SESUG 2022 - Paper 170

Cutting Edge Regression Methods: Ridge, LASSO, LOESS, and GAM

David J Corliss

Peace-Work, Plymouth, MI

ABSTRACT

This paper presents a brief introduction to recent advances in regression methods. Techniques demonstrated include ridge regression, LASSO, local polynomial regression (LOESS), and generalized additive models (GAM). Each method is presented separately, with a description of the SAS procedure used to implement them and recommendations for apply the methods in practical situations. A quick introduction to each method followed by two worked examples, with discussion of use cases, and options for SAS procedures and producing graphical output.

Ridge Regression

In regression analysis, multicollinearity occurs where some predictor variables are partly correlated. These are situations where the correlation between predictor variables to too strong to dismiss those with a weaker correlation to the outcome to dismiss, and yet the predictors are not very highly correlated so that they are effectively equivalent in the prediction. Where multiple variables make distinct, important contributions to a model but still have substantial correlation, the parameter estimates provided by ordinary regression may be less accurate. Penalized regression methods address this problem by creating a constraint, called a “penalty” to reduce the residual sum of the squares. This process, called shrinkage, reduces the impact of sampling variation at the cost of introducing a small amount of bias. Different penalized regression methods deploy different means for reducing the impact of multicollinearity. Balancing the benefits of shrinkage with the amount of bias optimizes the model parameters to increase the accuracy of the prediction.

In Ridge Regression, contributions from predictor variables are reduced by the *square of the magnitude* of the coefficients. This favors models with many strong contributors. In SAS, ridge regression is coded using PROC REG with the option RIDGE. The RIDGE option supports measurement and analysis of the amount of collinearity and parameterization of the ridge factor. The process is iterative, using a Ridge Factor that is specified using a starting value, ending value, and a step. The ridge regression process evaluates the model at each step from the starting value to the ending value.

<u>Example</u>: predicting the percentage of K-12 students in each state who are (homeless hs_pct_22) using economic and demographic factors. These include a number of factors related to poverty, including Gini Index for income distribution, the year over year change in the Gini Index, the poverty rate, high school gradation rate, and other factors which show some degree of correlation. Notice that the rate of homelessness for the current year is predicted using current and recent socioeconomic data, As the federal reporting of homelessness is retrospective by two years, this model predicts a current value that will not be available util a future date – in this instance, not for two years. In Time Series Analysis, this is an example of “Now-Casting”.

```
proc reg data=rm.homelessstudents ridge=0 to .04 by .005;
   outvif outest=ridgests plots(only)=ridge(unpack VIFaxis=log);
   model hs_pct_10 = GINI_10 GINI_pct_change Pov_Change_09_11
                     aa_pct hisp_latino_pct indian_alaskan_pct
                     high_school_grad_pct_10 mhi_09 mfi_acs_10;
run;
```

This implementation uses the REG Procedure with the RIDGE option, which specifies the parameterization of the Ridge Factor. The size of the ridge factor should always begin at 0 and seldom exceeds 0.4. In this case, the size of the step between iterations is 0.005, so there are 9 steps from 0.0 to 0.4.

SAS procedure options that may be helpful include:

<u>REG Statement Options</u>

- ridge – ridge parameter limits and step size
- outvif – output variance inflation factor => severity of multicollinearity
- outseb – output standard errors and parameter estimates

<u>PLOT Statement Options</u>

- all – lots of plots but many are seldom used
- ridge – shrinking by ridge parameter as the model converges

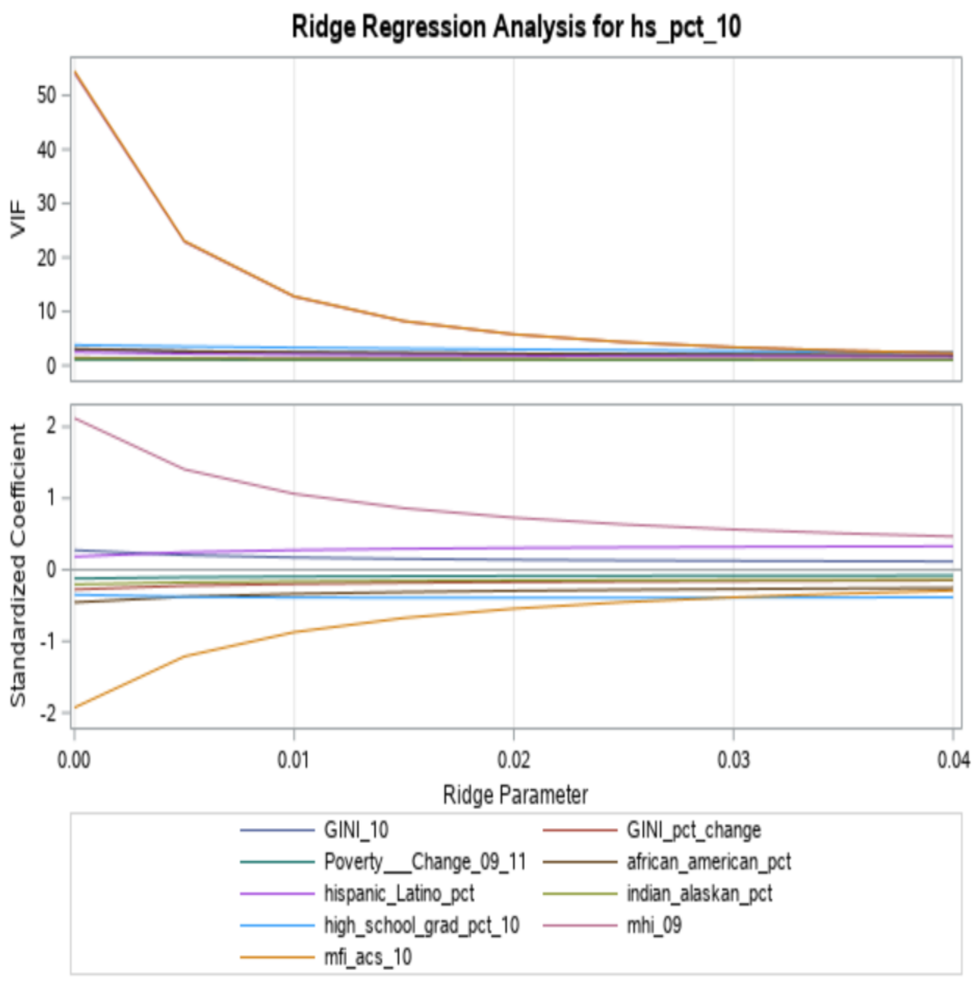
Output for this example:

The REG Procedure
Model: MODEL1
Dependent Variable: hs_pct_10 hs_pct_10

Number of Observations Read	52
Number of Observations Used	51
Number of Observations with Missing Values	1

Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	0.02665	0.00296	4.99	0.0002
Error	41	0.02435	0.00059384		
Corrected Total	50	0.05100			

Root MSE	0.02437	R-Square	0.5226
Dependent Mean	0.01949	Adj R-Sq	0.4178
Coeff Var	125.05331		



Ridge Regression – Another Example

This example uses the option OUTSEB for model parameters. In this application, a final run with the strongest variables from earlier runs performs well.

```
proc reg data=rm.homelessstudents ridge=0 to .04 by .005 outseb;
   outvif outest=ridgests plots(only)=ridge(unpack VIFaxis=log);
   model hs_pct_21 = GINI_pct_change african_american_pct
                     high_school_grad_pct_21 mhi_20 mfi_acs_21;
run;
```

Root MSE	0.02518	**R-Square**	0.4405
Dependent Mean	0.01949	**Adj R-Sq**	0.3783
Coeff Var	129.22766		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	0.56092	0.11549	4.86	<.0001
GINI_pct_change	GINI_pct_change	1	-0.51430	0.26674	-1.93	0.0602
african_american_pct	african_american_pct	1	-0.00106	0.00038824	-2.73	0.0090
high_school_grad_pct_10	high_school_grad_pct_10	1	-0.66107	0.14763	-4.48	<.0001
mhi_09	mhi_09	1	0.00000663	0.00000252	2.63	0.0116
mfi_acs_10	mfi_acs_10	1	-0.00000464	0.00000230	-2.02	0.0493

LASSO Regression

LASSO regression is also a penalized regression method. In LASSO, contributions from predictors are reduced by the *sum of the absolute values of the magnitude* of the coefficients, as opposed to the square of the magnitude used in Ridge Regression. This favors models with just a few correlated predictors. The acronym LASSO stands for Least Absolute Shrinkage and Selection Operator.

In SAS, LASSO regression is implemented using the GLMSELECT Procedure with the LASSO option selected in the MODEL statement. It supports multiple cross validation, a variety of methods for choosing variables for the model with the CHOOSE statement, and Bayesian analysis.

<u>Example</u>: LASSO regression is used to predict the salary of a professional athlete based on several performance characteristics which are correlated to salary to varying degrees. In this example, major League Baseball is used as the sport but any outcome with many multicollinear predictors will do. As the distribution of player salary is highly skewed by a small number of players making very large salaries, the predicted outcome is given here by the natural log of the salary. A hold-out sample of 30% of the records is randomly selected using the Fraction and Validate options.

```
 proc glmselect data=rm.sas_baseball plots=all;
   partition fraction(validate=.3);
   model logSalary = nAtBat nHits nHome nRuns nRBI nBB
                     yrMajor crAtBat crHits crHome crRuns
                     crRbi crBB nOuts nAssts nError
         / selection=lasso(stop=none choose=validate);
run;
```

SAS procedure options that may be helpful include:

<u>MODEL Statement Options</u>

- selection – note: this <u>must be set to lasso</u> to use this method

- stop – sets the criteria for when to stop variable selection, stop=none examines all variables in the MODEL statement

- choose – sets criteria for choosing the model; default is AICC, sbc is Schwarz Bayesian information criterion

<u>PARTITION Statement Options</u>

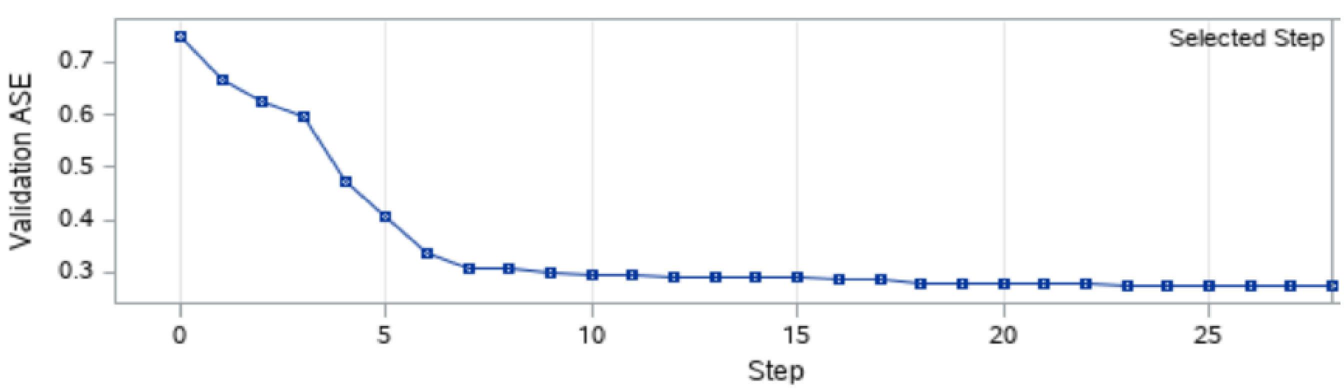
- validate – states the fraction for the validation sample

Output for this example:

Analysis of Variance				
Source	DF	Sum of Squares	Mean Square	F Value
Model	16	84.14448	5.25903	14.28
Error	161	59.30511	0.36835	
Corrected Total	177	143.44959		

Root MSE	0.60692
Dependent Mean	5.92151
R-Square	0.5866
Adj R-Sq	0.5455
AIC	18.36231
AICC	22.66420
SBC	-107.54737
ASE (Train)	0.33317
ASE (Validate)	0.27405

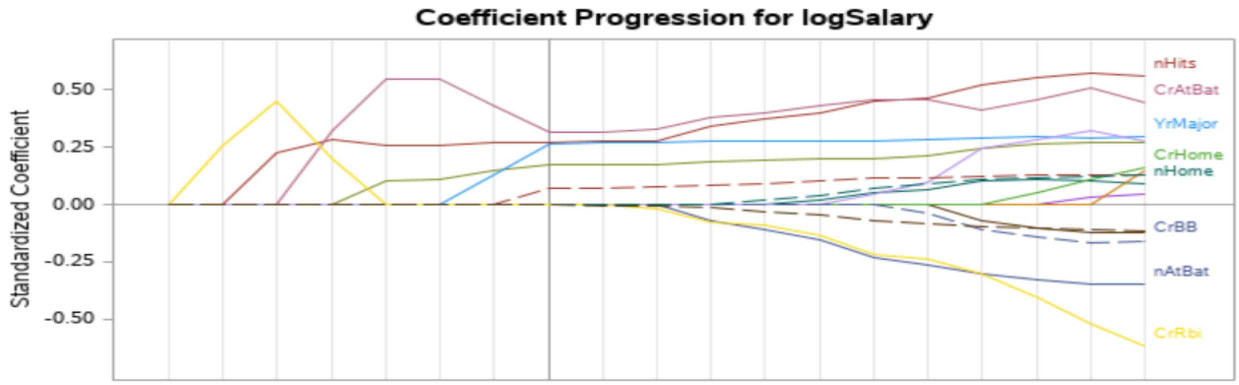
LASSO Selection Summary					
Step	Effect Entered	Effect Removed	Number Effects In	ASE	Validation ASE
0	Intercept		1	0.8059	0.7496
1	CrRuns		2	0.7087	0.6667
2	CrHits		3	0.6648	0.6268
3	CrRbi		4	0.6312	0.5963
4	nHits		5	0.5155	0.4746
5	nBB		6	0.4549	0.4082
6	YrMajor		7	0.3866	0.3343
7	nRBI		8	0.3650	0.3090
8	nRuns		9	0.3632	0.3067
9	nError		10	0.3514	0.2989
10	nOuts		11	0.3472	0.2953



LASSO Regression – Another Example

This example implements an Adaptive LASSO algorithm, which strengthens the ability to select just the strongest predictors. This example also uses Bayesian analysis, which is specified using <u>choose=sbc</u> on the SELECTION statement.

```
proc glmselect data=rm.sas_baseball plots=all;
   partition fraction(validate=.3);
   model logSalary = nAtBat nHits nHome nRuns nRBI nBB
                     yrMajor crAtBat crHits crHome crRuns
                     crRbi crBB nOuts nAssts nError
/ selection=lasso(adaptive stop=none choose=sbc);
run;
```



In the last plot, the relative contributions of different factors is tracked across the development of the output. Substantial contributions are made by just a few strong predictors, indicated this as a situation where LASSO should perform best. Where there is a larger number of strong predictors, Ridge Regression will usually work better. Best practice recommends starting with simple regression. Then, as strong interactions are observed, both Ridge and LASSO are tried and the results compared to identify the best solution.

Local Regression Using LOESS

Local Polynomial Regression are a group of non-parametric regression methods that combine multiple regression runs into a meta-model. While Local Regression was first developed by Savitsky and Golay in 1964, the extreme amount of computation required to execute these algorithms are prevented widespread application until recently. Instead of developing a model fit using all the available data equally, the algorithms are locally estimated. This makes the result at each point more sensitive to the closest data.

The SAS Procedure LOESS (Locally Estimated Scatterplot Smoothing) creates simple regress models from local data and combines them for an overall solution. It supports multiple dependent variables, multidimensional predictors and interpolation using kd trees.

<u>Example</u>: LOESS is used to map a relationship between income levels and the pass/fail rate on a state standard 8th grade reading test for school district in the Detroit metropolitan area. The district-level scores have a natural cut-off at 100%, resulting is a complex shape: at lower median household incomes, performance on the test increases with increasing income. Test scores are bounded but income is not, resulting in a tipping point beyond which additional income came increase educational enrichment but

doesn't result is a significant improvement in pass-fail rate. Local regression allows the identification of this tipping point, which is found at per capita income of $28,000 at the 2000 census ($48,000 in 2022 dollars). This example includes code for implementing the ODS output for the data visualizations.

```
ods graphics on;
proc loess data=rm.sem_eduction;
   ods output OutputStatistics=GasFit FitSummary=Summary;
   model MEAP8_Read = PCI_2000;
run;
ods graphics off;
```

SAS procedure options that may be helpful include:

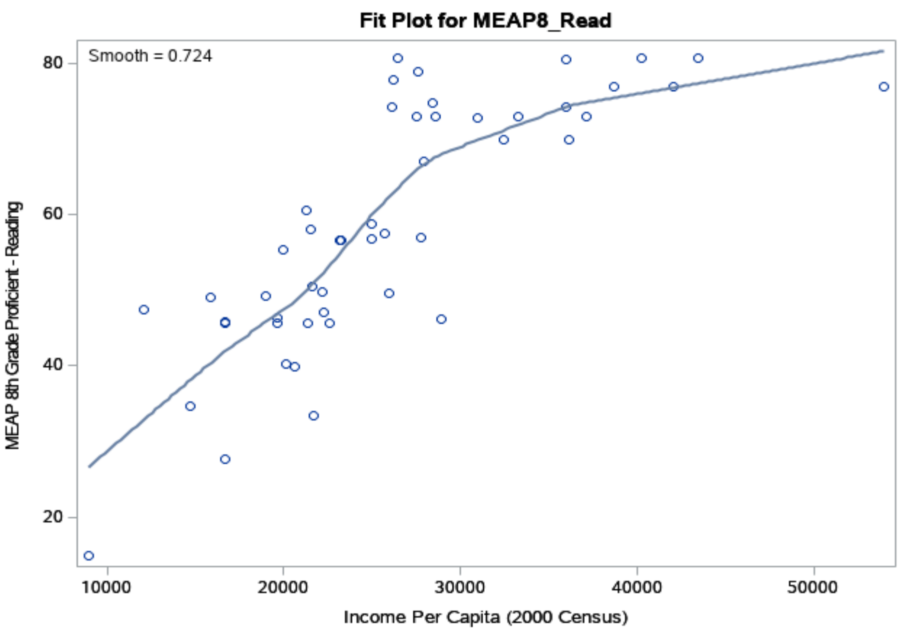
<u>MODEL Statement Options</u>

- degree – degree of the local polynomials (either 1 or 2)
- select= – specifies a smoothing method: AICC, AICC1, GCV, DF1, DF2, or DF3
- direct – requires direct fitting at every point
- std – outputs the standard of the mean predicted values

<u>SCORE Statement Options</u>

- clm – output confidence limits with the score

Output for this example:



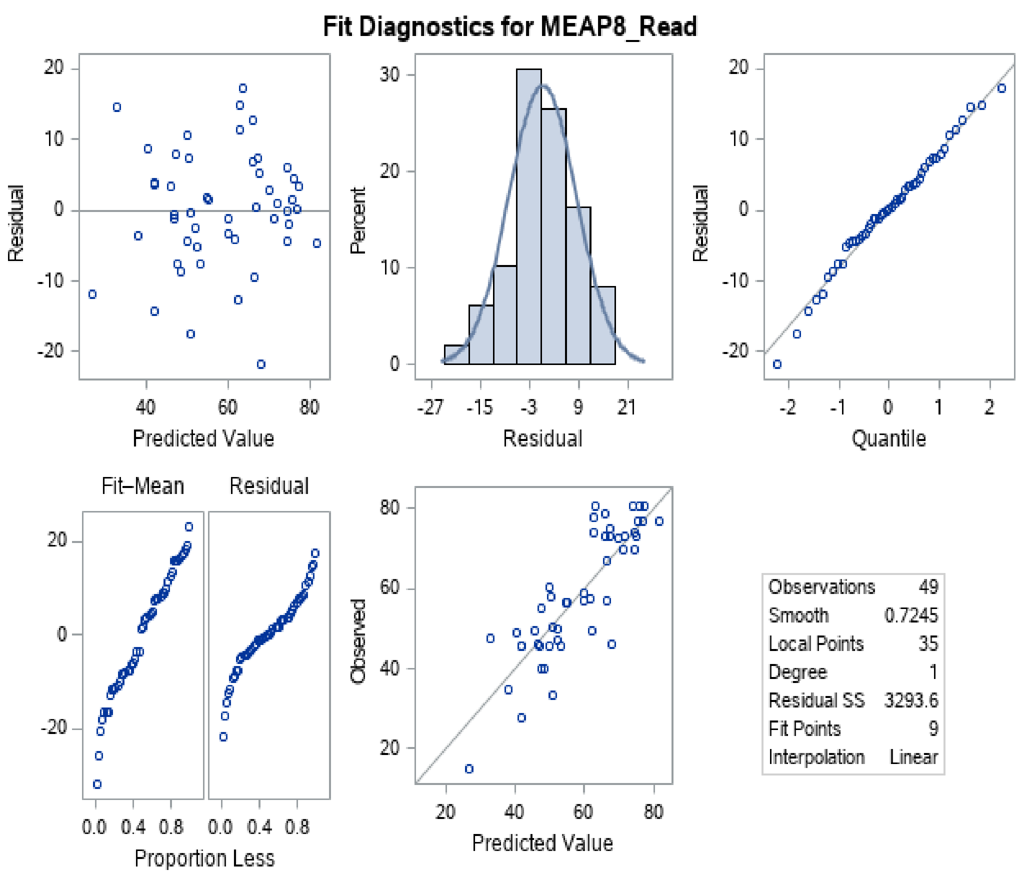
The LOESS Procedure
Selected Smoothing Parameter: 0.724
Dependent Variable: MEAP8_Read

Fit Summary	
Fit Method	kd Tree
Blending	Linear
Number of Observations	49
Number of Fitting Points	9
kd Tree Bucket Size	7
Degree of Local Polynomials	1
Smoothing Parameter	0.72449
Points in Local Neighborhood	35
Residual Sum of Squares	3293.62441
Trace[L]	4.26356
GCV	1.64570
AICC	5.45425

Optimal Smoothing Criterion	
AICC	Smoothing Parameter
5.45425	0.72449

The LOESS Procedure

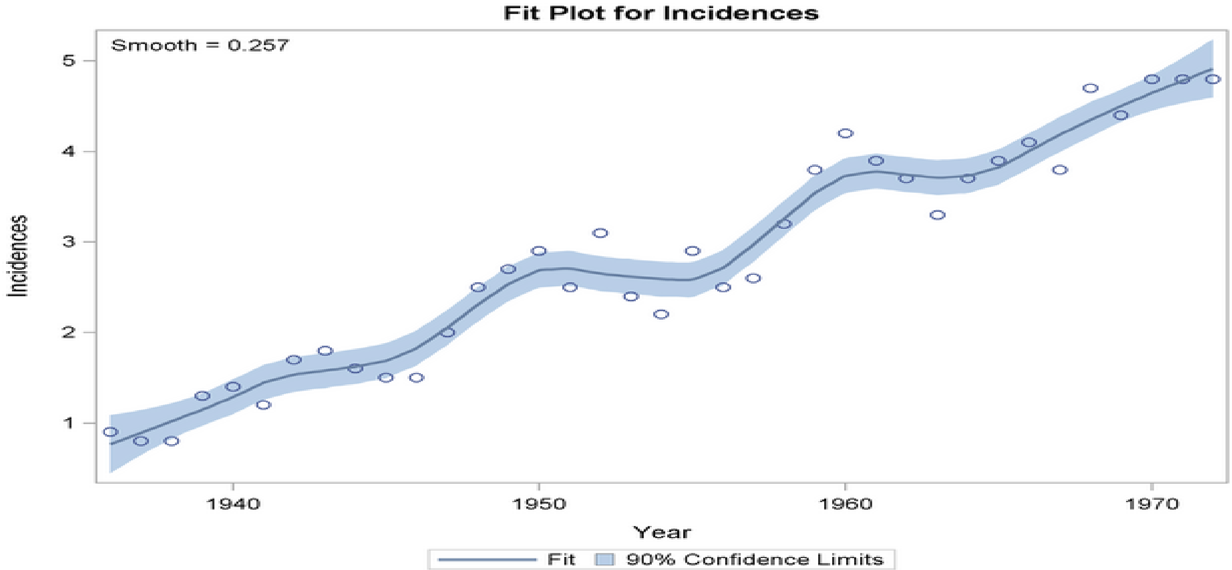
Independent Variable Scaling	
Scaling applied: None	
Statistic	Income Per Capita (2000 Census)
Minimum Value	8965
Maximum Value	53942



LOESS – Another Example

In this example, Local Regression is used with a smoothing factor to model a complex form without overfitting. LOESS was originally developed for smoothing applications such as this.

```
proc loess data=Melanoma;
   model Incidences=Year/clm alpha=0.1
run;
```



Generalized Additive Models (GAM)

In statistics, additive models are a non-parametric method using a one-dimensional smoother. The models are more flexible than standard linear regression and less subject to the Curse of Dimensionality. Generalized Additive Models, first developed by Trevor Hastie and Robert Tibshirani in 1990, combine features of genal linear models and additive models. The result allows better fitting of complex patterns while can be subject to a lack of applicability to other dataset.

In SAS, Generalized Additive Models are implemented using the GAM Procedure. The algorithm allows multiple independent non-parametric predictors, while the univariate smoothing provides finer details than is possible with the piece-wise LOESS procedure. The GAN Procedure supports non-parametric and semi-parametric models, and multidimensional predictor.

<u>Example</u>: forecasting the end-of-season ranking of a sports team. In major league baseball in North America, as with some other sports, a mid-year trade deadline prevents most exchanges of players between teams until after the end of the playing season. This results in a need to predict the ranking of teams of each team to guide strategy before the deadline passes. In this example, the win / loss rate of each team in each of the first three months of the season (April, May, and June) is used to predict the team ranking at the end of the season. These fields are moderately correlated.

```
proc gam data=rm.baseball plots(unpack)=all;
   model term_2017 = spline(RateApril) spline(RateMay)
                     spline(RateJune / method=gcv;
   output out=PredGAM p=Gam_p_;
run;
```

SAS procedure options that may be helpful include:

<u>GAM Statement Options</u>

- descending – reverses the sort order of the class variable
- plot= – plotting options: all, unpack

MODEL Statement Options

- anodev – smoothing options: refit, norefit, none
- maxiter – maximum number of estimation iterations
- method=gcv – smoothing parameter uses the generalized cross validation method

Output for this example:

The GAM Procedure
Dependent Variable: FinalPlace
Smoothing Model Component(s): spline(RateApril) spline(RateMay) spline(RateJune)

Summary of Input Data Set	
Number of Observations	30
Number of Missing Observations	0
Distribution	Gaussian
Link Function	Identity

Iteration Summary and Fit Statistics	
Final Number of Backfitting Iterations	7
Final Backfitting Criterion	4.3096977E-9
The Deviance of the Final Estimate	14.372176478

The backfitting algorithm converged.

Regression Model Analysis
Parameter Estimates

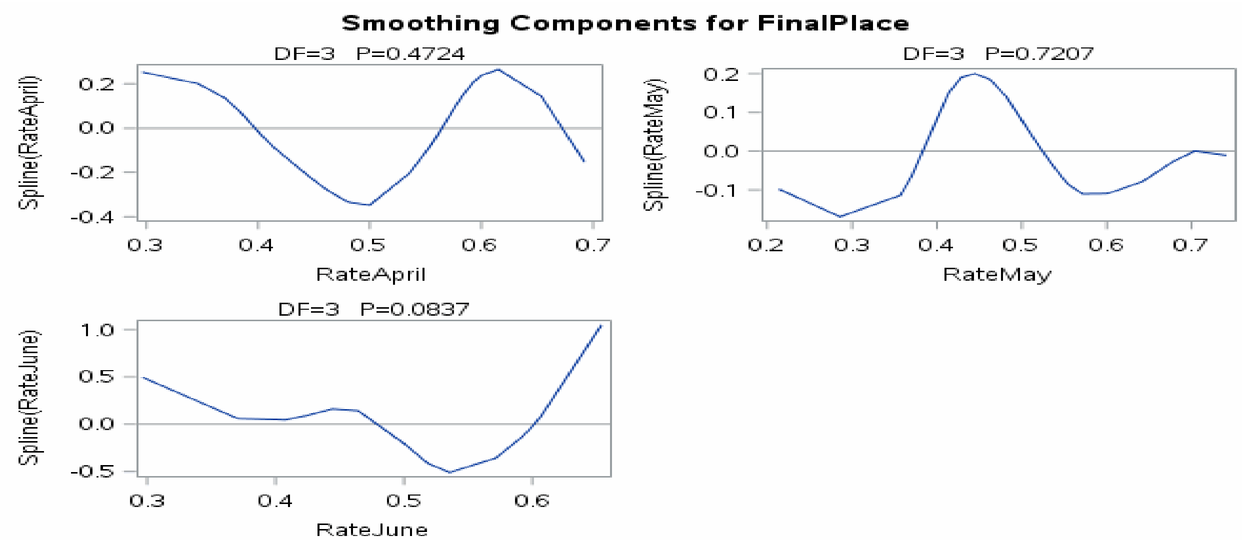
Parameter	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	10.07191	1.33201	7.56	<.0001
Linear(RateApril)	-5.57092	1.66769	-3.34	0.0039
Linear(RateMay)	-4.06923	1.42487	-2.86	0.0109
Linear(RateJune)	-4.63003	2.08142	-2.22	0.0400

Smoothing Model Analysis
Fit Summary for Smoothing Components

Component	Smoothing Parameter	DF	GCV	Num Unique Obs
Spline(RateApril)	0.972962	3.000000	0.791218	23
Spline(RateMay)	0.966501	3.000000	0.775891	22
Spline(RateJune)	0.841046	3.000000	0.586709	14

Smoothing Model Analysis
Approximate Analysis of Deviance

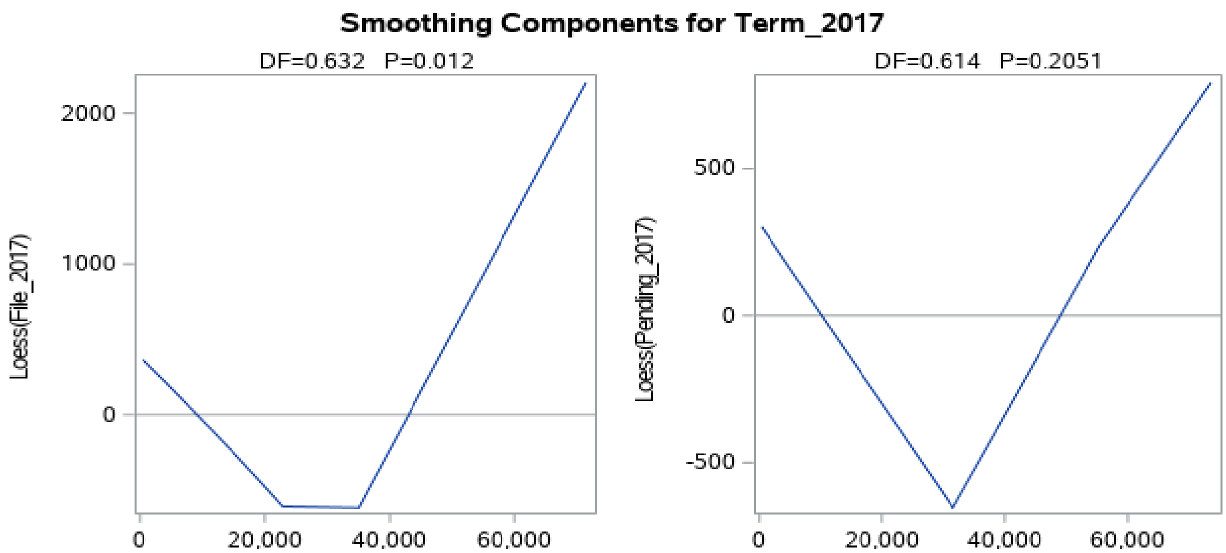
Source	DF	F Value	Pr > F
Spline(RateApril)	3.00000	0.88	0.4724
Spline(RateMay)	3.00000	0.45	0.7207
Spline(RateJune)	3.00000	2.63	0.0837



GAM – Another Example

In this example, an Generalized Additive Model used to fit a complex response surface without loss of detail to due piece-wise fitting in local regression. In this example, processing of cases through a bankruptcy court are considered to support planning required staffing levels. The number of cases at each of three successive stages are used: cases filed, pending, and terminated (completed).

```
ods graphics on;
proc gam data=rm.bankruptcy plots(unpack)=all;
   model term_2017 = loess(file_2017) loess(pending_2017) /
   method=gcv;
   output out=PredGAM p=Gam_p_;
run;
ods graphics off;
```



REFERENCES

Cohen, R. (2009). “Applications of the GLMSELECT Procedure for Megamodel Selection.” In Proceedings of the SAS Global Forum 2009 Conference. Cary, NC: SAS Institute Inc. http://support.sas.com/resources/ papers/proceedings09/259-2009.pdf

Gunes, F. (2015) “Penalized Regression Methods for Linear Models in SAS/STAT” In Proceedings of the SAS Global Forum 2015 Conference. Cary, NC: SAS Institute Inc.

Hastie, T. J. and Tibshirani, R. J. (1990). “Generalized Additive Models”, Chapman & Hall/CRC.

Kennedy, P., (2003), “A Guide to Econometrics” (Fifth ed.). Cambridge: The MIT Press. pp. 205–206. ISBN 0-262-61183-X.

Schreiber-Gregory, D. N., Jackson ,H. M., 2018, “Multicollinearity: What Is It , Why Should We Care , and How Can It Be Controlled ?”, Model Assisted Statistics and Applications, vol. 13, no. 4, pp. 359-365

Savitzky, A.; Golay, M.J.E. (1964). "Smoothing and Differentiation of Data by Simplified Least Squares Procedures". Analytical Chemistry. 36 (8): 1627–39.

Tibshirani, R., (1996). "Regression Shrinkage and Selection via the lasso". Journal of the Royal Statistical Society. Series B (methodological). Wiley. 58 (1): 267–88. JSTOR 2346178.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

David J Corliss
Peace-Work
davidjcorliss@peace-work.org