Simultaneous Variable and Sample Selection for PLS Calibrations Using a Robust Genetic Algorithm

Enric Comas*, Patrick Wiegand, Randy Pell

The Dow Chemical Company

ecomas@dow.com

8
Hu
HUMAN
7E+09

Problem Statement

- Building a multivariate calibration model begins with two main steps – assembling a set of calibration samples that are representative and have accurate reference values, and choosing appropriate variables with which to make a model
- Given that all concentrations in a dataset are accurate, many tools exist for choosing the best subset of variables for regression.
- Given the best set of variables, tools also exist to identify concentration outliers.
- If variables are not known, and some outliers may also be present, then it may not be possible to reliably identify either.

Implementation

- GA selects a set of variables to evaluate
- A robust PLS is done to identify outliers using a resampling approach
- Reduced sample set is used to generate a cross-validation error from standard PLS
- Variables corresponding to best SECV are propagated

Robust PLS Regression (Verboven and Hubert)

- Use Minimum Covariance Determinant Estimator to define location and scatter of points
- Use resampling and re-weighting based on scatter to determine distance from location for each point
- Outliers are defined as those samples with large orthogonal distance or large absolute concentration residual

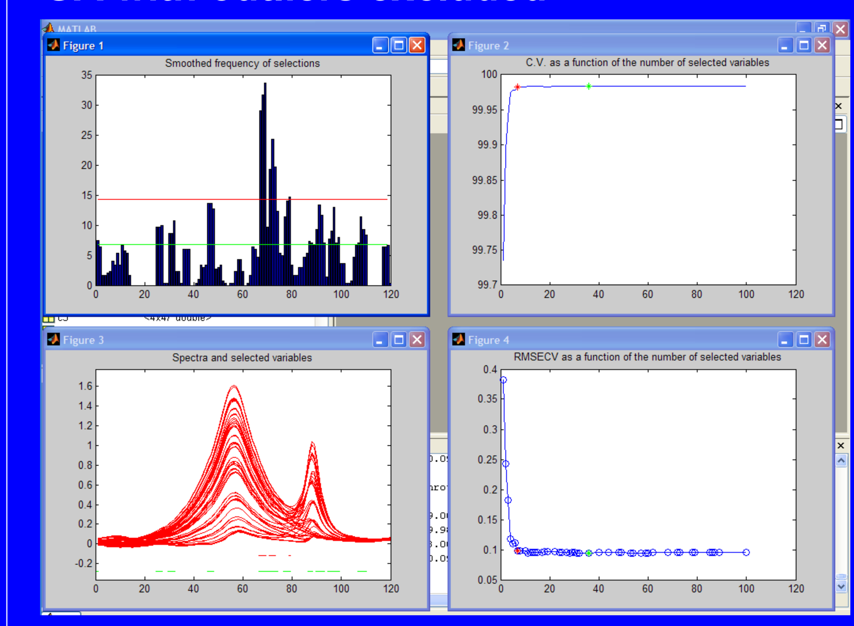
Variable Selection by GA (Leardi)

- Choose 30 subsets of variables (1st generation)
- Do PLS and get Std. Error of cross-validation for statistically valid number of factors.
- Do backward elimination of variables.
- Track variable selection frequency.
- Swap some of the variables for the best SECV combinations.
- Randomly add/delete some variables. Now have second generation. Repeat PLS, etc.
- Stop after 100 evaluations (= 1 "run")
- Do many runs and use variables with highest frequency of selection.

Concentration Outlier Dataset Description

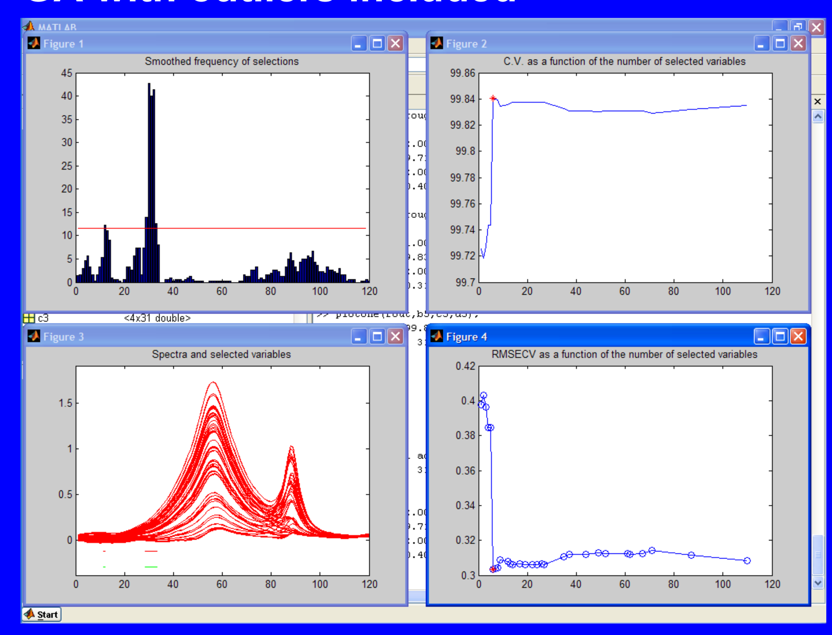
- A synthetic dataset was constructed from Mid-IR olefins spectra.
- Previously known concentration outliers were identified
- Spectra were reconstructed for these samples using results generated from other good samples
- Original bad concentrations were associated with the reconstructed spectra (samples 87-108)

GA with outliers excluded



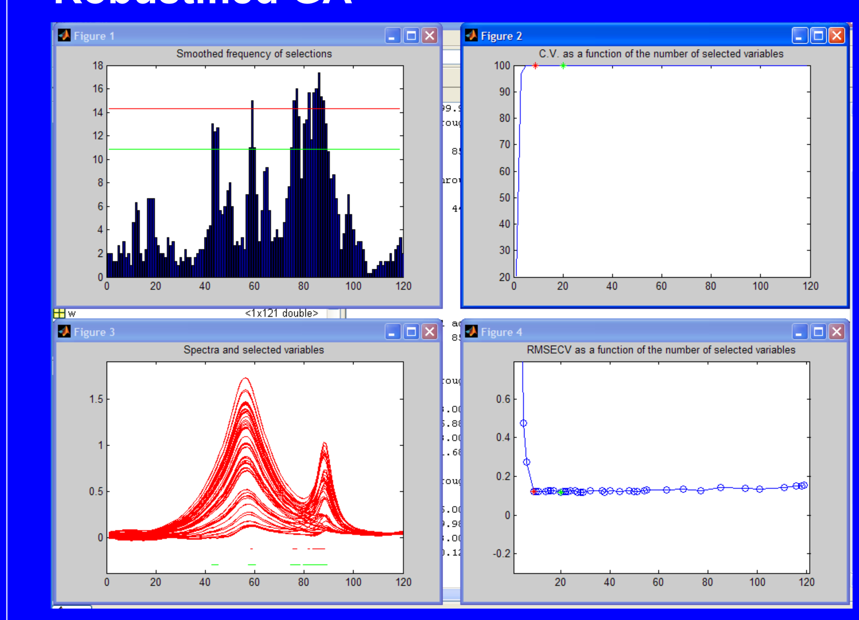
The normal GA algorithm developed by Leardi chooses regions near the inter-section of the two bands.

GA with outliers included



The same GA chooses regions to the left of the large band due to influence of outliers.

Robustified GA

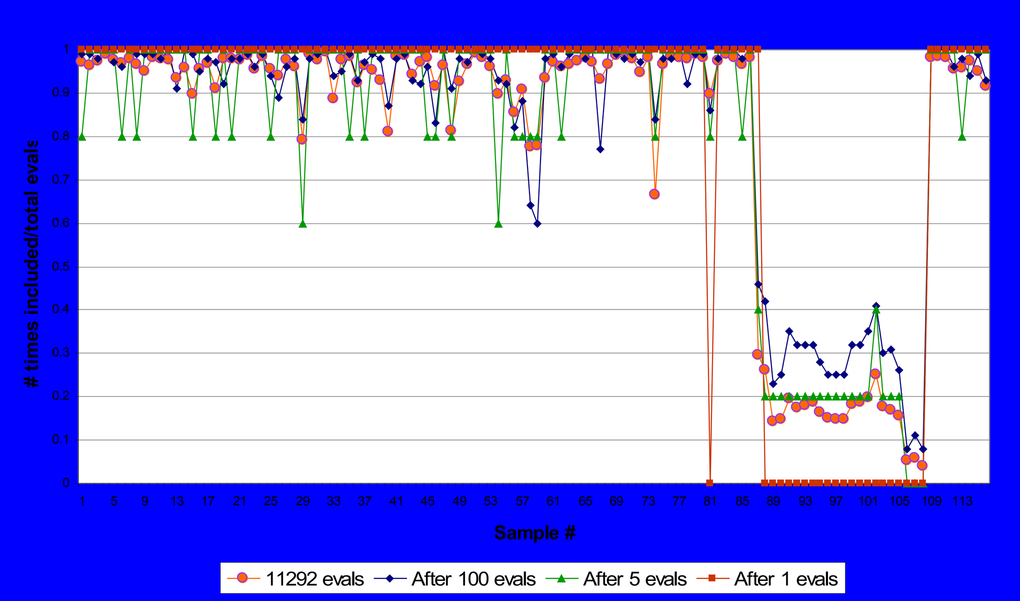


The robust GA chooses regions in approximately the same position as the normal GA operating on an outlier-free dataset.

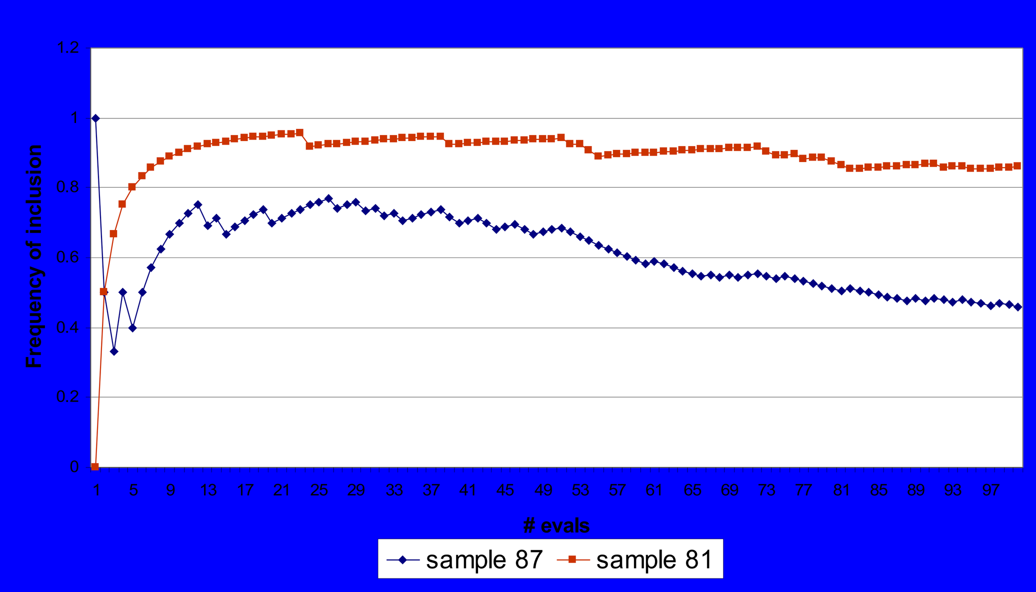
Can we increase the speed?

- Full robustification takes about 3 days to run
- Possible strategies:
  - Limit the number of evaluations in the GA
  - Only do robust GA until the outlier sample vector stabilizes (i.e. frequency of outlier identification)

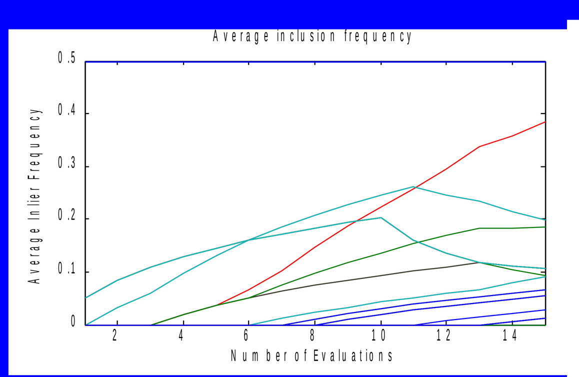
Sample inclusion frequency for # of evaluations (100 evals = 1 run)



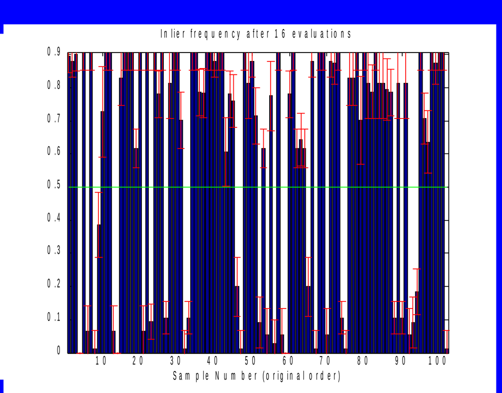
Behavior of borderline samples



Implementation of inclusion frequency stabilization

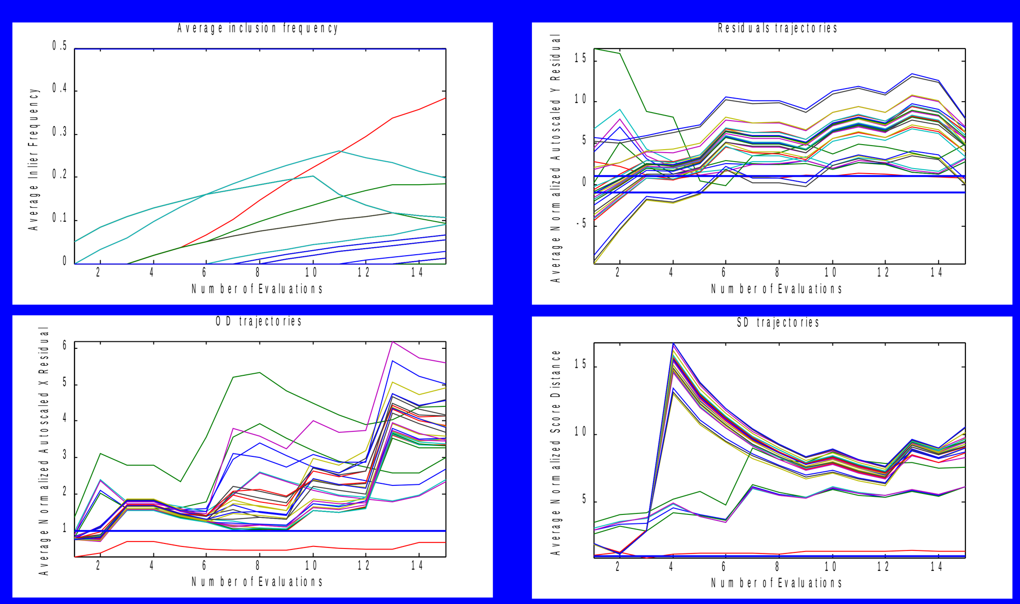


Average inclusion frequency (15-pt window) for outliers only from plastics dataset



All samples stabilized after 16 evaluations (cutoff of 0.5, 15-pt window)

Reasons for outlier designation
Samples rejected for residual or OD violation only – high Mahalanobis distance (or SD) OK



Conclusions

- Use of GA when outliers are present can lead to poor variable choices
- Addition of robust PLS to inner loop of GA can successfully identify outliers
- Robustified GA achieves performance similar to no outliers present
- Outlier identification stabilizes after a single run of GA and is not needed for further GA runs.

Future Work

- Testing with different datasets:
  - Spectral outliers present
  - Both spectral and concentration outliers present
- Totally GA approach
  - Simultaneous GA optimizations
    - Vector for variable selection
    - Vector for sample selection

Acknowledgements

- GA routines were developed through Dow support of Riccardo Leardi
- Robust routines were part of the Libra Matlab toolbox, developed by Mia Hubert