Macro **%shtscore** is primarily designed for score building when the dependent variable is binary.

There are several components in **%shtscore**:

1. Variable pre-scanning;
2. Smoothing binning;
3. Information index calculation;
4. Pair-wise correlation examination;
5. **Proc Logistic** application;
6. Multicollinearity examination;
7. Generating code for scoring purpose.

1. Variable Pre-scanning

Given an independent variable list, **%shtscore** excludes variables by several pre-determined exclusion criteria. The exclusion rules are defined separately for categorical and numerical variables.

For categorical variables:

a. A variable is excluded if the total number of different levels for that variable exceeds a user-specified number. It is set by default at 100. To change the default value, for example, to 50, the statement **"%let cat = 50;"** is inserted before "**%shtscore;**".
b. A variable is excluded if all observations assume an identical categorical value.
c. A variable is excluded if at least 98% (by default) of all observations assume an identical categorical value. To change the default value, for example, to 90%, the statement **"%let ppn = 90;"** is inserted before "**%shtscore;**".

For numerical variables:

d. A variable is excluded if the mean of the variable is greater than 9,999,999.
e. A variable is excluded if all observations assume an identical numerical value.
f. A variable is excluded if at least 98% (by default) of all observations assume an identical numerical value. To change the default value, for example, to 90%, the statement **"%let ppn = 90;"** is inserted before "**%shtscore;**".

2. Smoothing Binning

For categorical independent variables,

a. Given a categorical independent variable, for example, X with K levels, a new numerical variable X* is defined, also with K levels, each of which is equal to the average of the binary dependent variable for that level of X.
b. Grouping observations by their corresponding X* values according to the following steps:
    a. Define the effective data range for X*:
        i. Excluding each and every level of X* if the total number of observations corresponding to that level is less than 1% of the total sample size.
        ii. For the remaining levels of X*, the effective data range is the difference between the maximum and the minimum of the remaining levels.
    b. Grouping the levels of X* by proximity with a measure defined by the effective data range: any two observations with X* levels within 2% of "effective data range" are grouped together. The 2% here is by default. To change that, for example, to 3%, the statement **"%let difp = 0.03;"** is inserted before "**%shtscore;**".
    c. If the total number of groups by the previous step exceeds a pre-determined number (default=24), further grouping is deemed necessary. The further grouping is achieved by combining the two groups with the least difference in the average dependent variable value within group. Each further grouping leads to a number of groups one fewer than the previous generation. This process continues until the total number of groups is less or equal to 24 (default). To change the default, for example, to 10, the statement **"%let breaks = 10;"** is inserted before "**%shtscore;**".
    d. If the total number of observations within a group is less that 4% (default) of the sample size, further grouping is deemed necessary. In that case, the smallest group is then combined with one of the two neighboring groups whichever is closer in average value of the dependent variable within group. This process continues until the total number of observations in each and every group is no less than 4%. To change the default, for example, to 2%, the statement **"%let minkeepc = 0.02;"** is inserted before "**%shtscore;**".

For numerical independent variables, the grouping is achieved by two major steps:

a. Find the initial number, G, of groups satisfying the following criteria:
    a. The number of observations in each group must be greater or equal to 500 (default). This condition leads to a maximum possible number of groups, **Amax**. To change the default, for example, to 200, the statement **"%let Ynummin = 200;"** is inserted before "**%shtscore;**".

b. Each group must contain at least 25 (default) observations with dependent variable being "1". This condition leads to a maximum possible number of groups, Bmax. To change the default, for example, to 20, the statement **"%let** Ybadmin **= 20;"** is inserted before "**%shtscore;**".

c. There are two pre-determined maximum and minimum numbers of total groups, Yrangmax(default is 50) and Yrangmin(default is 10). The initial number of groups is determined by the following rule:

1. If min(Amax, Bmax)>= Yrangmin, then the initial number of groups is set to be min(Amax, Bmax, Yrangmax).
2. If min(Amax, Bmax)< Yrangmin, then the initial number of groups is set to be Yrangmin.

To change the default, for example, to 30, the statement **"%let** Yrangmax **= 30;" or "%let** Yrangmin **= 30;"** is inserted before "**%shtscore;**".

b. Initial Grouping. Initial grouping in achieved by:

a. Order the observations according to their X values, then group the observations into G equal-sized (N/G) subgroups. The marks for each subgroup are noted. This is done for each and every independent variable X. (Remark: Note that the ordering of the data can only be partially achieved since X may assume a same value for multiple observations. Grouping by percentage of observations can only find the X value of the bordering observation in a group. This value however may be shared by other observations in other (most likely a neighboring) groups. Therefore the initial grouping is achieved actually by a second step, using the group marks found in the first step, where a completely defined grouping is found. The grouping at the second step will almost surely result in groups of different size and the total group number will almost surely be different from G.)

b. The significant-digit of marks for each subgroup is set to 4 by default. To change that, for example, to 2 the statement **"%let sigdigit = 2;"** is inserted before "**%shtscore;**".

c. The average of the dependent variable within each subgroup is calculated. These averages are considered as the observed values of a new independent variable, X*.

C. Smooth Grouping. Smooth grouping is achieved by:

d. Defining the effective data range for X*:

1. Excluding each and every level of X* if the total number of observations corresponding that level is less than 1% of the total sample size.
2. For the remaining levels of X*, the effective data range is the difference between the maximum and the minimum of the remaining levels.

e. Grouping the levels of X* by proximity with a measure defined by the effective data range: any two observations with X* levels within 2% of "effective data range" are grouped together. The 2% here is by default. To change that, for example, to 3%, the statement **"%let difp = 0.03;"** is inserted before "**%shtscore;**".

f. Finding the zigzag points and smoothing them out by combining the groups where the zigzag points are with neighboring groups.
g. Determining the number of "peaks" and "valleys", and compare it to the pre-determined allowable number (default=1). Further smoothing is necessary if the default value is exceeded. To change the default value (1), for example, to 0, the statement **"%let peaks = 0;"** is inserted before "**%shtscore;**". "peaks=0" means that the model assumes a monotonic relationship between the dependent and each of the independent variables.
h. Transformation of X*. Three options exist for this transformation:
    1. Log odds.
    2. Averages of dependent variable.
    3. Weight of Evidence. (By default.)

    To change the default value (3), for example, to 1, the statement **"%let logodds = 1;"** is inserted before "**%shtscore;**".

    Note A: 3 rules for numerical variables:
    1. Each variable's negative and positive values are separately analyzed: Group 'I OWE YOU' will be analyzed separately from Group 'YOU OWE ME'.
    2. Value 0 is a separate group: Persons that have 0 balances will be analyzed in a separate group from the others.
    3. Value MISSING is a separate group for analysis: A person with an UNKNOWN value for a variable will not be grouped with those who have KNOWN values for that variable.

    Note B: by inserting **"%let linear = 1;"** before "**%shtscore;**", there will be no transformation for numerical variables but linear treatments:
    1. Performance-based missing value replacement: missing individual values are determined from overall values from the group and inserted.
    2. Outlier treatment: top and bottom "topping" by default is set to 2% (to change the default, inserting the statement **"%let p = 0.01;"** before "**%shtscore;**" will do top and bottom 1% "topping".)
    3. Final scorecard will be the sum of the products of the coefficients and variables.

3. Information Index Calculation

Now suppose all groups are found and smoothed. For each variable X, the amount of information it has pertaining to the dependent variable is calculated in the form of the information index.

Suppose that there are k groups for variable X. The information is calculated for each group by:

[(% of 0) – (% of 1)] * log [(% of 0) / (% of 1)]

Where (% of 0) is the percent of 0's (here 0 is the value of the dependent variable) in that group to the total number of 0's in the sample, and (% of 1) is the percent of 1's (here 1 is the value of the dependent variable) in that group to the total number of 1's in the sample.

The total information index value is the sum of all the group pieces given above.

4. Pair-wise Correlation Examination

For every pair of X*, the correlation is calculated and compare to a pre-determined value (default=90%). If it exceeds 90%, then the variable with a higher information index value is kept and the other is excluded from further study. To change the default value (90%), for example, to 80%, the statement **"%let bicorr = 0.8;"** is inserted before "**%shtscore;**".

5. Proc Logistic application

After excluding the set of highly correlated X*'s, the macro sets aside the top 150 independent variables as a basis for step-wise logistic regression. By specifying "`%let max=30;`", this macro will select up to 30 independent variables among the 150 available.

Several parameters are controlled by users:

a. Yentry – the maximum threshold level of significance to allow an independent variable to be included in the logistic model. To change the default value (0.05), for example, to 0.01, the statement **"%let yentry = 0.01;"** is inserted before "**%shtscore;**".
b. Ystay – the maximum level of significance for an independent variable to stay in the model (after initial entry) in the subsequent step-wise model analysis. To change the default value (0.05), for example, to 0.01, the statement **"%let ystay = 0.01;"** is inserted before "**%shtscore;**".

6. Multicollinearity Examination – A Consistency Check for Each Variable

The macro checks the sign of the estimated coefficient of each and every independent variable in the multiple-regression model from the previous step against that in the simple logistic model with one independent variable. An agreement of the two signs (for each variable) keeps the variable in the model. Otherwise the variable is excluded.

By default, every time some variables are excluded by this criterion, the stepwise regression is repeated with a reduced list of total variables available. This process continues until all study variables in the model satisfy the consistency criterion.

Users can choose to skip this step by changing default value (1) to 0 using the statement **"%let nocollin = 0;"** before "**%shtscore;**".

7. Generating code for scoring purpose

The macro outputs SAS code to generate score under several models. With specified minimum variables (min=13, default) and maximum variables (max=30, default), several scoring models will be given for users to pick and choose. The parameter "max=" may change its value according to the result of the built-in PROC LOGISTIC variable selection mechanism. In any case, there will be several best models to choose from by the users. By default, the scorecard file is named as "x#" where the "x" is the default prefix and the "#" is the number of variables in the scorecard. To change the prefix to, for example "score", the statement **"%let x = score;"** is inserted before "**%shtscore;**".