

Chapter 602

Hybrid Appraisal Models

Introduction

This procedure is used to estimate the parameters of a hybrid appraisal model, based on the data of a number of properties. This model, with estimated parameters, can then be used to estimate the market value of a single property or a number of properties based on the attributes of each property. Because hybrid models are a combination of additive models and multiplicative models, multiple regression analysis techniques cannot be used to analyze these models. Instead, nonlinear regression methods with differential evolution techniques are used to determine the parameter estimates.

The Hybrid Appraisal Model

The hybrid model is a combination of both additive and multiplicative models. It relates the sale price of properties to various characteristics such as size (in square feet), lot size, construction quality, location, number of bathrooms, etc. Before going straight to the hybrid model, we will begin with the simplest appraisal models and then examine models with increasing complexity until we arrive at the hybrid model.

Model Form, Model Estimation, and Value Estimation from the Model

To make the discussion easier, we should first distinguish between the form of the model, the estimation of the model, and the final model for value estimation.

The model form describes the relationship between sale price and the attributes of the property without giving the values of the model parameters. Two examples of models where only the form is shown are

$$\text{Sale Price} = b_0 + b_1 \text{SQFT}$$

$$\text{Sale Price} = \text{QUAL}^{b_1} \times b_2^{\text{IRONMONT}} \times (b_3 \text{SQFT} + b_4 \text{Age})$$

Once the form of the model has been determined, the parameters of the model are estimated. To estimate the parameters, a (hopefully large) number of properties with known sale price are used. The software uses algorithms to find values of the parameters of the model such that the model does a good job at returning the sale price of the properties. The software goes through several iterations to find the parameter estimates that return the sale prices that are closest, on average, to the original sale prices. Two examples of models where the estimation phase has taken place are

$$\text{Sale Price} = \$51,284 + \$85.63 \times \text{SQFT}$$

$$\text{Sale Price} = \text{QUAL}^{1.071} \times 1.348^{\text{IRONMONT}} \times (\$93.41 \times \text{SQFT} - \$1,453.55 \times \text{Age})$$

Hybrid Appraisal Models

Once the model parameter estimates have been determined, the model becomes useful for estimating the market value of properties where the attributes are known, but a sale price is not available. The value estimation model differs only on the left side of the equation:

$$\text{Estimated Value} = \$51,284 + \$85.63 \times SQFT$$

$$\text{Estimated Value} = QUAL^{1.071} \times 1.348^{IRONMONT} \times (\$93.41 \times SQFT - \$1,453.55 \times Age)$$

Basic Appraisal Models

The following are some very basic appraisal models. They will be used as building blocks for more complex models.

One-Term Coefficient Model

The one-term coefficient model multiplies the value of the attribute by the coefficient to obtain the estimate of the sale price:

$$\text{Sale Price} = b \times \text{Attribute Value}$$

An example of an estimated model might look like

$$\text{Sale Price} = \$108.12 \times SQFT$$

More commonly, the model also has a baseline (intercept) value added:

$$\text{Sale Price} = b_0 + b_1 \times \text{Attribute Value}$$

An example of an estimated model with a baseline (intercept) value is

$$\text{Sale Price} = \$29,356 + \$94.29 \times SQFT$$

This model is typically analyzed with simple linear regression, a very common statistical tool for estimating model coefficients.

Basic Binary (Coefficient) Model

The binary model adds (or subtracts) a specified amount based on whether or not the attribute is included.

$$\text{Sale Price} = b_0 + b_1 \times \text{Attribute Value (Yes or No)}$$

An example of an estimated model is

$$\text{Sale Price} = \$176,845 + \$14,587 \times POOL$$

The value for *POOL* is '0' if the property does not have a pool, and the value for *POOL* is '1' if the property does have a pool. This model is also typically analyzed using simple linear regression. In the NCSS Hybrid Appraisal Models procedure, basic binary coefficient models can only be used by creating columns of 0's and 1's and treating them as a regular coefficient model term.

Basic Exponent Model

The basic exponent model estimates the sale price by raising the attribute value to an exponent.

$$\text{Sale Price} = \text{Attribute Value}^b$$

This type of model results in a curved relationship between Sale Price and the values of the attribute. An example of an estimated model is

$$\text{Sale Price} = \text{SQFT}^{1.583}$$

This model can still be solved using simple linear regression, but the logarithm of both sides of the equation must be used in order to do so.

Basic Binary (Exponent) Model

In the binary exponent model, the value (b_0) is multiplied by 1 if the attribute is a 'No' ($b_1^0 = 1$), or is multiplied by the coefficient (b_1) if the value is a 'yes' ($b_1^1 = b_1$).

$$\text{Sale Price} = b_0 \times b_1^{\text{Attribute Value (Yes or No)}}$$

An example of an estimated model is

$$\text{Sale Price} = \$194,821 \times 0.974^{\text{SKYGLADE}}$$

In this estimated model, properties in the *SKYGLADE* neighborhood are valued down by 2.6%. Properties not in the *SKYGLADE* neighborhood are valued at \$194,821.

Intermediate Appraisal Models

The additive and multiplicative models involve combining multiple terms of one (or two) of the basic appraisal model types.

Additive Models

Additive models are made by adding together a series of coefficient and/or binary (coefficient) terms. The model has the form:

$$\text{Sale Price} = b_0 + b_1 \times \text{Attr}_1 + b_2 \times \text{Attr}_2 + \dots + b_p \times \text{Attr}_p$$

An example of an estimated model might look like

$$\begin{aligned} \text{Sale Price} = & \$68,224 + \$77.51 \times \text{SQFT} + \$1.51 \times \text{LOTSIZE} - \$838.26 \times \text{AGE} + \$14,342 \times \text{HERRICK} \\ & - \$9,346 \times \text{SKYGLADE} + \$12,846 \times \text{POOL} \end{aligned}$$

In this example, *SQFT*, *LOTSIZE*, and *AGE* are continuous value terms, whereas *HERRICK*, *SKYGLADE*, and *POOL* are binary terms where the only possible values are 0 (No) and 1 (Yes). Additive models are usually estimated using multiple regression analysis tools.

Multiplicative Models

Multiplicative models are made by multiplying together a series of exponent and/or binary (exponent) terms. The model has the form:

$$\text{Sale Price} = b_0 \times \text{Attr}_1^{b_1} \times \text{Attr}_2^{b_2} \times \dots \times \text{Attr}_q^{b_q} \times b_{q+1}^{\text{Attr}(Y \text{ or } N)_{q+1}} \times b_{q+2}^{\text{Attr}(Y \text{ or } N)_{q+2}} \times \dots \\ \times b_p^{\text{Attr}(Y \text{ or } N)_p}$$

An example of an estimated model might look like

$$\text{Sale Price} = 18.93 \times \text{SQFT}^{0.954} \times \text{QUAL}^{1.114} \times \text{LOTSIZE}^{0.035} \times 1.068^{\text{HERRICK}} \times 0.974^{\text{SKYGLADE}}$$

Multiplicative Models can sometimes be estimated from property data using multiple regression techniques, but this is done by taking the logarithm of both sides of the equation before estimation. Taking the logarithm essentially converts the model into an additive model. After estimation, a transformation is used to convert the model back into its multiplicative form.

Component Hybrid Appraisal Models

Hybrid appraisal models are formed by combining additive and multiplicative models. In **NCSS**, the basic form of a hybrid model is

$$\text{Multiplicative Model} \times \text{Additive Model}$$

or

$$\text{Attr}_1^{b_1} \times \text{Attr}_2^{b_2} \times \dots \times b_q^{\text{Attr}(Y \text{ or } N)_q} \times b_{q+1}^{\text{Attr}(Y \text{ or } N)_{q+1}} \times \dots \times (b_p \times \text{Attr}_p + b_{p+1} \times \text{Attr}_{p+1} + \dots)$$

This type of model may be used to estimate the value of any component of the property. For example, the form of a hybrid model for the land value might look like

$$\text{Land Value} = \text{SoilQuality}^{b_1} \times b_2^{\text{Fenced}} \times (b_3 \times \text{LandSQFT} + b_4 \times \text{SLOPE})$$

Estimation of this model might yield the estimated model

$$\text{Land Value} = \text{SoilQuality}^{1.041} \times 1.067^{\text{Fenced}} \times (4.56 \times \text{LandSQFT} - 0.89 \times \text{SLOPE})$$

Similar models may be formed for the building structure, any other property additions, as well as for the overall property.

Full Property Hybrid Appraisal Models

In NCSS, a hybrid appraisal model combining all the (hybrid model) components of the property has the general form

$$\text{Sale Price} = \text{Hybrid Model}_{\text{Overall}} (\text{Hybrid Model}_{\text{Land}} + \text{Hybrid Model}_{\text{Building}} + \text{Hybrid Model}_{\text{Additions}} + \dots)$$

Each of the *Overall*, *Building*, *Land*, and *Additions* components of the model have the basic hybrid model form shown above in the Component Hybrid Appraisal Models section.

This general model gives the flexibility to model the building value, the land value, and other additions, each with hybrid models, and then adjust the sum of these components through a hybrid model of overall factors.

It is unlikely that every component (building, land, etc.) will use all types of model terms in construction of the model, but the flexibility is available according to the needs of the appraiser.

An example of the form of a full hybrid appraisal model might be

$$\begin{aligned} \text{Sale Price} = & b_1^{\text{HERRICK}} \times b_2^{\text{SKYGLADE}} \times && [\text{Overall}] \\ & (\text{SoilQuality}^{b_7} \times (b_8 \times \text{LandSQFT} + b_9 \times \text{SLOPE}) + && [\text{Land}] \\ & \text{QUAL}^{b_3} \times (b_4 \times \text{SQFT} + b_5 \times \text{KitchenUpgrades} + b_6 \times \text{Age}) + && [\text{Building}] \\ & \text{ShedQuality}^{b_{10}} \times (b_{11} \times \text{ShedSQFT} - b_{12} \times \text{ShedAge})) && [\text{Addition}] \end{aligned}$$

The estimated model might look like

$$\begin{aligned} \text{Sale Price} = & 1.053^{\text{HERRICK}} \times 0.98^{\text{SKYGLADE}} \times \\ & (\text{SoilQuality}^{1.013} \times (3.88 \times \text{LandSQFT} - 0.71 \times \text{SLOPE}) + \\ & \text{QUAL}^{1.033} \times (\$93.41 \times \text{SQFT} + \$2,318 \times \text{KitchenUpgrades} - \$1,453.55 \times \text{Age}) + \\ & \text{ShedQuality}^{1.026} \times (\$31.18 \times \text{ShedSQFT} - \$146.41 \times \text{ShedAge})) \end{aligned}$$

Hybrid Appraisal Model Estimation Details

The coefficients (b_1, b_2, \dots) of a hybrid appraisal model are estimated from a (hopefully large) number of properties where the attribute values are known, and the sale price is known. Whereas the coefficients in additive models (and some multiplicative models) may be estimated using multiple regression analysis (a closed form solution), the coefficients in hybrid models cannot. Instead, the coefficients must be estimated by nonlinear methods and (intelligent) trial and error.

General Algorithm

The general steps for finding the 'best' estimates of the coefficients of the hybrid model are

1. Obtain a set of starting values, one for each coefficient (b_1, b_2, \dots).
2. Compare the predicted sale prices from this model with the actual sale prices, using a summary statistic, such as the average absolute difference or average absolute percent error.
3. Adjust the values of each of the coefficients in a direction that (hopefully) improves prediction.
4. Again, compare the predicted sale prices to the actual sale prices.
5. Continue to repeat steps 3 and 4 until there is no more (or very little) change in the evaluation criterion (average absolute difference or similar). This is called convergence.

The method used in **NCSS** for adjusting the coefficients for each iteration is differential evolution. It differs slightly from the general algorithm described above, but the notion is similar. In the differential evolution algorithm, instead of evaluating a single model each time, a pool of models is examined at each iteration. This pool of models is allowed to evolve until it converges on a single model.

Differential Evolution

Differential evolution is one of a group of *genetic algorithms* (see for example, the recent book by Haupt (1998)). By studying how generations respond over time to their environment, mathematicians have discovered new, more robust, sets of algorithms for minimizing an objective function, e.g., average absolute difference or average absolute percent error. Differential evolution uses the concept of inheritance of coefficient values for a pool of models from a pool of models of the previous iteration. The differential evolution technique performs a similar function to that of the formerly popular feedback algorithm.

To begin the differential evolution algorithm, a small group of 'individuals' (or estimated models) must be formed. This is done by assigning the nonlinear regression coefficients to one individual and then randomly assigning the other individuals (model estimates) to a grid of values around this first individual. This is the initial population.

The next step is the evolution of the population. The population progresses through a series of *generations*. At each change in generation, depending on a member's well-being (low average absolute percent error), each population member may move on to the next generation or be replaced by a better member. For each member, a trial replacement is constructed as follows:

1. The best member of the population is found (smallest absolute difference).
2. The attributes of each replacement member are computed as a weighted average of those of the member and the best member. The amount of weight of the best member is controlled by the *inheritance factor*. This is a value between 0 and 1. The closer this value is to 1, the more the replacement member resembles the best member. The closer this value is to 0, the more the replacement member resembles their parent. The value of 0.85 seems to work in many cases.
3. As in living populations, *mutations* are permitted to occur at a given rate. When a mutation occurs, a particular trait is changed randomly. This tends to maintain diversity in the population. A mutation rate of about 30% (0.30) seems to work well.
4. The algorithm proceeds from generation to generation until the population seems to converge to a single individual. The number of generations needed is specific to each case. Usually, about 100 generations are needed for the algorithm to converge.

Minimization of Error

There are various criteria upon which the estimated models may be evaluated. Each criterion should focus on minimizing the distance between the predicted sale prices and the actual sale prices. In the NCSS Hybrid Appraisal Models procedure, there are eight options for minimization criteria.

- **Minimize Squared Errors (Nonlinear Regression)**

This is the classical approach and gives reasonable estimates relatively quickly. This method tends to emphasize relatively expensive properties as compared to less expensive properties. It also is used as a starting value for the other minimization techniques.

- **Minimize the Average |Percent Error|**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize the average of the absolute percent errors. These percent errors are the difference between the actual and predicted sale prices divided by the actual price. This method gives equal weight to all properties, regardless of price.

The function minimized is

$$MAPE = \frac{\sum_{properties} \left| \frac{100(actual - predicted)}{actual} \right|}{N}$$

where $|X|$ represents the absolute (positive) value of X and $\sum_{properties} X$ is interpreted as the sum of the values of all properties. Note that this is the quantity minimized by typical *feedback* algorithms.

If you want to use a solution that minimizes the percent errors, this is the solution that we recommend.

- **Minimize the Maximum |Percent Error|**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize the maximum of the absolute percent errors. These percent errors are the difference between the actual and predicted sale prices divided by the actual price.

- **Minimize the Median |Percent Error|**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize the median of the absolute percent errors. These percent errors are the difference between the actual and predicted sales price divided by the actual price.

- **Minimize the Percentile |Percent Error|**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize a designated percentile of the absolute percent errors. These percent errors are the difference between the actual and predicted sale prices divided by the actual price. The percentile is specified in the Min Percentile box.

- **Minimize the Average |Error|**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize the average of the absolute errors. These absolute errors are the absolute values of the differences between the actual and predicted sale prices.

Hybrid Appraisal Models

- **Minimize the Median | Error |**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize the median of the absolute errors. These errors are the absolute values of the differences between the actual and predicted sale prices.

- **Minimize the Percentile | Error |**

Using the genetic search algorithm called *differential evolution*, this method finds estimates that minimize a designated percentile of the absolute errors. These errors are the absolute values of the differences between the actual and predicted sale prices. The percentile is specified in the Min Percentile box.

Data Structure

Each column of the spreadsheet (dataset) represents a property attribute, and each row represents a property. A sale price column is required. At least one (but likely more) attribute column(s) is needed to run the Hybrid Appraisal Models procedure. A column may contain a continuous range of values, such as square feet or number of bathrooms, or a set of discrete values, such as neighborhood or style.

The following dataset of residential property sales gives an example of what a hybrid appraisal model dataset may look like.

Recent Sales Dataset (Subset)

Sale_Price	Main_SF	Walls_Type	Baths	BS_SF_Fin	Age	Pool	Garage	Lot_SF	Lake_Front	NBHD
147900	2612	Brick	2.5	0	23	0	2	14778	Yes	Park Grove
184000	2478	Siding	2.5	0	26	0	2	8465	Yes	Park Grove
225000	2617	Wood	2.5	0	26	0	2	8277	Yes	Park Grove
108561	2354	Siding	2.5	0	26	0	2	8277	Yes	Park Grove
165500	2603	Siding	2.5	0	24	0	2	8277	Yes	Park Grove
191000	2549	Brick	2.5	510	22	0	2	10280	Yes	Park Grove
.
.
.
261000	2177	Siding	2.5	0	8	0	3	8170	Yes	Wood Village
260000	2337	Siding	2.5	0	8	0	3	8312	Yes	Wood Village
268900	2413	Brick	2.5	0	8	0	3	14238	Yes	Wood Village
281000	2015	Brick	2.5	0	8	0	2	9800	Yes	Wood Village
300000	2453	Brick	3.5	936	9	0	3	9361	Yes	Wood Village
225000	2536	Siding	2.5	0	9	0	2	9367	Yes	Wood Village

Recent Sales Dataset Column Definitions

Sale_Price	Purchase price	Pool	0 = No pool, 1 = Pool
Main_SF	Non-basement square feet	Garage	Number of attached garage spaces
Walls_Type	Material of exterior walls	Lot_SF	Size of lot in square feet
Baths	Number of (finished) bathrooms	Lake_Front	Lake front property
BS_SF_Fin	Finished basement square feet	NBHD	Name of subdivision
Age	Age in years of the residence		

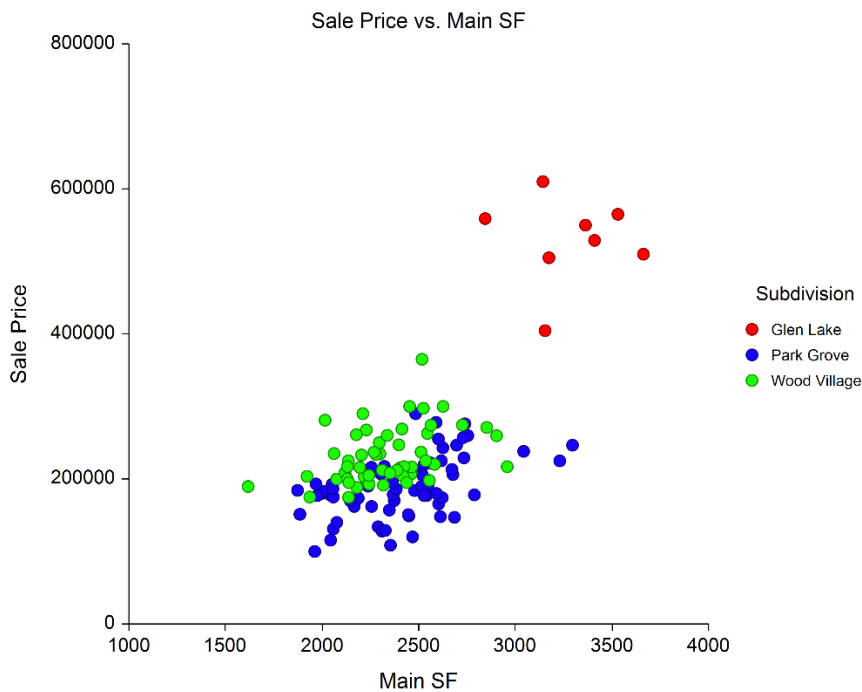
Model Building and Model Estimation Practical Notes

Because hybrid models don't have a closed form solution, iterative methods must be used to determine the estimated coefficients of the models. While these methods allow for increased flexibility in the types of models that may be considered, convergence on a 'best' model estimate is not guaranteed.

Model Building Notes

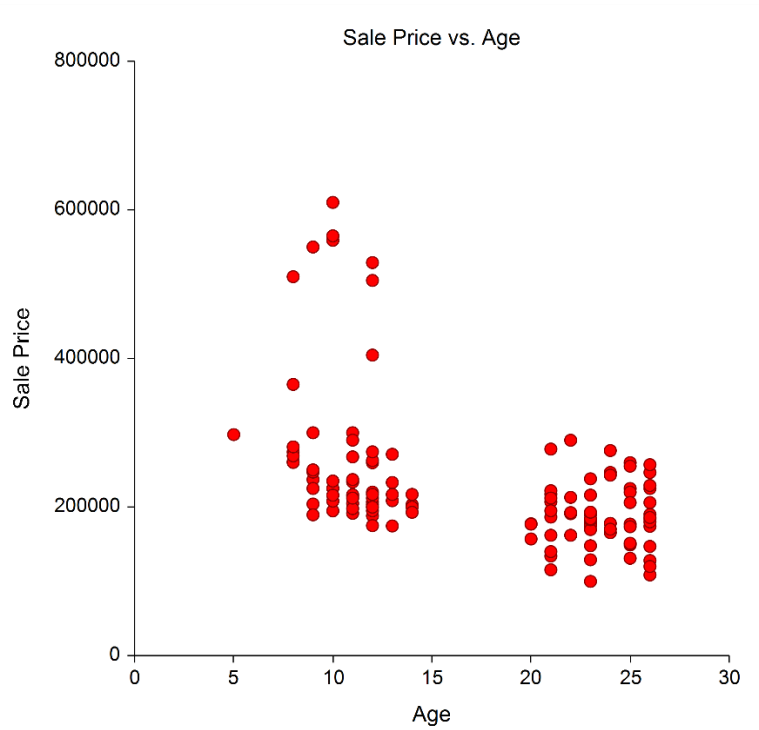
Plots of Sale Price versus Model Terms

Plots can aid in model building in two ways. First, plotting the sale price versus each term allows the appraiser to see any obvious problems in the relationship (e.g., extreme outliers or unexpected curves or direction). Second, the plots help the appraiser determine what to expect for each of the model coefficients (linear or curved, positive or negative relationship). However, since the plots can only be used to examine one or two terms at a time, they do not tell the whole story about how all the terms interact in their relationship with the sale price.

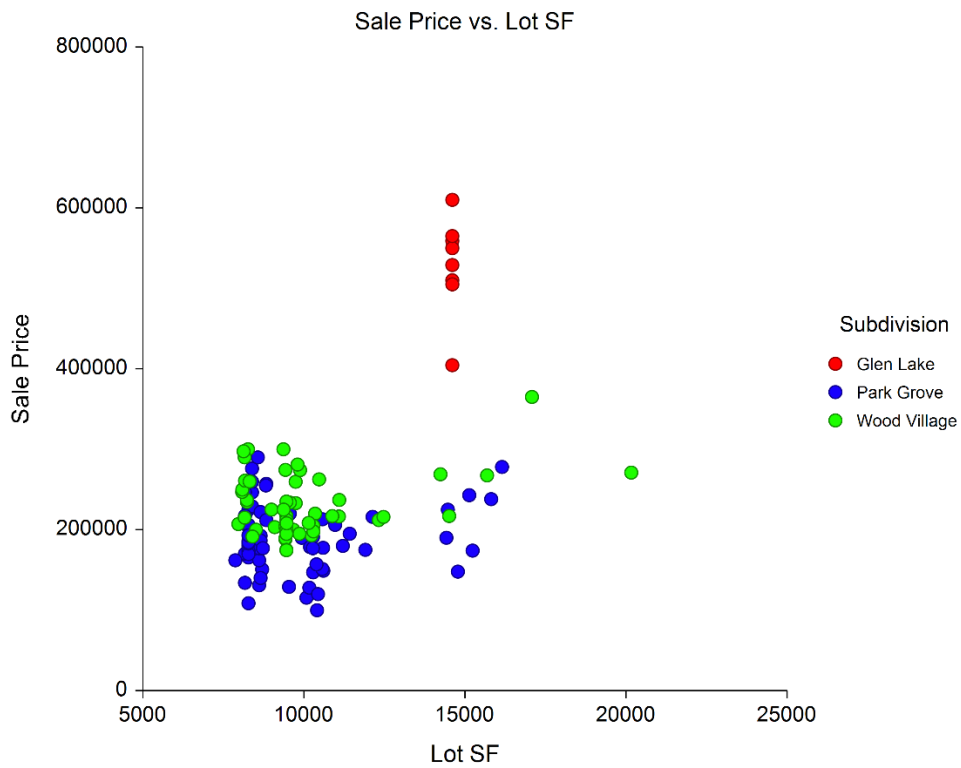


This scatter plot from data of the Recent Sales dataset shows the relationship between sale price and non-basement square feet for each of the three subdivisions. This plot quickly shows what to expect from the model for these terms.

Hybrid Appraisal Models



A plot of sale price versus age shows the relationship of lower values for older properties.



Perhaps surprisingly, the size of the lot seems to have very little effect on price, after accounting for subdivision.

Minimum, Maximum, and Starting Values

The plots (and common sense) can give a feel for the general region that is expected for the coefficients. The minimum, maximum, and starting values should be set accordingly. For example, the default minimum and maximum values for the coefficient for the age of the property might be 0.001 (minimum) to a large number (maximum). However, the plot, as well as experience, would tell us that the coefficient for age might be negative (properties sometimes decrease in value as they age). In this case, we should change the minimum allowed coefficient to be a significantly negative number, to include the option for the algorithm to find a negative relationship for age. The starting value might be adjusted accordingly as well.

Nonlinear Regression and Differential Evolution Option Notes

When confronted with the series of nonlinear regression and differential evolution options in the procedure, the task of setting proper values may seem daunting. Ideally, the default set of options would always yield convergence and a 'best' estimated model. Unfortunately, in practice, convergence is sometimes not achieved with the default options.

If the model estimation does not converge, the output should be examined to see if convergence failed during the nonlinear regression (initial) estimation, or during the differential evolution process. If the failure occurs during nonlinear regression estimation, adjustments to the nonlinear regression options should be the focus. Each option should be considered, perhaps one by one, with an adjustment to the value and a re-run of the procedure. The option notes will hopefully give some direction in how the option affects the process.

Similarly, if convergence fails during the differential evolution process, the corresponding options should be adjusted and the analysis re-run until convergence is achieved.

We have found that in some cases, the nature of the data does not give a stable solution, even though the algorithms converge. For this reason, we recommend that the analysis be run more than once, with the same settings, even when the run seems to complete normally. In the cases where repeated runs give different results (perhaps with substantially varying coefficient estimates), there may be problems in the dataset itself causing the issue. One example would be an extreme outlier, or a group of outliers. Another example would be too few properties of a given type, or too few properties in general. Visualization of the data through scatter plots is recommended in this case, to determine problematic properties. Likely those rows will need to be removed.

You can feel more comfortable with the stability of the results if multiple runs with the same settings give the same results, or multiple runs with varying settings give the same results. Changing the estimation method (minimization criteria) will likely give different coefficient estimates, but it is hoped that the differences aren't too extreme.

Missing Values

Rows with missing values for any of the columns in the analysis are ignored. That is, the whole row is removed from the analysis when there is a missing value for any used column in that row.

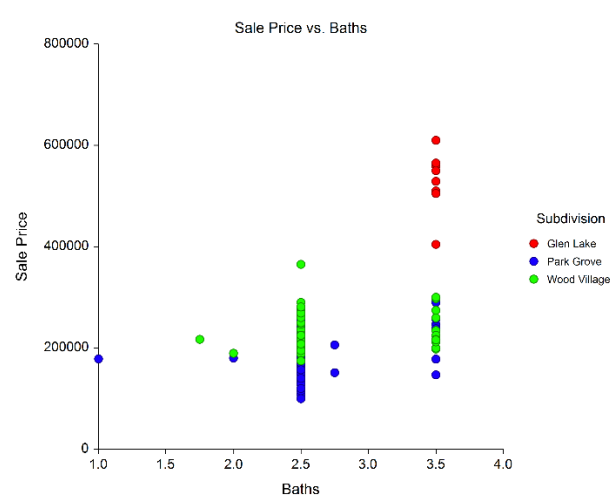
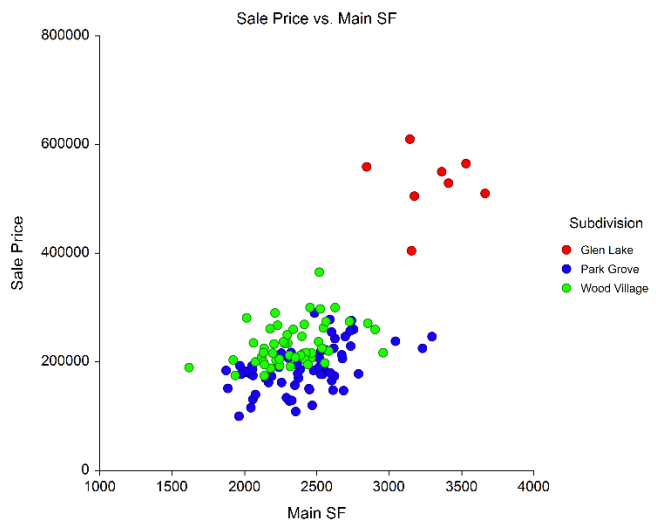
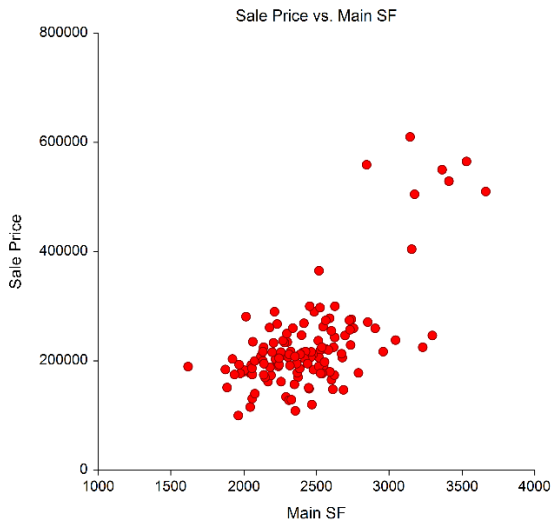
When the value of the sale price is missing (i.e., it is left blank), but values for all other used columns are non-missing, the estimated sale price for that row is generated (see Estimated Values for Estimation Rows report).

Example 1 – Hybrid Appraisal Model

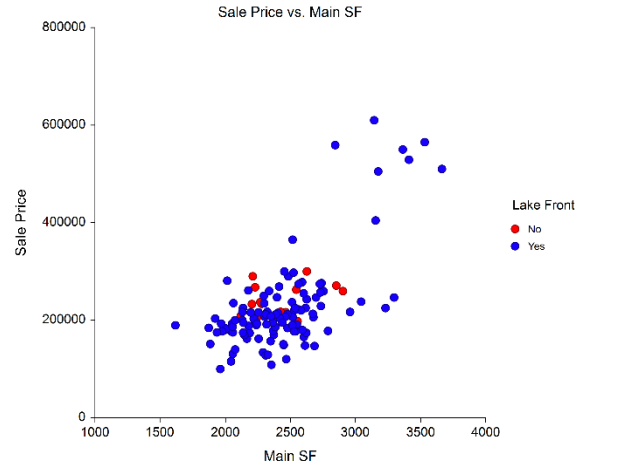
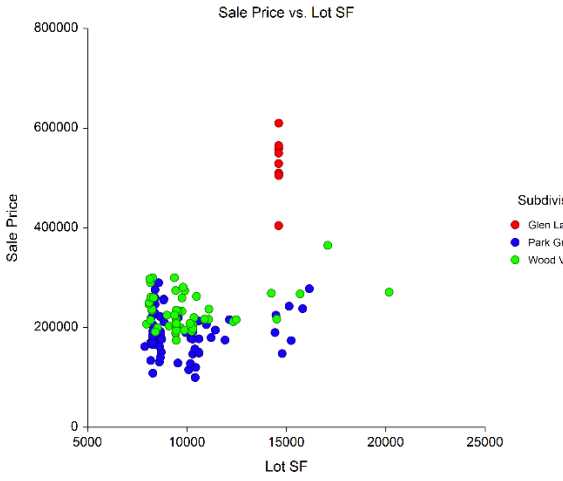
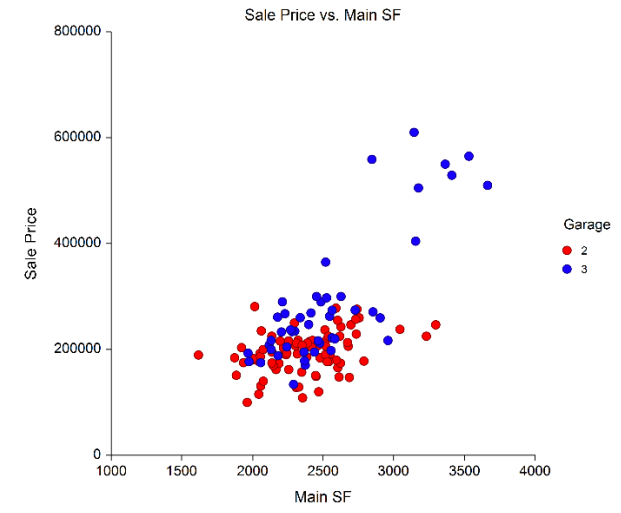
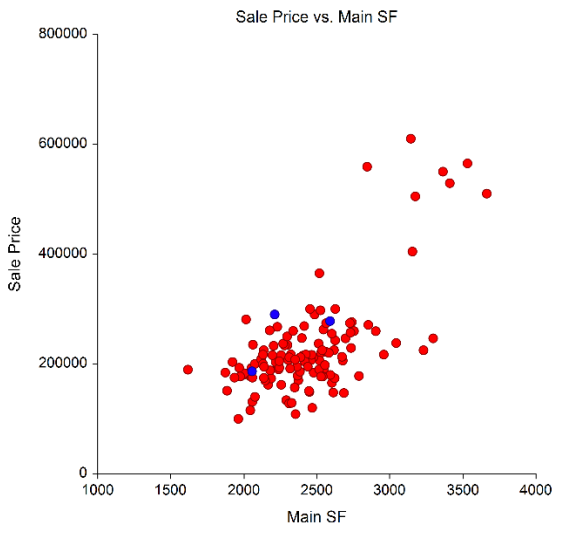
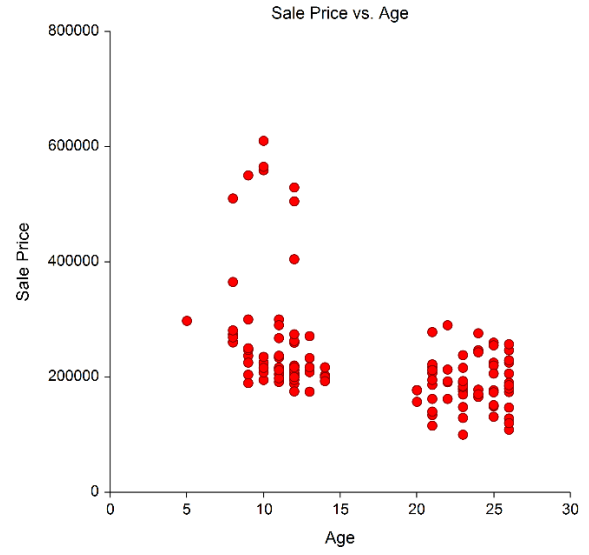
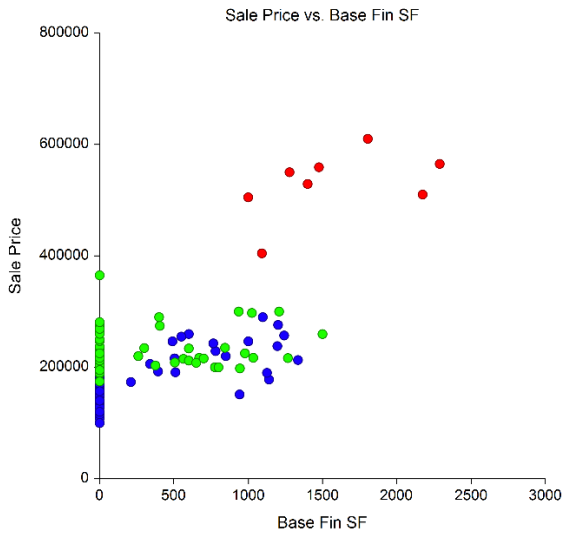
This section presents an example of estimating the parameters of a hybrid appraisal model based on the Recent Sales dataset. The Recent Sales dataset contains the sale price and attribute information about 125 properties. The property values of 3 properties without sale price information are to be estimated. The attribute values for these 3 properties are given in the last three rows (126, 127, and 128) of the dataset.

Pre-Modeling Analysis: Relationship Visualization

Before going straight to the full hybrid appraisal model, the graphical relationship of the sale price with each of the attributes of the dataset is examined. This is done using the Scatterplots procedure (Graphics > Scatterplots > Scatterplots). Sale_Price is entered as the Vertical Variable. Each of the numeric columns may be entered, in turn, for the Horizontal Variable. Columns with categorical values may be entered as a Grouping (Symbol) Variable for some cases. Labels are selected for Variable Names, for easier reading.



Hybrid Appraisal Models



Hybrid Appraisal Models

While these plots do not give a full picture of all the interactions that occur among all the attributes, they do give a feel for the relationship of each attribute with the sale price. Some of the obvious tendencies include a fairly clear subdivision difference, a negative trend for age, little or no trend for basement square feet or lot size, and a limited number of properties with pools (making the pool term difficult to estimate).

Considering the available attributes of the properties and the type of data in each column (numeric or categorical), a reasonable general form for the model is

$$\text{Sale Price} = \text{Hybrid Model}_{\text{Overall}} (\text{Hybrid Model}_{\text{Land}} + \text{Hybrid Model}_{\text{Building}} + \text{Hybrid Model}_{\text{Garage}} + \text{Hybrid Model}_{\text{Pool}})$$

The model form for each component that will be used in this example is

$$\text{Hybrid Model}_{\text{Overall}} = b_1^{\text{NBHD=Glen Lake}} \times b_2^{\text{NBHD=Park Grove}} \times b_3^{\text{Lake Front=Yes}}$$

$$\text{Hybrid Model}_{\text{Land}} = b_4 \times \text{LotSQFT}$$

$$\text{Hybrid Model}_{\text{Building}} = b_5^{\text{Wall Type=Brick}} \times b_6^{\text{Wall Type=Siding}} \times (b_7 \times \text{MainSQFT} + b_8 \times \text{BATHS} + b_9 \times \text{BaseSQFT} + b_{10} \times \text{AGE})$$

$$\text{Hybrid Model}_{\text{Garage}} = b_{11} \times \text{GARAGE}$$

$$\text{Hybrid Model}_{\text{Pool}} = b_{12} \times \text{POOL}$$

Thus, the general form of the full model is

$$\begin{aligned} \text{Sale Price} = & b_1^{\text{NBHD=Glen Lake}} \times b_2^{\text{NBHD=Park Grove}} \times b_3^{\text{Lake Front=Yes}} \times (b_4 \times \text{LotSQFT} + \\ & b_5^{\text{Wall Type=Brick}} \times b_6^{\text{Wall Type=Siding}} \times (b_7 \times \text{MainSQFT} + b_8 \times \text{BATHS} + b_9 \times \text{BaseSQFT} + \\ & b_{10} \times \text{AGE}) + b_{11} \times \text{GARAGE} + b_{12} \times \text{POOL}) \end{aligned}$$

The reference value for *NBHD* is Wood Village and the reference value for *Wall Type* is Wood.

The best model fit (best set of estimated parameters) will be determined as the model with the lowest average absolute percent error.

Setup

To run this example, complete the following steps:

1 Open the Recent Sales example dataset

- From the File menu of the NCSS Data window, select **Open Example Data**.
- Select **Recent Sales** and click **OK**.

2 Specify the Hybrid Appraisal Models procedure options

- Find and open the **Hybrid Appraisal Models** procedure using the menus or the Procedure Navigator.
- The settings for this example are listed below and are stored in the **Example 1** settings file. To load these settings to the procedure window, click **Open Example Settings File** in the Help Center or File menu.

Hybrid Appraisal Models

Model Columns Tab

Sale Price Column	Sale_Price
Component 1	Overall
Column	NBHD
Type.....	Bin Exp
Component 2	Overall
Column	Lake_Front(No)
Type.....	Bin Exp
Component 3	Land
Column	Lot_SF
Type.....	Coef
Component 4	Bldg1
Column	Walls_Type
Type.....	Bin Exp
Component 5	Bldg1
Column	Main_SF
Type.....	Coef
Component 6	Bldg1
Column	Baths
Type.....	Coef
Component 7	Bldg1
Column	BS_SF_Fin
Type.....	Coef
Component 8	Bldg1
Column	Age
Type.....	Coef
Min Start Max.....	-B 0 B
Component 9	Garage
Column	Garage
Type.....	Coef
Component 10	Pool
Column	Pool
Type.....	Coef

Reports Tab

Run Summary Report	Checked
Nonlinear Regression Iteration Report.....	Checked
Differential Evolution Iteration Report	Checked
Model Specification and Estimation Report	Checked
Model (Component Form).....	Checked
Estimated Model (Reading Form)	Checked
Estimated Model (Transformation Form)	Checked
Appraisal Ratio Report.....	Checked
Estimated Values for Estimation Rows	Checked
Estimated Values and Residuals Report	Checked

Hybrid Appraisal Models

Poorly Estimated Properties Report.....**Checked**
 Percent Error Cutoff.....**30**
 Sale Price and Residual Decimal Places.....**0**

Report Options (*in the Toolbar*)

Variable Labels.....**Column Names**

3 Run the procedure

- Click the **Run** button to perform the calculations and generate the output.

Run Summary Report

Run Summary Report

Item	Value
Model (Component Form)	Sale_Price = Overall(Land + Building + Garage + Pool)
Estimation Method	Minimize Average Percent Error (AAPE)
Final Value of AAPE	11.72
R-Squared (from Nonlinear Reg.)	0.8802978
Random Seed	7408926
Number of Columns Used	11 (One Sale Price column plus 10 columns for the model.)
Number of Parameters in Model	12
Number of Rows for Model	125
Number of Nonl. Reg. Iterations	21
Number of Diff. Evol. Iterations	1

This report displays summary information about the model estimation process.

Model (Component Form)

This shows the general form of the model that was estimated. In the context of the documentation formulas, this would be

$$Sale\ Price = Hybrid\ Model_{Overall} (Hybrid\ Model_{Land} + Hybrid\ Model_{Building} + Hybrid\ Model_{Garage} + Hybrid\ Model_{Pool})$$

Estimation Method

The estimation method indicates the criteria that is minimized during the model estimation process.

Final Value of AAPE

This shows the final (minimum) value of the absolute average percent error. The result 11.72 may be interpreted as the average percent difference of the estimated sale price from the actual sale price.

R-Squared (from Nonlinear Reg.)

This is the R-Squared that was achieved by the nonlinear regression portion of the model search. There is no direct R-Squared defined for nonlinear regression. This is a pseudo R-Squared constructed to approximate the usual R-Squared value used in multiple regression. The following generalization of the usual R-Squared formula is used:

$$R\text{-Squared} = (ModelSS - MeanSS)/(TotalSS - MeanSS)$$

where *MeanSS* is the sum of squares due to the mean, *ModelSS* is the sum of squares due to the model, and *TotalSS* is the total (uncorrected) sum of squares of Y (the dependent variable, i.e., sale price).

This version of R-Squared tells you how well the model performs after removing the influence of the mean of Y. Since many nonlinear models do not explicitly include a parameter for the mean of Y, this R-Squared may be negative (in which case we set it to zero) or difficult to interpret. However, if you think of it as a direct extension of the R-Squared that you use in multiple regression, it will serve well for comparative purposes.

Random Number Seed

The random number seed is shown so that if you want to duplicate these results, you can enter this random number seed.

Number of Columns Used

This is the total number of columns of the dataset used in the estimation process.

Number of Parameters in Model

This is the number of parameters that were estimated by the model.

Number of Rows for Model

This is the number of rows from the dataset that were used to estimate the model. It does not count the rows for which the sale price is estimated.

Number of Nonl. Reg. Iterations

This is the number of iterations used by the nonlinear regression procedure.

Number of Diff. Evol. Iterations

This is the number of iterations used by the differential evolution algorithm.

Nonlinear Regression Iteration Section

Nonlinear Regression Iteration Section

Minimization of the Sum of Squared Errors

Iteration Number	Sum of Squared Errors	Message
0	6.801228E+12	
1	5.772725E+12	Stepsize reduced to 0.09796838 by bounds. Lower bound active on B6.
2	2.204183E+12	Freeing parameter B6. Stepsize reduced to 0.7163185 by bounds. Stepsize reduced to 0.7214025 by bounds. Stepsize reduced to 0.7646232 by bounds. Stepsize reduced to 0.9217642 by bounds.
3	8.483604E+11	
4	4.4043E+11	
5	3.029399E+11	
6	2.252354E+11	
.	.	
.	.	
.	.	
18	1.250265E+11	
19	1.250265E+11	
20	1.250265E+11	
21	1.250265E+11	

Convergence criterion met.

This report displays the sum of squared errors for each iteration of the nonlinear regression process. It allows you to observe the algorithm's progress toward the solution. If you do not see the message 'Convergence criterion met' at the bottom, it means that the algorithm did not terminate normally, and you should take corrective action—which usually means that you should increase the maximum number of iterations or provide different starting values.

The Message column gives notes of adjustments that are made in the algorithm during the iteration process.

Differential Evolution Iteration Section

Differential Evolution Iteration Section

Minimization of the Average |Percent Error| (AAPE)

Iteration Number	Value of AAPE
0	11.72251
1	11.72251

This report displays the value of the criteria that is being minimum by the differential evolution algorithm. In this example, the criteria is the average absolute percent error between the actual and estimated sale price. The differential evolution search converged very quickly. Apparently, the nonlinear regression result was very close to or exactly the average absolute percent error estimate.

Model Specification and Estimation Section

Model Specification and Estimation Section

Model (Component Form): Sale_Price = Overall(Land + Building + Garage + Pool)

Component Estimates

Parm. Name	Component	Type	Column or Term	Parameter Estimate	Starting Value	Parameter Bounds
B1	Overall	Bin Exp (B^X)	NBHD="Glen Lake"	1.399654	1	0 to 5
B2	Overall	Bin Exp (B^X)	NBHD="Park Grove"	0.8573917	1	0 to 5
B3	Overall	Bin Exp (B^X)	Lake_Front="Yes"	1.040628	1	0 to 5
B4	Land	Coef (B*X)	Lot_SF	1.453138	1	0.001 to B
B5	Building	Bin Exp (B^X)	Walls_Type="Brick"	1.007641	1	0 to 5
B6	Building	Bin Exp (B^X)	Walls_Type="Siding"	0.805634	1	0 to 5
B7	Building	Coef (B*X)	Main_SF	61.75835	1	0.001 to B
B8	Building	Coef (B*X)	Baths	10721.02	1	0.001 to B
B9	Building	Coef (B*X)	BS_SF_Fin	19.66311	1	0.001 to B
B10	Building	Coef (B*X)	Age	-1374.53	0	-B to B
B11	Garage	Coef (B*X)	Garage	27416.4	1	0.001 to B
B12	Pool	Coef (B*X)	Pool	62428.48	1	0.001 to B

This report displays the details of the estimation of each parameter in the model.

Parm. Name

The name of the parameter is shown on this line.

Component

This shows the component of the model term. The general form of the model is shown at the top of the section.

Type

This shows the type (form) of the model term as specified on the Model Columns tab.

Column or Term

This gives the name of column from the dataset. Note that for columns where the type is binary exponential, a separate term is generated for each non-reference value. For example, the term for NBHD="Glen Lake" has a 1 whenever NBHD is Glen Lake, and 0 otherwise.

Parameter Estimate

This is the estimated value of the parameter in the hybrid model. Note that these values should not be analyzed individually, but together as a group. Parameter estimates may change, even dramatically, if the model is re-estimated with a change in the form of the model (e.g., including or excluding terms).

Starting Value

These are the values (as specified on the Model Columns tab) used by the nonlinear regression algorithm in the first iteration. Since the differential evolution algorithm uses the nonlinear regression estimation results as its starting values, these values have little influence on the results of the differential evolution algorithm. Instead, they influence the speed of convergence.

Parameter Bounds

These are the user-specified limits for the parameter estimates. If you notice an estimate that is equal to one of its bounds, the term should be carefully analyzed to determine if the bound should be relaxed to allow a wider search range.

Model (Component Form) Section

Model (Component Form) Section	
Component	Detail
Overall	(B1)^(NBHD="Glen Lake")*(B2)^(NBHD="Park Grove")*(B3)^(Lake_Front="Yes")
Land	(B4)*Lot_SF
Building	(B5)^(Walls_Type="Brick")*(B6)^(Walls_Type="Siding")*((B7)*Main_SF+(B8)*Baths +(B9)*BS_SF_Fin+(B10)*Age)
Garage	(B11)*Garage
Pool	(B12)*Pool

Model (Component Form): Sale_Price = Overall(Land + Building + Garage + Pool)

This report displays the terms that make up each component. The parameters to be estimated are B1, B2, ...

This report gives the output form of

$$Hybrid Model_{Overall} = b_1^{NBHD=Glen Lake} \times b_2^{NBHD=Park Grove} \times b_3^{Lake Front=Yes}$$

$$Hybrid Model_{Land} = b_4 \times LotSQFT$$

$$Hybrid Model_{Building} = b_5^{Wall Type=Brick} \times b_6^{Wall Type=Siding} \times (b_7 \times MainSQFT + b_8 \times BATHS + b_9 \times BaseSQFT + b_{10} \times AGE)$$

$$Hybrid Model_{Garage} = b_{11} \times GARAGE$$

$$Hybrid Model_{Pool} = b_{12} \times POOL$$

Component

The name of the higher-level component of the hybrid model.

Detail

The terms of the component as they are used in the model.

Estimated Model (Reading Form) Report

Estimated Model (Reading Form) Report

Estimated Market Value =

$(1.40)^{\text{NBHD}=\text{Glen Lake}} * (0.86)^{\text{NBHD}=\text{Park Grove}} * (1.04)^{\text{Lake Front}=\text{Yes}} * ((1.45) * \text{Lot_SF})$
 $+ ((1.01)^{\text{Walls_Type}=\text{Brick}} * (0.81)^{\text{Walls_Type}=\text{Siding}} * (61.76) * \text{Main_SF} + (10721.02) * \text{Baths}$
 $+ (19.66) * \text{BS_SF_Fin} + (-1374.53) * \text{Age}) + ((27416.40) * \text{Garage} + ((62428.48) * \text{Pool}))$

This report shows the model in reading form. The number of decimal places for the parameter estimates is set by the user. In the documentation formula form, this model would look like

$$\text{Sale Price} = 1.40^{\text{NBHD}=\text{Glen Lake}} \times 0.86^{\text{NBHD}=\text{Park Grove}} \times 1.04^{\text{Lake Front}=\text{Yes}} \times (1.45 \times \text{LotSQFT} + 1.01^{\text{Wall Type}=\text{Brick}} \times 0.81^{\text{Wall Type}=\text{Siding}} \times (61.76 \times \text{MainSQFT} + 10721.02 \times \text{BATHS} + 19.66 \times \text{BaseSQFT} - 1374.53 \times \text{AGE}) + 27416.40 \times \text{GARAGE} + 62428.48 \times \text{POOL})$$

Estimated Model (Transformation Form) Report

Estimated Model (Transformation Form) Report

Estimated Market Value =

$(1.39965383923748)^{\text{NBHD}=\text{Glen Lake}} * (0.857391728281951)^{\text{NBHD}=\text{Park Grove}} * (1.04062843036637)^{\text{Lake Front}=\text{Yes}} * (((1.45313794582045) * \text{Lot_SF}) + ((1.00764055666536)^{\text{Walls_Type}=\text{Brick}} * (0.805633962785495)^{\text{Walls_Type}=\text{Siding}}) * ((61.7583507785408) * \text{Main_SF} + (10721.0156090402) * \text{Baths} + (19.6631092449847) * \text{BS_SF_Fin} + (-1374.52982803653) * \text{Age}) + ((27416.4016963174) * \text{Garage}) + ((62428.4812000623) * \text{Pool}))$

This is the model with full precision parameter estimates. This expression may be copied onto the Clipboard and pasted into a transformation cell of the dataset to estimate other properties. This expression is always provided in double precision.

Appraisal Ratio Section

Appraisal Ratio Section				
Statistic	Actual Sale_Price	Estimated Sale_Price	Ratio	Percent Error
Number of Properties	125	125	125	125
Mean	228407	227882	1.02	11.72
Minimum	99900	139343	0.70	0.39
Lower Quartile	179900	180239	0.90	3.78
Median	208000	208785	1.00	9.98
Upper Quartile	246500	238826	1.09	15.70
Maximum	610000	584393	1.46	46.20
Range	510100	445050	0.76	45.81
I. Q. Range	66600	58588	0.19	11.92
Variance	8423219100	7656332267	0.02	97.77
Std. Deviation	91778	87500	0.15	9.89
Ave Dev. from Median	53924	49361	0.12	7.30
Coef. of Variation x 100	40.18	38.40	15.02	84.35
Coef. of Dispersion x 100	25.92	23.64	11.67	73.13
Weighted Mean			1.00	
Price Related Differential			1.02	

This report provides some of the basic statistics of an appraisal ratio study. For a much more comprehensive ratio study analysis, the estimated values should be stored in a column of the spreadsheet (see the Storage tab), and the Appraisal Ratio Studies procedure should be used.

The following are definitions that are not found in the Descriptive Statistics procedure.

Ave |Dev. from Median|

The average of the absolute values of the deviations from the median.

Coef. of Dispersion (COD)

This is 100 times the average absolute deviation about the median divided by the median.

Coef. of Variation (COV)

This is 100 times the standard deviation divided by the mean.

Weighted Mean

The weighted ratio mean is the mean of the estimated values divided by the mean of the actual values.

Price Related Differential (PRD)

The price related differential is the mean ratio divided by the weighted mean ratio. It provides a measure of assessment regressivity or progressivity. A PRD greater than 1.0 indicates that the more expensive properties are under appraised. A PRD less than one indicates that the more expensive properties are over appraised.

Estimated Values for Estimation Rows Section

Estimated Values for Estimation Rows Section

Row No.	Estimated Sale Price
126	225691
127	228452
128	200963

This section shows the estimated sale price for all rows where the attribute data is given, but the sale price column value is left blank.

Estimated Values and Residuals Section

Estimated Values and Residuals Section

Row No.	Actual Sale Price	Estimated Sale Price	Actual - Estimated (Residual)	Estimated / Actual (Ratio)	Percent Error
1	147900	208785	-60885	1.41	41.17
2	184000	163480	20520	0.89	11.15
3	225000	195885	29115	0.87	12.94
4	108561	157732	-49171	1.45	45.29
5	165500	170761	-5261	1.03	3.18
6	191000	209706	-18706	1.10	9.79
7	276000	227546	48454	0.82	17.56
8	178000	238826	-60826	1.34	34.17
9	229000	216159	12841	0.94	5.61
10	246500	243432	3068	0.99	1.24
.
.
.
120	261000	223913	37087	0.86	14.21
121	260000	232412	27588	0.89	10.61
122	268900	279958	-11058	1.04	4.11
123	281000	218943	62057	0.78	22.08
124	300000	304272	-4272	1.01	1.42
125	225000	214628	10372	0.95	4.61
126		225691			
127		228452			
128		200963			

This report shows the actual and estimated sale prices as well as 3 measures of disagreement. Assessors commonly study the ratio and/or the absolute percent error of estimated values to determine the quality or accuracy of appraisal.

Poorly Estimated Properties Section

Poorly Estimated Properties Section

Properties with absolute percent error greater than 30% are displayed.
 Absolute Percent Error = $100 * |(Actual - Estimated) / Actual|$.

Row No.	Actual Sale Price	Estimated Sale Price	Actual - Estimated (Residual)	Estimated / Actual (Ratio)	Percent Error
1	147900	208785	-60885	1.41	41.17
4	108561	157732	-49171	1.45	45.29
8	178000	238826	-60826	1.34	34.17
16	115500	151244	-35744	1.31	30.95
18	134000	184150	-50150	1.37	37.43
39	99900	146054	-46154	1.46	46.20
45	255000	178203	76797	0.70	30.12
47	120000	165590	-45590	1.38	37.99
85	197900	257331	-59431	1.30	30.03
104	216900	282717	-65817	1.30	30.34

This report shows those rows with a large (percentage) difference from the estimated sale price to the actual sale price. The percent error cutoff, as set on the Reports tab, is 30%. Each row in this report should be analyzed to determine if there is some underlying explanation as to why the estimation is so poor. In some cases, it may be reasonable to try re-estimating the same model without these poorly estimated properties, to determine their influence.

Storing Estimated Values to the Spreadsheet

Although no values were stored to the dataset in this example, the option is available to store the estimated values, residuals (actual sale price – estimated sale price), or ratios (estimated sale price / actual sale price) to the spreadsheet. This is done by entering a column name or number on the Storage tab of the procedure and then running the procedure.

Model Refinement

Determining the appropriateness of each model term is more difficult when using hybrid models as compared to multiple regression analysis. In multiple regression, the influence of each term may be measured directly, and even tested. In hybrid models, one can only look at the change in the minimization criteria when including or excluding a term. The tradeoff is flexibility. Hybrid models are more flexible.

Two statistics to consider when comparing hybrid model estimations are the final value of the minimized criteria (e.g., Final Value of AAPE) and the R-squared from nonlinear regression. When inclusion or exclusion of a term changes either of these values dramatically, it is likely an important term to include in the model.

Parameter estimates can also be monitored to determine if they make practical sense. For example, the plots tell us that the Glen Lake subdivision sale prices are generally much higher, and the Park Grove subdivision sale prices are lower. We would expect the parameter estimates to reflect this observation. In this example the Glen Lake parameter estimate is 1.40, while the Park Grove parameter estimate is 0.86. These multipliers are consistent with the observed plots.

Hybrid Appraisal Models

On the other hand, the pool coefficient estimate of 62428.48 may be of concern, since it is such a large estimate, and only three properties were used in the estimation process. If the pool term is removed from the model, the model estimate changes from

Estimated Model**Estimated Market Value =**

$$(1.40)^{(\text{NBHD}=\text{"Glen Lake"})} * (0.86)^{(\text{NBHD}=\text{"Park Grove"})} * (1.04)^{(\text{Lake_Front}=\text{"Yes"})} * (((1.45) * \text{Lot_SF}) + ((1.01)^{(\text{Walls_Type}=\text{"Brick"})} * (0.81)^{(\text{Walls_Type}=\text{"Siding"})} * ((61.76) * \text{Main_SF} + (10721.02) * \text{Baths} + (19.66) * \text{BS_SF_Fin} + (-1374.53) * \text{Age})) + ((27416.40) * \text{Garage}) + ((62428.48) * \text{Pool}))$$

to

Estimated Model**Estimated Market Value =**

$$(1.41)^{(\text{NBHD}=\text{"Glen Lake"})} * (0.88)^{(\text{NBHD}=\text{"Park Grove"})} * (1.02)^{(\text{Lake_Front}=\text{"Yes"})} * (((1.73) * \text{Lot_SF}) + ((1.01)^{(\text{Walls_Type}=\text{"Brick"})} * (0.81)^{(\text{Walls_Type}=\text{"Siding"})} * ((61.86) * \text{Main_SF} + (12086.52) * \text{Baths} + (17.62) * \text{BS_SF_Fin} + (-1740.03) * \text{Age})) + ((27807.47) * \text{Garage}))$$

and the Final Value of AAPE changes from 11.72 to 12.17. The R-Squared changes from 0.880 to 0.871. Given this information, if \$62,428 seems well outside the reasonable range for the effect of a pool, one may consider running the model without this term. The model should always be re-run when adding or removing a term. The change in all the other estimated parameters can be seen by comparing the estimated parameters of the before and after models.