Verisk Analytics

Interaction Detection in GLM – a Case Study

THE SCIENCE OF **RISK**SM

Chun Li, PhD
ISO Innovative Analytics

March 2012

Agenda

- Case study
- Approaches
  - Proc Genmod, GAM in R, Proc Arbor
- Details

Case Study

- Personal Auto loss prediction
  - Pure premium prediction (GLM – Tweedie)
  - Inputs:
    - Environment components
    - Vehicle components
    - Driver components
    - Household components
  - Need to detect interactions among the components

Components

Environment components (freq and sev each)

- Traffic density
- Traffic composition
- Traffic generators
- Weather
- Experience and Trend

Driver components

- Driver chars (age, gender, marital, good student etc)
- Violation history
- Claim history

Vehicle components

- ISO Symbol relativity
- Price new relativity
- Model year relativity
- Body style and dimension (COL)
- Performance and safety (COL)
- Theft (COMP)
- Weather (COMP)
- Animal (COMP)
- Glass (COMP)
- All other perils (COMP)

Driver components

- Usage/mileage
- Household composition

Challenges

- There are many different approaches in interaction detection
- We are constrained by:
  - a GLM model in SAS
  - large dataset (>1 million)
  - large number of interaction pairs
  - Interpretability is required

Approach

Step 0
- Build main effect model
- Aim to model the residue using interaction terms

Step I
- Automated pair wise selection
- Based on standalone contribution

Step II
- Manual selection from Step I results
- Based on marginal contribution in GLM

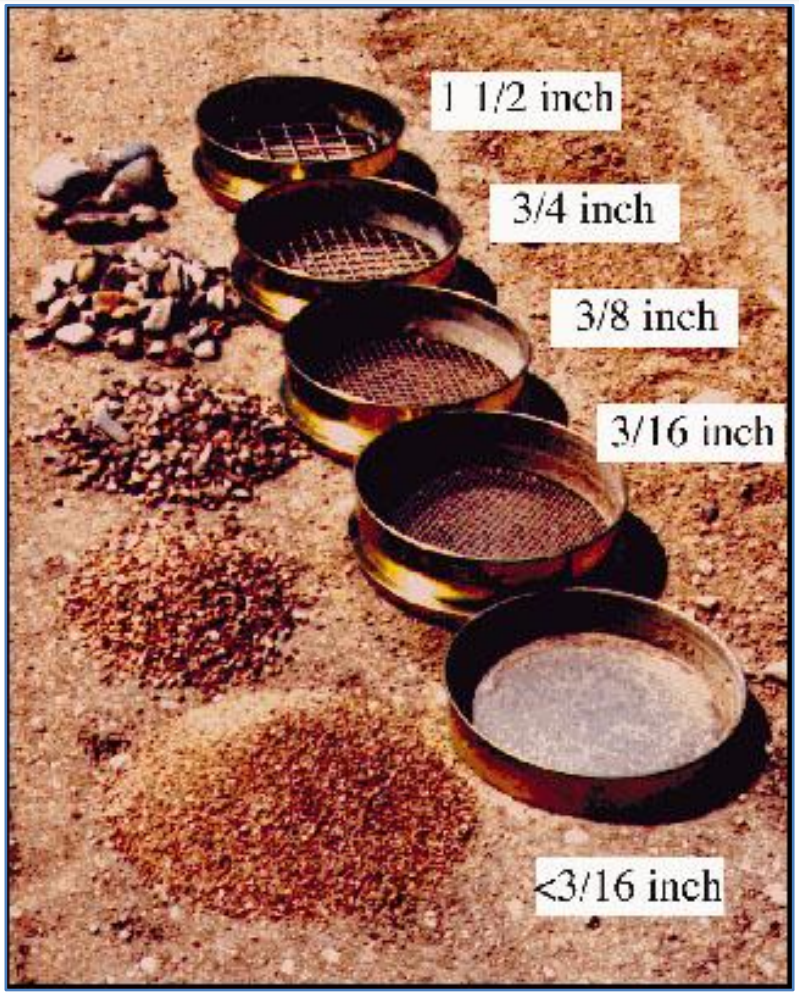
Step III
- Validation/Refinement/Finalization

**** We'll be focusing on Step I***

Step I - Details

The purpose of step I is to sort out significant interaction pairs from insignificant ones so that we can focus on those that have higher potentials.

The main idea is to add each pair to the model to predict the residual, measure the contribution, and rank order the list of pairs based on their contribution.



Step I - Details

Three methods are used

- Use Proc Genmod in SAS
- Use GAM in R
- Use Proc Arbor (Regression Tree) in SAS

Proc Genmod in SAS

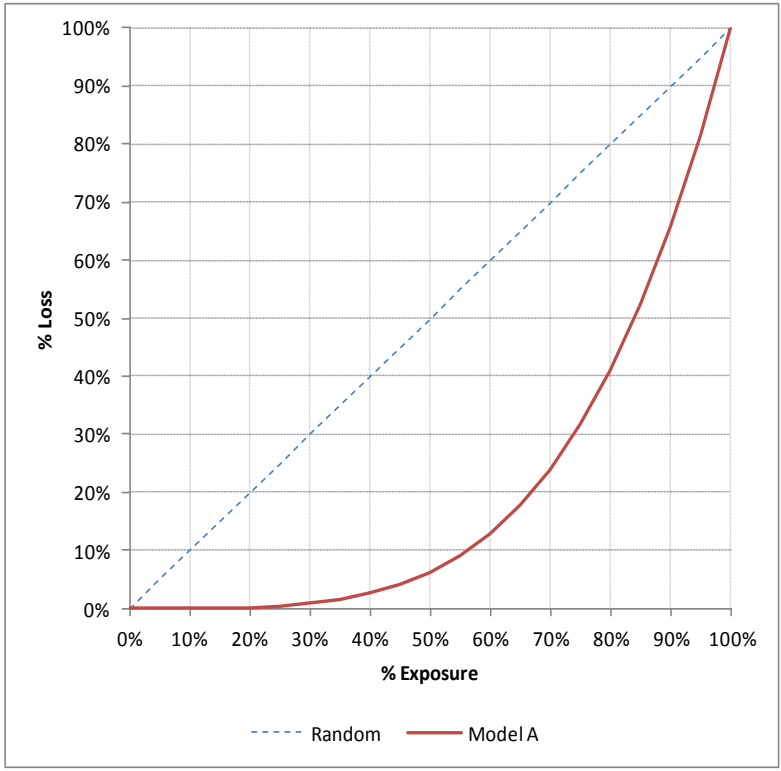
- Use main effect model as offset
- Add the interaction pair to the model
- Use ‘Increase in Gini’ as the measurement of lift
- Created SAS macro to loop through the list of all pairs and output the list ranked by the lift from high to low

Gini Definition

- A measurement of model predictiveness
- More on rank ordering than spot value

Definition:

- ✓ Sort the population by model score
- ✓ Calculate the % of accumulative exposure and loss from low to high score
- ✓ Chart the points using % exposure as x-axis and % loss as y-axis
- ✓ Gini is defined as 2 times the area between the diagonal line and the curve



Proc Genmod in SAS

- Interaction terms
  - Classed for both
  - One classed and one linear
  - Both linear

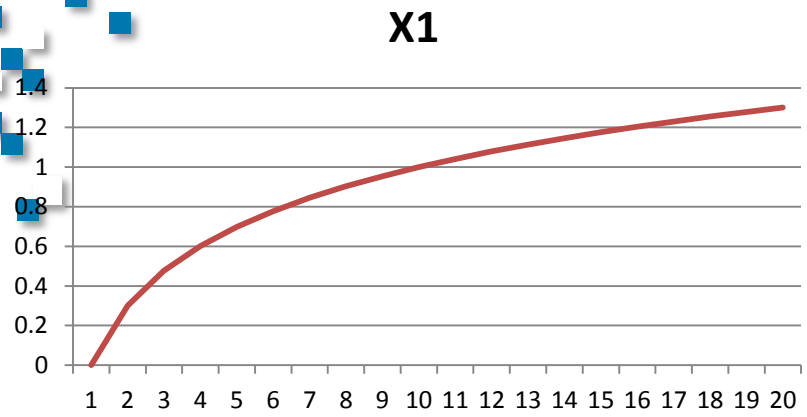
The linear assumption is based on the fact that the components (or sometimes, the log transformation of the components) are developed in the way that they have linear relationship with the target.

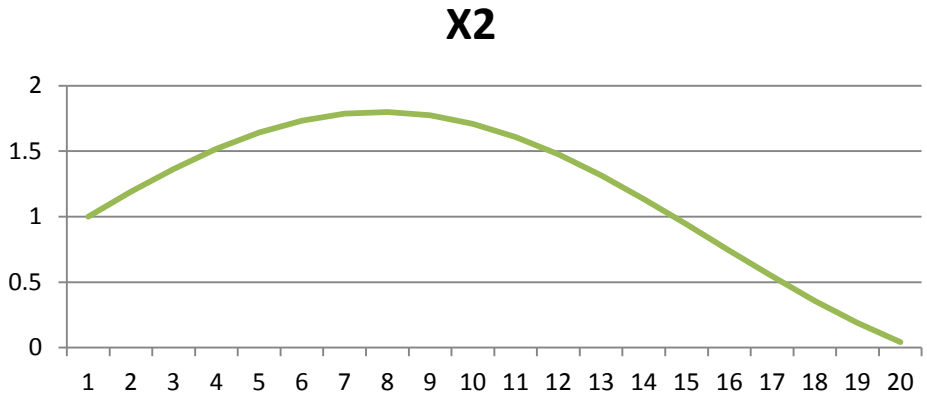
GAM in R

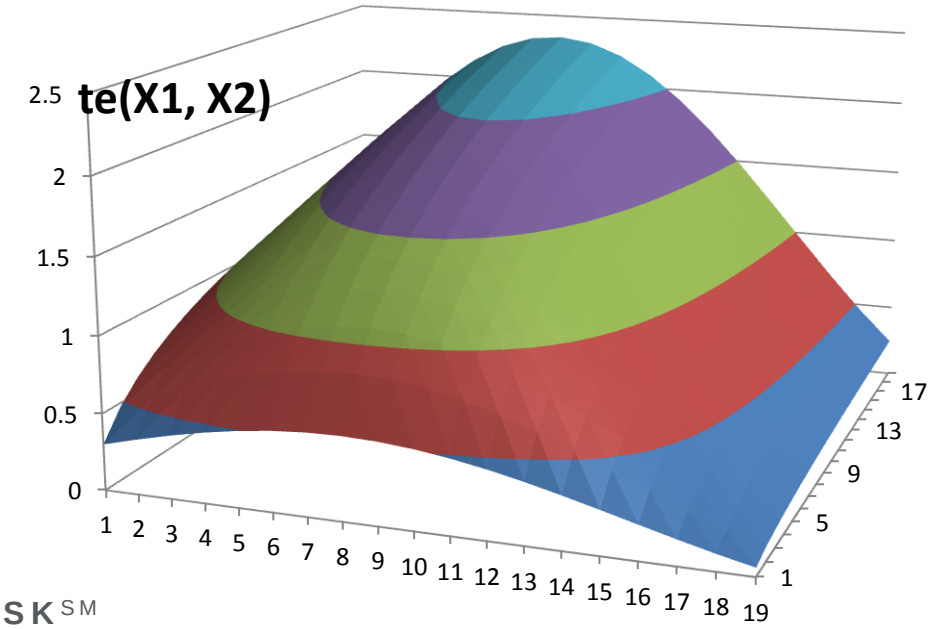
GAM = Generalized Additive Model

- R package: mgcv
- Able to do Tweedie distribution with Log link
- Takes weight
- Fits splines
- Multi-dimentional smoothing for interactions
  - Smooth classes: s(a, b)
  - Tensor product smooths: te(a, b)

Illustration of interaction surface







GAM in R

- Use main effect model as offset
- Add the interaction pair to the model
- Use ‘Decrease in AIC’ as the measurement of lift
- Created R process to loop through the list of all pairs and output the list ranked by the lift from high to low

Proc Arbor in SAS

Proc Arbor in SAS

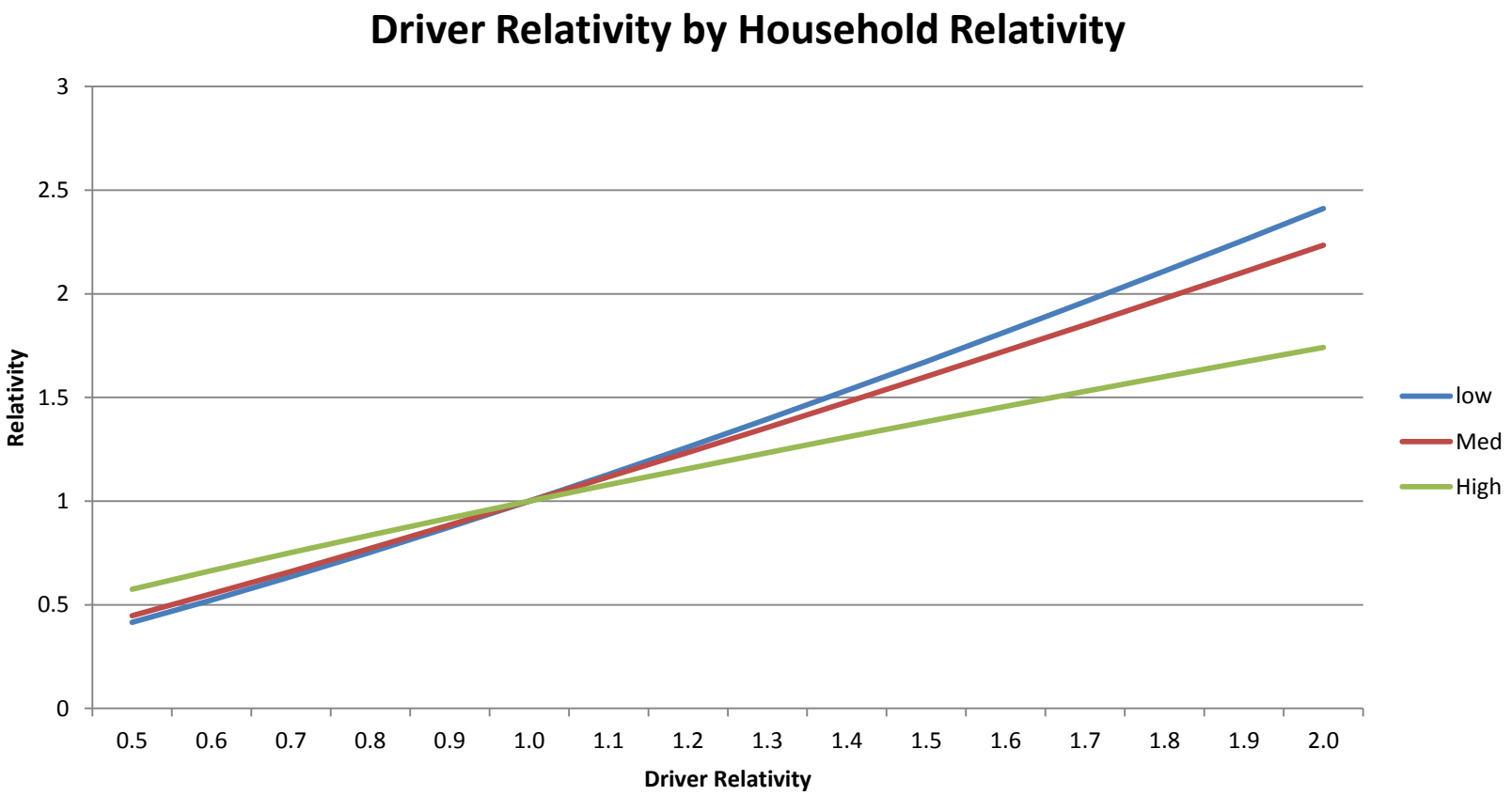
- The same algorithm behind EMiner's Decision Tree Note
- It allows programmable process
  - Loop through a list
  - Build a model
  - Evaluate the model performance

Proc Arbor in SAS

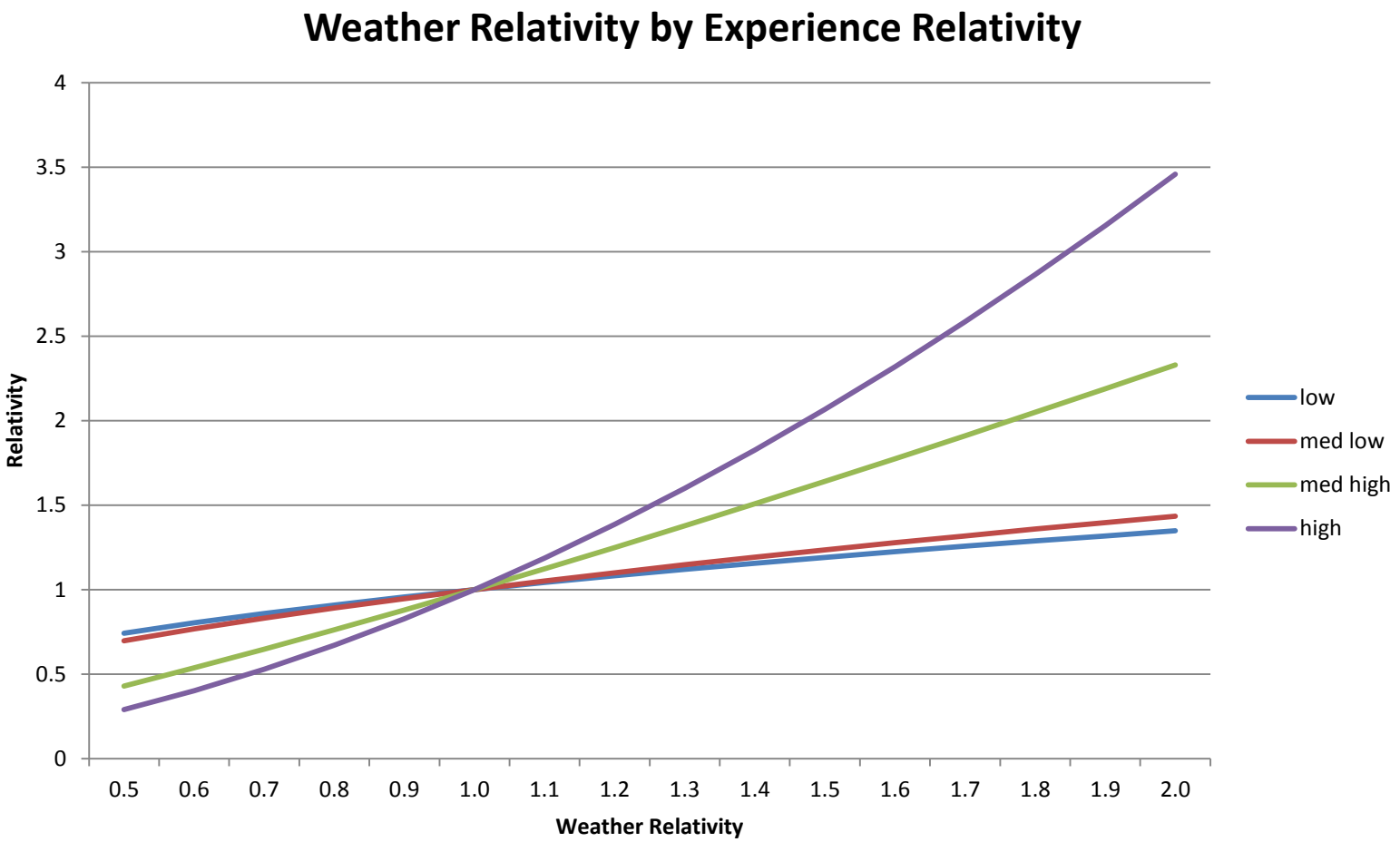
Proc Arbor in SAS

- Use residual of main effect mode as target
- Build regression tree using the pair of variables
- Measurement of lift
  - RSE_N=sqrt(MSE*Leaf_Count)
- Created SAS macro to loop through the list of all pairs and output the list ranked by the lift from high to low

Example – Collision Coverage



Example – Collision Coverage



Summary

- Most of the significant pairs are captured by proc Genmod method
  - Closest to the final model format
- Both GAM in R and proc Arbor detect some additional significant interaction pairs, but also give some false positives
  - Need to convert to the format that Proc Genmod can handle

Q & A

Questions?