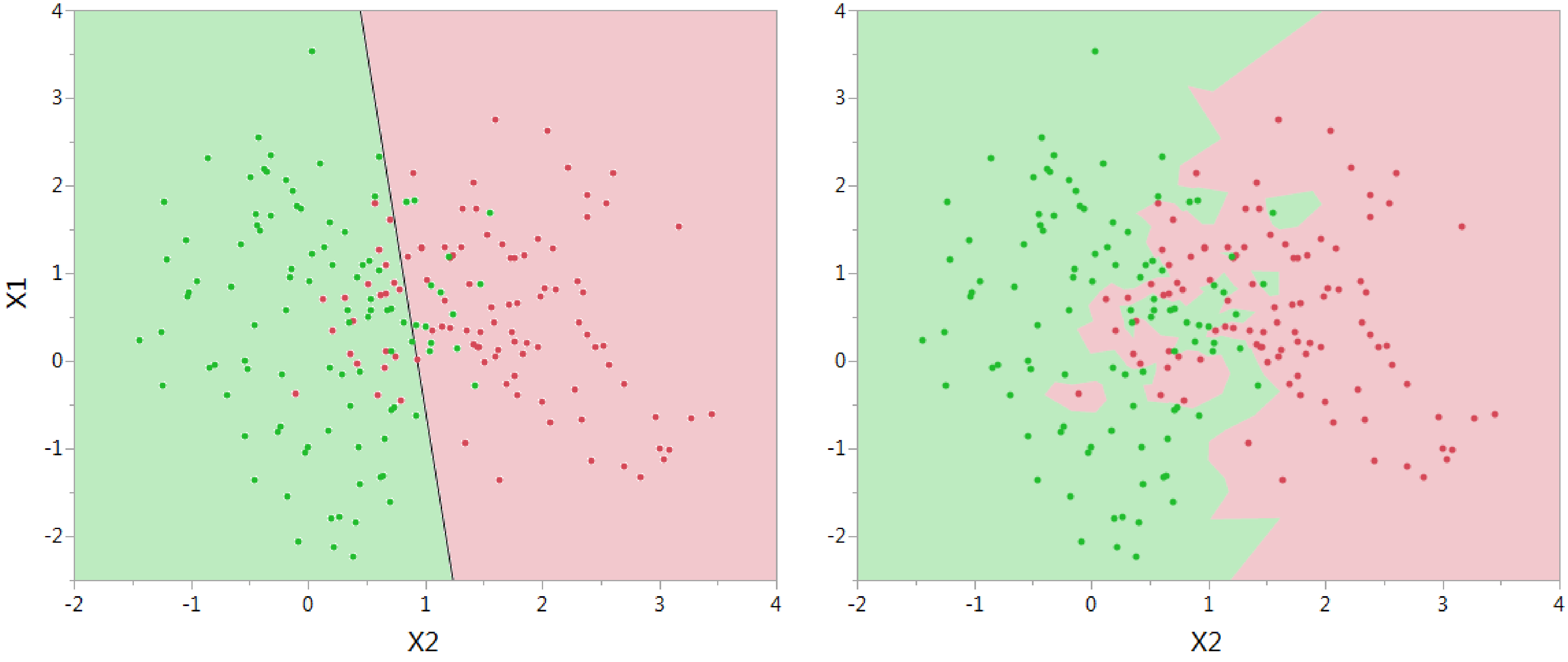
  - $\mathbf{Y}$ is one or more continuous or categorical response outcomes
  - $\mathbf{X}$ is one or more continuous or categorical predictors
  - $f(\mathbf{X})$ describes predictable variation in $\mathbf{Y}$ (signal)
  - $\mathbf{E}$ describes non-predictable variation in $\mathbf{Y}$ (noise)
- The mathematical form of $f(\mathbf{X})$ can be based on domain knowledge or mathematical convenience.

What is a predictive model?

- A type of statistical model where the focus is on predicting $\mathbf{Y}$ independent of the form used for $f(\mathbf{X})$.
  - There is less concern about the form of the model – parameter estimation isn't important. The focus is on how well it predicts.
  - Very flexible models are used to allow for a greater range of possibilities.
  - http://en.wikipedia.org/wiki/Predictive_modelling

What is a predictive model?

- Two Examples:



Regression

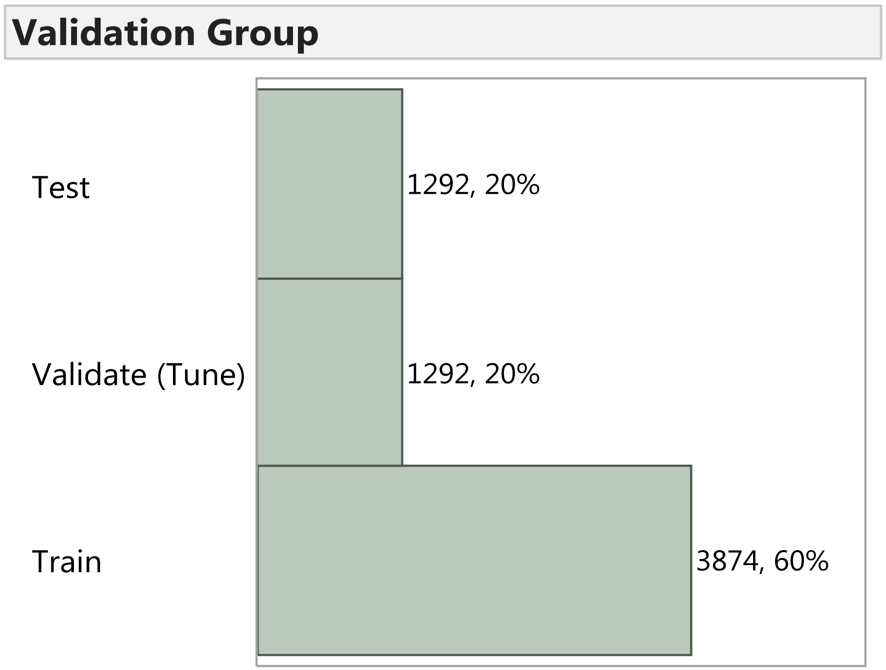
Nearest Neighbor

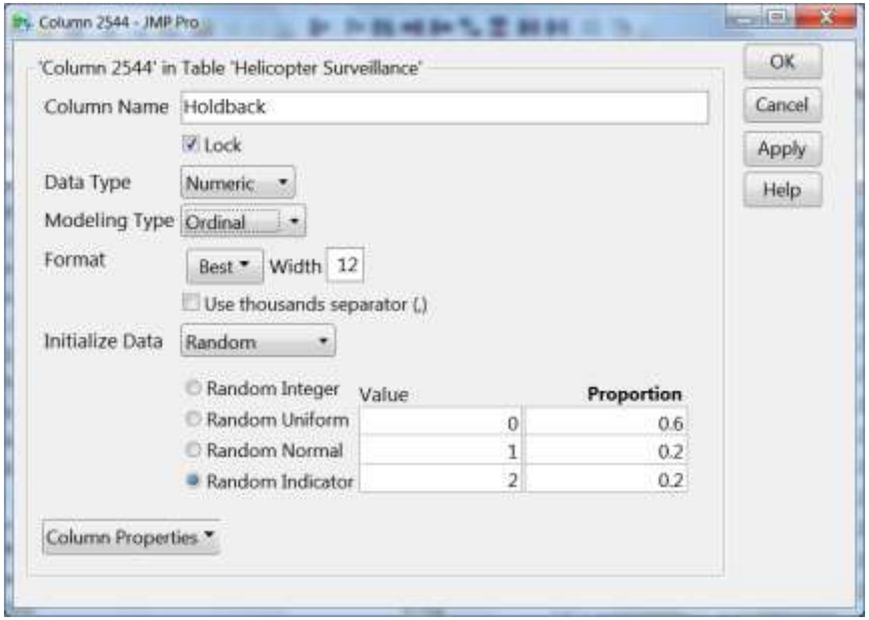
Preventing Model Overfitting

- If the model is flexible what guards against overfitting (i.e., producing predictions that are too optimistic)?
  - Put another way, how do we protect from trying to model the noise variability as part of $f(\mathbf{X})$?
- Solution – Hold back part of the data, using it to check against overfitting. Break the data into two or three sets:
  - The training set is used to ***build*** or ***fit*** the model
  - The validation set is used to ***select*** model by determining when the model is becoming too complex – it ***tunes*** the parameters
  - The test set is often used to ***evaluate*** how well model predicts independent of training and validation sets
  - Common methods include random holdback and k-fold crossvalidation

Honest Assessment Approach
Using Train, Validate (Tune), and Test Subsets

Used in model selection and estimating its prediction error on new data





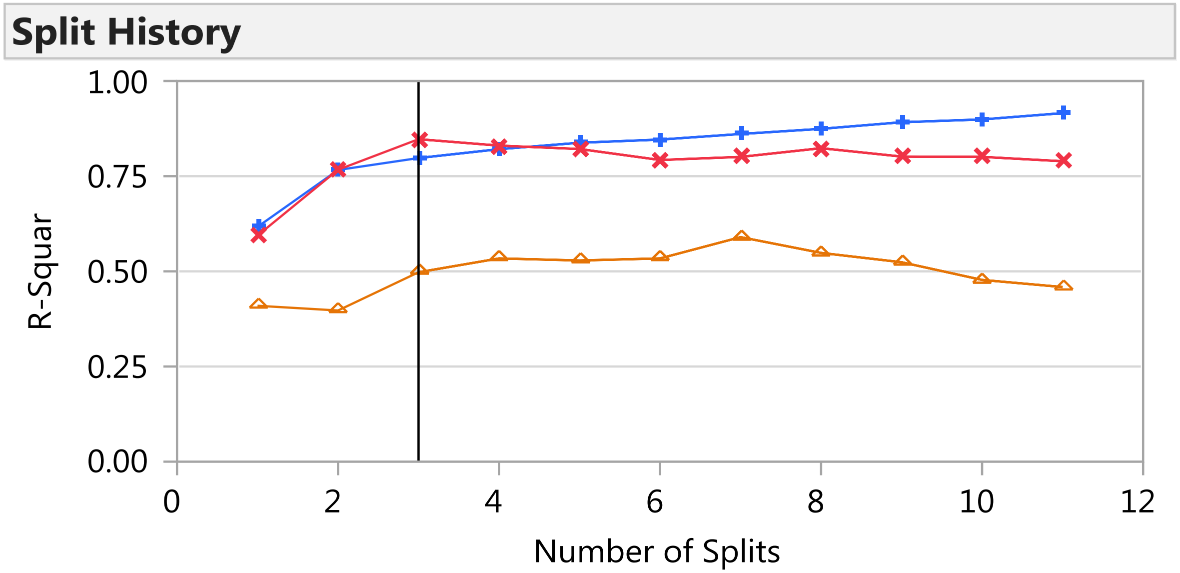
The Elements of Statistical Learning – Data Mining, Inference, and Prediction

Hastie, Tibshirani, and Friedman – 2001

(Chapter 7: Model Assessment and Selection)

Honest Assessment Approach Using Train, Validate (Tune), and Test Subsets

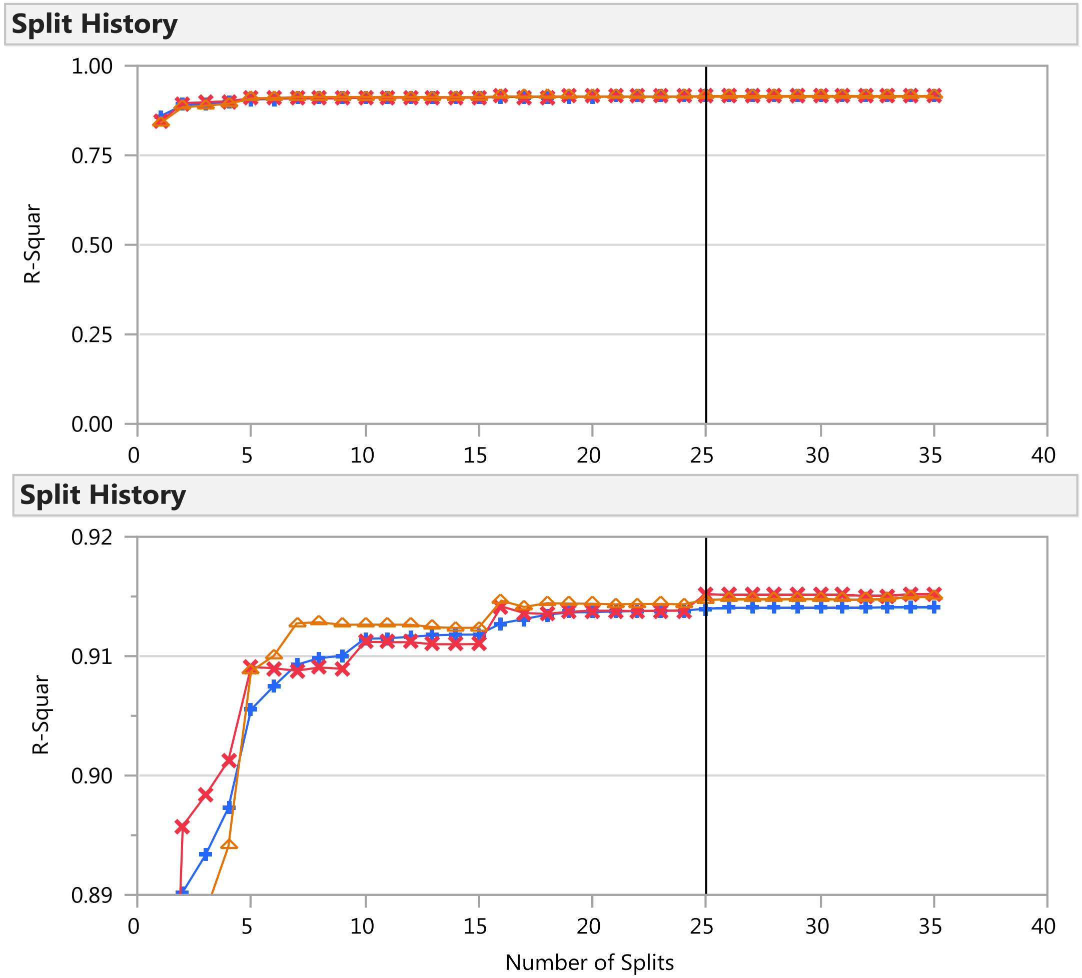
Train, Validate, Test R-Square vs. #Splits Decision Tree Model (569 rows of breast cancer data)



Validation Data in Red
Test Data in Orange

Honest Assessment Approach
Using Train, Validate (Tune), and Test Subsets

Train, Validate, Test R-Square vs. #Splits Decision Tree Model (6458 rows of simulation data for helicopter flying surveillance.)



Validation Data in Red

Test Data in Orange

Decision Trees

- Also known as Recursive Partitioning, CHAID, CART
- Models are a series of nested IF() statements, where each condition in the IF() statement can be viewed as a separate branch in a tree.
- Branches are chosen so that the difference in the average response (or average response rate) between paired branches is maximized.
  - For all factors bin factor values or levels into two buckets such that the means of the two buckets are as far apart as possible.
  - Split on factor with the biggest difference in bucket means.
- Tree models are “grown” by adding more branches to the tree so the more of the variability in the response is explained by the model

Decision Tree Step-by-Step

Goal is to predict “Rejects” & “Accepts””

Overall Accept Rate is 84.44%
Overall Reject Rate is 15.56%

RSquare
0.000

All Rows

Count	G^2
90	77.800668

Level	Rate	Prob
Accep	0.8444	0.8444
Reject	0.1556	0.1556

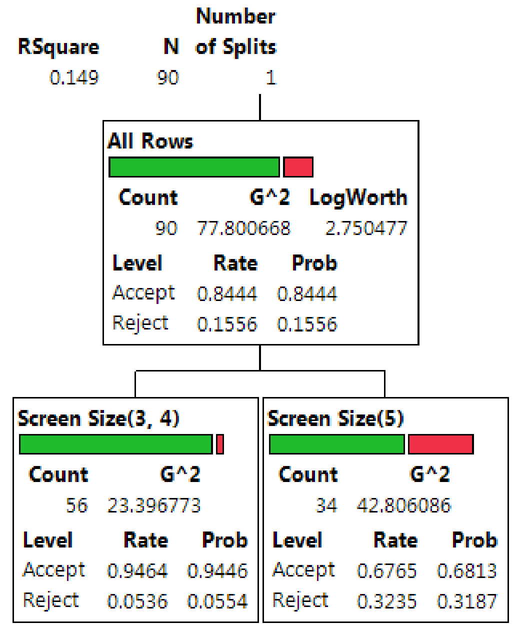
Candidates

Term	Candidate G^2		LogWorth	Cut Point
API Particle Size	4.04050319		0.986886932	Small,Large
Mill Time	10.63219688		1.912625603	11
Screen Size	11.59780917	>	2.750476973	3,4
MgSt Supplier	1.99715970		0.802459554	Jones Inc
Lactose Supplier	1.07597470		0.523458492	James Ind
Sugar Supplier	3.99502860		1.340705011	Sour
Talc Supplier	0.00000000		0.000000000	Rough
Blend Time	2.46622023		0.066048548	15.887
Blend Speed	6.86574102		0.717212865	60.772
Compressor	0.00153207		0.013776004	COMPRESS
Force	7.53188562		0.855446810	24.691
Coating Supplie	0.82675321		0.217072294	Mac
Coating Viscosit	4.66879353		0.322714711	96.413
Inlet Temp	7.28399996		0.803171227	106.39
Exhaust Temp	7.17119361		0.779703315	68.592
Spray Rate	15.01998363	<	2.736639439	403.26
Atom. Pressure	3.36570749		0.149475063	58.787

Candidate “X’s”

- Search through each of these
- Examine Splits for each unique level in each X
- Find Split that maximizes “LogWorth”
  - Will find split that maximizes difference in proportions of the target variable

Decision Tree Step-by-Step



1st Split:

Optimal Split Screen Size 3 & 4 vs. Screen Size 5

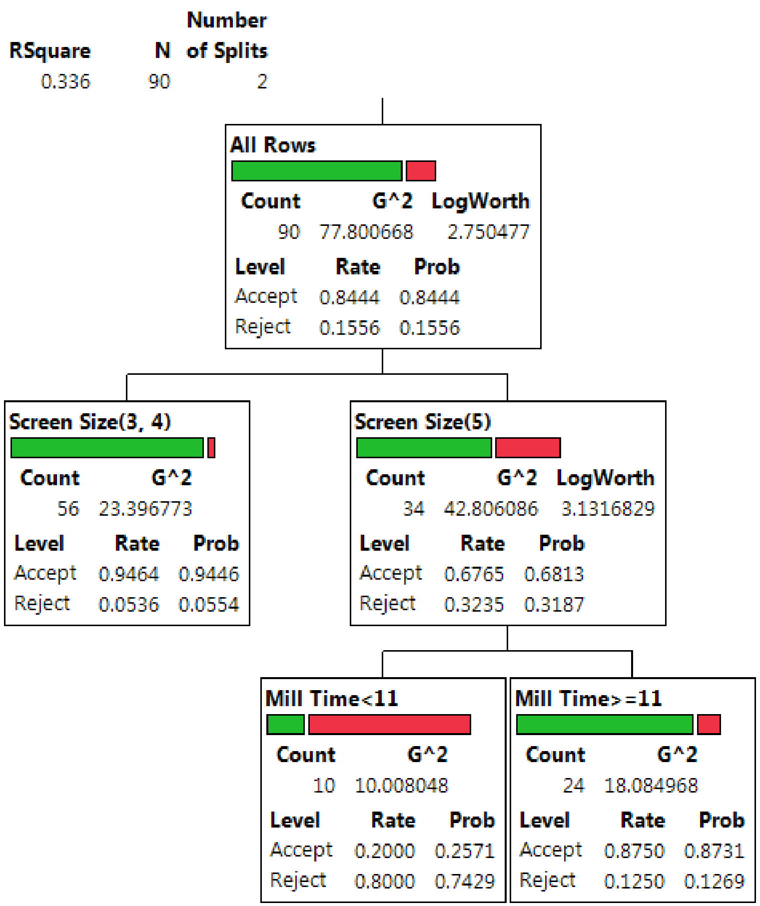
Notice the difference in the rates in each branch of the tree

Repeat “Split Search” across both “Partitions” of the data. Find optimal split across both branches.

Decision Tree (Step by Step)

2nd split on Mill Time
(< 11 vs. >= 11)

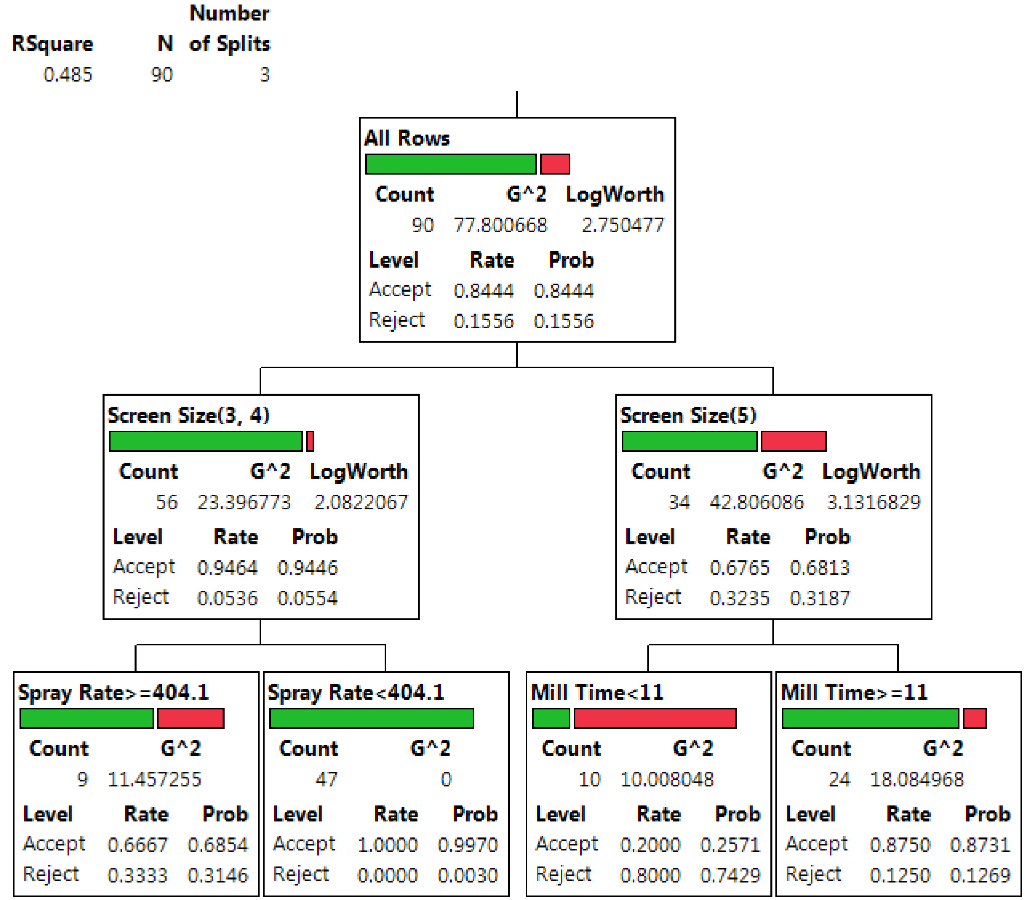
Notice variation in proportion of “1” in each branch



Decision Tree (Step by Step)

3rd split on Spray Rate (>= 404.1 vs. < 404.1))

Notice variation in proportion of “1” in each branch

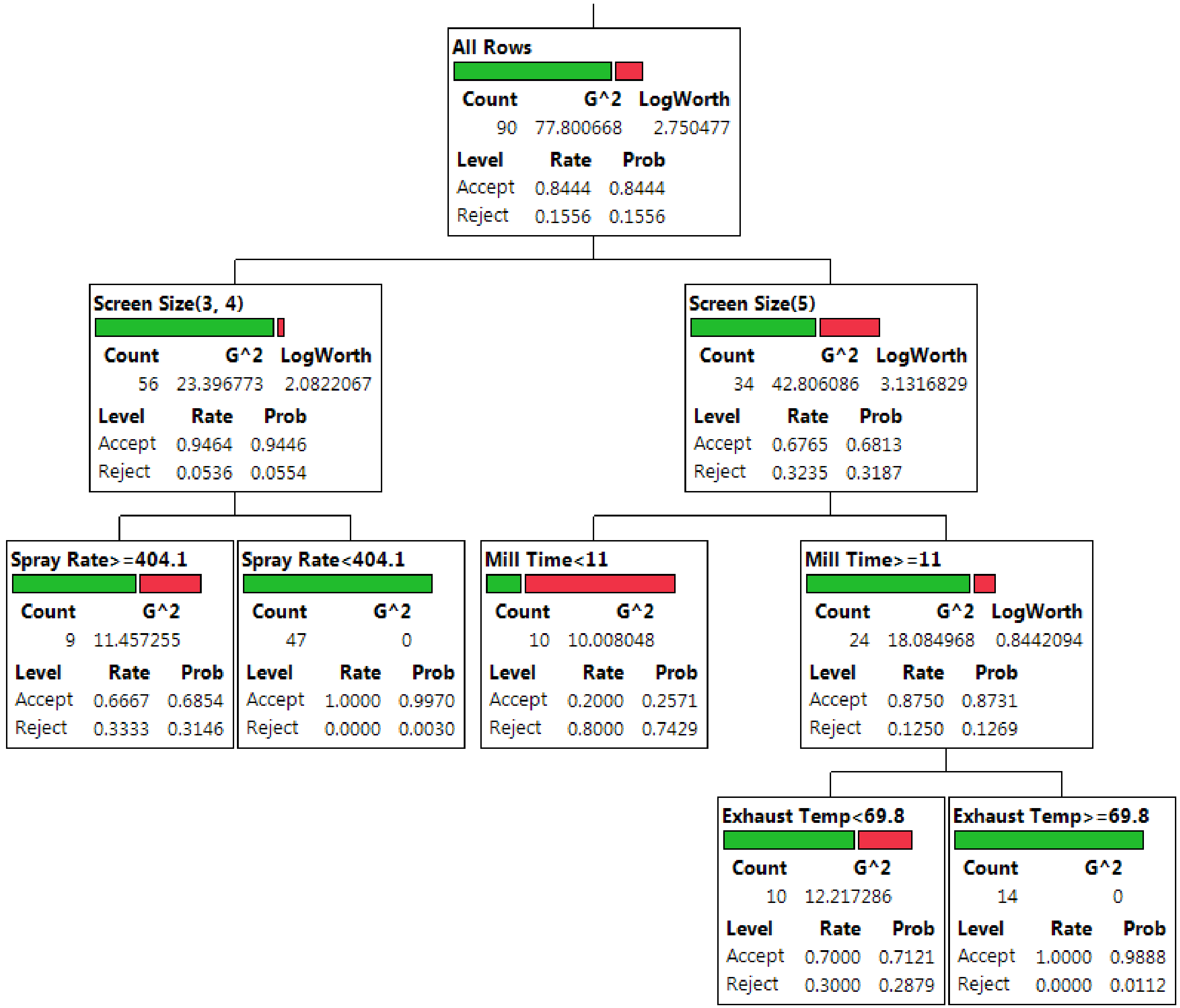


Decision Tree (Step by Step)

4th split on Exhaust Temp (< 69.8 vs. >= 69.8)

Notice variation in proportion of “1” in each branch

RSquare	N	Number of Splits
0.557	90	4

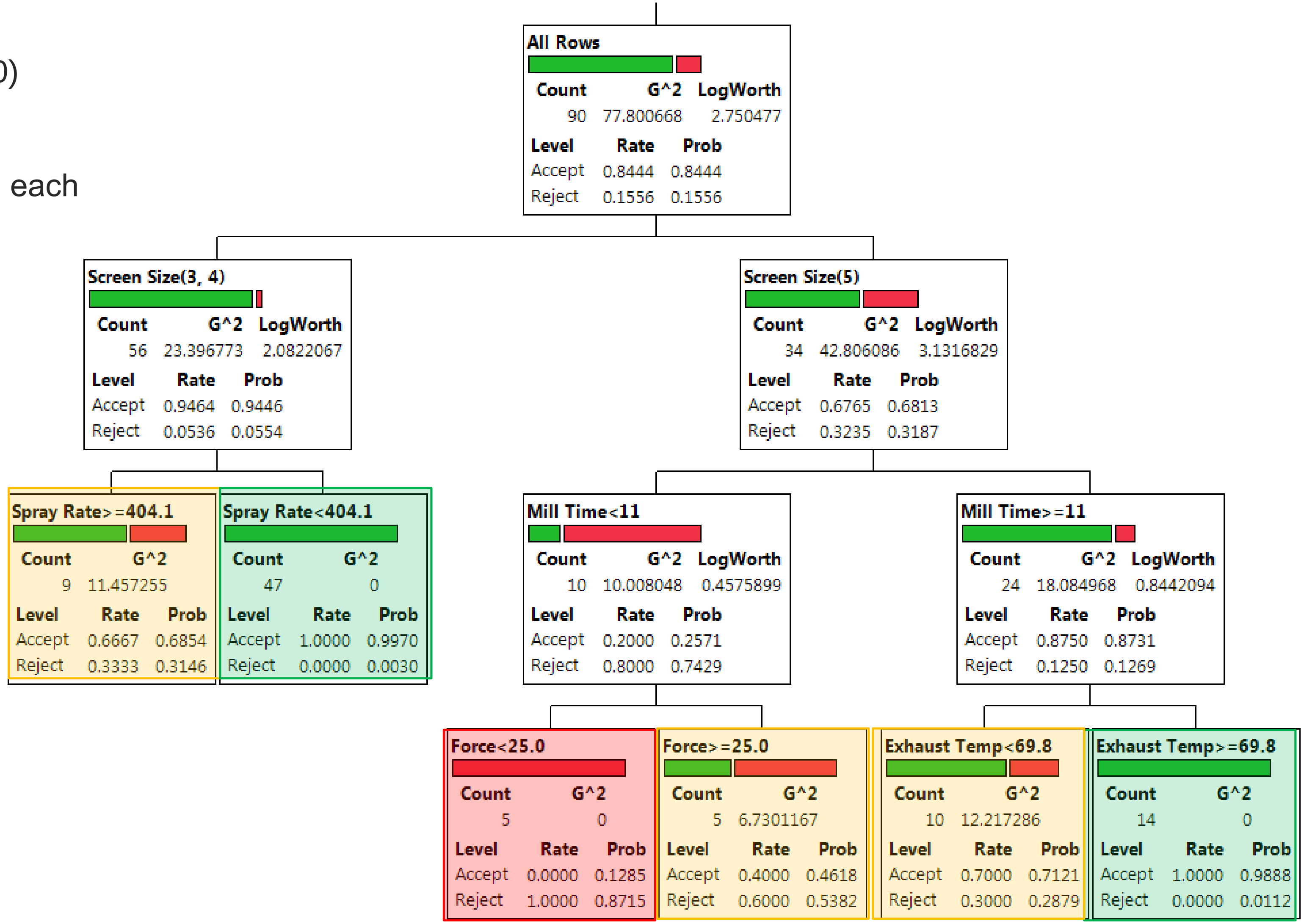


Decision Tree (Step by Step)

RSquare	N	Number of Splits
0.583	90	5

5th split on Force
(< 25.0 vs. >= 25.0)

Notice variation in proportion of “1” in each branch

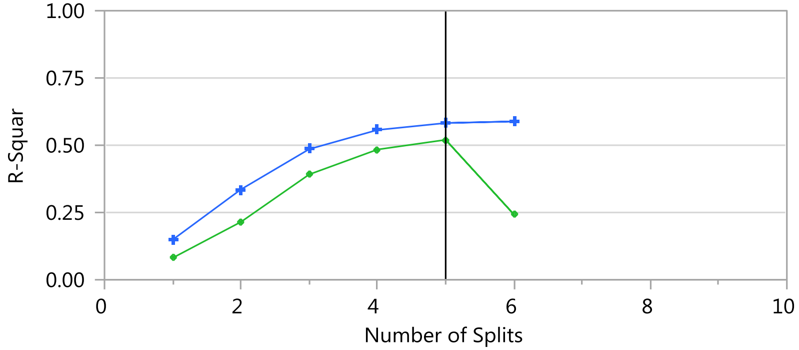


Decision Tree (Step by Step)

Crossvalidation

k-fold		-2LogLike	RSquare
5	Folde	37.3288048	0.5202
	Overa	30.4046577	0.5825

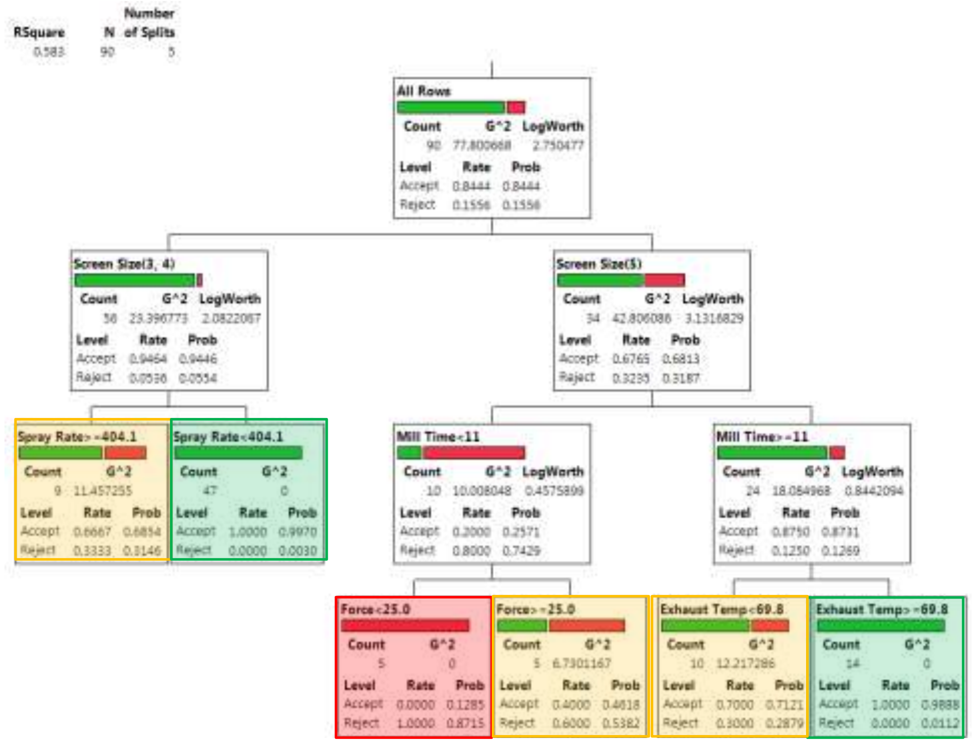
Split History



K-Fold in Green

Column Contributions

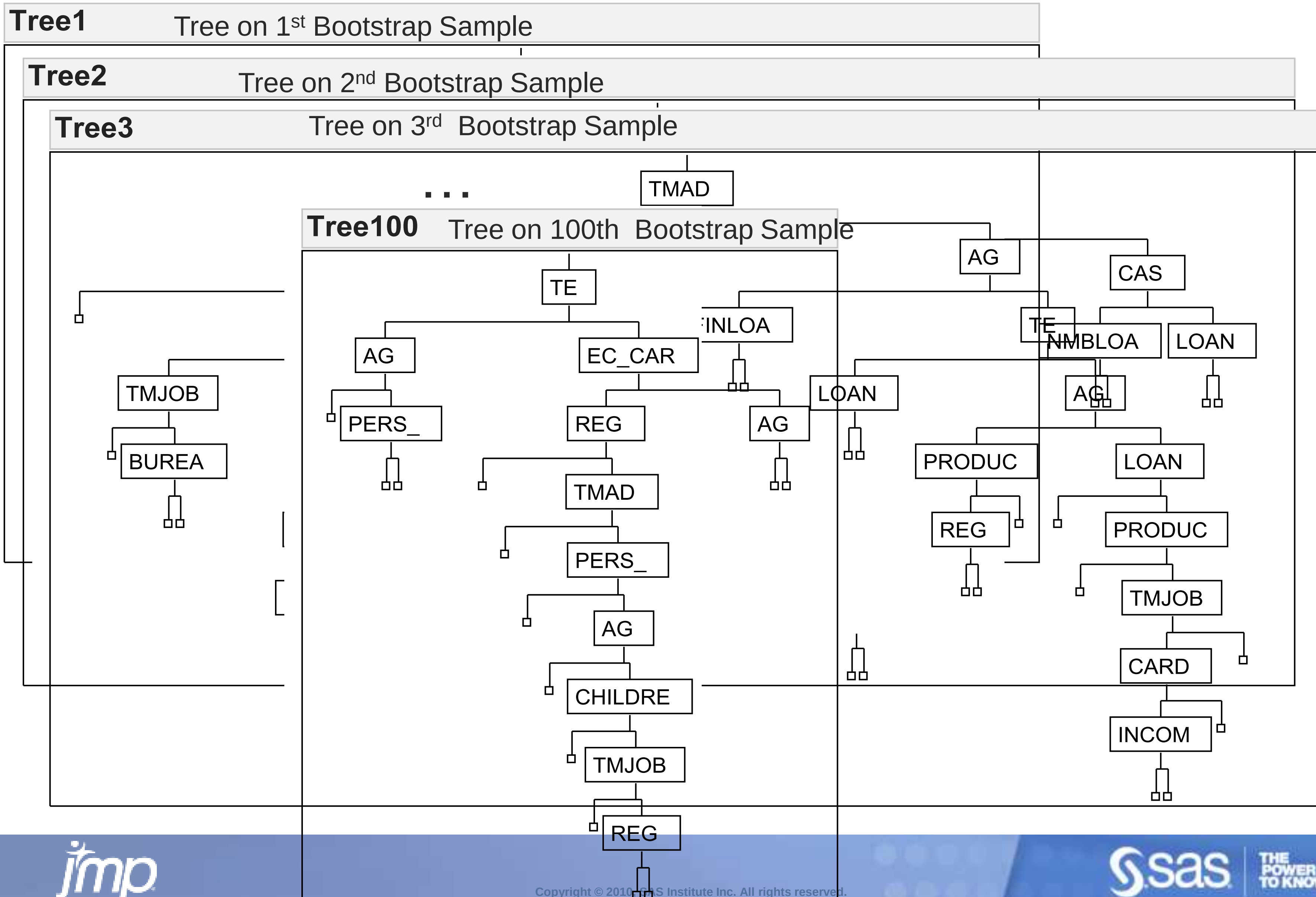
Term	Number of Splits	G^2	Portion
Mill Time	1	14.7130695	0.3104
Spray Rate	1	11.9395178	0.2519
Screen Size	1	11.5978092	0.2447
Exhaust Temp	1	5.8676817	0.1238
Force	1	3.2779318	0.0692
API Particle Size	0	0	0.0000
MgSt Supplier	0	0	0.0000
Lactose Supplier	0	0	0.0000
Sugar Supplier	0	0	0.0000
Talc Supplier	0	0	0.0000
Blend Time	0	0	0.0000
Blend Speed	0	0	0.0000
Compressor	0	0	0.0000
Coating Supplie	0	0	0.0000
Coating Viscosit	0	0	0.0000
Inlet Temp	0	0	0.0000
Atom. Pressure	0	0	0.0000



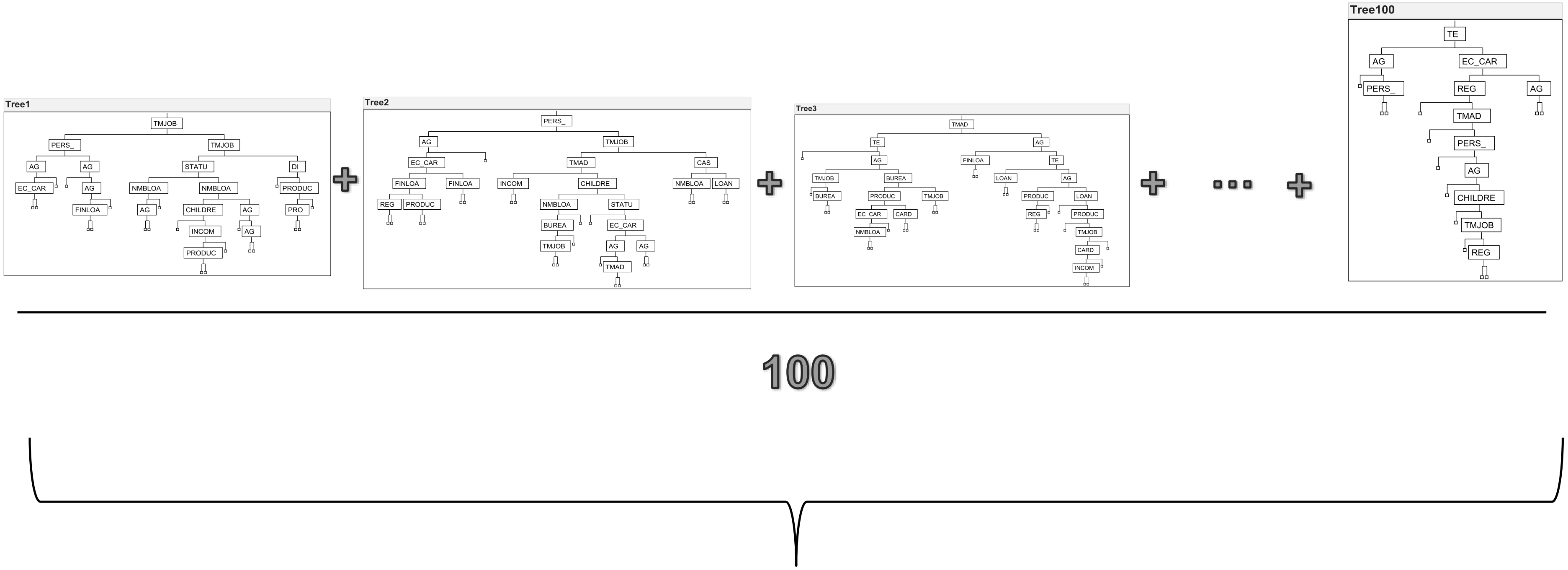
Bootstrap Forest

- Bootstrap Forest
  - For each tree, take a random sample of the predictor variables (***with replacement***) – e.g. pick half of the variables. Build out a decision tree on that subset of variables.
  - Make many trees and average their predictions (bagging)
  - This is also know as a random forest technique
  - Works very well on wide tables.
- Can be used for ***both*** predictive modeling and variable selection.
- Allows for dominant variables to be excluded from some trees giving less dominant – but still important – variables a chance to be selected.
- Valuable approach for screening variables for use with other modeling methods – e.g. neural networks.

See the Trees in the Forest



Average the Trees in the Forest



Bootstrap Forest Model

Similar results for helicopter simulation data

DECISION TREE - 6 FACTORS BOOTSTRAP FOREST

Column Contributions

Term	Number of Splits	SS	Portion
InsurgentCamouflag	50	1328.61688	0.9338
TigerSpeedRelative	36	31.1106368	0.0219
Tiger1_Distance	48	28.8649626	0.0203
TigerHeight	48	22.2499023	0.0156
num_INS2_AK47	40	8.36974799	0.0059
ConvoySpeed	32	3.6452873	0.0026

	RSquare	RMSE	N
Training	0.914	0.1170121	3874
Validatio	0.915	0.1132062	1292
Test	0.915	0.1148662	1292

DECISION TREE - 6 FACTORS

Column Contributions

Term	Number of Splits	SS	Portion
InsurgentCamouflag	6	553.514843	0.9819
TigerHeight	4	5.23947275	0.0093
ConvoySpeed	6	2.66493548	0.0047
TigerSpeedRelative	3	1.58563474	0.0028
num_INS2_AK47	4	0.66588349	0.0012
Tiger1_Distance	2	0.06006294	0.0001

	RSquare	RMSE	N
Training	0.914	0.1170276	3874
Validatio	0.915	0.1132339	1292
Test	0.915	0.1147605	1292

NOT so similar results for cyber attack data

DECISION TREE - 11 FACTORS BOOTSTRAP FOREST

Measure	Training	Validation	Test
Entropy RSquare	0.9816	0.9798	0.9807
Generalized RSquar	0.9975	0.9972	0.9974
Mean -Log p	0.0296	0.0324	0.0312
RMSE	0.0834	0.0888	0.0868
Mean Abs Dev	0.0235	0.0253	0.0247
Misclassification Rat	0.0042	0.0055	0.0048

Column Contributions

Term	Number of Splits	G^2	Portion
service	313	6647269.76	0.3546
dst_bytes	318	2378144.67	0.1269
src_bytes	642	2343701.45	0.1250
dst_host_srv_count	545	1371395.91	0.0732
count	384	1361411.35	0.0726
dst_host_diff_srv_rate	435	988535.468	0.0527
flag	190	889445.342	0.0475
dst_host_same_src_port_rat	402	881707.319	0.0470
dst_host_count	435	700494.072	0.0374
srv_count	287	669775.801	0.0357
duration	222	511537.238	0.0273

DECISION TREE - 11 FACTORS

Measure	Training	Validation	Test
Entropy RSquare	0.9486	0.8149	0.6335
Generalized RSquar	0.9925	0.9661	0.9061
Mean -Log p	0.0828	0.2979	0.5898
RMSE	0.1426	0.2127	0.2811
Mean Abs Dev	0.0387	0.0637	0.0969
Misclassification Rat	0.0230	0.0495	0.0821

Column Contributions

Term	Number of Splits	G^2	Portion
service	5	630992.402	0.5475
dst_bytes	4	128894.607	0.1118
dst_host_diff_srv_rate	3	115626.455	0.1003
src_bytes	8	97103.0428	0.0843
dst_host_count	2	71772.3696	0.0623
count	3	68716.3668	0.0596
dst_host_same_src_port_rat	3	19974.724	0.0173
dst_host_srv_count	1	10836.2482	0.0094
duration	1	5450.42578	0.0047
flag	1	3066.0292	0.0027
srv_count	0	0	0.0000

Columns Contributions for Bootstrap Forest Analysis of Cyber Data – Variable Selection w/44 Factors – 3 of which were Random Data!

Column Contributions

Term	Number of Splits	G^2	Portion
service	450	10603400.8	0.2831
dst_bytes	382	5308498.33	0.1417
src_bytes	820	4771327.16	0.1274
count	337	2700247.28	0.0721
dst_host_srv_count	528	1990388.66	0.0531
dst_host_diff_srv_rate	415	1575488.06	0.0421
flag	168	1153015.42	0.0308
srv_count	238	1115688.05	0.0298
dst_host_serror_rate	175	1060259.19	0.0283
duration	276	991351.909	0.0265
dst_host_count	499	714300.159	0.0191
dst_host_same_src_port_rat	389	616742.634	0.0165
hot	159	535399.996	0.0143
same_srv_rate	103	422795.794	0.0113
dst_host_same_srv_rate	334	421699.768	0.0113
diff_srv_rate	145	382986.204	0.0102
serror_rate	65	365667.013	0.0098
dst_host_rerror_rate	233	318445.492	0.0085
dst_host_srv_serror_rate	117	308717.284	0.0082
logged_in	40	305603.637	0.0082
srv_serror_rate	30	219339.913	0.0059
root_shell	32	203921.266	0.0054
dst_host_srv_diff_host_rate	253	196905.011	0.0053
Random Uniform	228	195145.878	0.0052
dst_host_srv_rerror_rate	81	153228.513	0.0041
protocol_type	53	152857.046	0.0041
is_guest_login	12	137886.036	0.0037
Random Normal	194	110253.474	0.0029
num_compromised	39	76703.4706	0.0020
num_file_creations	20	75279.6937	0.0020
wrong_fragment	29	72313.7688	0.0019
rerror_rate	45	59525.1111	0.0016
num_root	23	41990.5367	0.0011
Random Integer	146	21117.3276	0.0006
srv_diff_host_rate	33	17448.0232	0.0005
num_failed_logins	7	17407.5895	0.0005
srv_rerror_rate	30	16080.2873	0.0004
num_access_files	11	11528.8834	0.0003
num_shells	11	8067.77994	0.0002
urgent	4	3131.15585	0.0001
su_attempted	1	42.7170189	0.0000
land	0	0	0.0000
num_outbound_cmds	0	0	0.0000
is_host_login	0	0	0.0000

Column Contributions

Term	Number of Splits	G^2	Portion
service	450	10603400.8	0.2831
dst_bytes	382	5308498.33	0.1417
src_bytes	820	4771327.16	0.1274
count	337	2700247.28	0.0721
dst_host_srv_count	528	1990388.66	0.0531
dst_host_diff_srv_rate	415	1575488.06	0.0421
flag	168	1153015.42	0.0308
srv_count	238	1115688.05	0.0298
dst_host_serror_rate	175	1060259.19	0.0283
duration	276	991351.909	0.0265
dst_host_count	499	714300.159	0.0191
dst_host_same_src_port_rat	389	616742.634	0.0165
hot	159	535399.996	0.0143
same_srv_rate	103	422795.794	0.0113
dst_host_same_srv_rate	334	421699.768	0.0113

Top 11 of 44

Model Validation-Set Summaries

The fit below was the best of these models fit.

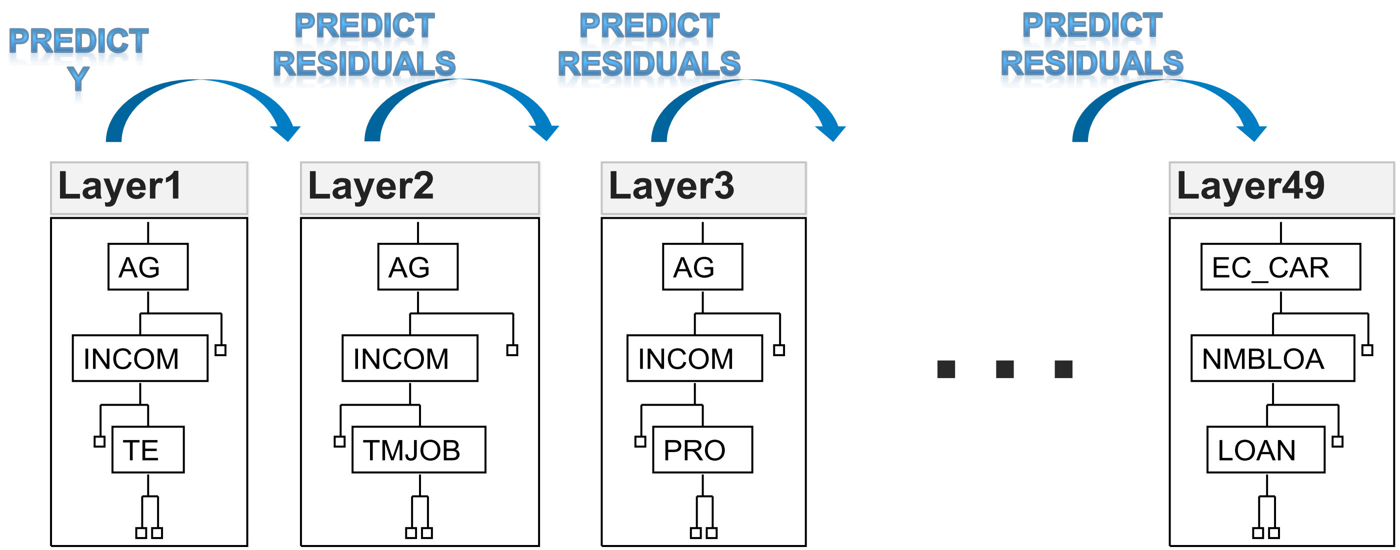
N Terms	N Trees	Entropy RSquare	Misclassification Rate	Avg -Log p	RMS Error	Avg Abs Error
11	200	0.9786	0.0040	0.0336	0.0856	0.0279
14	53	0.9811	0.0040	0.0297	0.0816	0.0243
18	48	0.9831	0.0039	0.0265	0.0770	0.0215

Term	Number of Splits	G^2	Portion
Random Uniform	228	195145.878	0.0052

Boosted Tree

- Beginning with the first tree (layer) build a small simple tree.
- From the residuals of the first tree, build another small simple tree.
- This continues until a specified number of layers has been fit, or a determination has been made that adding successive layers doesn't improve the fit of the model.
- The final model is the weighted accumulation of all of the model layers.

Boosted Tree Illustrated



Models M1 M2 M3 M49

Final Model

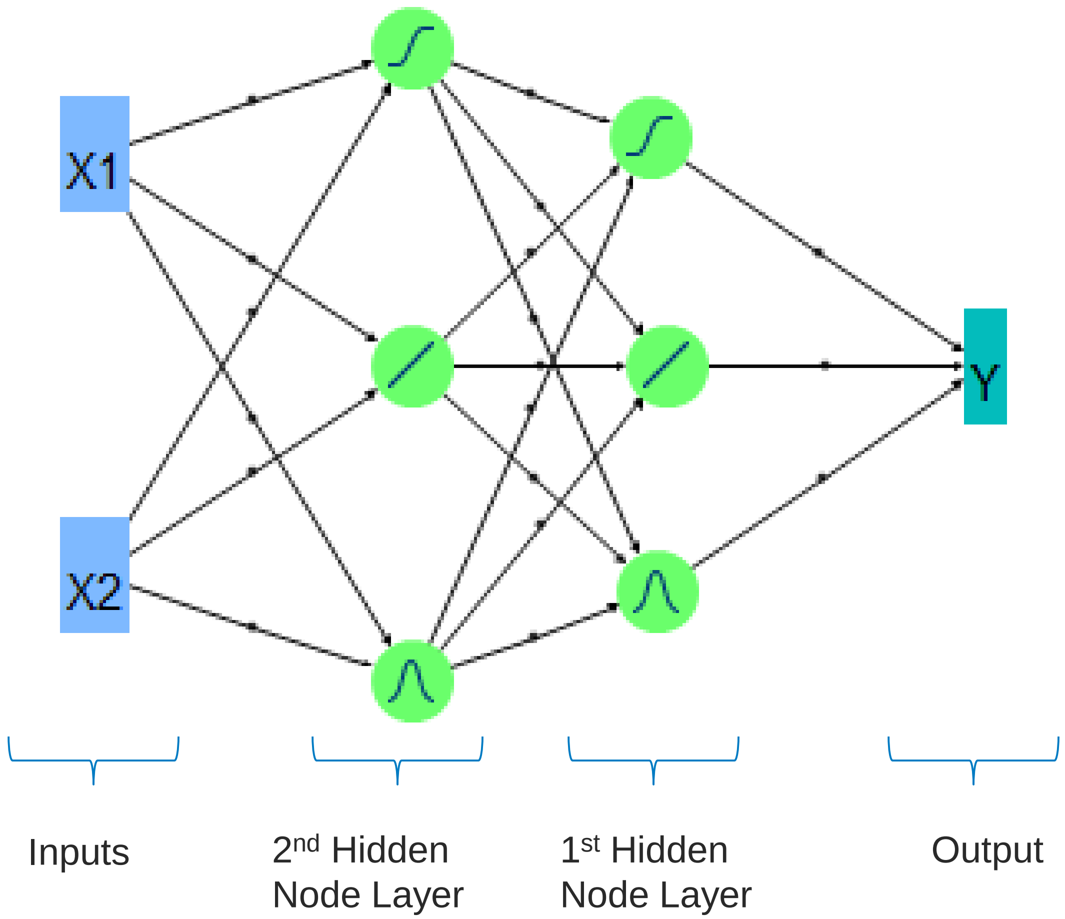
$$M = M1 + \varepsilon \cdot M2 + \varepsilon \cdot M3 + \cdots + \varepsilon \cdot M49$$

ε is the learning rate

Neural Networks

- Neural Networks are highly flexible nonlinear models.
- A neural network can be viewed as a weighted sum of nonlinear functions applied to linear models.
  - The nonlinear functions are called activation functions. Each function is considered a (hidden) node.
  - The nonlinear functions are grouped in layers. There may be more than one layer.
- Consider a generic example where there is a response Y and two predictors $X1$ and $X2$. An example type of neural network that can be fit to this data is given in the diagram that follows

Example Neural Network Diagram



Neural Networks

- Big Picture
  - Can model:
    - » Continuous and categorical predictors
    - » Continuous and categorical responses
    - » Multiple responses (simultaneously)
  - Can be numerically challenging and time consuming to fit
  - NN models are very prone to overfitting if you are not careful
    - » There are several ways to help prevent overfitting
      - » Some type of validation is required

Choosing the Best Model

- In many situations you would try many different types of modeling methods
- Even within each modeling method, there are options to create different models
  - In Stepwise, the base/full model specification can be varied
  - In Bootstrap Forest, the number of trees and number of terms sample per split
  - In Boosted Tree, the learning rate, number of layers, and base tree size
  - In Neural, the specification of the model, as well as the use of boosting
- So how can you choose the “best”, most useful model?

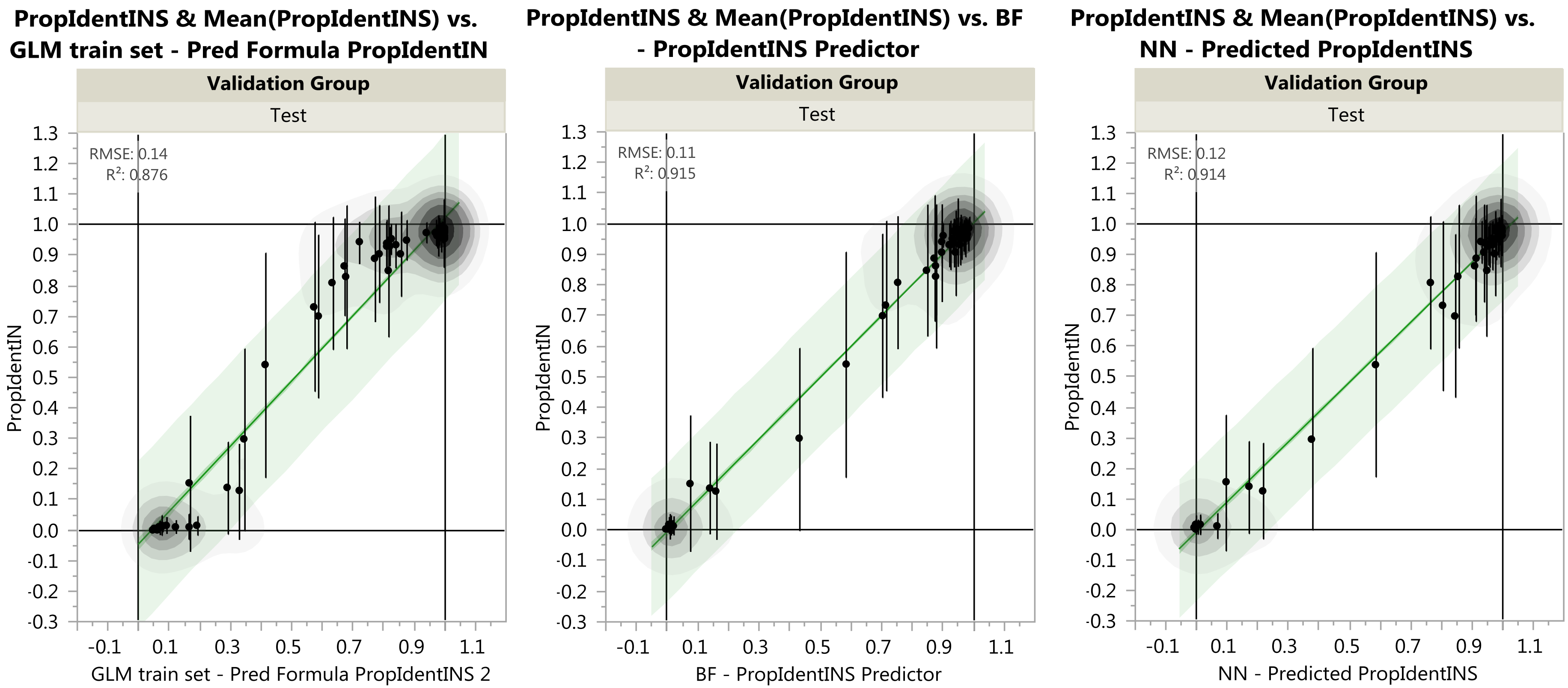
The Importance of the Test Set

- One of the most important uses of having a training, validation, AND **test set** is that you can use the test set to assess each model on the same basis.
- Using the test set allows you to compare competing models on the basis of model quality metrics
  - R^2
  - Misclassification Rate
  - Actual vs. Prediction (Confusion Matrix)
  - ROC (Receiver Operating Characteristics) Curves and AUC (Area Under Curve – of ROC Curve)

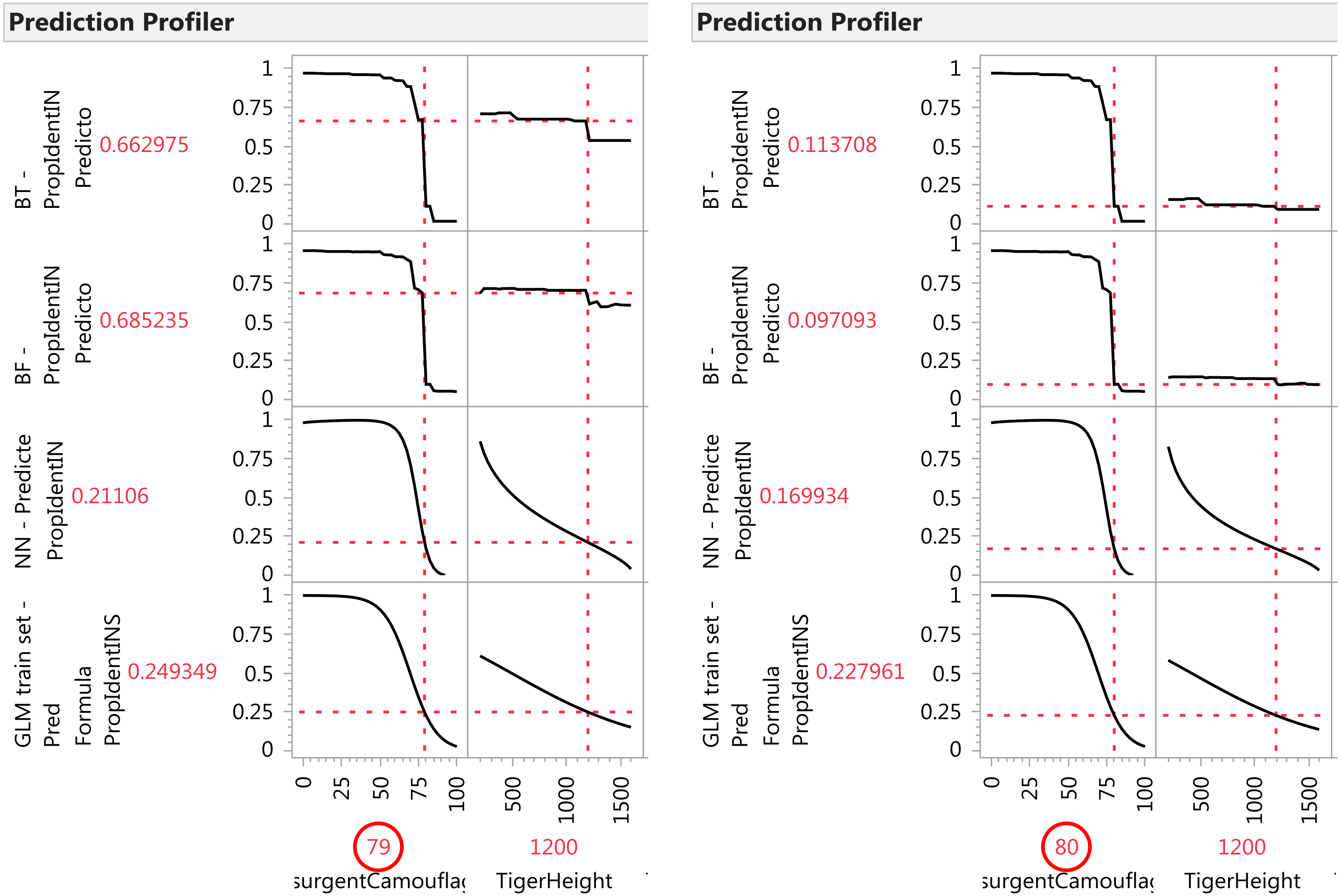
Measures of Fit for PropIdentINS

Metrics for Just the Test Subset

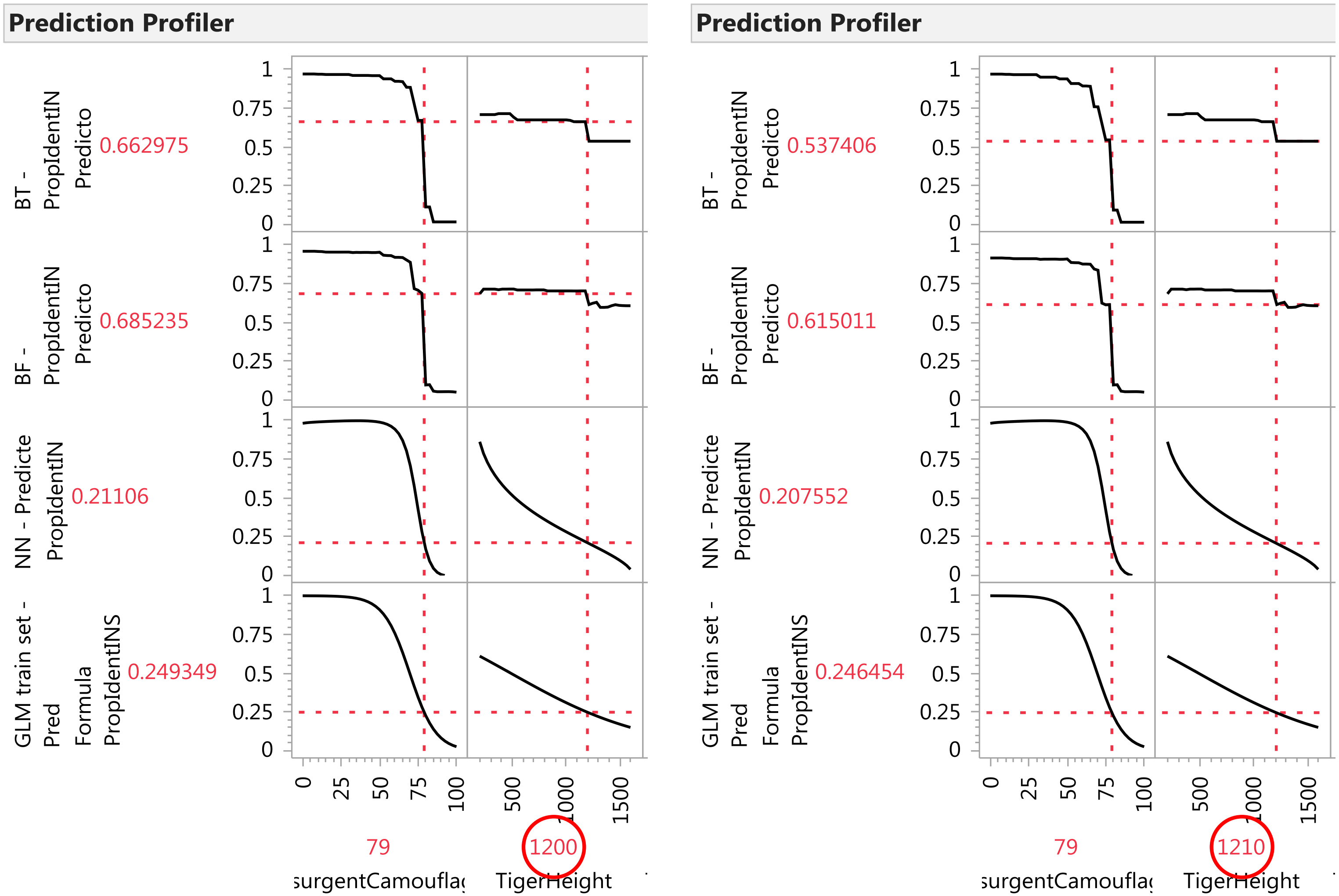
Predictor	Creator	.2.4.6.8	RSquare	RASE	AAE	Freq
GLM ALL Data Pred Formula PropIdentINS			0.8736	0.1397	0.0917	1292
Partition K-Fold PropIdentINS Predictor	Partition		0.9172	0.1131	0.0595	1292
BF - PropIdentINS Predictor			0.9149	0.1147	0.0609	1292
BT - PropIdentINS Predictor			0.9130	0.1159	0.0619	1292
NN Single Layer 33% Predicted PropIdentIN	Neural		0.9069	0.1199	0.0560	1292
NN - Predicted PropIdentINS			0.9105	0.1176	0.0570	1292
Probability(PropIdentINS=1)	Fit Generalize		0.8719	0.1407	0.0925	1292



Change Camouflage from 79 to 80
Decision Tree Predictions Drop by 6X



Change Tiger Height from 1200 to 1210
Decision Tree Predictions Drop by 10% to 20%!



Summary

- Learned about different data mining/model building methods
  - BUT, before building any models, use an “Honest Assessment” division of data into Train, Validate(Tune) and Test subsets to make models robust to overfitting AND comparisons of models fair
- Discussed creation of and showed results for some of these models fitting simulation data of helicopter surveillance
  - Decision Tree – Partition, Bootstrap Forest, Boosted Tree
  - Neural Net – Single Layer, Dual Layer, Boosted Neural
  - Generalized Linear Model (GLM) – Binomial Dist. w/Logit Link
- Evaluate and compare to choose best predictor
  - Table of metrics including R-Square
  - Plots of Actual vs. Predicted for the Test subset
- Gain insight into simulation model
  - Compare Prediction Profilers for different models – some are “smooth” models and some have “cut points”

Thanks.

Questions or comments?

tom.donnelly@jmp.com