CASE STUDY: Spline Approximation of Two-Dimension Data (spline_sum, st_pen, meanabs_pen, logexp_sum, logistic)

Background

This case study approximates two-dimension data by a spline using PSG function spline_sum. Input data for building spline:
- $\vec{x}$ = vector of independent values,
- $\vec{y}$ = vectors of dependent values,
- $D$ = degree of the spline,
- $K$ = number of polynomial pieces in the spline,
- $S$ = smoothing degree of the spline.

PSG risk functions, Standard Deviation (st_pen), Mean Absolute Error (meanabs_pen), and Maximum Likelihood for Logistic Regression (logexp_sum) are minimized to find the best approximation.

A matrix with data (so called Matrix of Scenarios) contains vectors $\vec{x}$ and $\vec{y}$ and a matrix with parameters specifies $D$, $K$, $S$.

Notations

- $J$ = number of points (observations) of independent variable, $j = index for points, j = 1, ..., J$;
- $x_j$ = point of independent variable, $j = 1, ..., J$. Points $x_j$ are ordered, i.e., if $j_1 < j_2$ then $x_{j_1} \leq x_{j_2}$;
- $y_j$ = point of dependent variable corresponding to the point $x_j$, $j = 1, ..., J$;
- $\vec{x} = (x_1, x_2, ..., x_J)$ = vector of points $x_j$;
- $\vec{y} = (y_1, y_2, ..., y_J)$ = vector of points $y_j$;
- $D$ = degree of spline, $D \geq 0$, integer;
- $K$ = number of polynomial pieces in the spline, $K > 0$, integer;
- $S$ = smoothing degree of a spline, $0 \leq S \leq D$, integer;
- $I = K \cdot (D + 1) = number of unknown coefficients of polynomial pieces in a spline;
- $a_{dk} = decision variable = coefficient for degreed in polynomial piecek, d = 0, ..., D, k = 1, ..., K$;
- $\vec{a} = (a_{01}, a_{11}, ..., a_{D1}, a_{02}, a_{12}, ..., a_{D2}, ..., a_{0K}, a_{1K}, ..., a_{DK}) = vector of decision variables (coefficients).

Important! These decision variables are not part of input data. They are generated by PSG automatically.

Names of decision variables are based on names of independent factors.

$X = \{X_0, X_1, ..., X_K\}$ = set of points (knots) partitioning segment $[x_1, x_J]$ in sub-segments, $k = 1, ..., K$; every sub-segment $[X_{k-1}, X_k]$ contains at least one point $x_j$ ($X_0 = x_1, X_K = x_J$);

$J_k = sub-set of indexes j = 1, ..., J$ corresponding to sub-segment $[X_{k-1}, X_k]$, $J_k = \{j|x_j \in [X_{k-1}, X_k]\}$;

$L_j(\vec{a}) = y_j - G_j^0(\vec{a}) = y_j - \sum_{d=0}^{D} a_{dk} \cdot x_j^d = Loss Function value at point x_j, j \in J_k, k = 1, ..., K$;

$G_j^0(\vec{a}) = \sum_{d=0}^{D} a_{dk} \cdot x_j^d = Gain Functions with zero scenario benchmark at point x_j, j \in J_k, k = 1, ..., K$;

spline_sum($D, K, S, \vec{x}, \vec{y}, \vec{a}$) = $\{L_1(\vec{a}), L_2(\vec{a}), ..., L_J(\vec{a})\}$ = PSG function. Spline_sum generates a set of loss scenarios $L_j(\vec{a})$ using initial data and smoothing constraint;
\textbf{Optimization Problem 1}  
minimizing Standard Penalty for building spline  

\[
\text{min}_\alpha \text{st_pen}(\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a}))
\]

\textit{calculation of Mean Absolute Penalty and Spline Sum for built spline}

\begin{align*}
\text{calculate} \\
\text{meanabs_pen}(\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a})) \\
\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a})
\end{align*}

\textbf{Optimization Problem 2}  
minimizing Mean Absolute Penalty for building spline  

\[
\text{min}_\alpha \text{meanabs_pen}(\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a}))
\]

\textit{calculation of Standard Penalty and Spline Sum for built spline}

\begin{align*}
\text{calculate} \\
\text{st_pen}(\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a})) \\
\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a})
\end{align*}

\textbf{Optimization Problem 3}  
maximizing Logarithms Exponents Sum for building spline  

\[
\text{max}_\alpha \text{logexp_sum}(\text{spline_sum}(D, K, \tilde{x}, \tilde{y}, \tilde{a}))
\]

\textit{calculation of Logarithms Exponents Sum and Logistic for built spline}
calculate

\[ \logexp_{\text{sum}}(\text{spline}_{\text{sum}}(D, K, S, \tilde{x}, \tilde{y}, \tilde{a})) \]

\[ \logistic(\text{spline}_{\text{sum}}(D, K, S, \tilde{x}, \tilde{y}, \tilde{a})) \]

**Solution Output**

For the considered optimization problems, solvers provide a standard output containing values of objective, constraints, and functions, and a point with solution vector containing generated decision variables \( a_{dk} \). In addition to the standard output, the solver gives \( \text{Spline}_{\text{sum}} \) function report containing two additional matrices: matrix with knots and matrix with quant. The first matrix contains knots \( \{X_1, ..., X_{K-1}\} \), the second one contains numbers of points \( x_j \) in every sub-segment \([X_{k-1}, X_k]\). Names of these two matrices are based on the name of matrix of input data.

**Solved Problem 1**

Here are four graphs presenting data and spline approximation obtained with \( \text{Optimization Problem 1} \). Data include 4,371 points connected by blue lines. The graphs show splines with different number of spline pieces: 30, 10, 5, 2.
**Solved Problem 2**

Here are four graphs presenting data and spline approximation obtained with *Optimization Problem 2*. Data include 4,371 points connected by blue lines. Majority of points lie in the segment [-2, 4]. The graphs show splines with different number of spline pieces: 30, 10, 5, 2.
**Solved Problem 3**

Here are four graphs presenting data and spline approximation obtained with *Optimization Problem 3*. Data include 149 points with values 0 and 1 connected by blue lines. Majority of points lie in the segment \([-2, 2]\). The graphs show spline approximations with different number of spline pieces: 30, 10, 5, 2.

Red curve shows the function of \( x \) calculated by using logistic function \( \frac{e^{S(x)}}{1 + e^{S(x)}} \), where \( S(x) \) is the polynomial piecewise spline. This function (red curve) approximates probability depending on the independent factor \( x \) according to values of dependent variable in the dataset.