Catalog of Curves
for Curve Fitting

Biometrics Information Handbook Series

CATALOGUE OF CURVES
FOR CURVE FITTING

by

Vera Sit
Melanie Poulin-Costello

Series Editors

Wendy Bergerud
Vera Sit

Ministry of Forests
Research Program

1994

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

TABLES

FIGURES

1 INTRODUCTION

This handbook is a collection of linear and non-linear models for fitting experimental data. It is intended to help researchers fit appropriate curves to their data.

Curve fitting, also known as regression analysis, is a common technique for modelling data. The simplest use of a regression model is to summarize the observed relationships in a particular set of data. More importantly, regression models are developed to describe the physical, chemical and biological processes in a system (Rawlings 1988).

This handbook is organized in the following manner:

- Section 2 defines the terminology and briefly describes the general idea of regression.

- Section 3 discusses the use of the SAS procedures `PROC REG` and `PROC NLIN` for linear and non-linear curve fitting.

- Section 4 presents eight classes of curves frequently used for modelling data. Each class contains several curves which are described in detail. For each curve, the equation, the derivatives, and the linearized form of the equation are provided, as well as sample plots and SAS programs for fitting the curve.

- Section 5 explains how to use this handbook for curve fitting. Some strategies for selecting starting values and the concept of convergence are also discussed. Two examples are given to illustrate the curve-fitting procedures.

- Section 6 provides a brief introduction to various basic attributes of curves. Also included are the corresponding algebra and calculus for identifying these attributes.

A strong statistical or calculus background is not required to use this handbook. Although the discussions and examples are based on SAS programs and assume some knowledge of programming, they should be of general interest for anyone fitting curves.

This handbook does not provide an in-depth discussion of regression analysis. Rather, it should be used in conjunction with a reliable text on linear and non-linear regression such as Ratkowsky (1983) or Rawlings (1988).

2 REGRESSION ANALYSIS

In regression, a model or a mathematical equation is developed to describe the behaviour of a variable of interest. The variable may be the growth rate of a tree, body weight of a grizzly bear, abundance of fish, or percent cover of vegetation. This is called the **dependent variable** and is denoted by Y. Other variables that are thought to provide information on the behaviour of the dependent variable are incorporated into the equation as explanatory variables. These variables (e.g., seedling diameter, chest size of a grizzly bear, volume of cover provided by large woody debris in streams, or light intensity) are called **independent variables** and are denoted by X. They are assumed to be known without error.

In addition to the independent variables, a regression equation also involves unknown coefficients called **parameters** that define the behaviour of the model. Parameters are denoted by lowercase letters (a, b, c, etc). A **response curve** (or "response surface", when many independent variables are involved) represents the true relationship between the dependent and independent variables. In regression analysis, a model is developed to approximate this response curve — a process that estimates parameters from the available data. A regression model has the general form:

$$Y = f(X) + E$$

where $f(X)$ is the expected model and E is the error term. For simplicity, the error term E is omitted in subsequent sections.

A **linear model** is one that is linear in the parameters — that is, each term in the model contains only one parameter which is a multiplicative constant of the independent variable (Rawlings 1988, p. 374). All other models are non-linear. For example:

$$Y = a\ X + b$$

and
$$Y = c\ X^2 + d\ X + g$$

are linear models with parameters **a** and **b**, and **c**, **d**, and **g**, respectively. But:

$$Y = a\ X^b$$

and
$$Y = a\ \sin(bX)$$

are non-linear models.

Some non-linear models are **intrinsically linear**. This means they can be made linear with an appropriate transformation. For example, the model:

$$Y = a\ e^{bx}$$

is intrinsically linear. It can be converted to linear form with the natural logarithm (ln) transform:

$$\ln(Y) = \ln(a) + b\ X$$

However, some non-linear models cannot be linearized. The model:

$$Y = \sin(bX)$$

is an example.

Linear models are very restrictive and are usually used for first-order approximations to true relationships. On the other hand, non-linear models such as the inverse polynomial models, exponential growth models, logistic model and Gompertz model, are often more realistic and flexible. In some cases, non-linear models will have fewer parameters to be estimated and are therefore simpler than linear models.

Once a curve or model has been fitted (i.e., the parameters are estimated) to a set of data, it can be used to estimate Y for each value of X. The deviation or difference of the observed Y from its estimate is called a **residual**. It is a measure of the agreement between the model and the data.

The parameters in a model can be estimated by the **least squares method** or the **maximum likelihood method**. With the least squares method the fitted model should have the smallest possible sum of squared residuals, while with the maximum likelihood method the estimated parameters should maximize the likelihood of obtaining the particular sample. Under the usual assumptions that the residuals are independent with zero mean and common variance σ^2, the least squares estimators will be the best (minimum variance) among all possible linear unbiased estimators. When the normality assumption is satisfied, least squares estimators are also maximum likelihood estimators.

3 CURVE FITTING WITH SAS

Two SAS procedures are available for curve fitting — `PROC REG` and `PROC NLIN`. `PROC REG` is suitable for fitting linear models whereas `PROC NLIN` is used for fitting non-linear models. Intrinsically linear models can also be fitted with `PROC REG` provided that the appropriate transformations are known. Both SAS procedures use the least squares method for estimating the model parameters.

3.1 Linear Regression Using `PROC REG`

The following is an example of the `PROC REG` procedure:

```
PROC REG DATA = one;
  MODEL Y = X;
RUN;
```

The keywords `PROC REG` invoke the SAS procedure for linear regression. The option:

$$DATA = SASdataset;$$

names the SAS data set to be analyzed by `PROC REG`. The most recently created SAS data set is used if the `DATA =` option is omitted.

The model to be fitted is specified in the `MODEL` statement which has the form:

$$MODEL\ dependent\ variables = independent\ variables/options;$$

The explanatory variables are listed on the right-hand side of the equal sign. In the above example,

$$MODEL\ Y = X;$$

specifies the model $\qquad\qquad Y = a + b\ X$

If more than one dependent variable is listed on the left-hand side of the equal sign, then separate models will be fitted for each one. For example, the statement:

$$MODEL\ Y\ Z = X1\ X2;$$

requests two models to be fitted. These models are:

$$Y = a + b\ X1 + c\ X2$$

and $\qquad\qquad Z = m + n\ X1 + k\ X2$

The variables specified in the `MODEL` statement must be numeric; functions of variables are not allowed. Therefore, to fit the model:

$$Y = a + b\ X + c\ X^2$$

one must first create a new variable (`X2 = X*X`) in a data step, then issue the `MODEL` statement:

$$MODEL\ Y = X\ X2;$$

The estimated parameters and the fitted values can be saved in SAS data sets for later use. See Chapter 28 of the SAS/STAT User's Guide (SAS Institute Inc.1988a) for a complete description of this procedure.

3.2 Non-linear Regression Using `PROC REG`

Intrinsically linear models can be fitted with `PROC REG` if the appropriate transformations are known. For example, the model:

$$Y = a\ X^b$$

can be converted to linear form using the natural log (ln) transformation:

$$\ln(Y) = \ln(a) + b\ \ln(X)$$

or
$$Z = c + b\ U$$

where $Z = \ln(Y)$, $c = \ln(a)$, and $U = \ln(X)$. Before using `PROC REG`, the natural log transformation of Y and X must be performed in a data step. If the variables Y and X were stored in a SAS data set called OLD, then the following SAS code[1] could be used to do the regression:

```
DATA NEW;
  SET OLD;
  Z = LOG(Y); U = LOG(X);

PROC REG DATA = NEW;
  MODEL Z = U;

RUN;
```

Once the regression is completed, the fitted parameters can be transformed to the parameters in the non-linear model. Continuing with the example, the parameter **a** in the non-linear model can be obtained from the intercept of the fitted linear model with the exponential function:

$$a = e^c$$

Parameter **b** has the same value in both the linear and non-linear model.

When a linearized model is fitted with `PROC REG`, the error term is also transformed. In this example, the fitted regression model is:

$$\ln(Y) = \ln(a) + b\ \ln(X) + E \tag{1}$$

The error term E in this model is additive. When the model is transformed back to its non-linear form, it becomes:

$$Y = a\ X^b \cdot e^E \tag{2}$$

That is, the actual model fitted has a multiplicative error term. The SAS procedure `PROC NLIN`, however, fits the model:

$$Y = a\ X^b + E \tag{3}$$

If E in the linear model (1) is normally distributed, then e^E in the non-linear model (2) is log-normal distributed, which is different from the normal distribution assumed in the non-linear model (3). Because of the different error structures in models (2) and (3), different parameter estimates may result. An example is given in Section 5.6.

In this handbook, the linearized forms, the relationships between parameters in the linear and non-linear forms, and the SAS programs for fitting the linearized models are provided for all intrinsically linear models.

3.3 Non-linear Regression Using `PROC NLIN`

Non-linear models are more difficult to specify and estimate than linear models. In SAS, they can be fitted with the procedure `NLIN`. To use `PROC NLIN`, one must write the regression expression, declare parameter names, guess the starting values of the parameters, and possibly specify the derivatives of the model with respect to the parameters. `PROC NLIN` is an iterative procedure with five methods available as options:

- gradient method
- Newton method

---

[1] In SAS, the function `LOG` is the natural logarithm (*ln*). Logarithm to the base 10 can be requested by the SAS function `LOG10`.

- modified Gauss-Newton method

- Marquardt method

- multivariate secant or false position (DUD) method

All methods except DUD (Does not Use Derivatives) require a partial derivative of the model with respect to each parameter (i.e., $\partial Y/\partial a$, $\partial Y/\partial b$, etc.) The Newton method also requires the second derivatives of the model with respect to each parameter (i.e., $\partial^2 Y/\partial a^2$, $\partial^2 Y/\partial b^2$, etc.).

When derivatives are provided, the Gauss-Newton is the default method of estimation; otherwise, DUD is the default method. Of the two default methods, the Gauss-Newton method is fastest.

To demonstrate the structure of a `PROC NLIN` step, let us suppose that the model $Y = a\, X^b$ is to be fitted to a set of data. The following is an example to carry out the regression:

```
PROC NLIN DATA = POINT;
  PARAMETER  A = 1.0  B = 0.5;
  MODEL Y = A * X ** B;
  DER.A = X ** B;
  DER.B = A * LOG(X) * X ** B;
RUN;
```

The `PROC NLIN` statement invokes the SAS procedure. It has several options.

$$DATA = SASdataset;$$

names the SAS data set to be analyzed by `PROC NLIN`. If the `DATA` = option is omitted then the most recently created SAS data set is used.

$$METHOD = iteration\ method;$$

specifies the iterative method `NLIN` uses. If the `METHOD` = option is omitted and the `DER` statement is specified, then `METHOD = GAUSS` is used. If the `METHOD` = option is not specified and the `DER` statement is absent, then `METHOD = DUD` is used.

Other options are available to control the iteration process. See Chapter 23 of the SAS/STAT User's Guide (SAS Institute Inc. 1988a) for more detail.

The `PARAMETER` statement defines the parameters and their starting values. It has the form:

$$PARAMETER\ parameter = starting\ values\ .\ .\ .\ ;$$

A range of starting values may also be specified with this statement. For example:

$$PARAMETER\ A = 0\ TO\ 15\ BY\ 5;$$

specifies four starting values (0, 5, 10, and 15) for A.

$$PARAMETER\ A = 1,\ 7,\ 9;$$

specifies three starting values (1, 7, and 9).

The `PARAMETER` statement must follow the `PROC NLIN` statement. See chapter 23 of the SAS/STAT User's Guide (SAS Institute Inc. 1988a) for more details.

The `MODEL` statement defines the equation to be fitted. It has the form:

$$MODEL\ dependent\ variable = expression;$$

The expression can be any valid SAS expression.

The `DER` statement specifies partial derivatives with respect to the parameters. It has the form:

$$DER.parameter = expression;$$

$$DER.parameter.parameter = expression;$$

Use the first statement for specifying first derivatives, and the second for second derivatives.

A special feature of `PROC NLIN` is that it allows the creation of new variables within the procedure. This is particularly useful when a certain expression is to be evaluated many times. In the regression example above, the expression X^b appears in the `MODEL` statement, the `DER.A` statement, and the `DER.B` statement. To simplify the program (and reduce computation time), we can add the statement

$$XB = X ** B;$$

to the program as follows:

```
PROC NLIN DATA = POINT;
  PARAMETER A = 1.0 B = 0.5;
  XB = X ** B;
  MODEL Y = A * XB;
  DER.A = XB;
  DER.B = A * LOG(X) * XB;
RUN;
```

The `PROC NLIN` codes for fitting the curves are provided in Section 4. Readers need only to rename the X and Y variables and choose the starting values for the parameters. The starting values can be determined by comparing a scatter plot of the experimental data with the sample plots shown throughout the handbook, and matching the description of the parameters in the handbook with the corresponding features shown in the scatter plot. This is further explained in the Section 5.2 and demonstrated in the examples.

4 A CATALOG OF CURVES

Curves for both linear and non-linear models are presented in this section. They are grouped into eight classes:

1. polynomials

2. inverse polynomials

3 exponential functions

4. power functions

5. combined exponential and power functions

6. logarithmic functions

7. trigonometric functions

8. common distributions

Each class contains several functions and each function is treated separately. The following information is provided:

The **functional form** is the equation of the model. It is stated as:

$$Y = f(X)$$

Standard functional names are used if possible. Otherwise, the functions are named sequentially as Type I or Type II. The first derivative of Y with respect to X $\left(\text{i.e. } \frac{dY}{dX} \right)$ is provided for determining special features of a curve such as the maximum, minimum, and point of inflection. These features and the technique to identify them are discussed in Section 6. The partial derivatives of Y with respect to model parameters $\left(\text{e.g. } \frac{\partial Y}{\partial a}, \frac{\partial Y}{\partial b} \right)$ are also given as they are required for `PROC NLIN` (but not for the derivative-free `[DUD]` method).

For intrinsically linear models, the linearized form and the relationship between the parameters in the linear and non-linear forms are supplied. Also included is a short description of the curve and the roles the parameters play in determining the behaviour of the curve. Sample SAS programs for doing linear or non-linear regression are provided for readers' convenience. To use these programs, the X's and Y's should be replaced by the appropriate variable names. In addition, appropriate starting values must be substituted if `PROC NLIN` is to be used (the values in the sample programs are arbitrary).

Finally, a number of graphs are presented for each curve. Each graph shows the impact of changing a parameter on the curve when other parameters are kept constant. These graphs are helpful for selecting models and parameter starting values.

The domain of a function is the set of X-values for which the function is defined. For most of the models presented in this section, the domain is the set of real numbers: $-\infty < X < \infty$.

4.1 Polynomials

A **polynomial of degree n** has the form:

$$Y = c_0 + c_1 X + c_2 X^2 + \ldots + c_{n-1} X^{n-1} + c_n X^n$$

where the c_i's are real numbers. They are the **parameters**, or **coefficients**, of the X terms. For example:

$$Y = 1 + 5X + 2X^2 \qquad (1)$$

is a polynomial of degree 2 with parameters $c_0 = 1$, $c_1 = 5$, and $c_2 = 2$. The equation:

$$Y = 4X^3 \qquad (2)$$

is a polynomial of degree 3 with parameters $c_0 = c_1 = c_2 = 0$ and $c_3 = 4$.

Polynomials are unbounded — that is, as X increases indefinitely, the function Y increases or decreases without limit.

A criterion for selecting a functional form is the number of extremums (i.e., maximum or minimum or both). A polynomial of degree n has at most n-1 extremums. For example, equation (1) is a polynomial of degree 2 with one minimum; equation (2) is a polynomial of degree 3 with no extremum. A polynomial of degree 1 is a straight line with no extremum.

A polynomial is linear with respect to its parameters, and therefore can be fitted with `PROC REG`. Polynomials are a common choice for curve fitting because they are simple to use. Lower degree polynomials are easier to interpret than higher degree polynomials. For this reason, only polynomials of degree one, two, and three are included here. Although polynomials can provide a very good fit to data, they extrapolate poorly and should be used with caution.

Any continuous response function can be approximated to any level of precision desired by a polynomial of appropriate degree. Because of this flexibility, it is easy to "overfit" a set of data with polynomial models. Thus, an excellent fit of a polynomial model (or for that matter, any model) cannot be interpreted as an indication that it is, in fact, the true model.

4.1.1 First degree polynomial: linear

Functional form: $\qquad Y = a + bX \qquad\qquad -\infty < X < \infty$

Derivatives: $\qquad \dfrac{dY}{dX} = b$

$$\dfrac{\partial Y}{\partial a} = 1 \qquad \dfrac{\partial Y}{\partial b} = X$$

Linearized model and parameters: $\qquad$ This functional form is already linear.

Description: $\qquad$ This is the equation of a straight line. Parameter **a** is the Y-intercept and it controls the vertical position of the line. Parameter **b** is the slope of the line.

Sample `PROC REG` **program:**

```
PROC REG DATA=LINEAR;
  MODEL Y = X;
RUN;
```

`PROC NLIN` is unnecessary because this is a linear model.

Functional form: $Y = a + bX$ $X > 0$

b = 2.0 vary a

a = 10
a = 5
a = 0
a = -5

a = 5.0 vary b

b = 5
b = 2
b = 0
b = -2

4.1.2 Second degree polynomial: quadratic

Functional form: $Y = a + bX + cX^2$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = b + 2cX$

$\dfrac{\partial Y}{\partial a} = 1 \qquad \dfrac{\partial Y}{\partial b} = X \qquad \dfrac{\partial Y}{\partial c} = X^2$

Linearized model and parameters: This functional form is already linear.
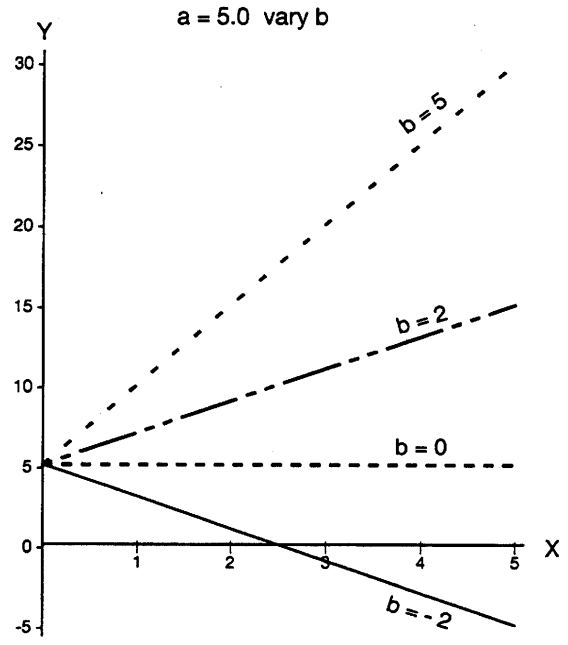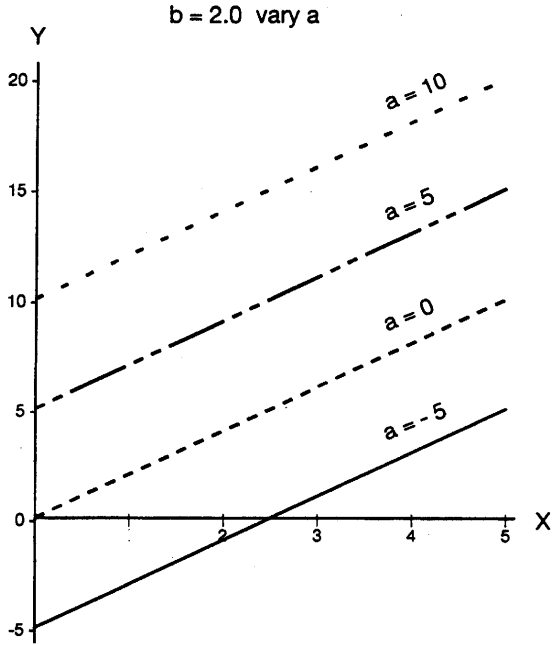
Description: The graph of a quadratic is called a **parabola**. It is commonly expressed in the above functional form. However, the roles of the parameters are easier to understand from the alternative non-linear form:

$$Y = A(X - B)^2 + C$$

The parameters in the two forms are related as follows:

$$a = AB^2 + C \qquad\qquad b = -2AB \qquad\qquad c = A$$

Also, $A = c \qquad\qquad B = -\dfrac{b}{2c} \qquad\qquad C = a - \dfrac{b^2}{4c}$

Parameter **A** controls the shape of the parabola such that:

- for $A > 0$, parabola is concave up
- for $A = 0$, parabola becomes the horizontal line $Y = C$
- for $A < 0$, parabola is concave down

The parabola is symmetric about $\mathbf{X = B}$, which is the location of the extremum on the X-axis. Parameter **C** shifts the parabola up and down the Y-axis. See Section 6.1 for an example using the second degree polynomial.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=PARAB;
  MODEL Y = X X2;
   *NOTE  X2 = X*X   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the alternative form:**

```
PROC NLIN DATA = PARAB;
  PARAMETERS A=5 B=10 C=30;
  XB = (X - B);
  MODEL Y = A*XB**2 + C;
    DER.A = XB**2;
    DER.B = -2*A*XB;
    DER.C = 1;
RUN;
```

Functional form: $Y = A(X - B)^2 + C$

B = 0 C = 0 vary A

A = 100 C = 50 vary B

A = 100 B = 0 vary C
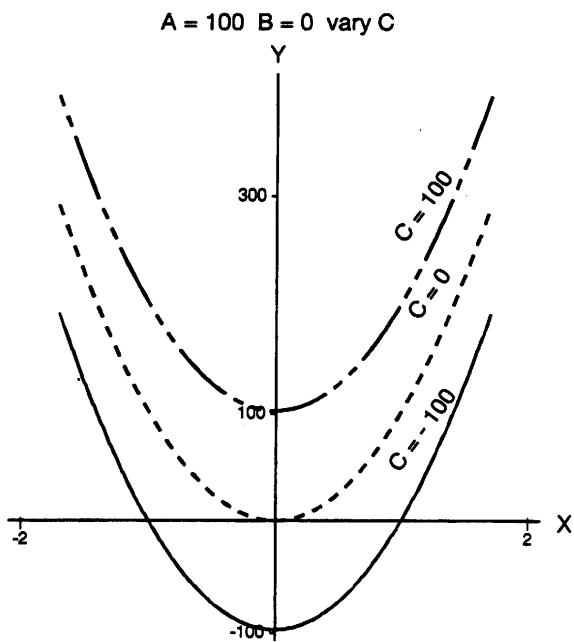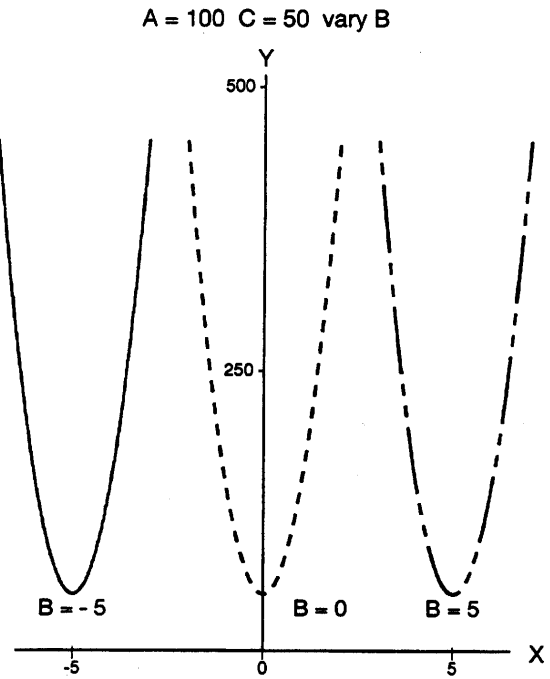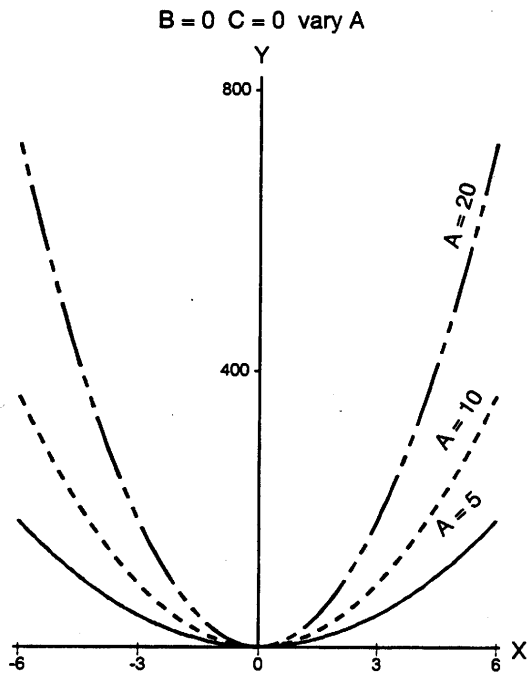
4.1.3 Third degree polynomial: cubic

Functional form: $\quad Y = a + bX + cX^2 + dX^3 \qquad\qquad -\infty < X < \infty$

Derivatives: $\quad \dfrac{dY}{dX} = b + 2cX + 3dX^2$

$$\frac{\partial Y}{\partial a} = 1 \qquad \frac{\partial Y}{\partial b} = X \qquad \frac{\partial Y}{\partial c} = X^2 \qquad \frac{\partial Y}{\partial d} = X^3$$

Linearized model and parameters: This functional form is already linear.

Description: The basic shape of a cubic is a sideways "S" ( ). Parameter **d** shifts the curve up and down the Y-axis. The parameters **a**, **b**, and **c** work together to make the S-shape flatter or deeper. While the above form is linear, the role of the parameters is more clearly understood in the alternative form:

$$Y = A(X - B)(X - C)\left(X - \frac{B+C}{2}\right) + D$$

The parameters in the two forms are related as follows:

$$a = D - ABC\left(\frac{B+C}{2}\right) \qquad\qquad b = A\,\frac{(B+C)^2}{2} + ABC$$

$$c = \frac{-3}{2}A(B+C) \qquad\qquad d = A$$

Also $\qquad\qquad A = d \qquad\qquad\qquad\qquad B = \frac{1}{2}\left[-\frac{2c}{3d} + \sqrt{\frac{4c^2}{9d^2} - \frac{4}{d}\left(b - \frac{2c^2}{9d}\right)}\,\right]$

$$C = \frac{1}{2}\left[-\frac{2c}{3d} - \sqrt{\frac{4c^2}{9d^2} - \frac{4}{d}\left(b - \frac{2c^2}{9d}\right)}\,\right] \qquad\qquad D = a - \frac{c}{3d}\left(b - \frac{2c^2}{9d}\right)$$

In the non-linear form of the cubic, parameter **A** scales the curve in the Y direction and **D** shifts the curve in the Y direction. Parameters **B** and **C** have the same effects on the shape of the curve; together they shift and squeeze the curve in the X direction. As parameters **B** and **C** are equivalent, only the graphs that show the effect of varying **B** are presented in the following pages.

Sample `PROC REG` **program for the functional form:**

```
PROC REG DATA = CUBIC;
  MODEL Y = X X2 X3;
  * X2 = X*X  AND  X3 = X*X*X   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the alternative form:**

```
PROC NLIN DATA = CUBIC2;
  PARAMETER  A= 0.8  B=-0.5  C=-0.4  D=2;
  XB=X-B;  XC=X-C;  BC2=(B+C)/2;
  MODEL  Y = A*XB*XC*(X-BC2)+D;
  DER.A = XB*XC((X-BC2);
  DER.B = A*XC*(B+0.5*C-1.5*X);
  DER.C = A*XB*(C+0.5*B-1.5*X);
  DER.D = 1;
RUN;
```

Functional form: $Y = A(X - B)(X - C)\left(X - \dfrac{B+C}{2}\right) + D$

B = 1 C = - 1 D = 0 vary A

A = 10
A = 1
A = -1
A = - 10

A = 5 B = 1 C = - 1 vary D

D = 3
D = 0
D = - 3

A = 5 C = - 1 D = 0 vary B > 0

B = 2
B = 1
B = 0.5

A = 5 C = 1 D = 0 vary B > 0

B = 0.1
B = 5
B = 2

Functional form: $Y = A(X - B)(X - C)\left(X - \dfrac{B + C}{2}\right) + D$

A = 1 C = -1 D = 0 vary B < 0

A = 1 C = 1 D = 0 vary B < 0
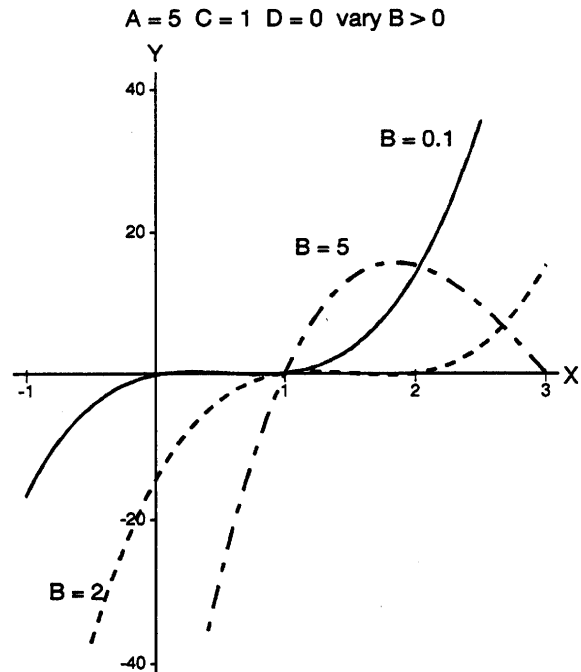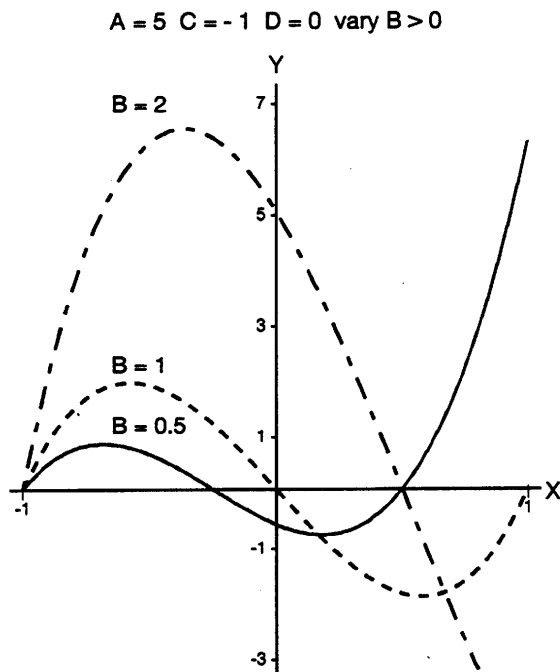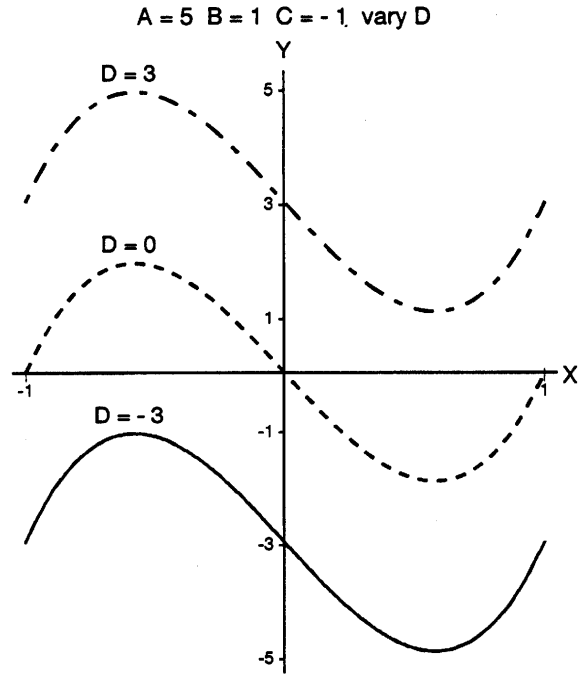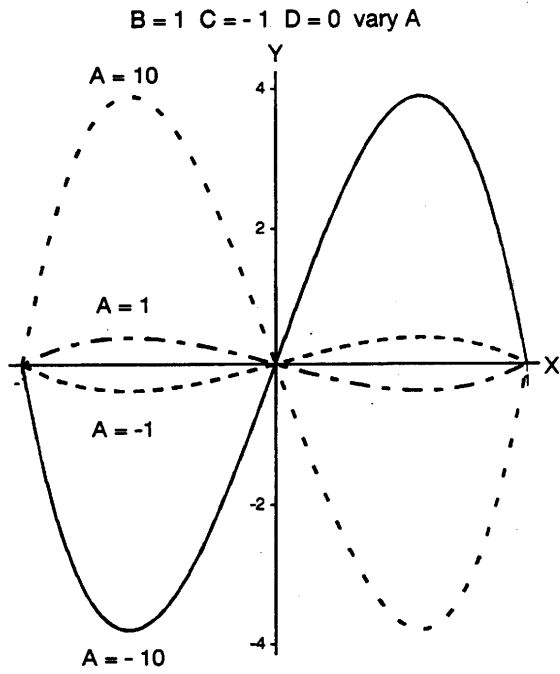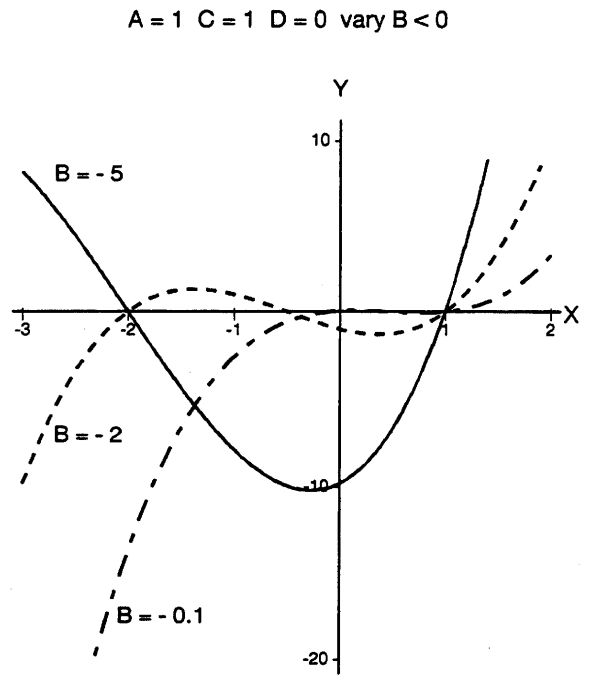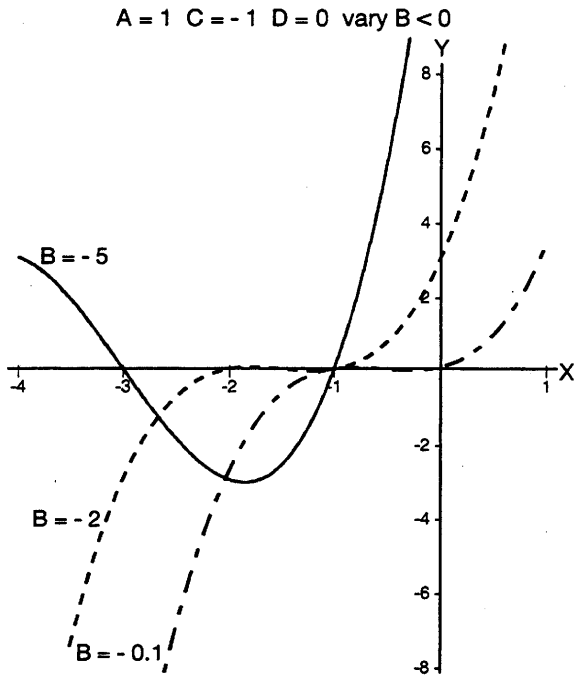
4.2 Inverse Polynomials

In biometrics applications, an increase in a stimulus often produces either a saturation effect or a toxic effect to the experimental subjects. With a saturation effect, the response reaches a finite amount and remains at that level without exceeding it. Therefore, an unbounded functional form, such as a polynomial, is not suitable. With a toxic effect, the response rises to a maximum then drops off, which could be modelled with a concave down second degree polynomial. However, there is usually no *a priori* reason for assuming any symmetry about the peak of the function and a parabola is not appropriate. Instead, these types of response could be modelled with inverse polynomials (Nelder 1966).

An **inverse polynomial of degree n** has the form:

$$\frac{X}{Y} = c_0 + c_1 X + c_2 X^2 + \ldots + c_{n-1} X^{n-1} + c_n X^n$$

or

$$Y = \frac{X}{c_0 + c_1 X + c_2 X^2 + \ldots + c_{n-1} X^{n-1} + c_n X^n}$$

where c_i's are the parameters.

Inverse polynomials are generally non-negative, bounded, and not necessarily symmetric in the second degree case. See Nelder (1966) for a complete description of inverse polynomials.

4.2.1 First degree inverse polynomial: hyperbola

Functional form: $Y = \dfrac{X}{a + bX}$ $X \neq -\dfrac{a}{b}$

Derivatives: $\dfrac{dY}{dX} = \dfrac{a}{(a + bX)^2}$

$\dfrac{\partial Y}{\partial a} = \dfrac{-X}{(a + bX)^2}$ $\dfrac{\partial Y}{\partial b} = \dfrac{-X^2}{(a + bX)^2}$

Linearized model and parameters: $\dfrac{1}{Y} = \dfrac{b_0}{X} + b_1$

$a = b_0$ $b = b_1$

Description: The first degree inverse polynomial has a horizontal asymptote at $Y = \dfrac{1}{b}$ and a vertical asymptote at $X = -\dfrac{a}{b}$. (See Section 6.3 for a description of asymptotes.)

Sample PROC REG **program for the linearized model:**

```
PROC REG DATA = FIRST;
  MODEL Y1 = X1;
  *NOTE X1 = 1/X   and ;
  *     Y1 = 1/Y   must be created in a previous data step;
RUN;
```

Sample PROC NLIN **program for the functional form:**

```
PROC NLIN DATA = FIRST;
  PARAMETERS A = 4.0 B = 0.5;
  ABX = 1/(A + B*X);
  XABX = X*ABX;
  XABXSQ = XABX*ABX;
  MODEL Y = XABX;
  DER.A = -1*XABXSQ;
  DER.B = -X*XABXSQ;
RUN;
```

Functional form: $Y = \dfrac{X}{a + bX}$

b = 0.5 vary a

a = 2

a = -2

a = 4 vary b X > 0

b = 0.1

b = 0.5
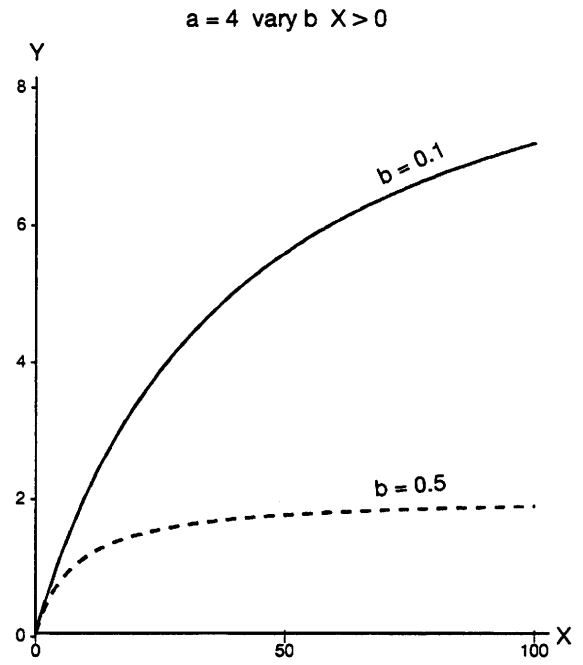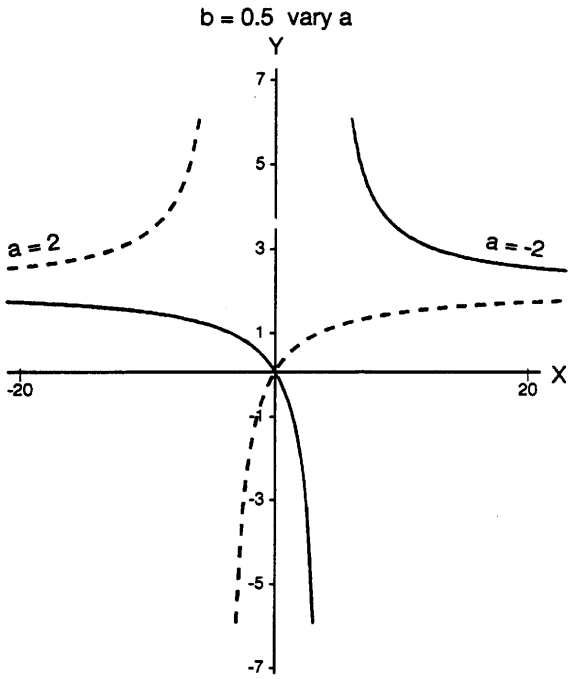
4.2.2 Second degree inverse polynomial: inverse quadratic

Functional form: $Y = \dfrac{X}{a + bX + cX^2}$ $a + bX + cX^2 \neq 0$

Derivatives: $\dfrac{dY}{dX} = \dfrac{a - cX^2}{(a + bX + cX^2)^2}$

$\dfrac{\partial Y}{\partial a} = \dfrac{-X}{(a + bX + cX^2)^2}$ $\dfrac{\partial Y}{\partial b} = \dfrac{-X^2}{(a + bX + cX^2)^2}$

$\dfrac{\partial Y}{\partial c} = \dfrac{-X^3}{(a + bX + cX^2)^2}$

Linearized models and parameters: $\dfrac{1}{Y} = \dfrac{b_0}{X} + b_1 + b_2 X$

$a = b_0$ $b = b_1$ $c = b_2$

Description: The inverse quadratic function has a shape similar to a parabola. It rises quickly for small values of X, and then falls gradually as X increases further. Parameters **a** and **c** describe the rising and falling extrema of the curve, respectively; for small X, $Y \approx X/a$, and for large X, $Y \approx 1/(cX)$. The extrema occur at $X = \pm\sqrt{a/c}$. Parameter **b** controls the height of the maximum. Ratkowsky (1990, p.3) calls this the Hailwood and Horrobin Model.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA = SECOND;
  MODEL Y1 = X1 X;
  *NOTE  X1 = 1/X   and;
  *      Y1 = 1/Y   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA = SECOND;
  PARAMETERS A = 2.0 B=0.5 C = 0.5;
  ABCX = 1/(A + B*X + C*X**2);
  ABCX2 = ABCX**2;
  MODEL Y = X*ABCX;
  DER.A = -X*ABCX2;
  DER.B = -X**2*ABCX2;
  DER.C = -X**3*ABCX2;
RUN;
```

Functional form: $Y = \dfrac{X}{a + bX + cX^2}$

b = 2 c = 0.5 vary a
X > 0
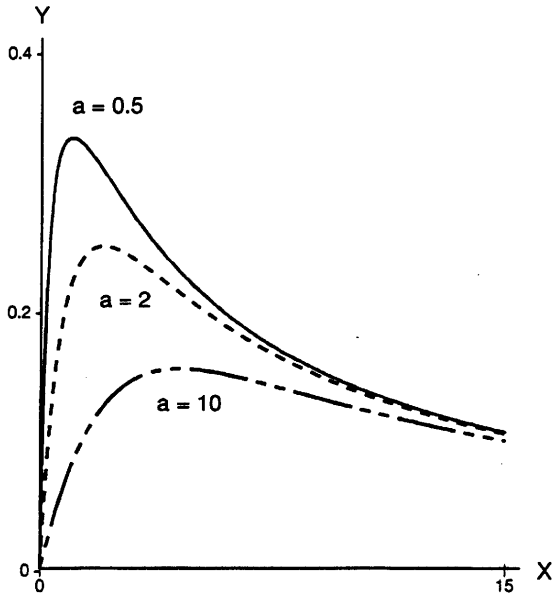
a = 2 c = 0.5 vary b
X < 0
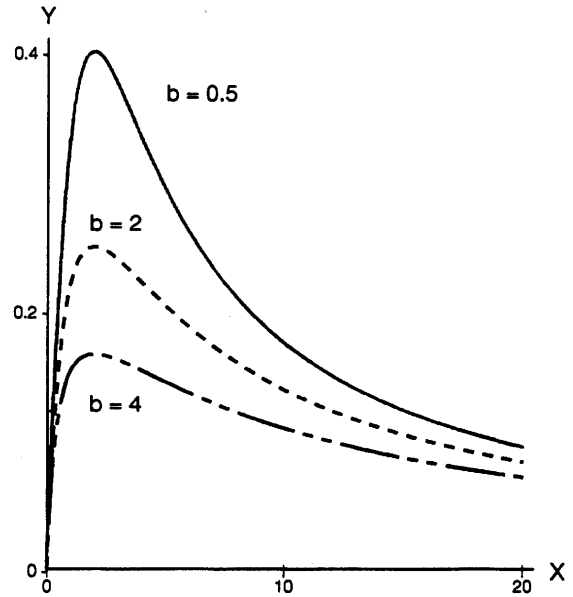
a = 2 b = 0.5 vary c
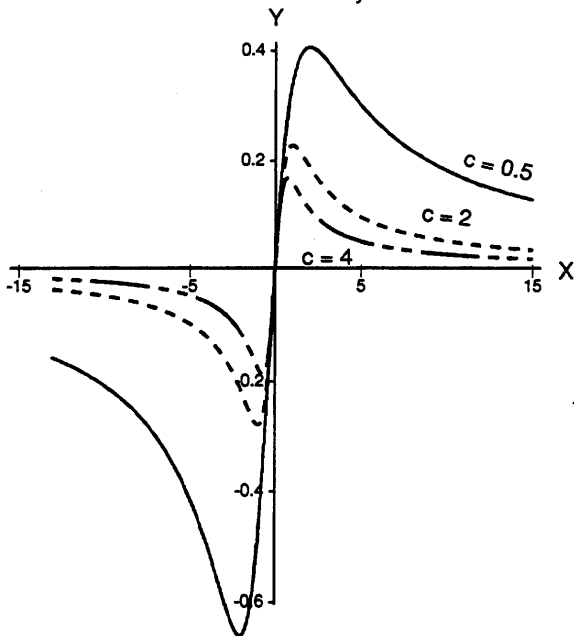
4.2.3 Third degree inverse polynomial: inverse cubic

Functional form:
$$Y = \frac{X}{a + bX + cX^2 + dX^3} \qquad a + bX + cX^2 + dX^3 \neq 0$$

Derivatives:
$$\frac{dY}{dX} = \frac{a - cX^2 - 2dX^3}{(a + bX + cX^2 + dX^3)^2}$$

$$\frac{\partial Y}{\partial a} = \frac{-X}{(a + bX + cX^2 + dX^3)^2}$$

$$\frac{\partial Y}{\partial b} = \frac{-X^2}{(a + bX + cX^2 + dX^3)^2}$$

$$\frac{\partial Y}{\partial c} = \frac{-X^3}{(a + bX + cX^2 + dX^3)^2}$$

$$\frac{\partial Y}{\partial d} = \frac{-X^4}{(a + bX + cX^2 + dX^3)^2}$$

Linearized model and parameters:
$$\frac{1}{Y} = \frac{b_0}{X} + b_1 + b_2 X + b_3 X^2$$

$$a = b_0 \qquad\qquad b = b_1 \qquad\qquad c = b_2 \qquad\qquad d = b_3$$

Description: The shape of the curve, the location and the value of the extrema are controlled by all four parameters.

Sample `PROC REG` program for the linearized model:

```
PROC REG DATA = THIRD;
  MODEL Y1 = X1 X X2;
  * Note:  X1 = 1/X, X2 = X*X and Y1 = 1/Y;
  *        must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` program for the functional form:

```
PROC NLIN DATA = THIRD;
  PARAMETERS A=4.0 B=0.5 C=0.5 D=2.0;
  ABCDX = 1/(A + B*X +C*X**2 D*X**3);
  ABCDX2 = ABCDX**2;
  MODEL Y = X*ABCDX;
  DER.A = -X*ABCDX2;
  DER.B = -X**2*ABCDX2;
  DER.C = -X**3*ABCDX2;
  DER.D = -X**4*ABCDX2;
RUN;
```

Functional form: $\quad Y = \dfrac{X}{a + bX + cX^2 + dX^3} \qquad X > 0$



b = 2  c = 0.5 d = 2  vary a



a = 0.5  c = 0.5  d = 2  vary b



a = 0.1  b = 2  d = 2  vary c



a = 0.1  b = 2  c = 0.5  vary d
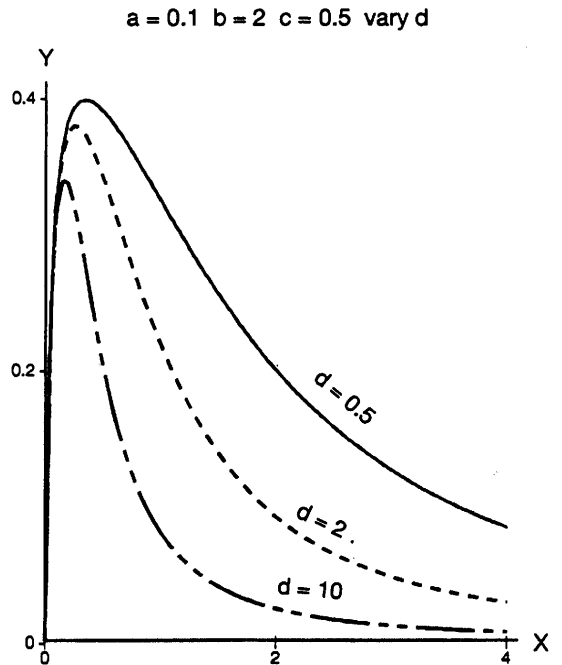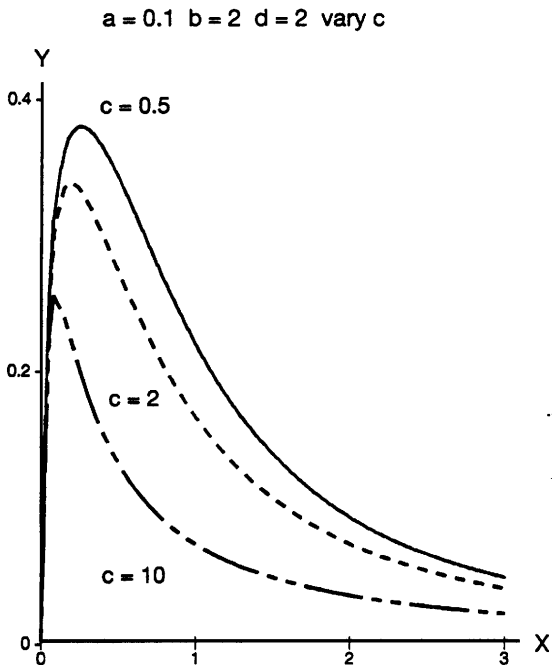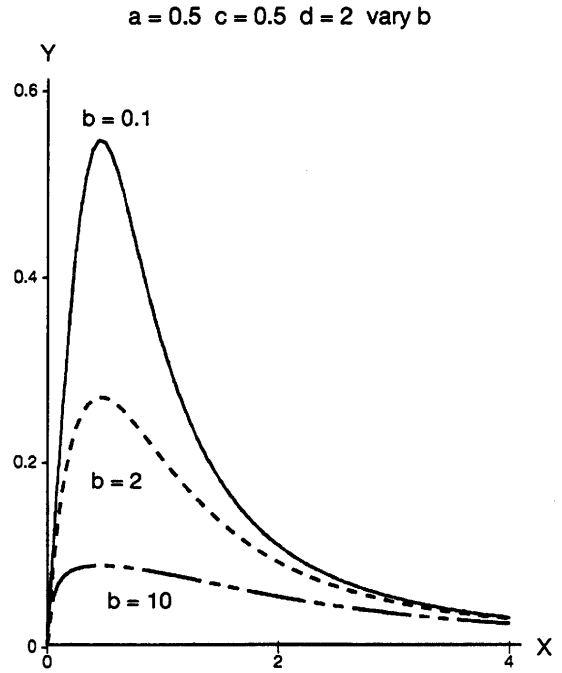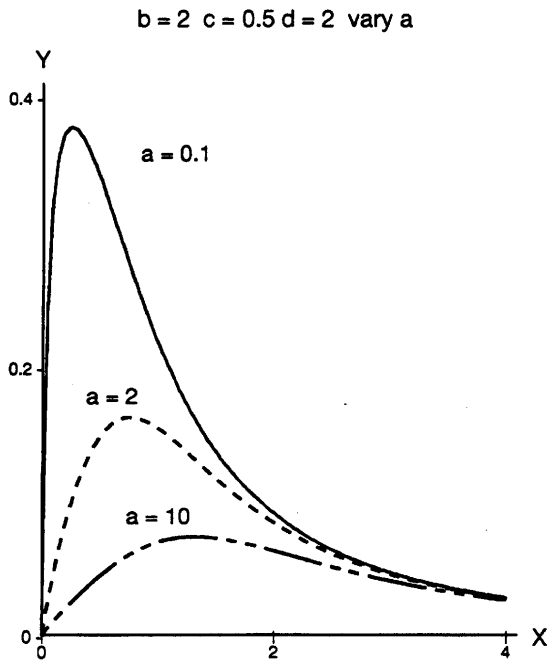
### 4.2.4  Rational function

**Functional form:**  $Y = \dfrac{a}{1 + b/X} + \dfrac{c}{1 + d/X}$      $X \neq -b$ and $X \neq -d$

**Derivatives:**  $\dfrac{dY}{dX} = \dfrac{ab}{(1 + b/X)^2 X^2} + \dfrac{cd}{(1 + d/X)^2 X^2} = \dfrac{ab}{(X + b)^2} + \dfrac{cd}{(X + d)^2}$

$\dfrac{\partial Y}{\partial a} = \dfrac{1}{1 + b/X}$          $\dfrac{\partial Y}{\partial b} = \dfrac{-a}{X(1 + b/X)^2}$

$\dfrac{\partial Y}{\partial c} = \dfrac{1}{1 + d/X}$          $\dfrac{\partial Y}{\partial d} = \dfrac{-c}{X(1 + d/X)^2}$

**Linearized model and parameters:** This function can not be linearized.

**Description:**  The curve has a horizontal asymptote at $Y = a + c$. Parameters **b** and **d** control the rate at which the curve approaches its asymptote. Note that varying **a** would have the same effect as varying **c**, and varying **b** would have the same effect as varying **d**. Also, the curve is a first degree inverse polynomial (Section 4.2.1), where **c** $= 0$ or **a** $= 0$.

**Sample** PROC NLIN **program for the functional form:**

```
PROC NLIN DATA=RAT;
  PARAMETERS A=1.0 B=1.0 C=1.0 D=1.0;
  DENOMA = 1/(1 + B/X);
  DENOMC = 1/(1 + D/X);
  ADENA = A*DENOMA;
  CDENC = C*DENOMC;
  MODEL Y = ADENA + CDENC;
  DER.A = DENOMA;
  DER.B = -1*ADENA*DENOMA/X;
  DER.C = DENOMC;
  DER.D = -1*CDENC*DENOMC/X;
RUN;
```

Functional form: $\quad Y = \dfrac{a}{1 + b/X} + \dfrac{c}{1 + d/X} \qquad X > 0$

b = c = d = 1 vary a

a = 2

a = 1

a = 0.5

a = c = d = 1 vary b

b = 0.1

b = 1

b = 10
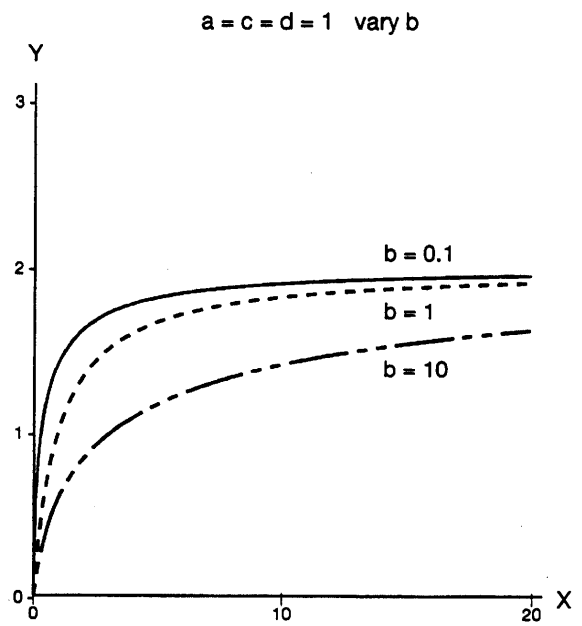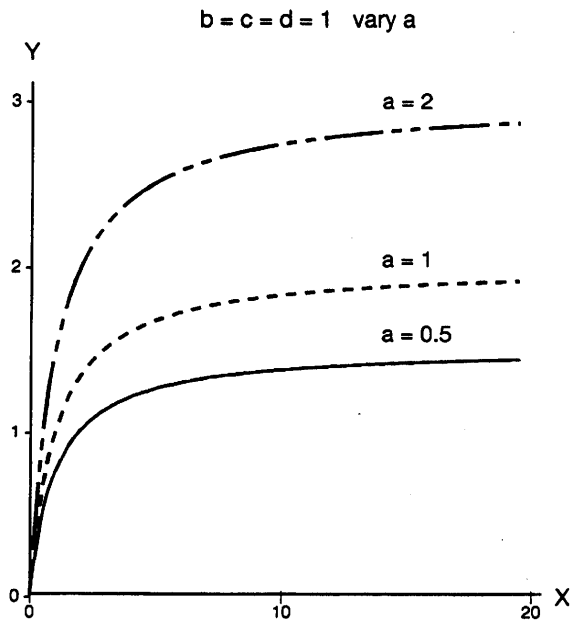
4.2.5 Mixed type function

Functional form: $Y = \dfrac{a}{X} + bX + c$ $X \neq 0$

Derivatives: $\dfrac{dY}{dX} = \dfrac{-a}{X^2} + b$

$$\frac{\partial Y}{\partial a} = \frac{1}{X} \qquad\qquad \frac{\partial Y}{\partial b} = X \qquad\qquad \frac{\partial Y}{\partial c} = 1$$

Linearized model and parameters: $Y = b_0 + b_1 X + b_2 X^{-1}$

$$a = b_2 \qquad\qquad b = b_1 \qquad\qquad c = b_0$$

Description: By finding a common denominator for the right-hand side, this functional form can also be written as:

$$Y = \frac{a + bX^2 + cX}{X}$$

Notice that this is the reciprocal of the second degree inverse polynomial. There is a vertical asymptote at $X = 0$. The parameters **a** and **b** control the shape of the curve while **c** shifts the curve up and down the Y-axis.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA = ABC;
  MODEL Y = X X1;
   *NOTE  X1 = 1/X   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA = ABC;
  PARAMETERS A=4.0 B=5.0 C=10.0;
  X1 = 1/X;
  MODEL Y = A*X1 + B*X + C;
  DER.A = X1;
  DER.B = X;
  DER.C = 1;
RUN;
```

Functional form: $Y = \dfrac{a}{X} + bX + c \qquad X > 0$

b = 5 c = 10 vary a

a = 4 c = 10 vary b

a = 4 b = 5 vary c
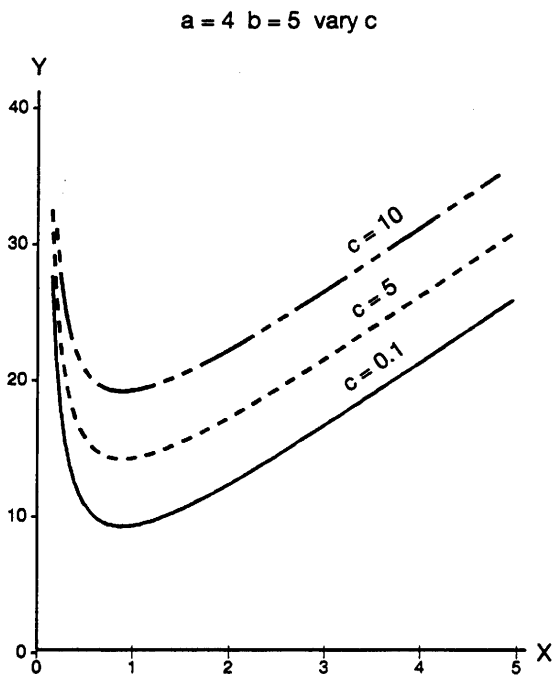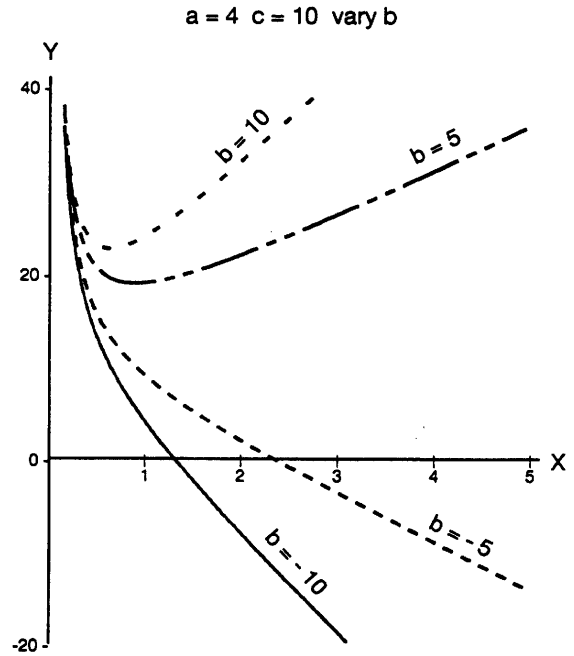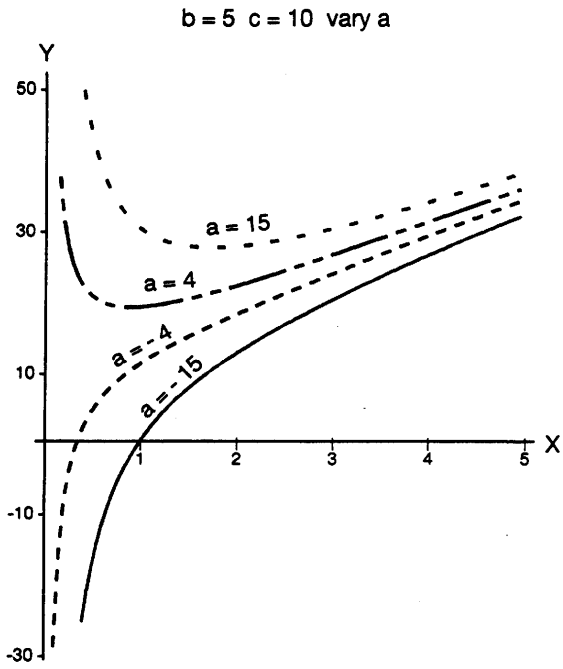
4.3 Exponential Functions

4.3.1 Type I exponential function

Functional form: $\qquad Y = ae^{bX}$ $\qquad\qquad -\infty < X < \infty$

Derivatives: $\qquad \dfrac{dY}{dX} = abe^{bX}$

$\qquad\qquad\qquad \dfrac{\partial Y}{\partial a} = e^{bX}$ $\qquad\qquad \dfrac{\partial Y}{\partial b} = aXe^{bX}$

Linearized model and parameters: $\qquad \ln(Y) = b_0 + b_1 X$

$\qquad\qquad\qquad a = e^{b_0}$ $\qquad\qquad\qquad\qquad b = b_1$

Description: The parameter **a** is the Y-intercept; the parameter **b** is the shape parameter of the curve. There is a horizontal asymptote at $Y = 0$. When $b < 0$, the curve approaches the asymptote as X approaches infinity; when **b** > 0, the curve approaches the asymptote as X approaches negative infinity. See Daniel and Woods (1980, Chapter 3) and Ratkowsky (1990, p. 86) for descriptions of the exponential function.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=EXPON;
  MODEL Y1 = X;
  ****  Y1 = LOG(Y)   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=EXPON;
  PARAMETERS A=5.0 B=2.0;
  EBX = EXP(B*X);
  MODEL Y = A*EBX;
  DER.A = EBX;
  DER.B = A*X*EBX;
RUN;
```

Functional form: $Y = ae^{bX}$

b = - 2 vary a

a = 5

a = 0.5

a = -0.5

a = -5

b = 2 vary a

a = 5,

a = 0.5

a = -0.5

a = -5

31
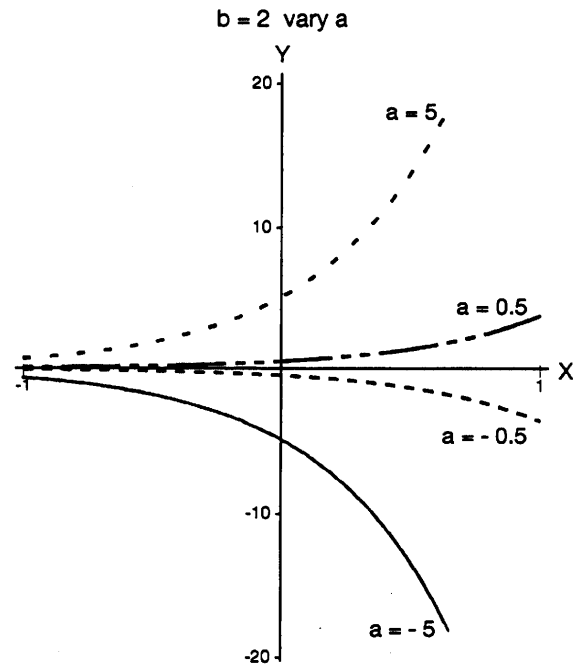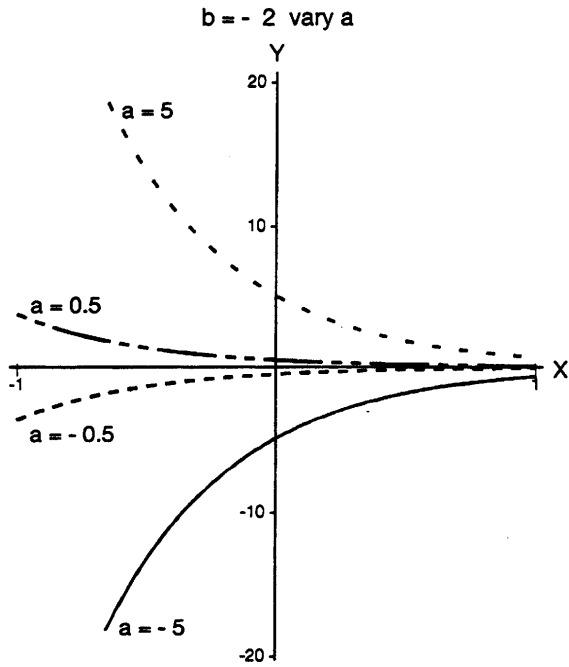
4.3.2 Type II exponential function

Functional form: $Y = e^{a - bX}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = -be^{a - bX}$

$\dfrac{\partial Y}{\partial a} = e^{a - bX}$ $\dfrac{\partial Y}{\partial b} = -Xe^{a - bX}$

Linearized model and parameters: $\ln(y) = b_0 + b_1 X$

$a = b_0$ $b = -b_1$

Description: This is a more modern form of the exponential function in Section 4.3.1. Ratkowsky (1990) describes this as a "close-to-linear" model because the estimated parameters possess the characteristics of those of a linear model — that is, the estimates are close to being unbiased, normally distributed, and having minimum variance. Since the parameter estimates have desirable statistical properties, this form of the exponential function is also used in the logistic curve and the Gompertz curve.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=EXPON;
  MODEL Y1 = X;
  ****  Y1 = LOG(Y)   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=EXPON;
  PARAMETERS A=5.0 B=2.0;
  EABX = EXP(A - B*X);
  MODEL Y = EABX;
  DER.A = EABX;
  DER.B = -X*EABX;
RUN;
```

Functional form: $Y = e^{a - bX}$

b = 2 vary a

a = 2
a = 0.5
a = -0.5
a = -2

b = -2 vary a

a = 2
a = 0.5
a = -0.5
a = -2
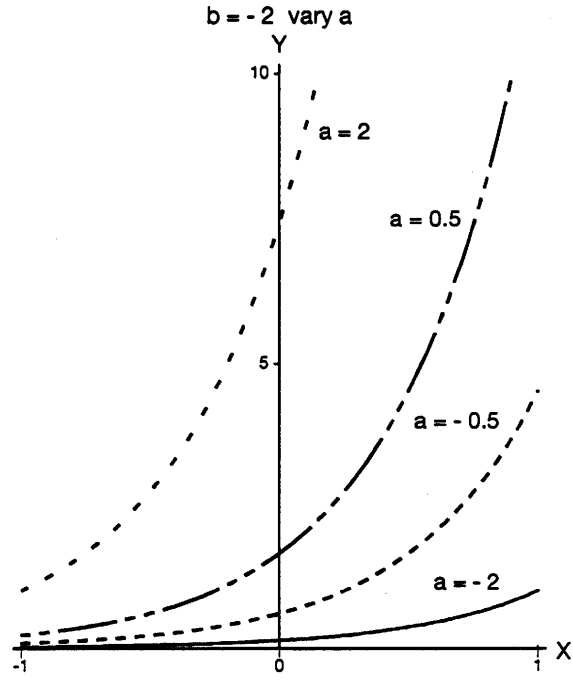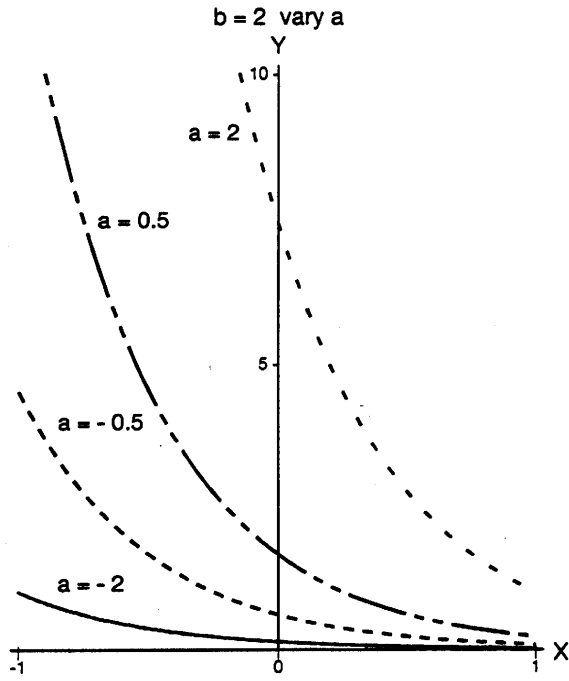
4.3.3 Type III exponential function

Functional form: $Y = ae^{b/X}$ $X \neq 0$

Derivatives: $\dfrac{dY}{dX} = \dfrac{-abe^{b/X}}{X^2}$

$\dfrac{\partial Y}{\partial a} = e^{b/X}$ $\dfrac{\partial Y}{\partial b} = \dfrac{a}{X}e^{b/X}$

Linearized model and parameters: $\ln(Y) = b_0 + \dfrac{b_1}{X}$

$a = e^{b_0}$ $b = b_1$

Description: In this functional form, X is non-linear in the exponent. This function has no maximum or minimum. It has a vertical asymptote at $X = 0$ and a horizontal asymptote at $Y = a$. The shape of the curve is concave up for **a** > 0 and **b** > 0, and concave down for **a** < 0 and **b** < 0. See Daniel and Woods (1980, Chapter 3) for another description of this exponential function.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=AEBX;
  MODEL Y1 = X1;
  ****  X1 = 1/X   must be created in a previous data step;
  ****  Y1 = LOG(Y)
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=EXPON;
  PARAMETERS A=5.0 B=2.0;
  EBX = EXP(B/X);
  MODEL Y = A*EBX;
  DER.A = EBX;
  DER.B = A*EBX/X;
RUN;
```

Functional form: $Y = ae^{b/X}$ $X > 0$
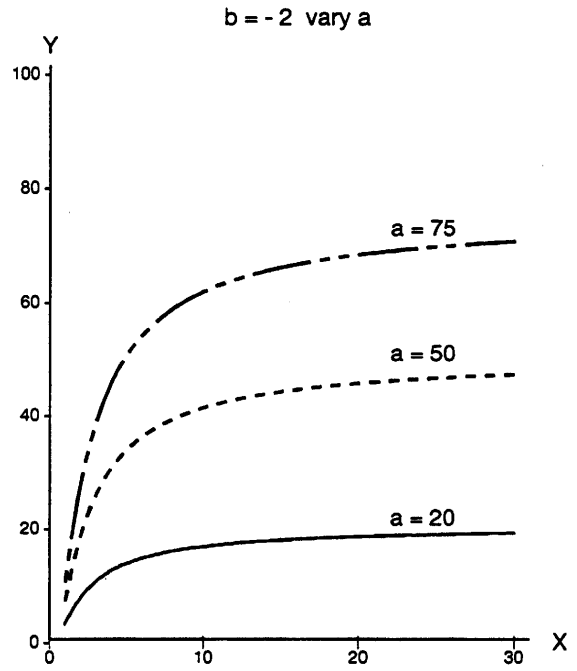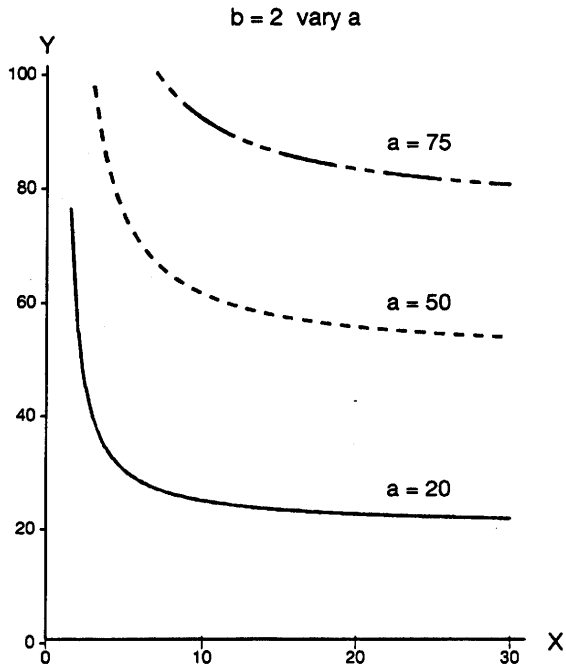
4.3.4 Type IV exponential function

Functional form: $Y = ab^X = ae^{X\ln(b)}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = ab^X \ln(b),\ b > 0$

$\dfrac{\partial Y}{\partial a} = b^X$ $\dfrac{\partial Y}{\partial b} = aXb^{X-1}$

Linearized model and parameters: For $b > 0$, $\ln(Y) = b_0 + b_1 X$

$a = \exp(b_0)$ $b = \exp(b_1)$

Description: This functional form is a more general form of the Type I exponential function (Section 4.3.1). Note that b^X is not always defined for $b < 0$. For example, the square root ($X = \frac{1}{2}$) of a negative number is undefined. Also, $Y = a$ at $X = 0$.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=MODEXP;
  MODEL Y1 = X;
  ****  Y1 = LOG(Y)   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=MODEXP;
  PARAMETERS A=4.0 B=1.25;
  BX = B**X;
  ABX = A*BX;
  MODEL Y = ABX;
  DER.A = BX;
  DER.B = X*ABX/B;
RUN;
```

36

Functional form: $Y = ab^X$ $X > 0$

b = 1.25 vary a

a = 4 vary b

37
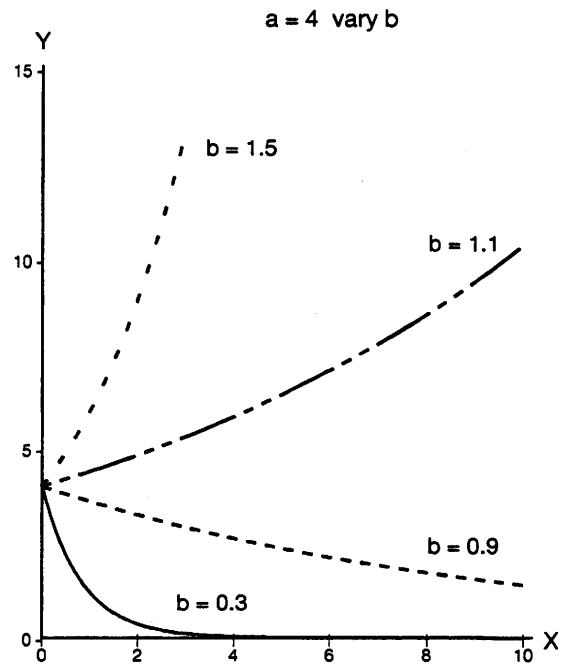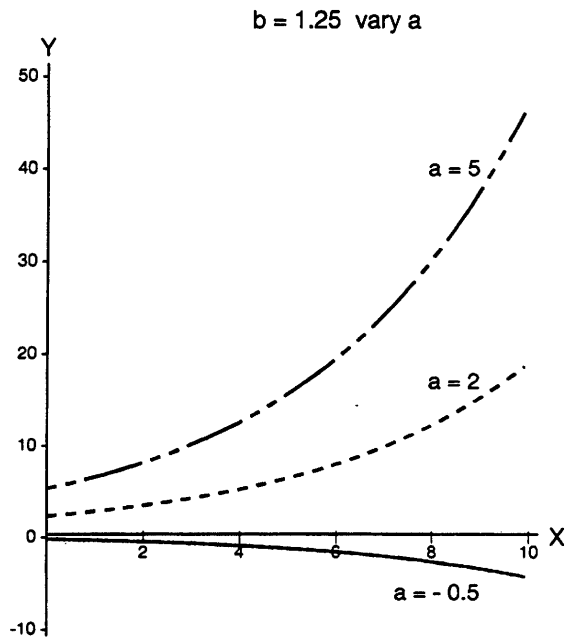
4.3.5 Type V exponential function

Functional form: $Y = ab^{(X-c)^2}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = 2a(X-c)b^{(X-c)^2}\ln(b),\ b > 0$

$\dfrac{\partial Y}{\partial a} = b^{(X-c)^2}$ $\dfrac{\partial Y}{\partial b} = a(X-c)^2 b^{(X-c)^2-1}$

$\dfrac{\partial Y}{\partial c} = -2a(X-c)\ln(b)b^{(X-c)^2}$

Linearized model and parameters: $\ln(Y) = b_0 + b_1 X + b_2 X^2$

$a = \exp\!\left(b_0 - \dfrac{b_1^{\,2}}{4b_2}\right)$ $b = \exp(b_2)$

$c = \dfrac{-b_1}{2b_2}$

Description: This functional form has a quadratic in the exponent. This curve is the same as the Normal distribution in Section 4.8.1 (mean μ and variance σ^2) when $a = \dfrac{1}{\sqrt{2\pi}\,\sigma}$, $b = e^{-1/(2\sigma^2)}$, and $c = \mu$. Parameter **a** controls the height (Y-value) of the maximum or minimum; parameter **c** controls the location (X-value) of the maximum or minimum. The curve has a minimum for **a** > 0 and **b** > 1, and a maximum for **a** > 0 and **b** < 1; it is straight line (i.e., $Y = a$) for **b** $= 1$.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=MODEXP;
  MODEL Y1 = X X2;
  ****  X2 = X*X      must be created in a previous data step;
  ****  Y1 = LOG(Y)
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=MODEXP;
  PARAMETERS A=2.0 B=0.1 C=1.0;
  XC = (X-C)**2;
  BXC = B**XC;
  ABXC = A*BXC;
  MODEL Y = ABXC;
  DER.A = BXC;
  DER.B = ABXC*XC/B;
  DER.C = -2*ABXC*(X-C)*LOG(B);
RUN;
```

Functional form: $Y = ab^{(X-c)^2}$

b = 0.2 c = 1 vary a

a = 2

a = 1

a = 0.5

a = 2 c = 1 vary b

b = 10

b = 2

b = 1

b = 0.01

b = 0.2

a = 0.5 b = 0.2 vary c

c = 1

c = 2
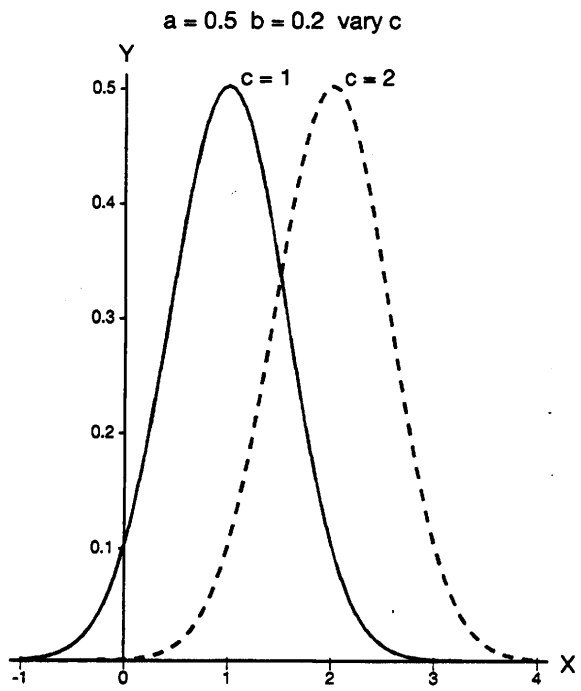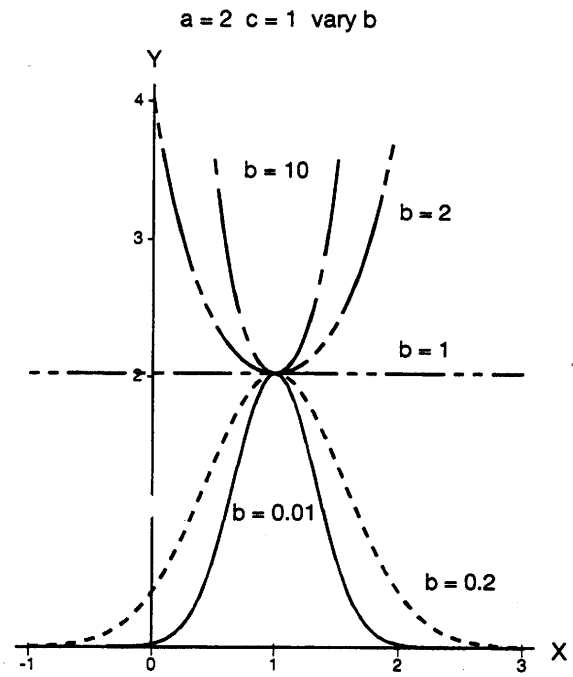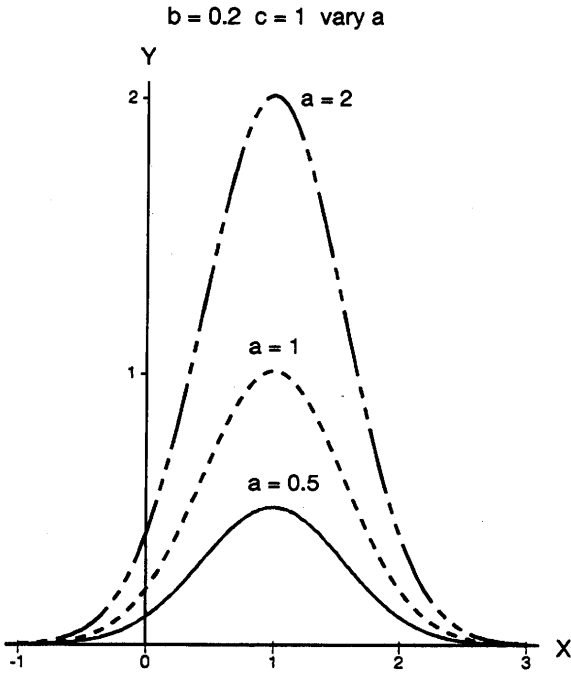
4.3.6 Type VI exponential function

Functional form: $Y = \dfrac{a(e^{-bX} - e^{-aX})}{a - b}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = \dfrac{a(ae^{-aX} - be^{-bX})}{a - b}$

$\dfrac{\partial Y}{\partial a} = \dfrac{1}{a-b}(e^{-aX}(aX - 1) + e^{-bX}) - \dfrac{a(e^{-bX} - e^{-aX})}{(a-b)^2}$

$\dfrac{\partial Y}{\partial b} = \dfrac{a(e^{-aX} - e^{-bX})}{(a-b)^2} - \dfrac{aXe^{-bX}}{a-b}$

Linearized model and parameters: This function can not be linearized.

Description: The parameters **a** and **b** work together to control the height of the maximum. Parameter **a** also shifts the maximum along the X-axis. See Daniel and Woods (1980, Chapter 3) for another description of this exponential function.

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=NLIN;
  PARAMETERS A=0.3 B=0.4;
  EAX = EXP(-A*X);
  EBX = EXP(-B*X);
  EAEB = EBX-EAX;
  AB = 1/(A-B);
  AEB = A*EAEB*AB
  MODEL Y = AEB;
  DER.A = AB*(EAX*(A*X-1) + EBX) - EAEB*A*AB*AB;
  DER.B = AEB*AB - A*X*EBX*AB;
RUN;
```

Functional form: $\qquad Y = \dfrac{a(e^{-bX} - e^{-aX})}{a - b} \qquad X > 0$

b = 0.4 vary a

a = 10

a = 2

a = 0.3

a = 0.3 vary b

b = - 0.5

b = 0.5

b = 3
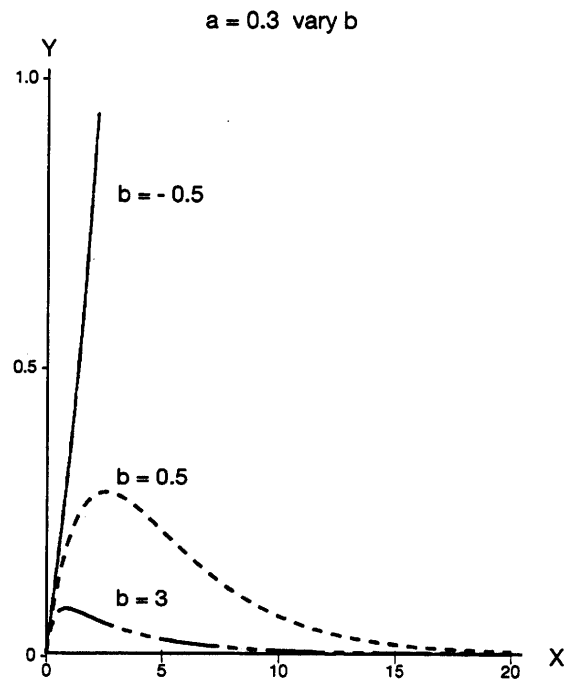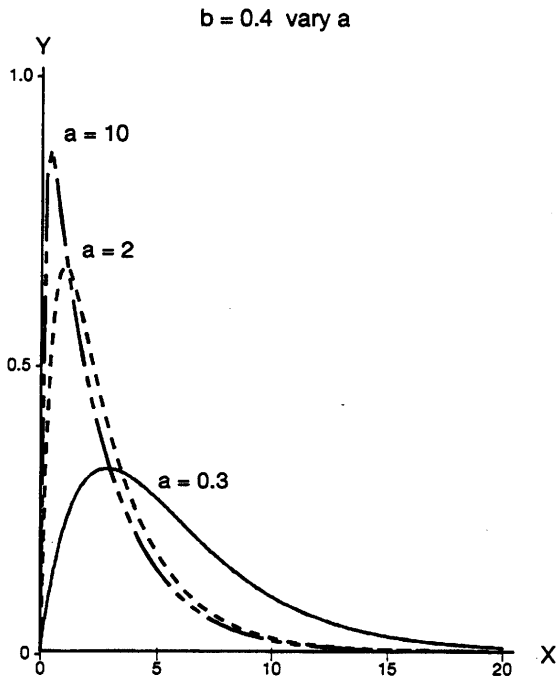
4.3.7 Schumacher's equation

Functional form: $\quad Y = e^{a + b/X} \qquad\qquad -\infty < X < \infty$

Derivatives: $\quad \dfrac{dY}{dX} = -\dfrac{b}{X^2}\, e^{a + b/X}$

$$\frac{\partial Y}{\partial a} = e^{a + b/X} \qquad\qquad \frac{\partial Y}{\partial b} = \frac{1}{X}\, e^{a + b/X}$$

Linearized model and parameters: $\quad \ln(Y) = b_0 + \dfrac{b_1}{X}$

$$a = b_0 \qquad\qquad b = b_1$$

Description: This equation was developed by F.X. Schumacher (1939) for modelling volume-yield of an even-aged timber stand. It is usually expressed in the linearized form. This equation assumes that the rate of change of Y is inversely proportional to X. In growth and yield terms, this assumption translates as decline in growth rate with age, a characteristic of an even-aged timber stand. Parameter **a** is the logarithm of the maximum Y-value as X approaches infinity. See Schumacher (1939) for more description and an application of this equation.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=SCHUMACH;
  MODEL Y1 = X1;
  ****  Y1 = LOG(Y)  and  X1 = 1/X ;
  ****  must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA = SCHUMACH;
  PARAMETER A = 0.1 B = 0.1;
  EX = EXP(A+B/X);
  MODEL Y = EX;
  DER.A = EX;
  DER.B = EX/X;
RUN;
```

Functional form: $Y = e^{a + b/X}$

b = 1 vary a   X < 0

a = 1

a = 0

a = -1

b = 1 vary a   X > 0

a = 1

a = 0

a = -1

a = 2 vary b   X < 0

b = -1

b = 0

b = 1

a = 2 vary b   X > 0

b = 1

b = 0

b = -1

43
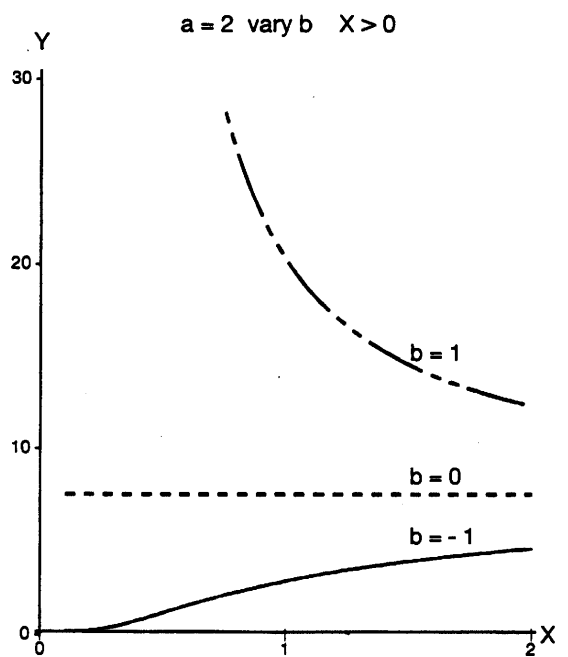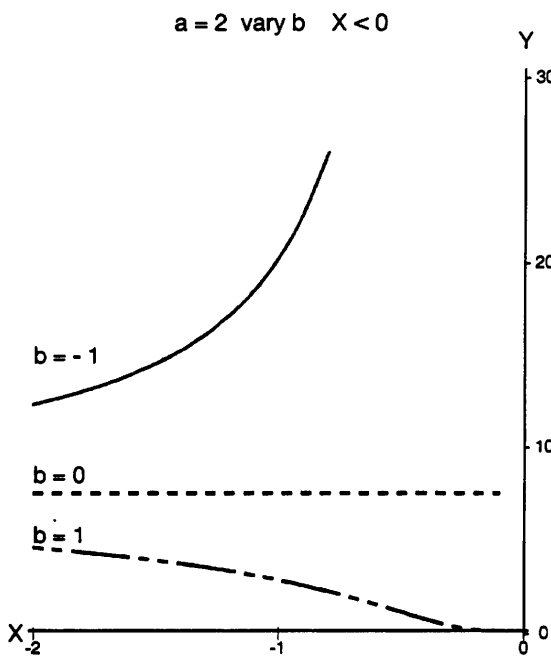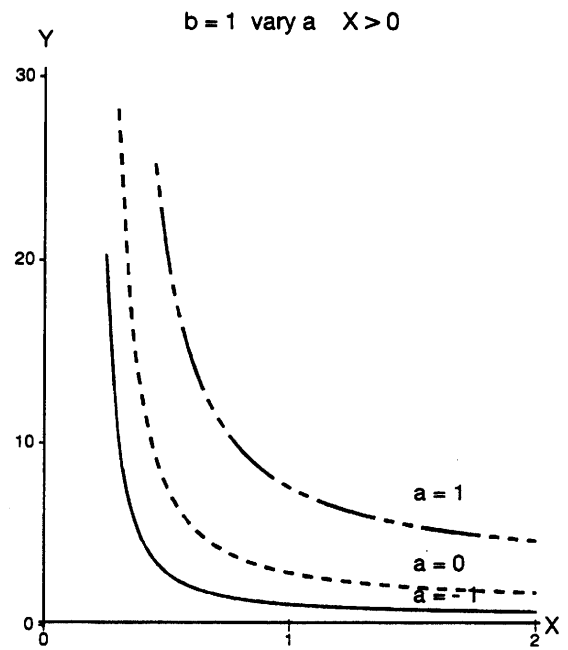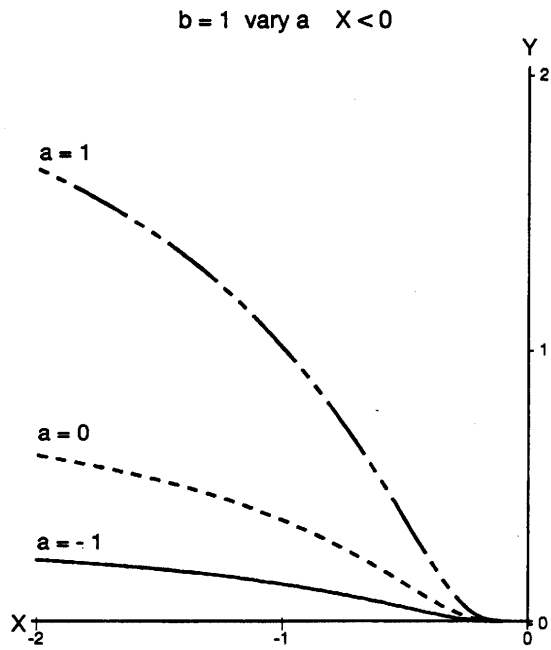
4.3.8 Modified Weibull equation

Functional form: $Y = 1 - e^{-aX^b}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = abX^{b-1}e^{-aX^b}$

$\dfrac{\partial Y}{\partial a} = X^b e^{-aX^b}$ $\dfrac{\partial Y}{\partial b} = aX^b e^{-aX^b} \ln(X)$

Linearized model and parameters: $\ln\left[-\dfrac{1}{a}\ln(Y)\right] = b_1 X$

$b = b_1$

This linearized model requires that **a** be known.

Description: This model is derived from the Weibull Probability function, originally developed by Weibull (1951). It is very flexible, and especially useful for fitting growth and yield data. Parameters **a** and **b** are scale and shape parameters, respectively. The curve passes through the origin ($X = 0$, $Y = 0$), and approaches a maximum asymptotically when X approaches infinity. It is similar to the Type I exponential function (Section 4.3.1) when **b** = 1. See Yang et al. (1978) for more description of the modified Weibull equation and examples. See Leech and Ferguson (1981) for a comparison of this model with other growth models.

Sample `PROC REG` **program for the linearized model with parameter a known:**

```
PROC REG DATA=WEIBULL;
  MODEL Y1 = X / NOINT; ** A no intercept model is requested;
  **** Y1 = LOG(-LOG(1-Y)/A)   must be created in a previous
                               data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=WEIBULL;
  PARAMETERS A=10 B=2.0;
  EABX = EXP(-A*X**B);
  MODEL Y = 1-EABX;
  DER.A = EXAB*X**B;
  DER.B = LOG(X)*A*EABX*X**B
RUN;
```

Functional form: $Y = 1 - e^{-aX^b}$

b = 3 vary a

a = 10

a = 1

a = 5

b = 3 vary a < 0

a = -10

a = -1

a = -5

a = 5 vary b : 0 < b < 1

b = 0.3

b = 0.5

b = 0.7

a = 5 vary b > 0

b = 1

b = 10

b = 3

b = 7
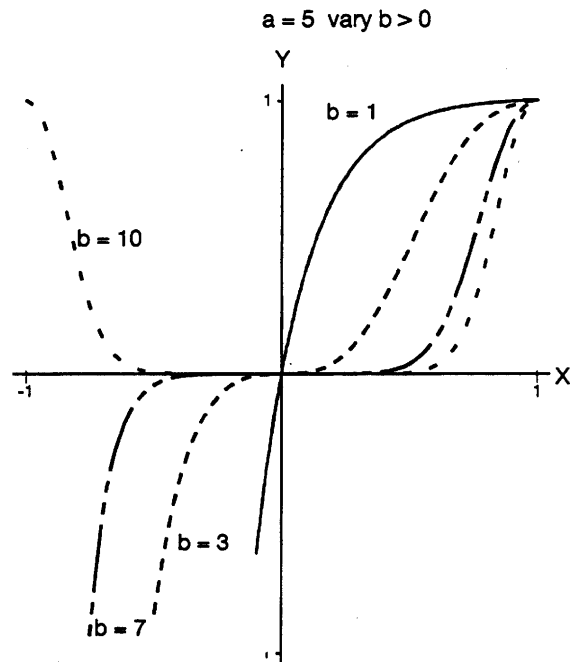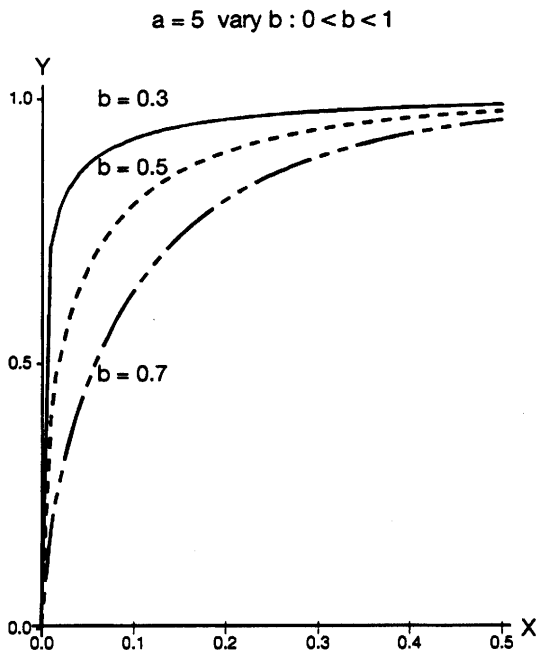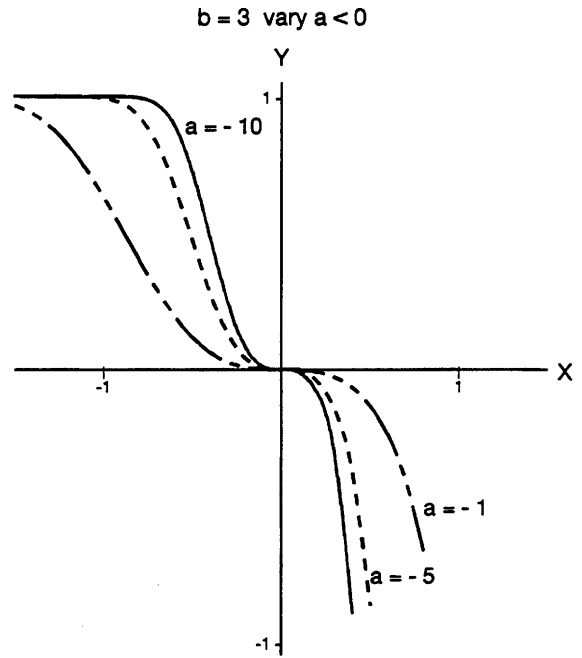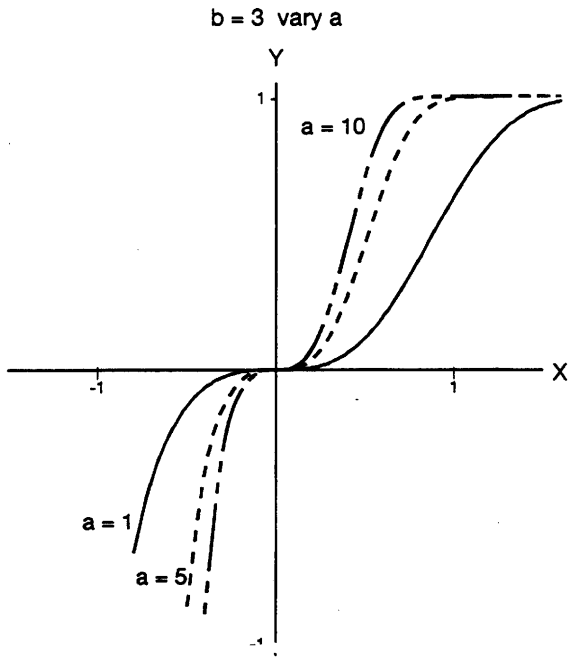
4.3.9 Chapman-Richard's equation

Functional form: $\quad Y = a\left[1 - e^{-bX}\right]^{c} \qquad\qquad -\infty < X < \infty$

Derivatives: $\quad \dfrac{dY}{dX} = abc\, e^{-bX}\left[1 - e^{-bX}\right]^{c-1}$

$$\frac{\partial Y}{\partial a} = \left[1 - e^{-bX}\right]^{c} \qquad\qquad\qquad \frac{\partial Y}{\partial b} = acX e^{-bX}\left[1 - e^{-bX}\right]^{c-1}$$

$$\frac{\partial Y}{\partial c} = a\left[1 - e^{-bX}\right]^{c}\ln\left[1 - e^{-bX}\right]$$

Linearized model and parameters: $\quad \ln\left[1 - \left(\dfrac{Y}{a}\right)^{1/c}\right] = b_1 X$

$$b = -b_1$$

This linearized model requires that parameters **a** and **c** be known.

Description: This model is also known as the Von Bertalanffy equation. It includes a number of common models. It is the **natural growth model** when **c = 1**:

$$Y = a(1 - e^{-bX})$$

It is the **logistic model** when c = −1:

$$Y = \frac{1}{a(1 - e^{-bX})}$$

See Richards (1959), Hunt (1982, Sections 6.2 and 6.5), and Ratkowsky (1990, pp. 109−110, 131) for more descriptions of this function. The following PROC REG step is written for the natural growth model (c = 1).

Sample PROC REG **program for the linearized model with parameter c = 1 and parameter a known:**

```
PROC REG DATA=NATURAL;
  MODEL Y1 = X / NOINT; ** A no intercept model is requested;
  ****  Y1 = LOG(1 - Y/A)  must be created in a previous data step;
RUN;
```

Sample PROC NLIN **program for the functional form:**

```
PROC NLIN DATA=NATURAL;
  PARAMETERS A=8.0 B=2.0 C=0.5;
  EBX = EXP(-B*X);
  EBX1 = 1 - EBX;
  EBXC = (EBX1)**C;
  MODEL Y = A*EBXC;
  DER.A = EBXC;
  DER.B = A*X*C*EBX*EBX1**(C-1);
  DER.C = A*EBXC*LOG(EBX1);
RUN;
```

$$Y = a(1 - e^{-bX})^C \qquad X > 0$$

b = 2 c = 1 vary a

a = 8

a = 2

a = -2

a = 8 c = 1 vary b

b = 2

b = 0.5

b = 0.1

b = -0.1

b = -0.5

a = 8 b = 2 vary c

c = 0.5

c = 1

c = 5
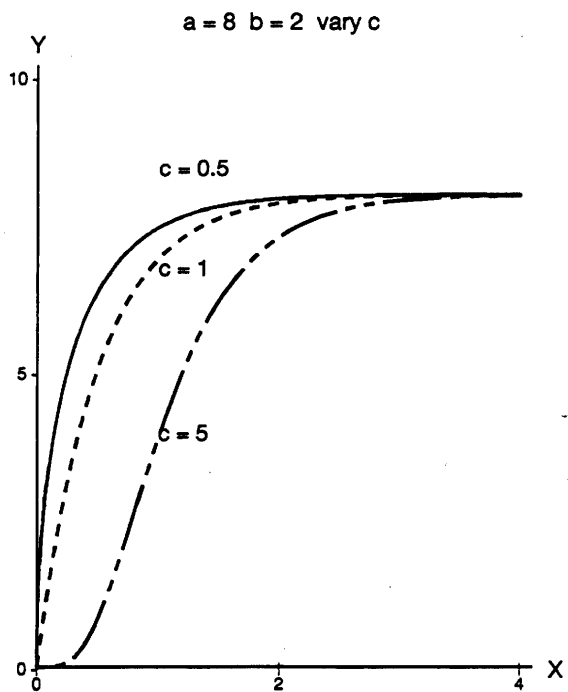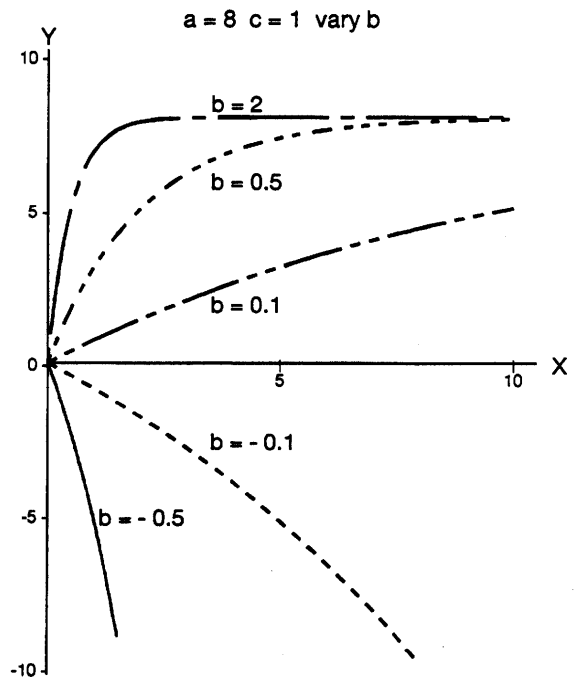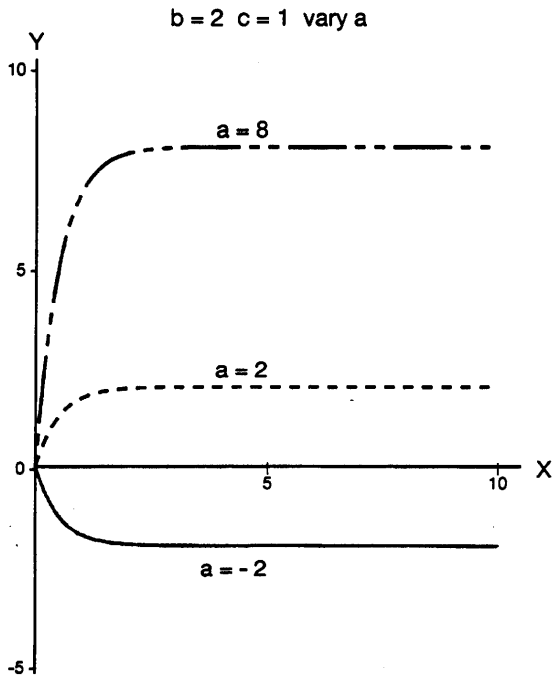
4.3.10 Generalized logistic function

Functional form: $Y = \dfrac{a}{d + e^{b-cX}}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = \dfrac{ace^{b-cX}}{(d + e^{b-cX})^2}$

$\dfrac{\partial Y}{\partial a} = \dfrac{1}{d + e^{b-cX}}$ $\dfrac{\partial Y}{\partial b} = \dfrac{-ae^{b-cX}}{(d + e^{b-cX})^2}$

$\dfrac{\partial Y}{\partial c} = \dfrac{aXe^{b-cX}}{(d + e^{b-cX})^2}$ $\dfrac{\partial Y}{\partial d} = \dfrac{-a}{(d + e^{b-cX})^2}$

Linearized model and parameters: $\ln\left[\dfrac{a}{Y} - d\right] = b_1 + b_2 X$

$b = b_1$ $c = -b_2$

The linearized form of the generalized Logistic function requires that both parameters **a** and **d** be known. Note also that $a/Y - d$ must be positive for the logarithm to be valid.

Description: This is a common function in biometrics applications. It has two horizontal asymptotes: at $Y = 0$ for $X \to \infty$ (off the graph), and at $Y = a/d$ for $X \to -\infty$ (c positive). This function has an inflection point at $X = b/c$. The characteristic shape of the logistic function is called a *sigmoid*. See Sweda (1984), Daniel and Woods (1980, Chapter 3), and Ratkowsky (1990, pp. 128, 137) for other descriptions of this function. Any textbook on modelling growth will also discuss the logistic function.

Sample `PROC REG` program for the linearized model with parameters a and d known:

```
PROC REG DATA=LOGISTIC;
  MODEL Y1 = X;
  **** Y1 = LOG(A/Y - D)   must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` program for the functional form:

```
PROC NLIN DATA=LOGIST;
  PARAMETERS A=2.0 B=2.0 C=2.0 D=2.0;
  DEN = 1/(D + EXP(B-C*X));
  EE  = A*EXP(B-C*X)*DEN**2;
  MODEL Y = A*DEN;
  DER.A = DEN;
  DER.B = -EE;
  DER.C = X*EE;
  DER.D = -A*DEN**2
RUN;
```

Functional Form: $Y = \dfrac{a}{d + e^{b-cX}}$ $X > 0$

b = 2 c = 0.5 d = 2 vary a

a = 5 c = 0.5 d = 2 vary b

a = 5 b = 2 d = 2 vary c

a = 2 b = 2 c = 0.5 vary d

49
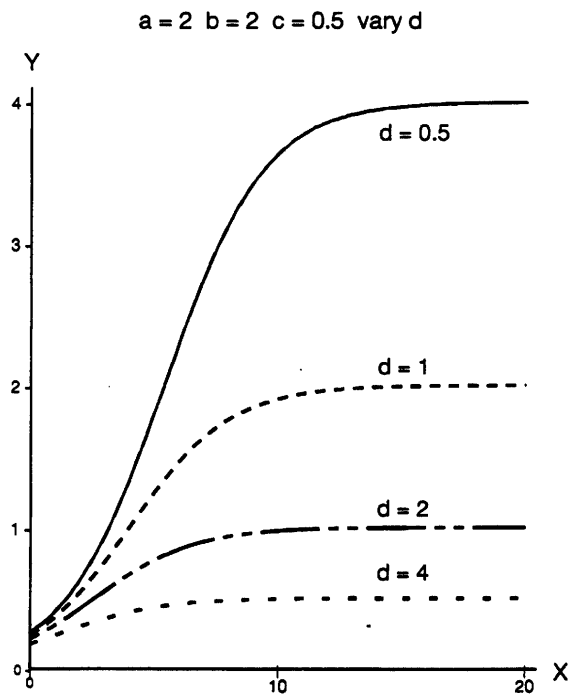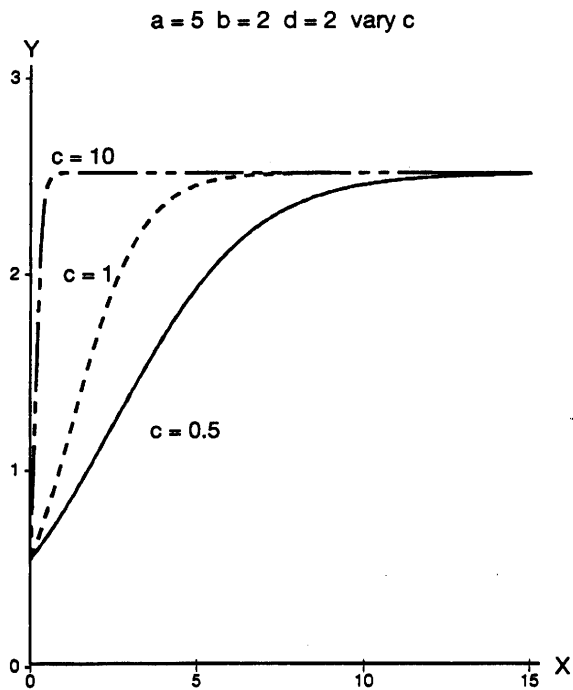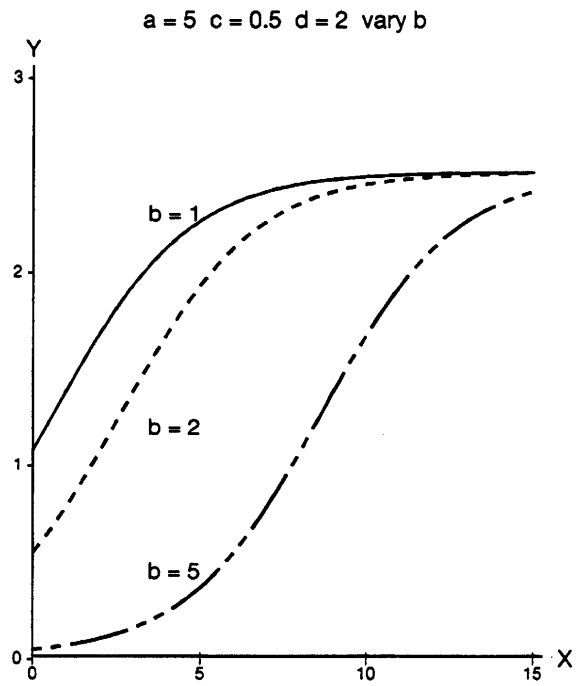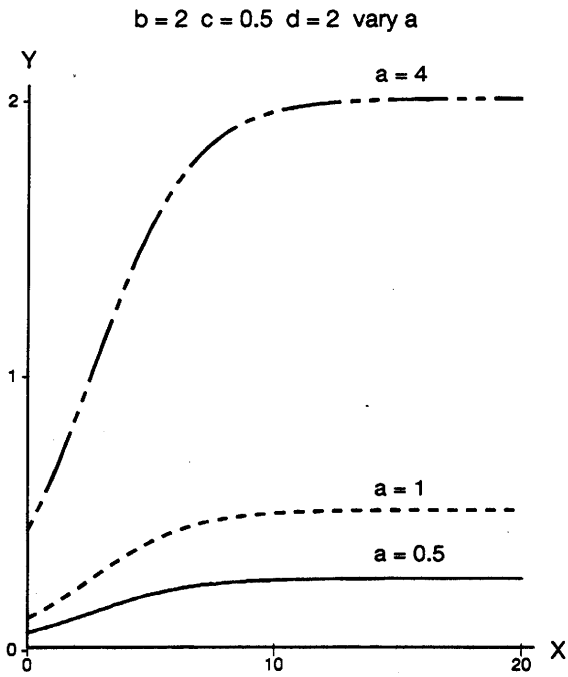
4.3.11 Logistic function

Functional form: $\quad Y = \dfrac{a}{1 + e^{b-cX}} \qquad\qquad -\infty < X < \infty$

Derivatives: $\quad \dfrac{dY}{dX} = \dfrac{ace^{b-cX}}{(1 + e^{b-cX})^2}$

$$\dfrac{\partial Y}{\partial a} = \dfrac{1}{1 + e^{b-cX}} \qquad\qquad \dfrac{\partial Y}{\partial b} = \dfrac{-ae^{b-cX}}{(1 + e^{b-cX})^2}$$

$$\dfrac{\partial Y}{\partial c} = \dfrac{aXe^{b-cX}}{(1 + e^{b-cX})^2}$$

Linearized model and parameters: $\quad \ln\left[\dfrac{a}{y} - 1\right] = b_1 + b_2 X$

$$b = b_1 \qquad\qquad\qquad c = -b_2$$

Description: This is a special case of the generalized logistic function (Section 4.3.10) with **d** = 1. It has the same shape as the generalized function. It is symmetric about the inflection point (X = **b/c**, Y = **a/2**) and has an asymptote at Y = **a**. See Sweda (1984), Daniel and Woods (1980, Chapter 3), Hunt (1982, Section 6.3), and Ratkowsky (1990, pp. 128, 137) for other descriptions of this function. Any text on modeling growth will also discuss the logistic function.

Sample `PROC REG` **program for the linearized model with parameter a known:**

```
PROC REG DATA = LOGISTIC;
  MODEL Y1 = X;
  ****  Y1 = LOG(A/Y - 1)    must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA = LOGISTIC;
  PARAMETERS A = 2.0 B=1.0 C=-0.5;
  E = EXP(B - C*X);
  MODEL Y = A/(1 + E);
  DER.A = 1/(1 + E);
  DER.B = -A*E/(1 + E)**2;
  DER.C = A*E*X/(1 + E)**2;
RUN;
```

Functional form: $Y = \dfrac{a}{1 + e^{b - cX}}$ $\qquad X > 0$

b = 1 c = 0.5 vary a

a = 5 c = 0.5 vary b

a = 5 b = 1 vary c > 0

a = 5 b = 1 vary c < 0
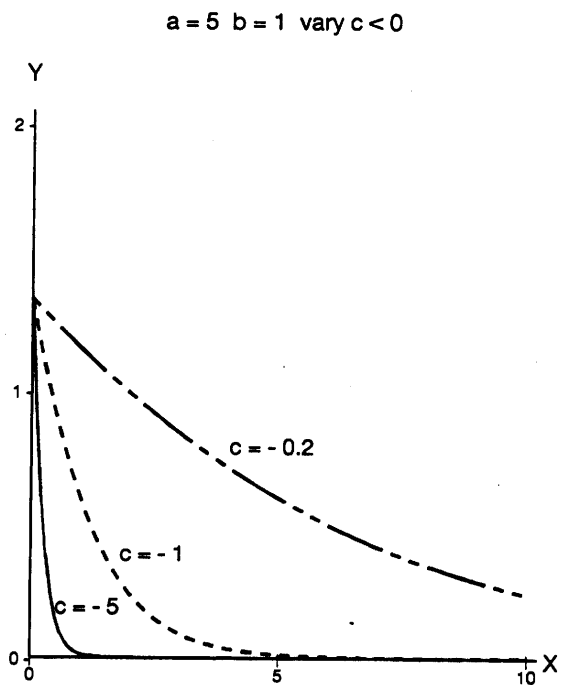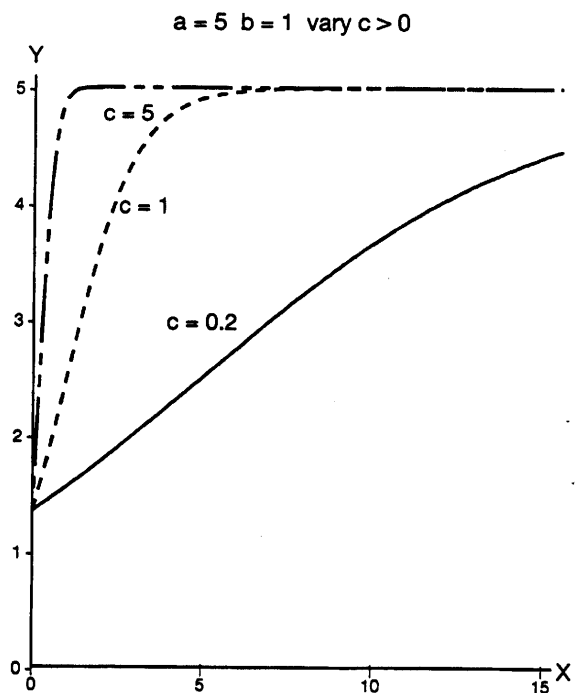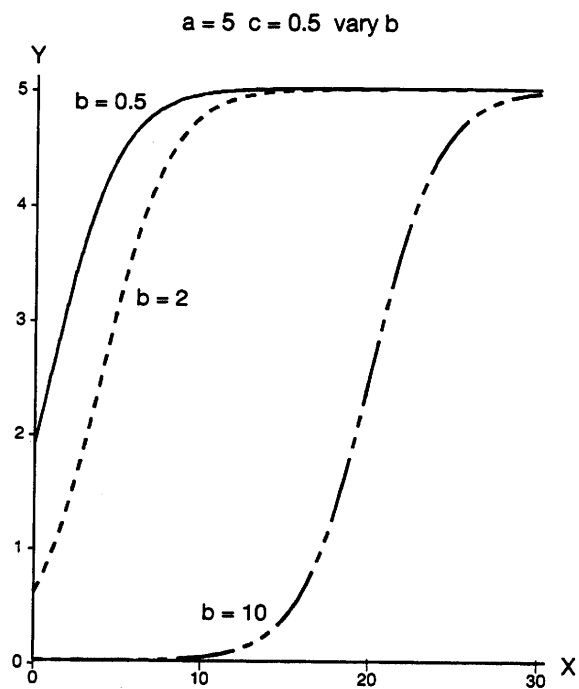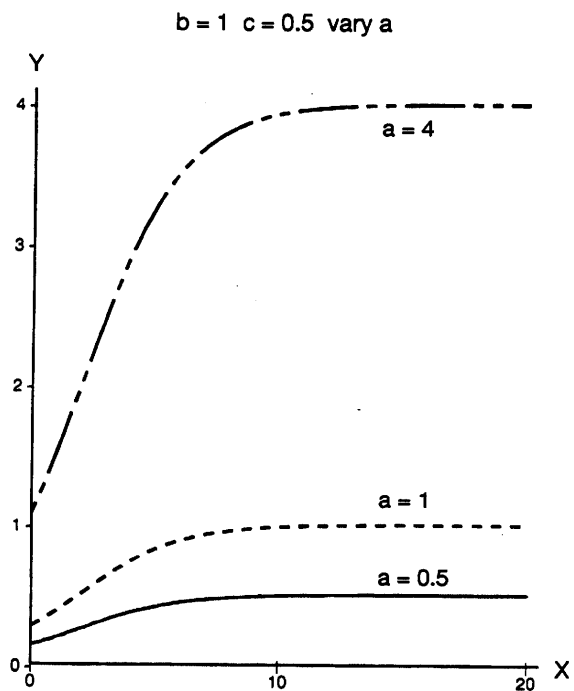
4.3.12 Gompertz function

Functional form: $Y = ae^{-e^{b-cX}}$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = ace^{-e^{b-cX}}e^{b-cX}$

$\dfrac{\partial Y}{\partial a} = e^{-e^{b-cX}}$ $\dfrac{\partial Y}{\partial b} = -ae^{-e^{b-cX}}e^{b-cX}$

$\dfrac{\partial Y}{\partial c} = aXe^{-e^{b-cX}}e^{b-cX}$

Linearized model and parameters: $\ln\left[-\ln\left[\dfrac{Y}{a}\right]\right] = b_1 + b_2 X$

$b = b_1$ $c = -b_2$

Description: This function is named after Benjamin Gompertz who derived it in 1825. Similar to the logistic function, the Gompertz function has asymptotes at $Y = 0$ and $Y = $ **a**. The Gompertz function is not symmetric about its point of inflection. Parameters **a** and **c** control the shape of the curve. Parameter **b** shifts the curve sideways along the X-axis. See Sweda (1984) and Hunt (1982) for more description of the Gompertz function.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=GOMP;
  MODEL Y1 = X;
  **** Y1 = LOG((-LOG(Y/A)) must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program:**

```
PROC NLIN DATA=GOMP;
  PARAMETERS A=6.0 B=2.0 C=1.0;
  BDX = EXP(B-C*X);
  BCBDX = EXP(-BDX);
  BBB = BDX*BCBDX*A;
  MODEL Y = BCBDX*A;
  DER.A = BCBDX;
  DER.B = -BBB;
  DER.C = X*BBB;
RUN;
```

Functional form: $Y = ae^{-e^{b-cX}}$ $\quad$ $X > 0$

b = 2 c = 1 vary a

a = 10

a = 6

a = 2

a = 2 c = 1 vary b

b = 5

b = 2

b = 0.5

a = 2 b = 2 vary c

c = 5

c = 1

c = 0.5

4.3.13 Schnute's equation

Functional form: $\qquad Y = \left[c^b + (d^b - c^b) \dfrac{1 - e^{-a(X - x1)}}{1 - e^{-a(x2 - x1)}} \right]^{1/b} \qquad\qquad -\infty < X < \infty$

Derivatives: Not available because the partial derivatives are not easily simplified to manageable forms.

Linearized model and parameters: This function can not be linearized.

Description: This four-parameter model was developed by J. Schnute (1981) for fisheries research. It is equivalent to the Chapman-Richards equation (Section 4.3.9), the logistic function (Section 4.3.11), and the Gompertz function (Section 4.3.12) in simple cases. It is more flexible than the Chapman-Richards equation because it does not impose an asymptotic trend. It can assume a wide range of characteristic shapes that describe asymptotic and non-asymptotic trends. Parameters **a** and **b** implicitly define the shape of the curve. Parameters **c** and **d** are Y-values at $X = x1$ and $X = x2$ respectively; **x1** and **x2** are the first and last values of the X-interval on which the function is to be fitted. See Bredenkamp and Gregoire (1988) for an example and more description of Schnute's equation.

Sample `PROC NLIN` **program:**

```
PROC NLIN DATA=SCHNUTE;
  PARAMETERS A=0.1 B=0.5 C=5 D=20;

/*** use the Y-values corresponding to the first and last
**** X-values as the starting values for C and D   ***/

  X1 = 0; *** X1 and X2 are the first and last values;
  X2 = 20; *** of the X-interval;
  EX = (1-EXP(-A*(X-X1)))/(1-EXP(-A*(X2-X1)));
  MODEL Y = (C**B + (D**B - C**B)*EX)**(1/B);
RUN;
```

Functional form: $$Y = \left[c^b + (d^b - c^b) \frac{1 - e^{-a(X - x1)}}{1 - e^{-a(x2 - x1)}} \right]^{1/b} \qquad X > 0$$

b = 0.5 vary a
x1 = 0 x2 = 20 c = y1 = 0 d = y2 = 80

a = 0.7
a = 0.1
a = - 0.1
a = - 0.7

a = 0.1 vary b
x1 = 0 x2 = 20 c = y1 = 0 d = y2 = 80

b = 5
b = 2
b = 1
b = 0.5
b = 0.1
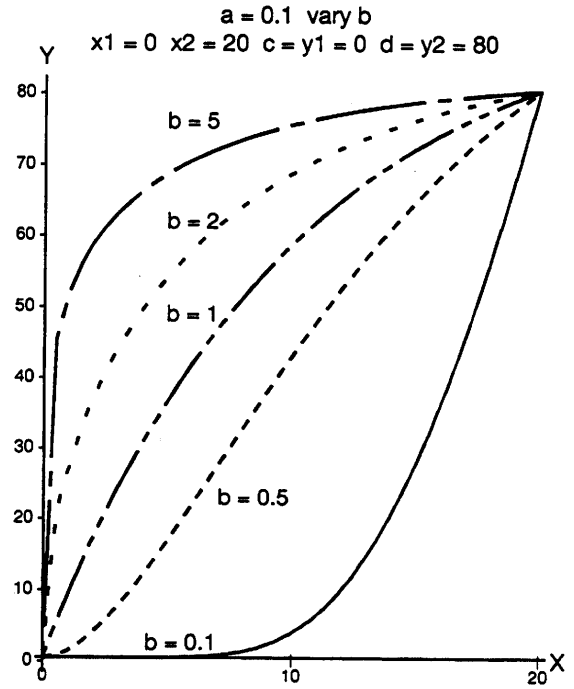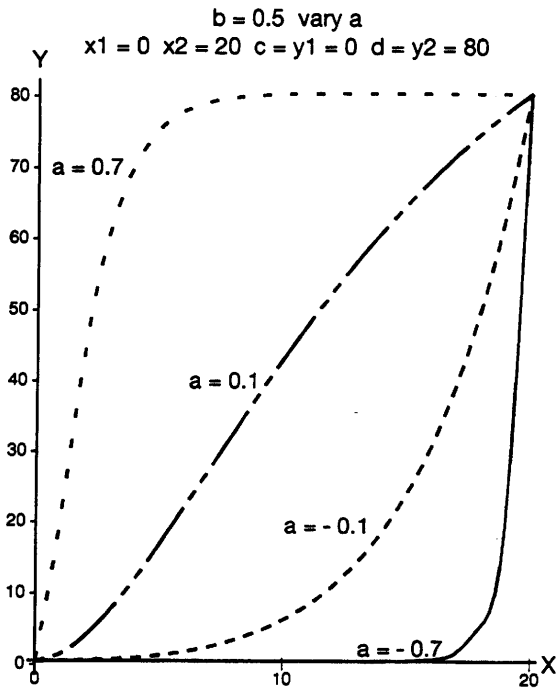
4.4 Power Function

Functional form: $Y = aX^b$ $-\infty < X < \infty$,   restricted X when b is not an integer

Derivatives: $\dfrac{dY}{dX} = abX^{b-1}$

$\dfrac{\partial Y}{\partial a} = X^b$ $\dfrac{\partial Y}{\partial b} = aX^b \ \ln(X)$

Linearized model and parameters: $\ln(Y) = b_0 + b_1 \ln(X)$

$a = e^{b_0}$ $b = b_1$

Description   The power function is also called the **allometric function**. When parameter **b** is an integer, the power function is a special case of a polynomial of degree **b**. This model is undefined for values of **X** < 0 when **b** is not an integer (e.g., undefined for $X < 0$ and $b = 0.5$). The parameter **a** controls the rate of increase or decrease of the curve. The parameter **b** controls the shape of the curve as follows (**a** > 0):

- for **b** > 1,   curve is concave up and increasing

- for $0 < $ **b** < 1,   curve is concave down and increasing

- for **b** < 0,   curve is concave up and decreasing

See Little and Hills (1978, Chapter 14), Daniel and Woods (1980, Chapter 3), and Ratkowsky (1990, p. 125) for more discussion on the power function.
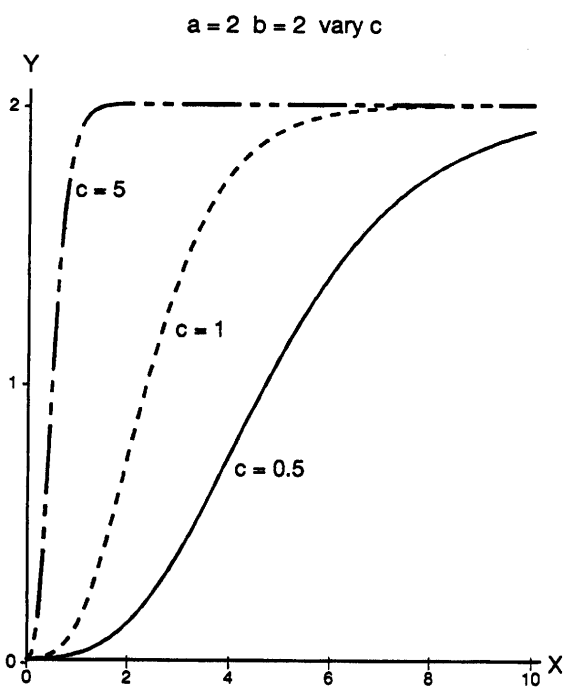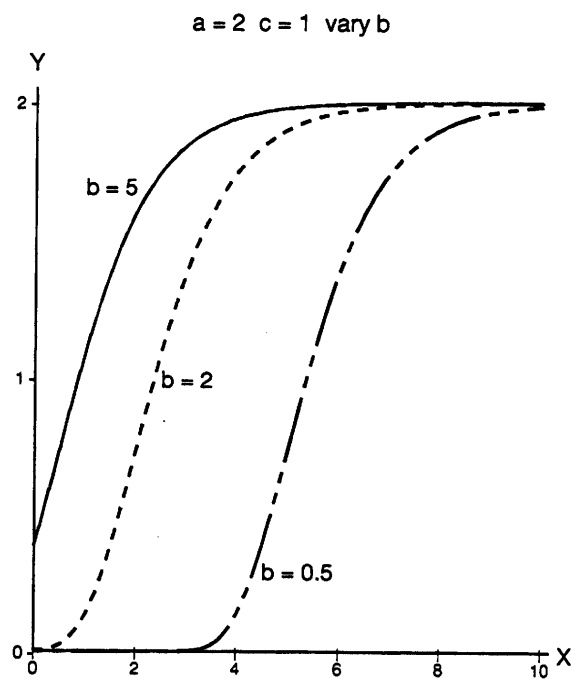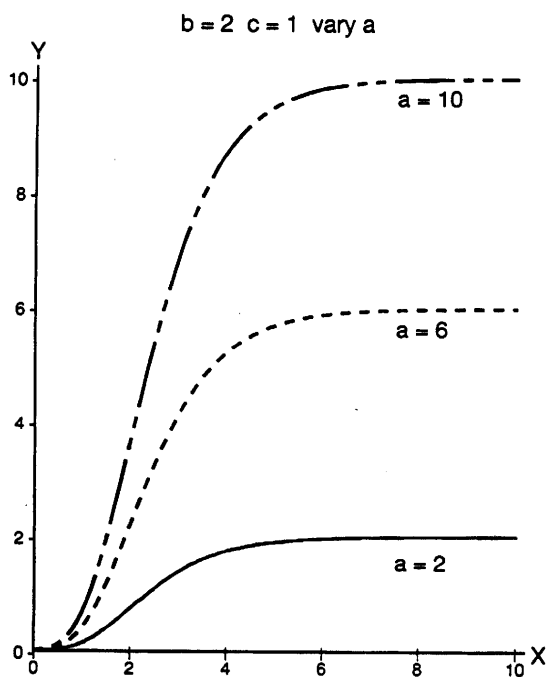
Sample PROC REG **program for the linearized model:**

```
PROC REG DATA=POWER;
  MODEL Y1 = X1;
  ****  X1 = LOG(X)    must be created in a previous data step;
  ****  Y1 = LOG(Y)
RUN;
```

Sample PROC NLIN **program for the functional form:**

```
PROC NLIN DATA=POWER;
  PARAMETERS A=2.0 B=2.0;
  XTOB = X**B;
  MODEL Y = A*XTOB;
  DER.A = XTOB;
  DER.B = A*XTOB*LOG(X);
RUN;
```

Functional form: $Y = aX^b$ $X > 0$

b > 0 : b = 2 vary a

a = 4

a = 1

a = 0.1

0 < b < 1 : b = 0.5 vary a

a = 10

a = 4

a = 2

b < -1 : b = -2 vary a

a = 4

a = 1

a = 0.3

-1 < b < 0 : b = -0.5 vary a

a = 2

a = 1

a = 0.5

4.5 Combined Exponential and Power Functions

4.5.1 Type I combined exponential and power function

Functional form: $Y = aX^b c^X$ $-\infty < X < \infty$,   X restricted when b is not an integer

Derivatives: $\dfrac{dY}{dX} = aX^b c^X \left[\ln(c) + \dfrac{b}{X} \right]$

$$\dfrac{\partial Y}{\partial a} = X^b c^X \qquad\qquad \dfrac{\partial Y}{\partial b} = aX^b c^X \ln(X) \qquad\qquad \dfrac{\partial Y}{\partial c} = aX^{b+1} c^{X-1}$$

Linearized model and parameters: $\ln(Y) = b_0 + b_1 \ln(X) + b_2 X$

$$a = e^{b_0} \qquad\qquad b = b_1 \qquad\qquad c = e^{b_2}$$

Description: This is a more general form of both the power function (Section 4.4) and the exponential function (Section 4.3.1). If interested in curves of this shape, the reader is encouraged to also fit the model in Section 4.5.2 as it has more desirable statistical properties. Note that:

- for $b = 0$, the function becomes $y = ac^X$, the exponential function (Section 4.3.4);
- for $c = 1$, the function becomes $y = aX^b$, the power function (Section 4.4.1);
- for $c = e$, the function becomes $y = aX^b e^X$, the model in Section 4.5.2 with $c = 1$;
- for $c < 0$, many values of c^X will be undefined.

This functional form produces graphs of very different shapes depending on the parameter values. In other words, the parameters **a**, **b** and **c** interact closely to control the shape of the graph, and none of the three parameters independently controls a characteristic of the curve.

Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA = MODALLO;
  MODEL Y1 = X1 X;
  ****  X1 = LOG(X)   must be created in a previous data step;
  ****  Y1 = LOG(Y)
RUN;
```

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=MODALLO;
  PARAMETERS A=4.0 B=2.0 C=0.5;
  XBXC = (X**B)*(C**X);
  AXBXC = A*XBXC;
  MODEL Y = AXBXC;
  DER.A = XBXC;
  DER.B = AXBXC*LOG(X);
  DER.C = AXBXC*X/C;
RUN;
```

Functional form: $Y = aX^b c^X$ $X > 0$

b > 0 : b = 2   c > 1 : c = 2
vary a

b > 0 : b = 2   0 < c < 1 : c = 0.5
vary a

b < 0 : b = -2   0 < c < 1 : c = 0.5
vary a

b < 0 : b = -2   c > 1 : c = 2
vary a
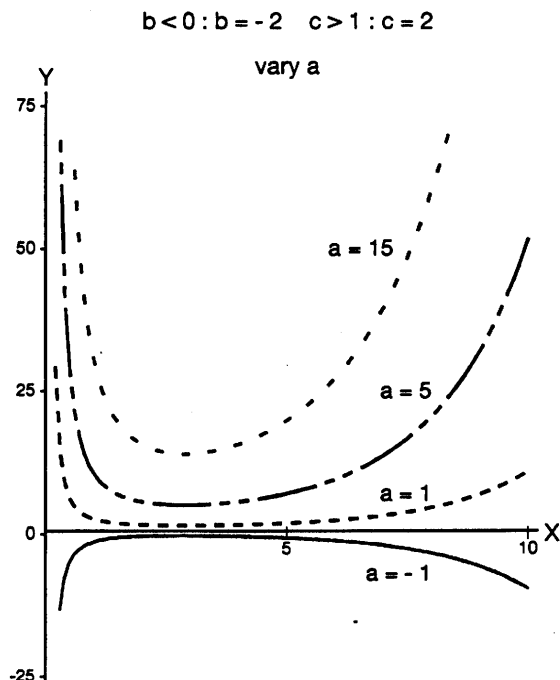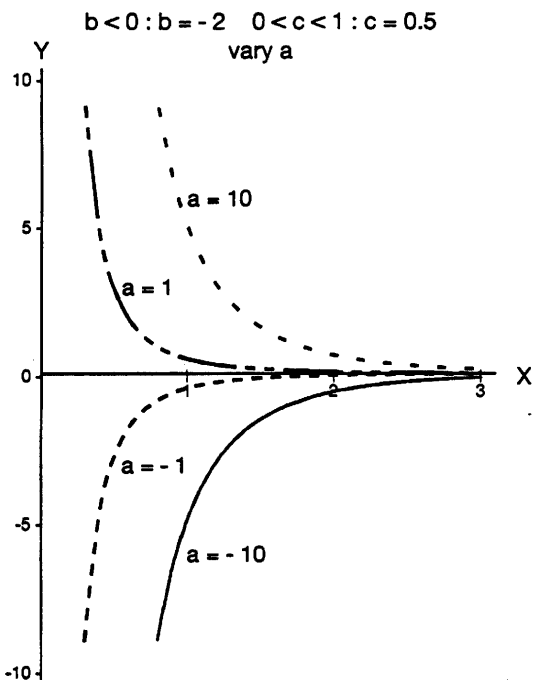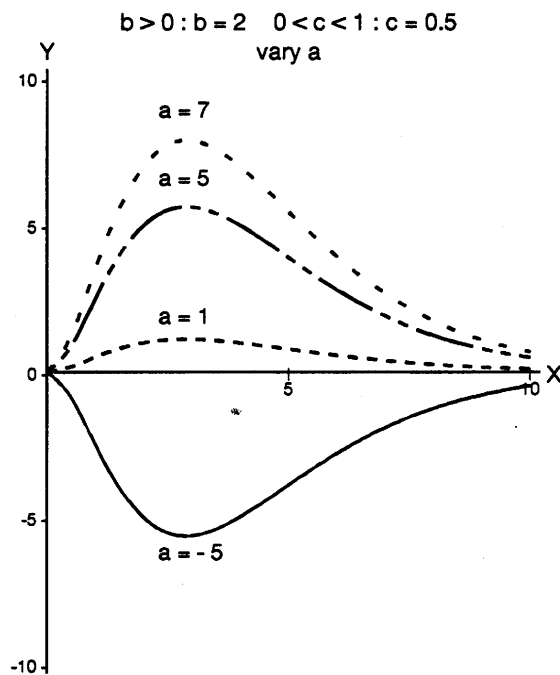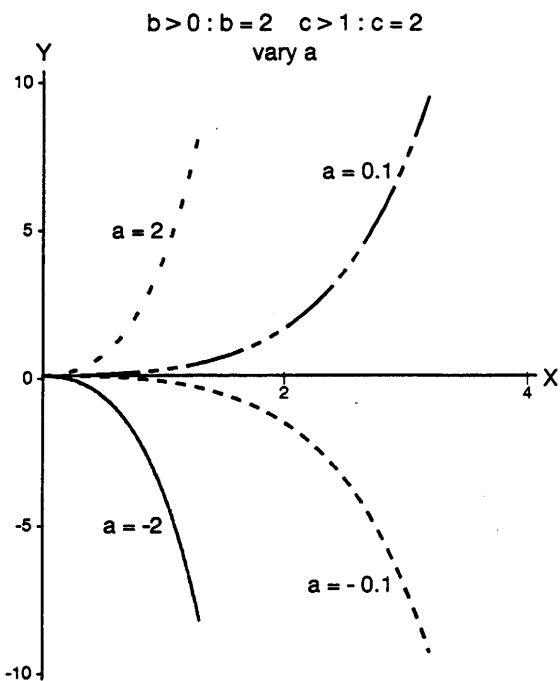
4.5.2 Type II combined exponential and power function

Functional form: $\quad Y = aX^b e^{cX}$ $\qquad\qquad -\infty < X < \infty,\quad$ restricted X when b is not an integer

Derivatives: $\quad \dfrac{dY}{dX} = aX^b e^{cX}\left(c + \dfrac{b}{X}\right)$

$$\frac{\partial Y}{\partial a} = X^b e^{cX} \qquad\qquad\qquad \frac{\partial Y}{\partial b} = aX^b e^{cX}\ln(X)$$

$$\frac{\partial Y}{\partial c} = aX^{b+1} e^{cX}$$

Linearized model and parameters: $\quad \ln(Y) = b_0 + b_1\ln(X) + b_2 X$

$$a = e^{b_0} \qquad\qquad b = b_1 \qquad\qquad c = b_2$$

Description: This function is a generalization of the power function (Section 4.4.1) and the exponential function (Section 4.3.1). This model is more specific and has better statistical properties than that of Type I (Section 4.5.1). The estimated parameters are close to being unbiased and normal, and have minimum variance. The readers are encouraged to fit the model in Section 4.5.1 and this model and to compare the results. Note:

- for $c = 0$, the function becomes $Y = aX^b$, the power function (Section 4.4.1)

- for $b = 0$, the function becomes $Y = ae^{cX}$, the exponential function (Section 4.3.1)

The parameters **a** and **b** interact closely to control whether the curve is increasing or decreasing. Changing the value of the parameter **c** causes dramatic changes in the shape of the curve (see graphs below). See Hoerl (1954) and Daniel and Woods (1980), Chapter 3, for further descriptions of this function. Ratkowsky (1990, pp. 162, 163) also discusses this function and the related function $Y = X^b e^{(a - bX)}$.
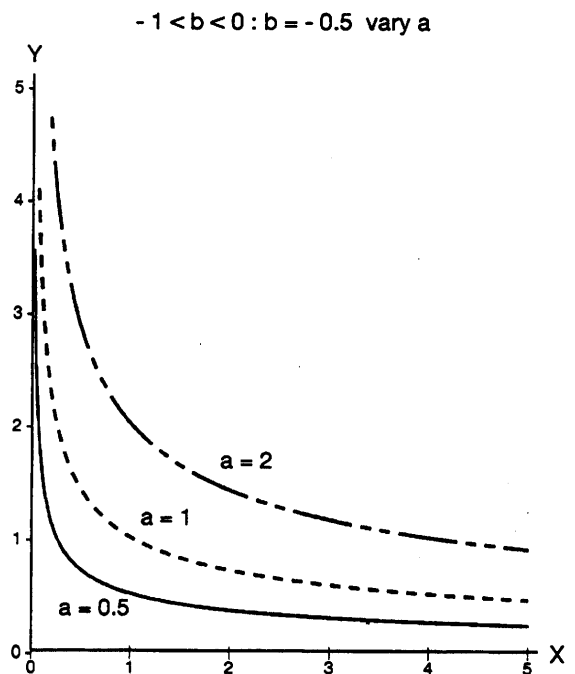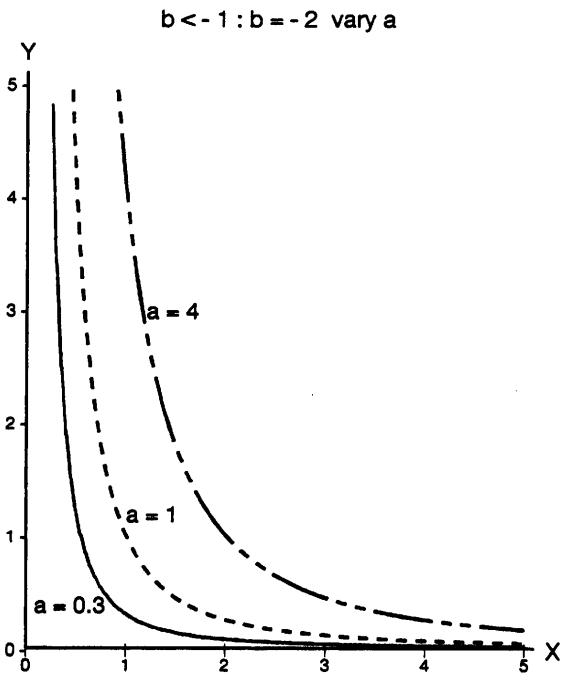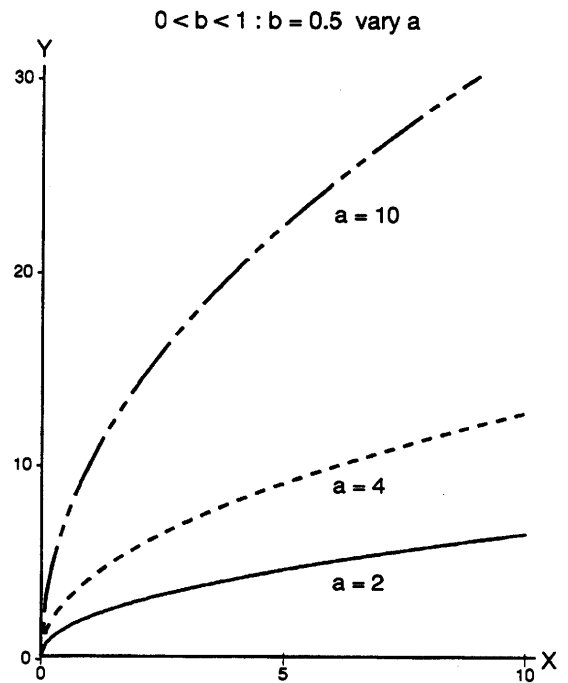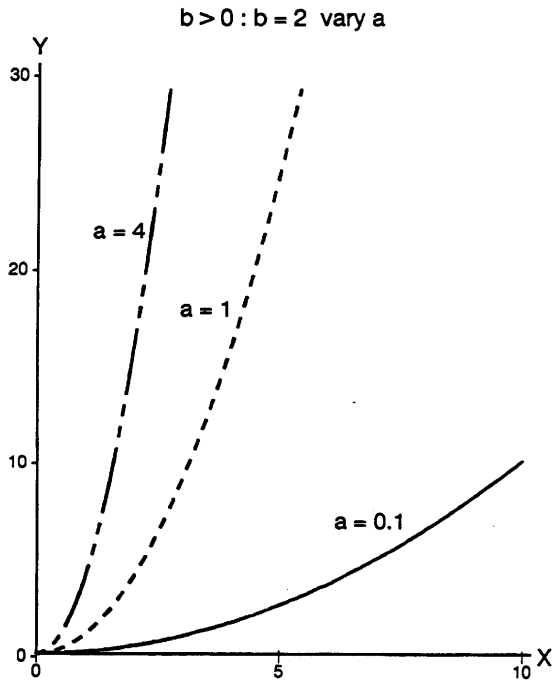
Sample `PROC REG` **program for the linearized model:**

```
PROC REG DATA=HOERL;
  MODEL Y1 = X1 X;
  **** X1=LOG(X) and Y1=LOG(Y) must be created in a previous data step;
RUN;
```

Sample `PROC NLIN` **program:**

```
PROC NLIN DATA=HOERL;
  PARAMETERS A=2.0 B=2.0 C=0.1;
  EBCX = X**B*EXP(C*X);
  MODEL Y = A*EBCX;
  DER.A = EBCX;
  DER.B = A*EBCX*LOG(X);
  DER.C = A*X*EBCX;
RUN;
```

Functional form: $Y = aX^b e^{cX}$ $X > 0$

b = - 1 c = 0.1 vary a
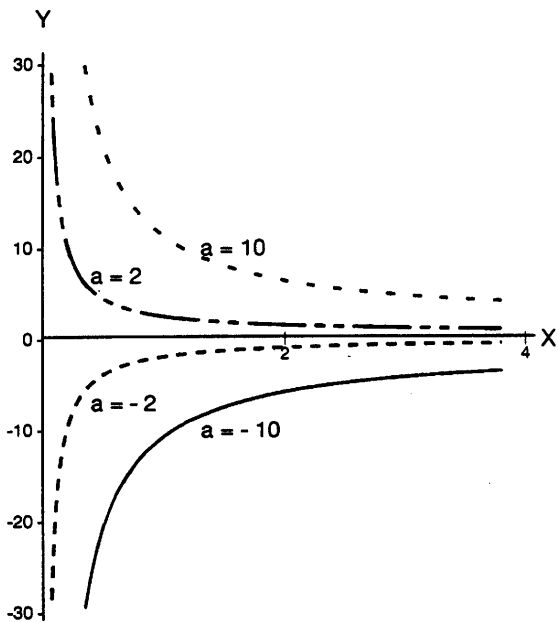
b = 2 c = 0.1 vary a

b = - 1 c = - 0.1 vary a

b = 2 c = - 0.2 vary a

61

Functional form: $Y = aX^b e^{cX}$ $\qquad X > 0$

a = 2 b = 2 vary c < 0
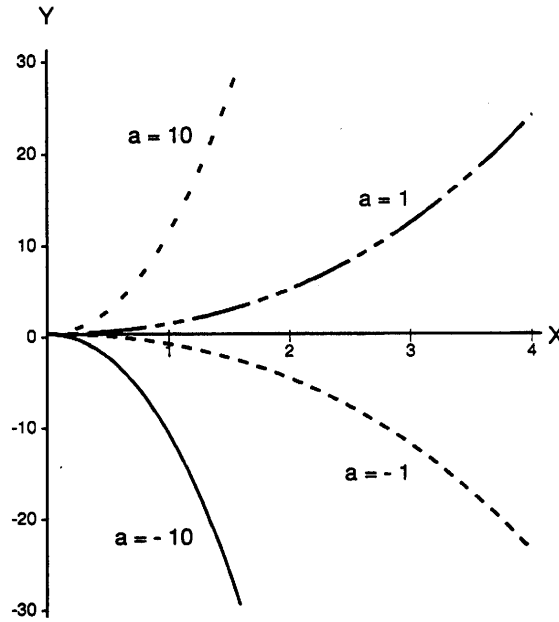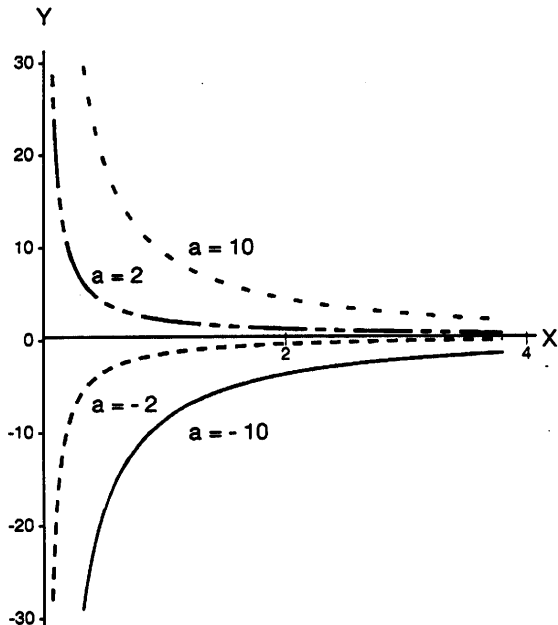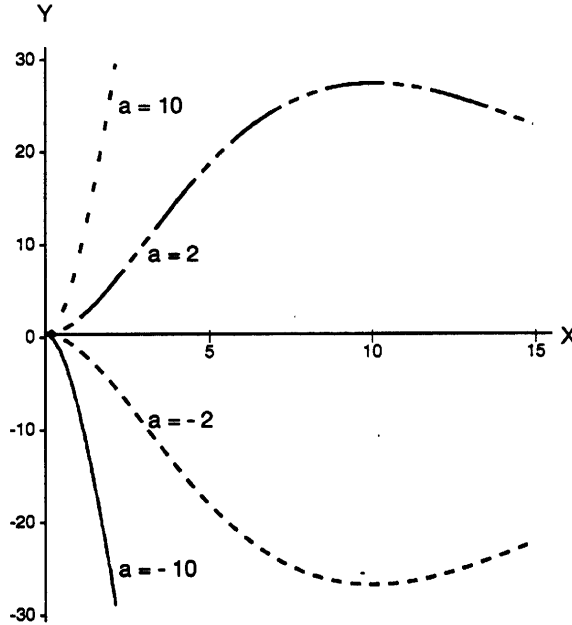
a = 2 b = -2 vary c > 0
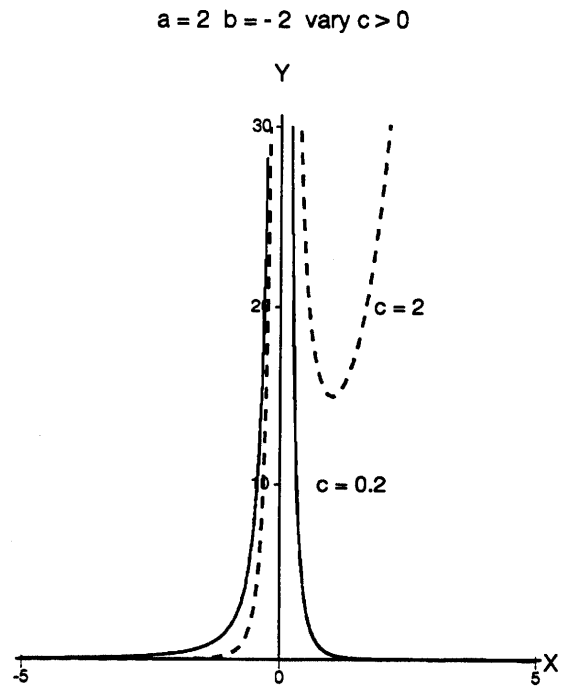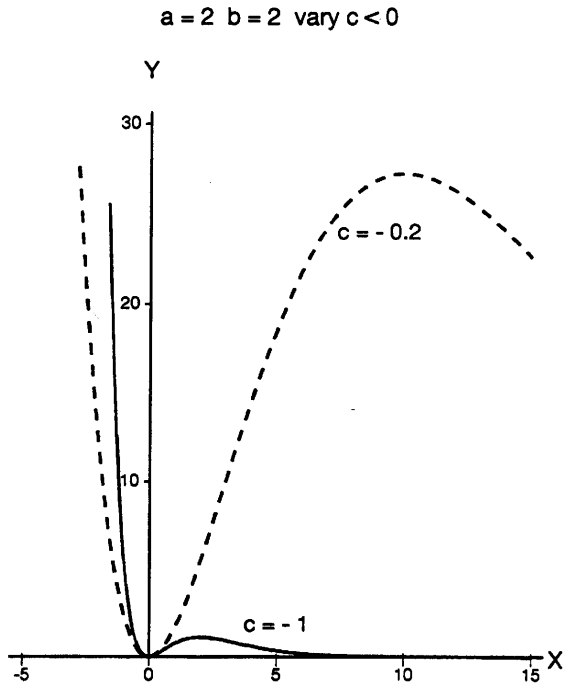
Note: as pointed out in the description, changing the value of **c** causes dramatic changes in the shape of the curve. The graph on the right has a vertical asymptote at $X = 0$ and is discontinuous.

4.5.3 Generalized Poisson function

Functional form: $Y = \left[\dfrac{b-X}{b-a}\right]^c \exp\left\{\dfrac{c}{d}\left[1 - \left[\dfrac{b-X}{b-a}\right]^d\right]\right\}$, X restricted when c or d not an integer

Derivatives: $\dfrac{dY}{dX} = \dfrac{c}{b-a}\left[\dfrac{b-X}{b-a}\right]^{c-1}\left[\left[\dfrac{b-X}{b-a}\right]^d - 1\right]\exp\left\{\dfrac{c}{d}\left[1 - \left[\dfrac{b-X}{b-a}\right]^d\right]\right\}$

$$\dfrac{\partial Y}{\partial a} = Y\left[\dfrac{c}{b-a} - c\left[\dfrac{b-X}{b-a}\right]^d \dfrac{1}{b-a}\right]$$

$$\dfrac{\partial Y}{\partial b} = Y\left[\dfrac{c}{b-X} - \dfrac{c}{b-a} - c\left[\dfrac{b-X}{b-a}\right]^{d-1}\dfrac{X-a}{(b-a)^2}\right]$$

$$\dfrac{\partial Y}{\partial c} = Y\left\{\ln\left[\dfrac{b-X}{b-a}\right] + \dfrac{1}{d}\left[1 - \left[\dfrac{b-X}{b-a}\right]^d\right]\right\}$$

$$\dfrac{\partial Y}{\partial d} = -Y\dfrac{c}{d}\left\{\left[\dfrac{b-X}{b-a}\right]^d \ln\left[\dfrac{b-X}{b-a}\right] + \dfrac{1}{d}\left[1 - \left[\dfrac{b-X}{b-a}\right]^d\right]\right\}$$

Linearized model and parameters: This function can not be linearized.

Description: This model is restrictive for general use since it is not always defined for values of $X > $ **b** when parameters **c** or **d** are not integers. For example, where parameters **c** and **d** vary, for **b** $= 12$ and **a** $= 5$ (as shown in the following graphs), the curves are not always defined for $X > 12$. Parameters **c** and **d** are shape parameters. The points (a,1) and (b,0) are on the curve.

Sample PROC NLIN **program for the functional form:**

```
PROC NLIN DATA = POISSON;
  PARAMETERS A=2.0 B=12.0 C=30.0 D=5.0;
  BA = 1/(B-A);
  CBA = C*BA;
  BASQ = BA*BA;
  BXBA = (B-X)*BA;
  BXBAD = BXBA**D;
  BD = (1-BXBAD)/D;
  BXBAC = (BXBA**C)*(EXP(C*BD));
  DBC = D*BXBAD/BXBA+C/BXBA;
  MODEL Y = BXBAC;
  DER.A = BXBAC*(CBA - C*BXBAD*BA);
  DER.B = BXBAC*(C/(B-X) - CBA - C*BXBAD*(X-A)*BASQ/BXBA);
  DER.C = BXBAC*(LOG(BXBA) + BD);
  DER.D = -BXBAC*(C/D)*(BXBAD*LOG(BXBA) + BD/D);
RUN;
```

Functional form: $Y = \left[\dfrac{b-X}{b-a}\right]^c \exp\left\{\dfrac{c}{d}\left[1 - \left[\dfrac{b-X}{b-a}\right]^d\right]\right\}$ $X > 0$

b = 12 c = 3 d = 5 vary a

a = 5 c = 3 d = 5 vary b

a = 5 b = 12 d = 5 vary c

a = 5 b = 12 c = 3 vary d
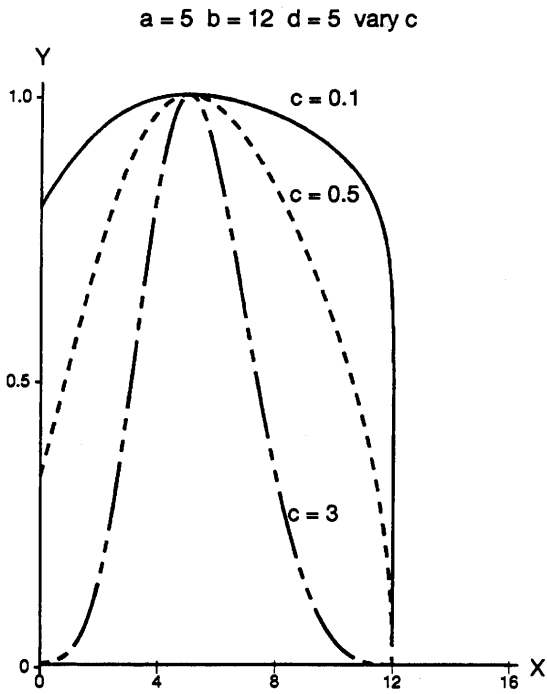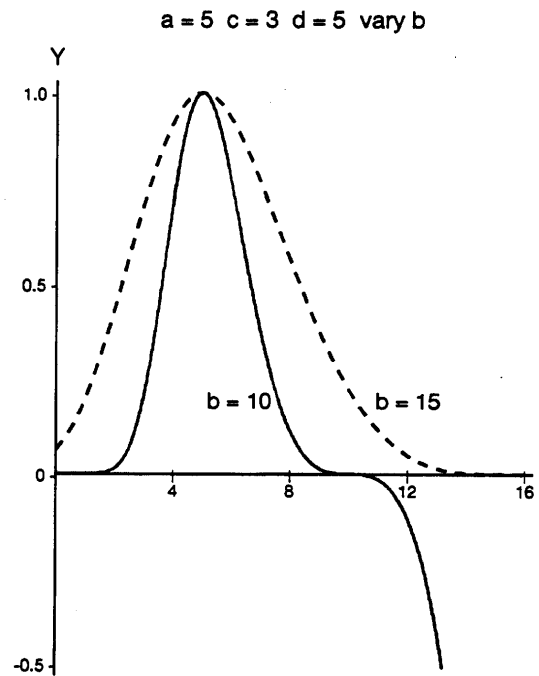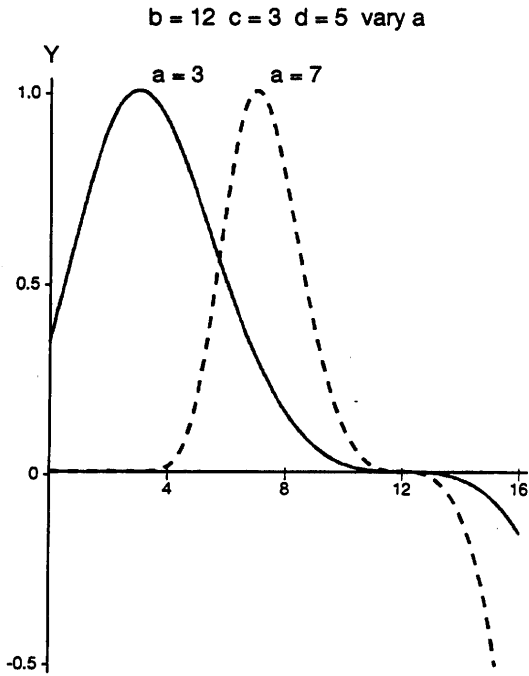
4.6 Logarithmic Function

Functional form: $Y = a + b \ln(X)$, $X > 0$

Derivatives: $\dfrac{dY}{dX} = \dfrac{b}{X}$

$\dfrac{\partial Y}{\partial a} = 1$ $\dfrac{\partial Y}{\partial b} = \ln(X)$

Linearized model and parameters: Already linear in $\ln(X)$.

Description: The parameter **a** shifts the curve up and down the Y-axis. The parameter **b** controls the shape of the curve as follows:

- for **b** > 0, the curve is concave down and increasing

- for **b** $= 0$, the curve is a horizontal line, $Y = $ **a**

- for **b** < 0, the curve is concave up and decreasing

Sample PROC REG **program for the linearized model:**

```
PROC REG DATA=LOGR;
  MODEL Y = X1;
  ****  X1 = LOG(X)   must be created in a previous data step;
RUN;
```

Sample PROC NLIN **program for the functional form:**

```
PROC NLIN DATA=LOGR;
  PARAMETERS A=5.0 B=2.0;
  LOGX = LOG(X);
  MODEL Y = A + B*LOGX;
  DER.A = 1;
  DER.B = LOGX;
RUN;
```

Functional form: $Y = a + b\ln(X)$ $X > 0$
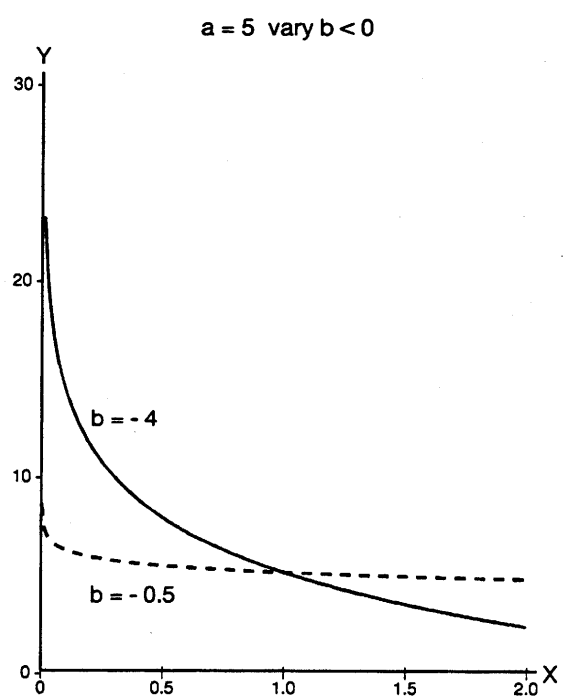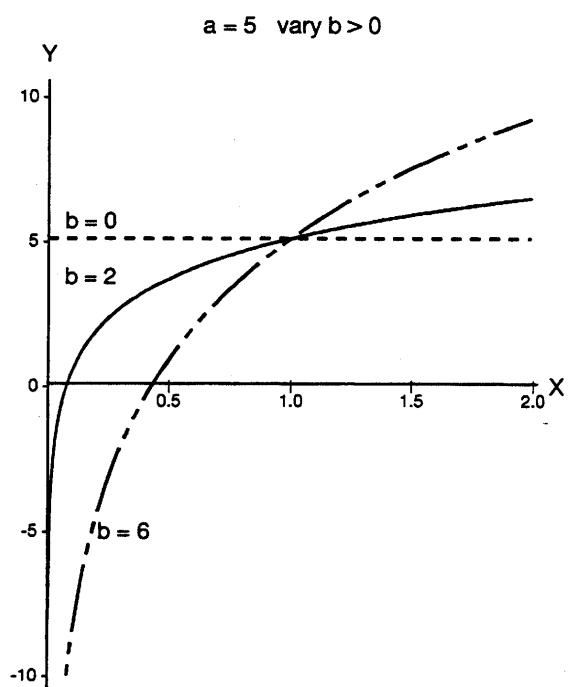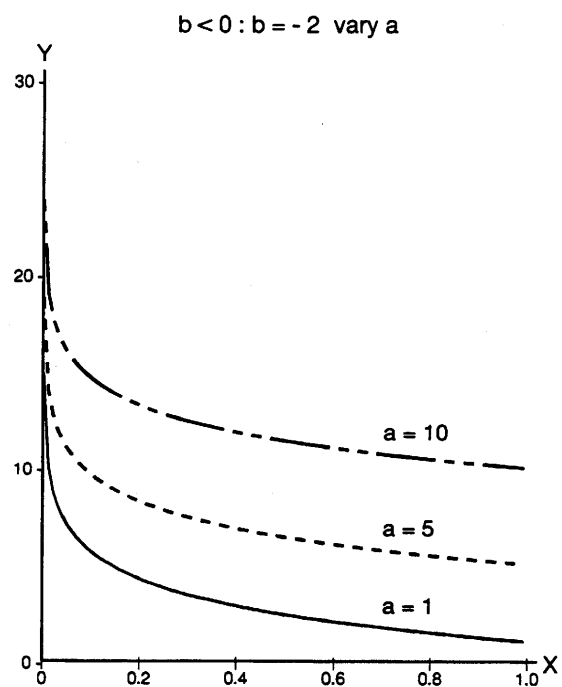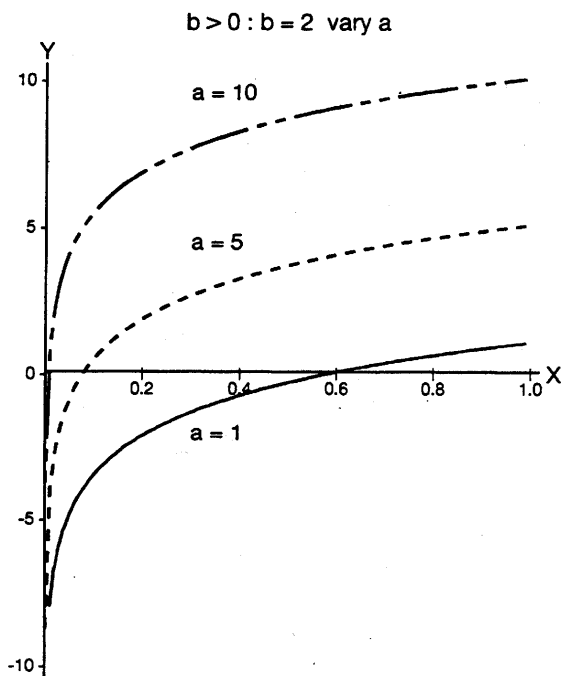
4.7 Trigonometric Functions

The cosine (cos) and sine (sin) functions are the most commonly used trigonometric functions. The two functions are identical except for a lag of 90° ($\pi/2$). That is,

$$\sin\left(X + \frac{\pi}{2}\right) = \cos(X)$$

$$\cos\left(X - \frac{\pi}{2}\right) = \sin(X)$$

The sine and cosine functions are also related in their derivatives as follows:

$$\frac{d(\sin(X))}{dX} = \cos(X)$$

$$\frac{d(\cos(X))}{dX} = -\sin(X)$$

The sine and cosine functions are periodical functions; that is, the shape of the curve repeats along the X-axis. The shape of such a curve is determined by amplitude and wavelength. The **amplitude** of the curve is the distance between the maximum and minimum Y-values. The **wavelength** of the curve is the length of each non-repeating pattern along the X-axis. The sine and cosine functions are symmetric about any maximum and minimum. The shape of a sine or cosine curve is often referred to as **sinusoidal**.

Many trigonometric identities exist that relate sine, cosine, tangent (tan) and their reciprocals: cosecant, secant, and cotangent. Two commonly used identities are:

$$\tan(X) = \frac{\sin(X)}{\cos(X)}$$

$$\sin^2(X) + \cos^2(X) = 1$$

For more information on trigonometry, see Edwards and Penney (1982) or other algebra and calculus textbooks.

4.7.1 Cosine function

Functional form: $\quad Y = a\,\cos\left[\dfrac{2\pi}{b}(X-c)\right]$ $\qquad\qquad -\infty < X < \infty$

Derivatives: $\quad \dfrac{dY}{dX} = \dfrac{-2\pi a}{b}\,\sin\left[\dfrac{2\pi}{b}(X-c)\right]$

$$\dfrac{\partial Y}{\partial a} = \cos\left[\dfrac{2\pi}{b}(X-c)\right] \qquad\qquad \dfrac{\partial Y}{\partial b} = (X-c)\dfrac{2\pi a}{b^2}\,\sin\left[\dfrac{2\pi}{b}(X-c)\right]$$

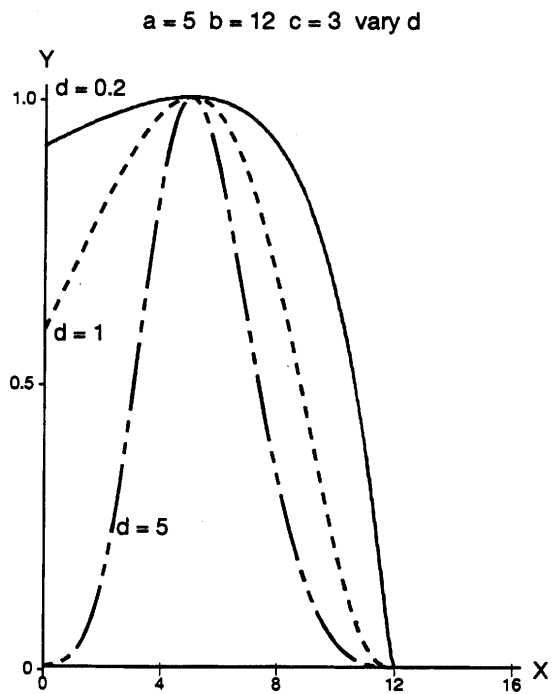$$\dfrac{\partial Y}{\partial c} = \dfrac{2\pi a}{b}\,\sin\left[\dfrac{2\pi}{b}(X-c)\right]$$

Linearized model and parameters: This function can not be linearized.

Description: The cosine function has an amplitude of 2**a** and a wavelength of **b**. The parameter **c** shifts the curve along the X-axis. The maximum is at **X** = **c** and again at **X** = **c** + **b**, and so on.

Sample PROC NLIN **program for the functional form:**

```
PROC NLIN DATA = COSINE;
  PARAMETERS A=1.0 B=10.0 C=0;
  PI = 3.1415926;
  BXC = 2*PI*(X-C)/B;
  COSBXC = COS(BXC);
  SINBXC = 2*PI*A*SIN(BXC)/B;
  MODEL Y = A*COSBXC;
  DER.A = COSBXC;
  DER.B = SINBXC*(X-C)/B;
  DER.C = SINBXC;
RUN;
```

Functional form: $\quad Y = a \cos\left[\dfrac{2\pi}{b}(X-c)\right] \qquad X > 0$

b = 10 c = 0 vary a
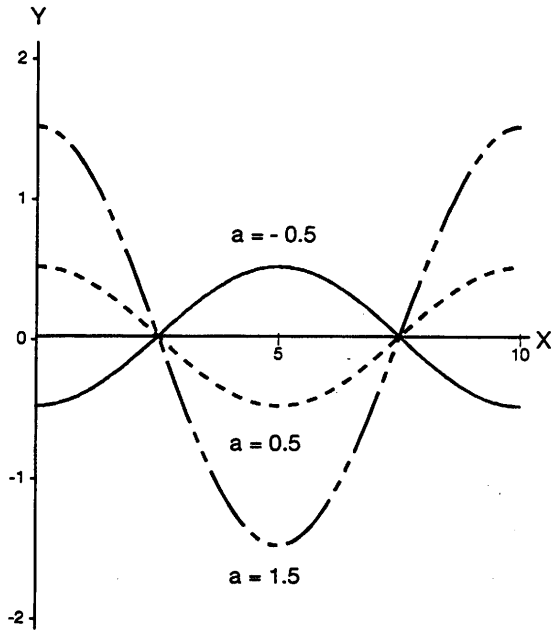
a = 1 c = 0 vary b
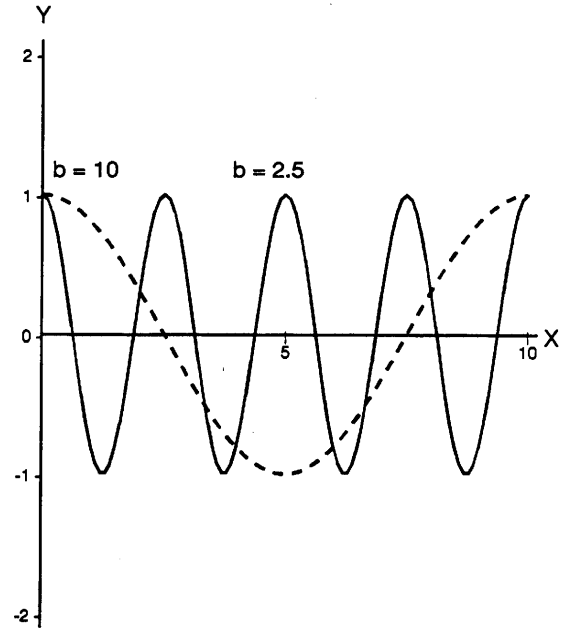
a = 1 b = 10 vary c
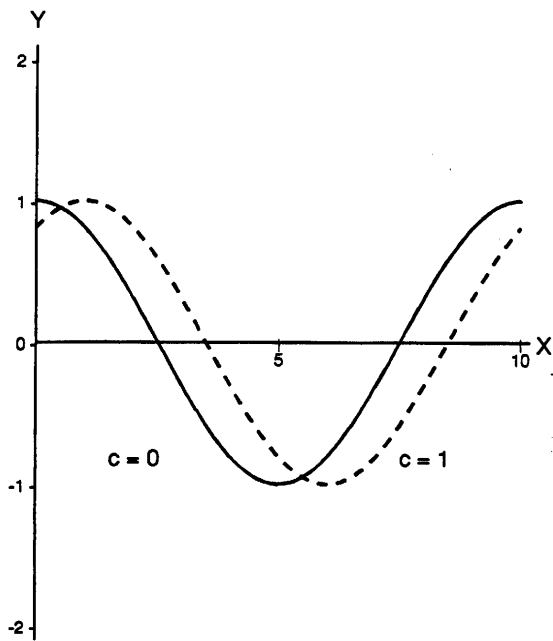
4.7.2 Sine function

Functional form: $Y = a \sin\left[\dfrac{2\pi}{b}(X - c)\right]$ $-\infty < X < \infty$

Derivatives: $\dfrac{dY}{dX} = \dfrac{2\pi a}{b} \cos\left[\dfrac{2\pi}{b}(X - c)\right]$

$\dfrac{\partial Y}{\partial a} = \sin\left[\dfrac{2\pi}{b}(X - c)\right]$ $\dfrac{\partial Y}{\partial b} = (X - c)\dfrac{-2\pi a}{b^2} \cos\left[\dfrac{2\pi}{b}(X - c)\right]$

$\dfrac{\partial Y}{\partial c} = -\dfrac{2\pi a}{b} \cos\left[\dfrac{2\pi}{b}(X - c)\right]$

Linearized model and parameters: This function can not be linearized.

Description: The sine function has an amplitude of 2**a** and wavelength of **b**. The parameter **c** shifts the curve along the X-axis with maximums at $X = c + b/4$, $X = c + 5b/4$, and $X = c + 9b/4$, and so on.

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=SINE;
  PARAMETERS A=1.0 B=10.0 C=0;
  PI = 3.1415926;
  BXC = 2*PI*(X-C)/B;
  SINBXC = SIN(BXC);
  COSBXC = 2*PI*A*COS(BXC)/B;
  MODEL Y = A*SINBXC;
  DER.A = SINBXC;
  DER.B = -COSBXC*(X-C)/B;
  DER.C = -COSBXC;
RUN;
```

Functional form: $Y = a \sin\left[\dfrac{2\pi}{b}(X - c)\right]$ $\quad X > 0$

b = 10 c = 0 vary a
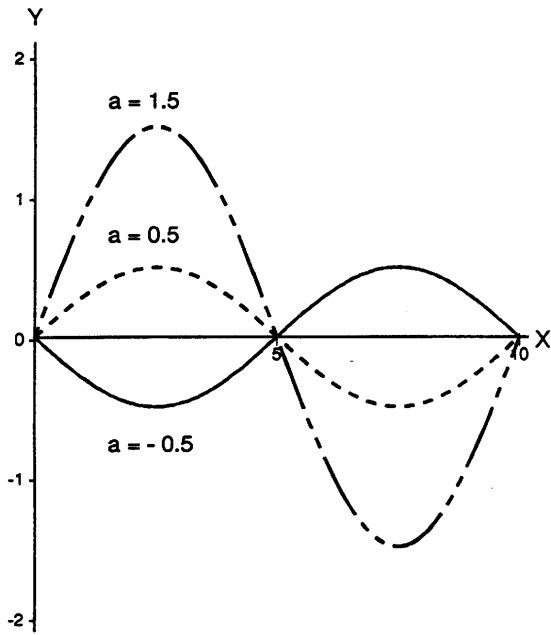
a = 1 c = 0 vary b
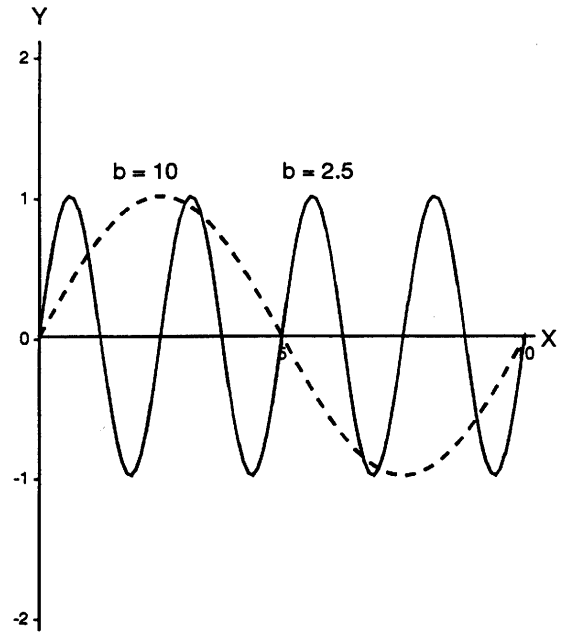
a = 1 b = 10 vary c
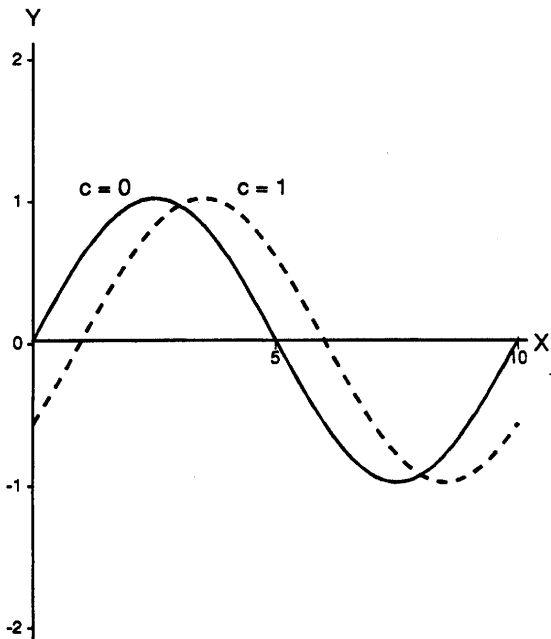
4.7.3 Arctangent function

Functional form: $\qquad Y = a + \dfrac{b}{\pi} \arctan\left[\pi c\left(X - d\right)\right] \qquad\qquad -\infty < X < \infty$

Derivatives: $\qquad \dfrac{dY}{dX} = bc\left[\dfrac{1}{1 + (\pi c(X - d))^2}\right]$

$$\dfrac{\partial Y}{\partial a} = 1 \qquad\qquad\qquad \dfrac{\partial Y}{\partial b} = \dfrac{1}{\pi}\arctan\left[\pi c\left(X - d\right)\right]$$

$$\dfrac{\partial Y}{\partial c} = \dfrac{b(X - d)}{1 + (\pi c(X - d))^2} \qquad\qquad \dfrac{\partial Y}{\partial d} = -bc\left[\dfrac{1}{1 + (\pi c(X - d))^2}\right]$$

Linearized model and parameters: This function can not be linearized.

Description: The arctangent function has an S shape. Parameter **a** shifts the curve up or down; parameter **b** controls the vertical distance between the minimum and the maximum; parameter **c** gives the slope of the curve at the point of inflection; and parameter **d** shifts the curve along the X-axis.

Sample `PROC NLIN` **program for the functional form:**

```
PROC NLIN DATA=ARCTANG;
  PARAMETERS A = 4.0 B= 0.5 C= 1.0 D = 1.0;
  PI = 3.1415926;
  CXD = PI*C*(X-D);
  CXD2 = B/(1+CXD**2);
  MODEL Y = A+B*ATAN(CXD)/PI;
  DER.A = 1;
  DER.B = ATAN(CXD)/PI;
  DER.C = (X-D)*CXD2;
  DER.D = -CXD2*C;
RUN;
```

Functional form: $Y = a + \dfrac{b}{\pi} \arctan\left[\pi c(X - d)\right]$ $\qquad X > 0$

b = 1 c = 1 d = 4 vary a

a = 0.5 c = 1 d = 4 vary b

a = 0.5 b = 1 d = 4 vary c

a = 0.5 b = 1 c = 1 vary d
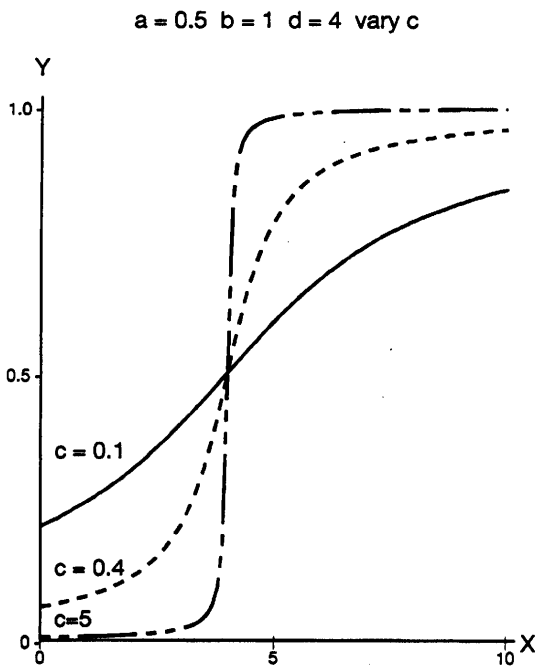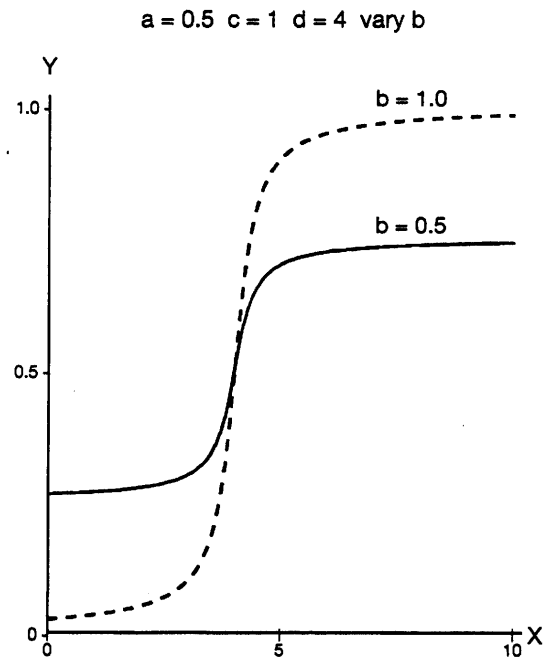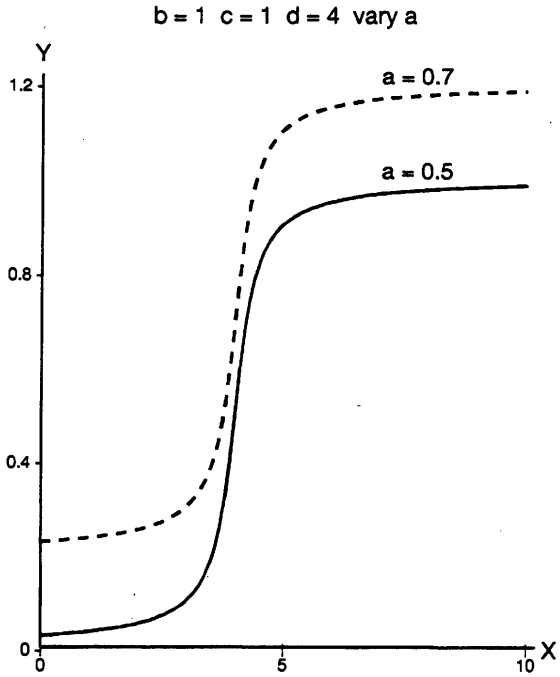
4.8 Common Distributions

To find the distribution of a set of data, frequency data can be fitted to probability distribution functions. The following distributions and accompanying notation are commonly used in statistics. A basic statistics textbook such as Freund and Walpole (1987) is recommended for the interested reader.

Normal	$N(\mu,\sigma^2)$	where μ is the mean and σ^2 is the variance
Student-t	t_a	where **a** is the degrees of freedom
Fisher F	$F_{a,b}$	where **a** and **b** are the numerator and denominator degrees of freedom
Chi-square	χ_a^2	where **a** is the degrees of freedom

The following definitions and properties apply to the preceding distributions.

Definition 1:   Let $Z_1, Z_2,\ldots, Z_n$ be independent $N(0,1)$ random variables, then:

$$Z_1^2 + Z_2^2 + \ldots + Z_n^2$$

has a Chi-square distribution with n degrees of freedom, and is denoted as:

$$Z_1^2 + Z_2^2 + \ldots + Z_n^2 \sim \chi_n^2$$

Note: The notation $\sim$ means "is distributed as."

Definition 2:   Let $Z \sim N(0,1)$ and $W \sim \chi_n^2$ be independent random variables, then:

$$T = \frac{Z}{\sqrt{W/n}} \sim t_n$$

That is, T has a Student-t distribution with **n** degrees of freedom.

Definition 3:   Let $W_1 \sim \chi_n^2$ and $W_2 \sim \chi_m^2$ be independent Chi-square variables, then:

$$F = \frac{W_1/n}{W_2/m} \sim F_{n,m}$$

That is, F has a Fisher F distribution with **n** and **m** degrees of freedom.

Properties:

1. If $W \sim t_n$, then $W^2 \sim F_{1,n}$.
2. If $W \sim F_{n,m}$, then $\frac{1}{W} \sim F_{m,n}$.

4.8.1 Normal distribution

Functional form: $Y = \dfrac{1}{\sqrt{2\pi b}} \exp\left[\dfrac{-(X-a)^2}{2b}\right]$ $-\infty < X < \infty$ $b > 0$

Derivatives: $\dfrac{dY}{dX} = \dfrac{-(X-a)}{\sqrt{2\pi}\, b^{1.5}} \exp\left[\dfrac{-(X-a)^2}{2b}\right]$

$\dfrac{\partial Y}{\partial a} = \dfrac{(X-a)}{\sqrt{2\pi}\, b^{1.5}} \exp\left[\dfrac{-(X-a)^2}{2b}\right]$

$\dfrac{\partial Y}{\partial b} = \dfrac{1}{2}\left[\dfrac{(X-a)^2}{b} - 1\right]\dfrac{1}{\sqrt{2\pi}\, b^{1.5}} \exp\left[\dfrac{-(X-a)^2}{2b}\right]$

Linearized model and parameters: This function cannot be linearized.

Description: The function is symmetric about $X = $ **a**. The parameter **a** is the mean of the distribution and is usually denoted by the Greek letter μ. The parameter **b** is the variance which is denoted by the Greek letter σ^2. The latter controls the height and width of the curve. The reader may also try to fit the model in Section 4.3.5 which has the same features as the normal distribution.

Sample `PROC NLIN` **Program for the functional form:**

```
PROC NLIN DATA=NORMAL;
  PARAMETERS A=0 B=1;
  PI = 3.14215926;
  XA = X-A;
  NORM = EXP(-0.5*XA**2/B)/SQRT(2*PI*B);
  MODEL  Y = NORM;
  DER.A = NORM*XA/B;
  DER.B = 0.5*(XA**2/B-1)*NORM/B;
RUN;
```

Functional form: $Y = \dfrac{1}{\sqrt{2\pi b}} \exp\left[\dfrac{-1}{2b}(X-a)^2\right]$

b = 1 vary a

a = 0 vary b
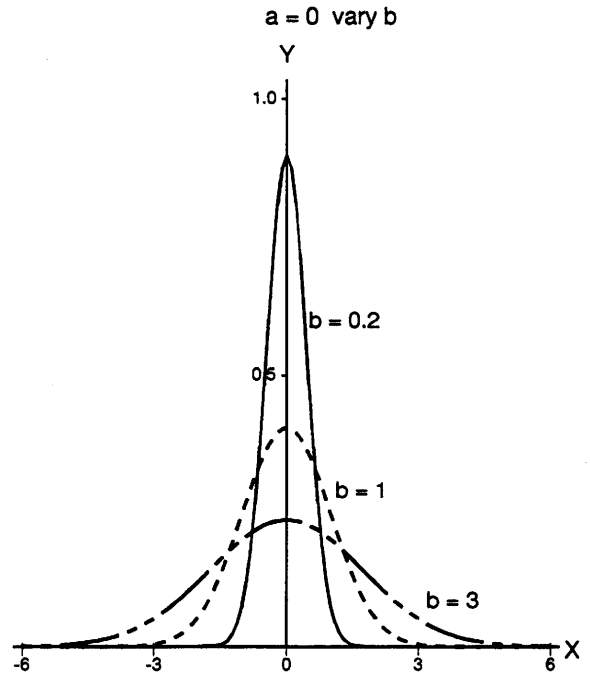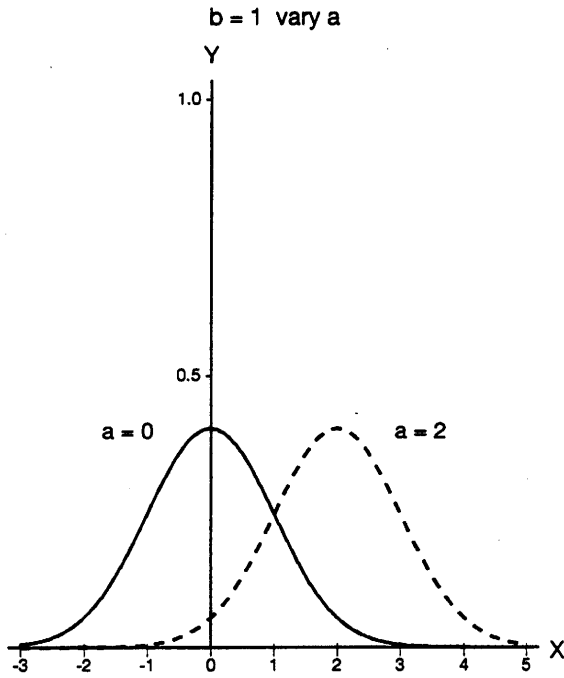
4.8.2 Student-*t* distribution

Functional form:
$$Y = \frac{\Gamma\left[\dfrac{a+1}{2}\right]}{\sqrt{a\pi}\ \Gamma\left[\dfrac{a}{2}\right]}\left\{1 + \frac{X^2}{a}\right\}^{-\frac{a+1}{2}} \qquad -\infty < X < \infty, \qquad a > 0$$

Derivative:
$$\frac{dY}{dX} = \frac{-(a+1)\,X\Gamma\left[\dfrac{a+1}{2}\right]}{a\,\sqrt{a\pi}\ \Gamma\left[\dfrac{a}{2}\right]}\left\{1 + \frac{X^2}{a}\right\}^{-\frac{a+3}{2}}$$
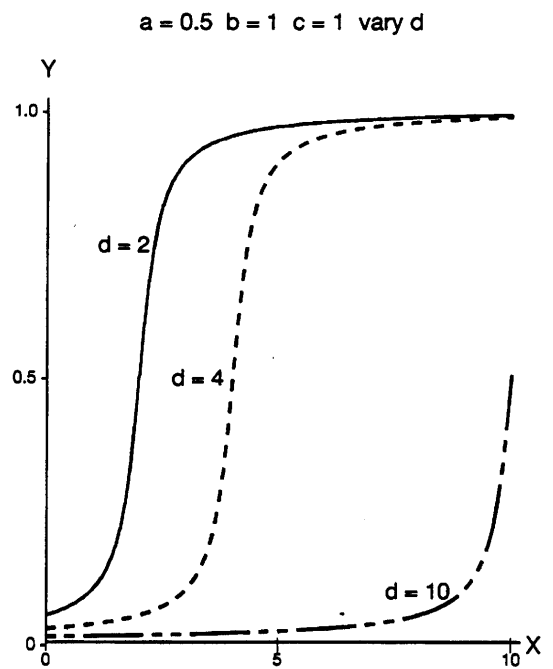
Linearized model and parameters: This function can not be linearized.

Description: This model is also known as the **central student-*t*** distribution. It looks more like a normal distribution as parameter **a** gets larger. The parameter **a** is called the **degrees of freedom**; the curve is taller and narrower for larger values of **a**. See Appendix 1 for a description of $\Gamma(X)$, the Gamma function.

Sample `PROC NLIN` **program for the functional form:**

Because the derivative of the Gamma function is not easily calculated, it is more straightforward, in this case, to use the `DUD` method within `NLIN`. `DUD` is the default method used when no derivatives are given in the program. The SAS function `GAMMA` can be used to evaluate the gamma function. For example, use `GAMMA(7)` to evaluate $\Gamma(7)$.

```
PROC NLIN DATA=STUDT;
  PARAMETERS A = 5.0;
  PI = 3.14215926;
  A12 = (A+1)/2;
  G = GAMMA(A/2);
  G1 = GAMMA(A12);
  MODEL Y = G1*(1 + X*X/A)**(-A12) / (SQRT(A*PI) * G);
RUN;
```

Functional Form: $$Y = \frac{\Gamma\left[\dfrac{a+1}{2}\right]}{\sqrt{a\pi}\;\;\Gamma\left[\dfrac{a}{2}\right]}\left\{1+\frac{X^2}{a}\right\}^{-\frac{a+1}{2}}$$

vary a
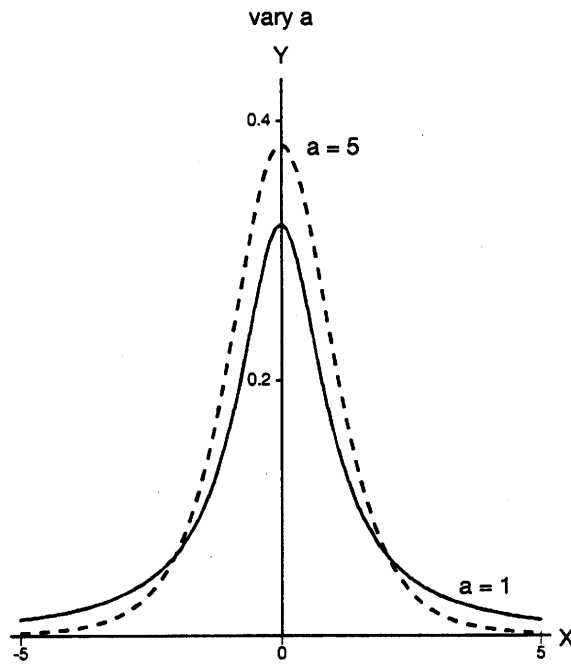
4.8.3 Fisher *F* distribution

Functional form:
$$Y = \left[\frac{a}{b}\right]^{\frac{a}{2}} \frac{\Gamma\left[\frac{a+b}{2}\right]}{\Gamma\left[\frac{a}{2}\right]\Gamma\left[\frac{b}{2}\right]} X^{\frac{a}{2}-1}\left[1 + \frac{a}{b}X\right]^{-\frac{a+b}{2}} \qquad X > 0$$

Derivative:
$$\frac{dY}{dX} = \left[\frac{a}{b}\right]^{\frac{a}{2}} \frac{\Gamma\left[\frac{a+b}{2}\right]}{\Gamma\left[\frac{a}{2}\right]\Gamma\left[\frac{b}{2}\right]} X^{\frac{a}{2}-1}\left[1 + \frac{a}{b}X\right]^{-\frac{a+b}{2}} \left\{\left[\frac{a}{2}-1\right]X^{-1} - \left[\frac{a}{b}\right]\left[\frac{a+b}{2}\right]\left[1+\frac{a}{b}X\right]^{-1}\right\}$$

Linearized model and parameters: This function can not be linearized.

Description: This model is also known as the **central *F* distribution**. It looks more like a Chi-square, χ^2, distribution as parameters **a** and **b** get larger. Parameters **a** and **b** are the **numerator** and **denominator degrees of freedom** respectively. Parameter **a** controls the rate of the curve's decrease while **b** controls the height of the curve.

Sample `PROC NLIN` **program for the functional form:**

Because the derivative of the Gamma function is not easily calculated, it is more straightforward, in this case, to use the `DUD` method within `NLIN`. `DUD` is the default method used when no derivatives are given in the program. The SAS function `GAMMA` can be used to evaluate the gamma function. For example, use `GAMMA(7)` to evaluate $\Gamma(7)$.

```
PROC NLIN DATA=FISHER;
  PARAMETERS A=3 B=5;
  AB = A/B;
  A2 = A/2;
  AB2 = (A+B)/2;
  GA = GAMMA(A2);
  GB = GAMMA(B/2);
  GAB = GAMMA(AB2);
  MODEL Y = (AB**A2) * (GAB/GA/GB) * X**(A2-1)
          * (1 + AB*X)**(-AB2);
RUN;
```

Functional form: $Y = \left[\dfrac{a}{b}\right]^{\frac{a}{2}} \dfrac{\Gamma\left[\dfrac{a+b}{2}\right]}{\Gamma\left[\dfrac{a}{2}\right]\Gamma\left[\dfrac{b}{2}\right]} X^{\frac{a}{2}-1} \left[1 + \dfrac{a}{b}X\right]^{-\frac{a+b}{2}}$ $\qquad X > 0$

b = 8 vary a

a = 8 vary b
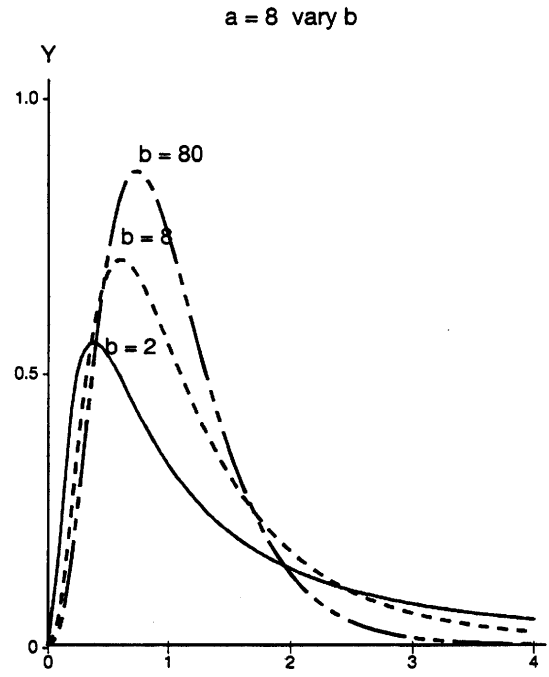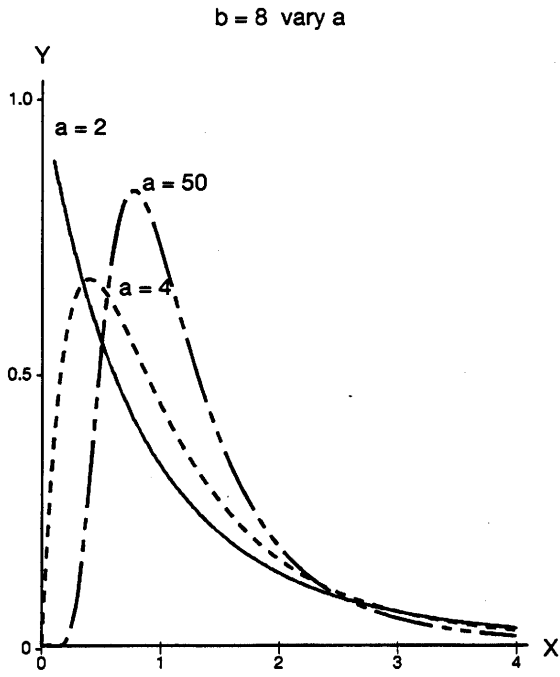
4.8.4 Chi-square distribution

Functional form:
$$Y = \frac{X^{\frac{a}{2}-1} \exp\left[-\frac{X}{2}\right]}{2^{a/2} \, \Gamma\left[\frac{a}{2}\right]} \qquad\qquad X \geq 0, \; a \geq 0$$

Derivative:
$$\frac{dY}{dX} = \frac{1}{2^{a/2} \, \Gamma\left[\frac{a}{2}\right]} X^{\frac{a}{2}-1} \exp\left[-\frac{X}{2}\right] \left\{ \left[\frac{a}{2}-1\right]\frac{1}{X} - \frac{1}{2} \right\}$$

Linearized model and parameters: This function can not be linearized.

Description: The Chi-square distribution with **a** degrees of freedom is commonly denoted by the Greek symbol χ_a^2. The parameter **a** represents the degrees of freedom and controls the height and the shape of curve. See Appendix 1 for a description of the Gamma function, $\Gamma(X)$.

Sample `PROC NLIN` **program for the functional form:**

Because the derivative of the Gamma function is not easily calculated, it is more straightforward, in this case, to use the `DUD` method within `NLIN`. `DUD` is the default method used when no derivatives are given in the program. The SAS function `GAMMA` can be used to evaluate the gamma function. For example, use `GAMMA(7)` to evaluate $\Gamma(7)$.

```
PROC NLIN DATA=CHISQ;
  PARAMETERS A=11.0;
  A2 = A/2;
  GA = GAMMA(A2);
  MODEL Y = (X**(A2-1) * EXP(-X/2))/(2**A2 * GA);
RUN;
```

Functional form:
$$Y = \frac{1}{2^{a/2} \; \Gamma\left[\frac{a}{2}\right]} X^{\frac{a}{2}-1} \exp\left[-\frac{X}{2}\right]$$

vary a
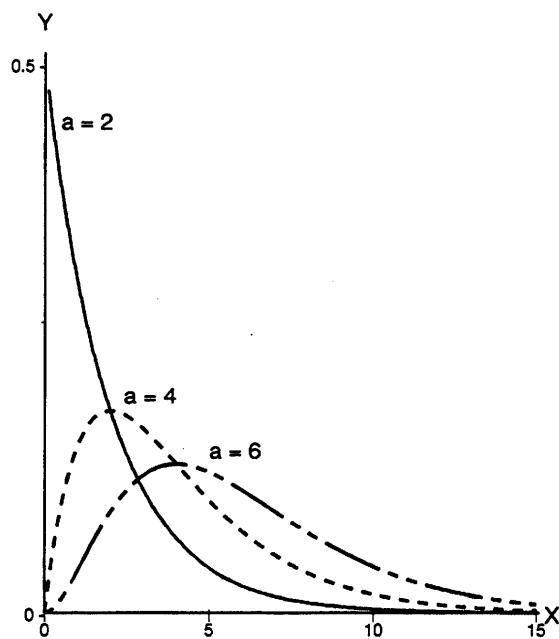
5 CURVE FITTING METHODS

This section explains how to use this handbook to fit curves. The concept of convergence is discussed and some suggestions to correct nonconvergence are provided. Also the criteria for comparing models are briefly described. Two examples are presented to demonstrate the curve fitting process. This is an introduction only and further information should be sought in books such as Hoerl (1954), Ratkowsky (1983), Gallant (1987), and Rawlings (1989).

5.1 How to Select Models

Choosing a model for a set of data can be a difficult task. It is important to incorporate prior information of the system's behaviour into the model. The following steps describe the procedure for selecting a model from those contained in this handbook.

First, plot the data, dependent variable (Y) versus independent variable (X). This will provide some information on the relationship between the dependent and independent variables.

There are several ways to select candidate functional forms to fit a particular data set. A thorough review of the literature may indicate models that have been successful in the past. Theoretical considerations are also important in biological modelling. If the behaviour of the data is known *a priori* (for example, growth rates are often represented by exponential functions), then choose a curve that reflects this knowledge. If no previous research has been done and nothing is known about the data, then the selection of the model must be based on the raw data.

On the scatter plot, draw a smooth curve that fits the data well, noting where the data is increasing, decreasing, or symmetric, and where the maximums, minimums, bounds, or concavity occur. This curve should provide the general shape of the regression model. Examine the graphs provided in this catalog and select a curve that is similar in shape to the graph of the raw data. Consider the following:

- How well does the curve mimic the data?
- How complex is the function? Keep in mind that a simple function is easier to use and interpret.
- Can the parameters of the function be interpreted?

Once a curve is selected, use the sample SAS programs for `PROC NLIN` or `PROC REG` to perform curve fitting.

5.2 How to Choose Starting Values for `PROC NLIN`

`PROC NLIN` requires that starting values for the parameters be estimated. The closer the starting values are to the true parameter values, the more likely that the fitting process will be successful. Finding an appropriate set of starting values can be tricky. Fortunately, there are some strategies for choosing the starting values.

1. Read the descriptions of the parameters carefully, then try to match them with features on the scatter plot.

   For example, suppose the quadratic function in Section 4.1.2 is to be fitted. According to the description in that section, the parameter **B** in the alternative form is the location of the extremum on the X-axis. If a maximum is found near X = 10 on the scatter plot then **B** = 10 would be an appropriate starting value.

2. Compare the scatter plot with the sample graphs provided. This will help to eliminate improper starting values.

   Continuing with the example in point one, the description of the quadratic function states that parameter **A** must be negative for a quadratic curve with a maximum. The size of parameter **A** also controls the spread of the curve so that a large negative value will give a sharp maximum. Therefore, if the scatter plot of experimental data shows a flatter maximum, then a small negative value could be used as the starting value for parameter **A**.

3. If the curve to be fitted can be linearized, it may be possible to obtain starting values of the parameters from `PROC REG`.

   As an example, suppose the logistic function in Section 4.3.11 is to be fitted to a set of data showing a horizontal asymptote at $Y = 15$. This asymptote suggests that $a = 15$ is a reasonable starting value. With this assumption, the Y-variable can be transformed according to the left-hand side of the linearized model $(\ln(a/Y - 1))$. Then a linear regression with `PROC REG` can be performed on the transformed Y- and X-variables. The estimated parameters (slope and intercept) from this linear regression can be used as starting values for parameters **b** and **c** of the logistic curve. This technique is demonstrated in Sit (1992).

4. If there is no clue at all then take an educated guess. Try a range of starting values. Sometimes the parameter estimates from the last iteration of an unsuccessful run would be good starting values for the parameters in the next run.

5.3 What is Convergence?

The `PROC NLIN` procedure in SAS uses an iterative algorithm to estimate the model parameters. At each iteration, the residual sum of squares (SSE)[2] is evaluated. The procedure is said to have **converged** when the SSE is at a minimum. This is detected when the change in the SSE from the previous to the present iteration is less that a preset constant (10^{-8} by default). If the procedure has converged, then the parameter estimates in the last iteration would be the parameter estimates of the model.

If there are local minima on the SSE response curve and the starting values are far from the true parameter values, convergence to the local minimum rather than to the global minimum may occur (see Section 6.5 for definitions of local and global minimum). To check that convergence is to a global minimum, use different sets of starting values to see whether the same solution is reached in all cases. Always plot the fitted model over the raw data to ensure that the fit is reasonable.

In some cases, convergence may not be obtainable. Possible reasons are that the model is inappropriate or the derivatives are incorrect. Rawlings (1988) points out three other reasons for nonconvergence:

1. The model is over-defined. That is, it has too many parameters or is unnecessarily complex.

2. There is insufficient data to fully characterize the response curve. Note that a model may appear over-defined in this case.

3. The model is poorly parameterized with several parameters playing similar roles. This could lead to a similar fit from different combinations of parameter values and may be reflected in highly correlated (0.98 or higher) parameter estimates.

If `PROC NLIN` fails to converge, try the following:

- check the derivatives to make sure that they are properly specified;

- use a different set of starting values — even the last set of iterative estimates may do;

- use a different iterative method — `METHOD = MARQUARDT` sometimes works when the default method (Gauss-Newton) does not; or

- use a different model.

Rawlings (1988) shows how to diagnose possible causes of nonconvergence (see Example 14.2). As well, Section 5.5 presents an example of a nonconvergence fit and provides steps to correct the problem.

5.4 Model Comparisons and Model Validation

Several common criteria are available to compare regression models. These include: residual mean squares MSE; coefficient of determination (R^2); and adjusted coefficient of determination (R^2_{adj}).

---

[2] SSE is sometimes denoted as SSR (Handbook No. 1). Both notations are commonly used in the literature. SSE is used here because SAS outputs use this notation.

Residual mean squares, MSE, is defined as the residual sum of squares (SSE) divided by its degrees of freedom. It is an estimate of the variance of Y(σ^2). MSE is automatically calculated when `PROC REG` or `PROC NLIN` is used. A model with small MSE is more desirable.

The coefficient of determination, R^2, is the proportion of corrected total sum of squares of the dependent variable that is "explained" by the independent variable(s) in the model (Rawlings 1988):

$$R^2 = \frac{SS(\text{regression})}{SS(\text{corrected total})}$$

The larger the R^2, the more variation that is accounted for by the model. Note that the regression SS displayed in a `PROC NLIN` output is uncorrected for the mean. Therefore, R^2 must be computed with the following equation:

$$R^2 = 1 - \frac{SS(\text{residual})}{SS(\text{corrected total})}$$

The adjusted coefficient of determination (R^2_{adj}) is a rescaling of R^2 by the degrees of freedom so that it involves a ratio of mean squares rather than sums of squares. Similar to R^2, it should be computed from the residual mean squares:

$$R^2_{adj} = 1 - \frac{MS(\text{residual})}{MS(\text{corrected total})}$$

The adjusted coefficient of determination is more comparable than R^2 for models that involve different numbers of parameters. A model with large R^2_{adj} is more favourable. Because of its structure, the R^2_{adj} criterion often leads to the same conclusion as the MSE criterion. See Section 5.6 for a comparison of two models from the statistical and subjective points of view.

Once a regression model is chosen, the model should be validated to confirm its effectiveness. Validation of the model requires comparing the fitted equation with an independent set of data. See Rawlings (1988, Section 7.6) for an in-depth discussion of model validation.

5.5 Example 1

This example is taken from Little and Hills (1978). A researcher is interested in finding the relationship between the yield in kilograms of lima beans and the time of harvest. In an experiment, the yield is recorded for various harvest dates. The data is shown in Table 1.

TABLE 1. Lima bean yield data

Date (X)	Yield (Y)
0	27.4
4	39.3
7	46.2
10	47.8
13	44.5
18	24.5

The variable Y represents the yield in kilograms of lima beans; the variable X represents the number of days from the initial harvest at $X = 0$. To investigate the relationship, we first plot (Figure 1) yield (Y) versus harvest date (X) from the data in Table 1.
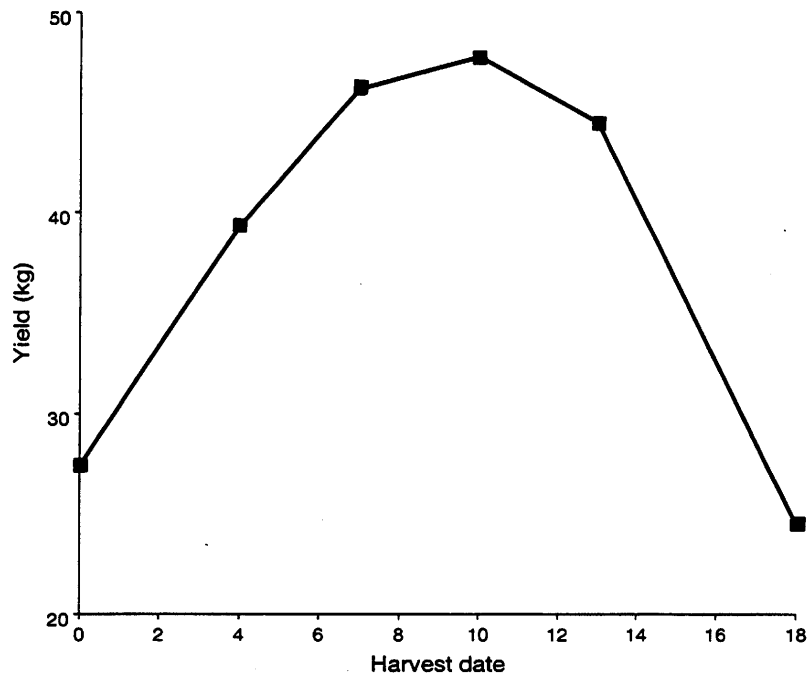
FIGURE 1. Lima bean yield versus harvest dates.

No other experimental work has been done on lima bean yield, so nothing is known about the relationship between yield and date of harvest. Therefore curve selection must be based on the scatter plot.

The graphed data in Figure 1 resembles the shape of a parabola. According to Section 4.1.2, either `PROC REG` or `PROC NLIN` can be used to fit the data. The two approaches are demonstrated below.

5.5.1 Fitting the model using `PROC REG`

The linearized model of the parabola has the form:

$$Y = a + bX + cX^2$$

We can fit the model by performing a multiple regression on Y, lima bean yield, with X, the harvest date, and X^2 as the independent variables. In the following SAS program, a variable called DATE2 is created in the data step for the X^2 values. The `PROC REG` step in this program is identical to the sample program given in Section 4.1.2 except for the change in variable names.

SAS program using PROC REG

```
TITLE1  'LIMA BEAN EXAMPLE';
TITLE2  'SECOND DEGREE POLYNOMIAL FIT - PROC REG';

DATA BEAN;
  INPUT  DATE YIELD;
  DATE2 = DATE**2;
  CARDS;
  0   27.4
  4   39.3
  7   46.2
  10  47.8
  13  44.5
  18  24.5
;
PROC REG DATA=BEAN;
  MODEL YIELD = DATE DATE2;

RUN;
```

SAS output

<div align="center">

LIMA BEAN EXAMPLE

SECOND DEGREE POLYNOMIAL FIT - PROC REG

</div>

Model: MODEL1
Dependent Variable: YIELD

<div align="center">

Analysis of Variance

</div>

Source	DF	Sum of Squares	Mean Square	F Value	Prob>F
Model	2	493.57666	246.78833	95.265	0.0019
Error	3	7.77167	2.59056		
C Total	5	501.34833			

Root MSE	1.60952	R-square	0.9845	
Dep Mean	38.28333	Adj R-sq	0.9742	
C.V.	4.20423			

<div align="center">

Parameter Estimates

</div>

Variable	DF	Parameter Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
INTERCEP	1	26.331776	1.47950636	17.798	0.0004
DATE	1	4.783178	0.36857319	12.978	0.0010
DATE2	1	-0.269021	0.01950494	-13.792	0.0008

The parameter estimates are: $a = 26.3$, $b = 4.78$, and $c = -0.269$.

The fitted model is: $Y = 26.3 + 4.78X - 0.269X^2$

5.5.2 Fitting the model using `PROC NLIN`

Either the functional form or the alternative form of the quadratic can be fitted with `PROC NLIN`. In this example, we will use the alternative form:

$$Y = A(X - B)^2 + C$$

`PROC NLIN` uses an iterative algorithm to estimate the parameters and requires starting values for the first iteration. The closer the starting values are to the true values, the faster the convergence. Appropriate starting values can be found by examining the data plot in Figure 1.

Starting value for A

As stated in Section 4.1.2, if **A** is less than zero then the parabola is concave down. Therefore, we can try **A** $= -1$ as the starting value.

Starting value for B

The maximum or the minimum of a parabola occurs at **X** $=$ **B** and **Y** $=$ **C**. We can estimate from Figure 1 that the maximum of the curve is approximately at **X** $= 10$ and **Y** $= 47.8$. Therefore, we can try **B** $= 10$ and **C** $= 48$ as the starting values.

The following is the SAS program for fitting the alternative form of the parabola. The `PROC NLIN` step is identical to the sample program in Section 4.1.2, except for the two statements (in bold italics) for assigning proper values to X and Y.

SAS program using `PROC NLIN`

```
TITLE1    'LIMA BEAN EXAMPLE';
TITLE2    'SECOND DEGREE POLYNOMIAL FIT - PROC NLIN';

DATA BEAN;
  INPUT DATE YIELD;
  CARDS;
  0   27.4
  4   39.3
  7   46.2
  10  47.8
  13  44.5
  18  24.5
;
PROC NLIN DATA=BEAN;
  PARAMETERS A=-.1 B=10 C=48;
  X = DATE;
  Y = YIELD;
  XB = (X-B);
  MODEL Y = A*XB**2 + C;
  DER.A = XB**2;
  DER.B = -2*A*XB;
  DER.C = 1;
RUN;
```

SAS output

```
                              LIMA BEAN EXAMPLE
                   SECOND DEGREE POLYNOMIAL FIT – PROC NLIN


                   Non-Linear Least Squares Iterative Phase
                  Dependent Variable YIELD Method: Gauss-Newton
         Iter          A               B                C   Sum of Squares
            0     -0.100000       10.000000        48.000000     438.390000
            1     -0.269021        7.013777        47.261441     225.703710
            2     -0.269021        8.889967        46.645943      13.152283
            3     -0.269021        8.889967        47.592921       7.771674
     WARNING: Step size shows no improvement.
     WARNING: PROC NLIN failed to converge.
```

```
Non-Linear Least Squares Summary Statistics      Dependent Variable YIELD


         Source               DF      Sum of Squares    Mean Square


         Regression            3       9287.2583255    3095.7527752
         Residual              3          7.7716745       2.5905582
         Uncorrected Total     6       9295.0300000


         (Corrected Total)     5        501.3483333

WARNING: PROC NLIN failed to converge.
```

```
      Parameter        Estimate      Asymptotic                 Asymptotic 95 %
                                      Std. Error             Confidence Interval
                                                          Lower            Upper
             A      -0.26902111              0     -0.269021108    -0.269021108
             B       8.88996693              0      8.889966930     8.889966930
             C      47.59292137              0     47.592921371    47.592921371
```

```
                        Asymptotic Correlation Matrix

      Corr                           A               B                 C
        --------------------------------------------------------------------
             A                       1     -0.039491723      -0.71679787
             B              -0.039491723              1       0.0547551726
             C               -0.71679787     0.0547551726              1
```

There is a warning in the SAS output stating that the iterations in PROC NLIN did not converge. As discussed in Section 5.1, nonconvergence can be caused by:

- the starting values are too far from the true parameter values;
- the derivatives given in PROC NLIN are incorrect; or
- the model chosen is incorrect.

We can safely eliminate the last two causes in this example. To solve the problem, we can a try different fitting method with the METHOD = option, or we can try a different set of starting values. Checking the last two iterations, we see that the estimates for **A** and **B** are quite stable, but the estimate for **C** and the sum of squares (SSE) value fluctuate slightly. The small SSE (7.772) suggests that the estimates are quite close to the true values. Sometimes the last SSE computed is so close to its minimum that new parameter estimates cannot be found to further reduce the SSE. When this happens, PROC NLIN gives up and declares nonconvergence. We can re-run the program using a different set of starting values. In this example, we will use the estimates from the last iteration as the starting values. That is, we will replace the PARAMETER statement with the following:

PARAMETER A = −0.26 B = 8.9 C = 47.6;

The following is the iteration portion of the output using this set of starting values.

```
                      LIMA BEAN EXAMPLE                              1
                SECOND DEGREE POLYNOMIAL FIT - PROC NLIN


                 Non-Linear Least Squares Iterative Phase
               Dependent Variable YIELD Method: Gauss-Newton
     Iter           A              B              C   Sum of Squares
        0      -0.260000       8.900000      47.600000       8.943748
        1      -0.269021       8.889619      47.592894       7.771682
        2      -0.269021       8.889967      47.592921       7.771674
        3      -0.269021       8.889967      47.592921       7.771674
NOTE: Convergence criterion met.
```

The convergence criterion is met and the fitted model is:

$$Y = -0.269(X - 8.89)^2 + 47.6$$

It is left to the readers to check that this fitted model is the same as the model obtained from PROC REG. Readers can refer to the books listed in the references for discussions about interpreting regression results from SAS output or determining the goodness-of-fit of the model.

5.6 Example 2

The city of San Diego, California wants to model its population growth over time. The population of the city in the last 11 decades is shown in Table 2.

TABLE 2. San Diego population data from 1860 to 1960

Year of Census	Decades from 1860 (X)	Population (Y)
1860	0	731
1870	1	2 300
1880	2	2 636
1890	3	16 159
1900	4	17 700
1910	5	39 578
1920	6	74 361
1930	7	147 995
1940	8	203 341
1950	9	334 387
1960	10	573 224

The variable X is the number of decades from the first census in 1860 ($X = 0$ for the first census). The variable Y is the city's population. Figure 2 shows a scatter plot of this data.
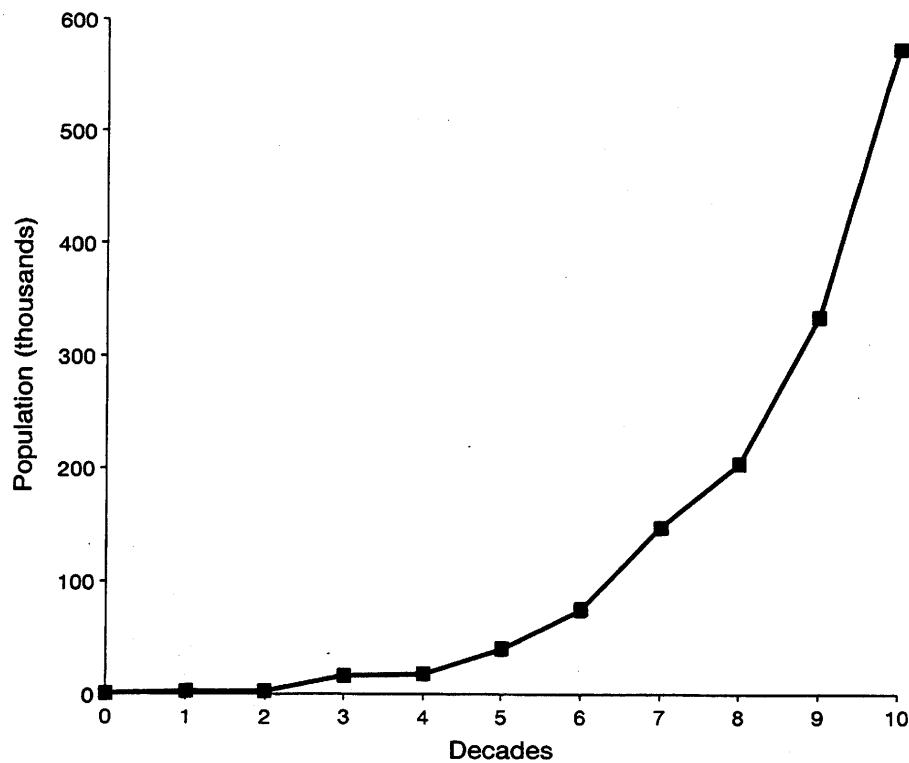
FIGURE 2. San Diego population versus decades from first census

94

No previous information about the population growth in San Diego is assumed. Nevertheless, growth data can often be represented by an **exponential function**, which is also suggested by the scatter plot. The type IV exponential form in Section 4.3.4 is the most general form with a minimum number of parameters and both `PROC REG` and `PROC NLIN` can be used to fit the data.

5.6.1 Fitting the model using `PROC REG`

The linearized form of model:

$$Y = ab^X$$

is

$$\ln(Y) = A + BX$$

where $a = e^A$ and $b = e^B$. The following is an SAS program to fit the linearized model.

SAS program using `PROC REG`

```
TITLE1 'SAN DIEGO POPULATION EXAMPLE';
TITLE2 'EXPONENTIAL GROWTH FUNCTION - PROC REG';

DATA POPN;
INPUT DECADE POPN;
LOGP = LOG(POPN);
CARDS;
0   731
1   2300
2   2636
3   16159
4   17700
5   39578
6   74361
7   147995
8   203341
9   334387
10 573224
;
PROC REG;
  MODEL LOGP = DECADE;
RUN;
```

SAS output

Model: MODEL1
Dependent Variable: LOGP

                          Analysis of Variance

                              Sum of   Mean
Source   DF   Squares   Square   F Value   Prob>F

Model   1   47.31372   47.31372   333.001   0.0001
Error   9   1.27874   0.14208
C Total   10   48.59247

        Root MSE   0.37694   R-square   0.9737
        Dep Mean   10.32664   Adj R-sq   0.9708
        C.V.   3.65016

                          Parameter Estimates

                    Parameter   Standard   T for H0:
Variable   DF   Estimate   Error   Parameter=0   Prob > |T|

INTERCEP   1   7.047444   0.21262209   33.145   0.0001
DECADE   1   0.655839   0.03593969   18.248   0.0001

The parameter estimates are $A = 7.05$ and $B = 0.656$. The fitted equation is:

$$\ln(Y) = 7.05 + 0.656X$$

Transforming the estimates to the non-linear form:

$$a = e^{7.05} = 1150$$

$$b = e^{0.656} = 1.93$$

The fitted model is:

$$Y = 1150\,(1.93)^X$$

5.6.2. Fitting the model using PROC NLIN

We can use the estimates from PROC REG as the starting values for the parameters. However, for the sake of illustration, we will choose starting values based on the plotted curve in Figure 2.

Starting value for a

For the model $Y = ab^X$, at $X = 0$, $Y = \mathbf{a}$, reasonable starting value is $\mathbf{a} = 700$. (In table 2, population $= 731$ in 1860.)

Starting value for b

The graphs in Section 4.3.4 indicated that as **b** gets bigger, the curves get steeper. We can begin with b = 1.5. The PROC NLIN step in the following program is identical to the sample program in Section 4.3.4 except for the two statements (in bold italics) for assigning proper values to X and Y.

SAS program using PROC NLIN

```
TITLE1  'SAN DIEGO POPULATION EXAMPLE';
TITLE2  'EXPONENTIAL GROWTH MODEL – PROC NLIN';

DATA POPN;
  INPUT DECADE POPN;
  CARDS;
 0  731
 1  2300
 2  2636
 3  16159
 4  17700
 5  39578
 6  74361
 7  147995
 8  203341
 9  334387
10  573224
;
PROC  NLIN  DATA=POPN;
  PARAMETERS A=700 B=1.5;
  X = DECADE;
  Y = POPN;
  MODEL Y = A*B**X;
  DER.A = B**X;
  DER.B = A*X*B**(X-1);
RUN;
```

SAS output

<pre>
                    SAN DIEGO POPULATION EXAMPLE
                  EXPONENTIAL GROWTH MODEL - PROC NLIN

                 Non-Linear Least Squares Iterative Phase
                 Dependent Variable Y   Method: Gauss-Newton
       Iter   A   B   Sum of Squares
        0   700.000000   1.500000   437333614628
        1   898.141784   1.947689   22313337398
        2   1527.141021   1.792251   10147766051
        3   1922.094536   1.749936   8883368862
        4   2630.817000   1.694216   7330093063
        5   3555.319008   1.653680   1863046080
        6   3594.316548   1.659922   848171927
        7   3597.263952   1.659614   847713812
        8   3597.204591   1.659616   847713811
NOTE: Convergence criterion met.

    Non-Linear Least Squares Summary Statistics   Dependent Variable Y

    Source   DF   Sum of Squares   Mean Square

    Regression   2   510485940983   255242970492
    Residual   9   847713811   94190423
    Uncorrected Total   11   511333654794

    (Corrected Total)   10   329978413181
    Parameter   Estimate   AsymptoticAsymptotic 95 %
                                 Std. Error   Confidence Interval
                                                   Lower   Upper
         A   3597.204591 489.99575478 2488.7477532 4705.6614287
         B   1.659616   0.02385320   1.6056563   1.7135766

                  Asymptotic Correlation Matrix
              Corr   A   B

                             A   1 -0.995015492
                             B -0.995015492   1
</pre>

The parameter estimates are **a** = 3600 and **b** = 1.66 and the fitted model is:

$$Y = 3600 \, (1.66^X)$$

Notice that the fitted model from PROC REG is quite different from the fitted model from PROC NLIN. This is because the two SAS procedures use different statistical models — PROC REG uses a model with multiplicative errors while PROC NLIN uses a model with additive errors. In other words, PROC REG minimizes the transformed residuals whereas PROC NLIN minimizes the untransformed residuals (see Section 3.2 for discussion). Which model is better? The answer depends on the statistical and subjective considerations.

5.6.3 Statistical considerations for a "better" model

Two or more models can be compared statistically on the basis of the residual mean squares (MSE), the coefficient of determination (R^2), or the adjusted coefficient of determination (R^2_{adj}). These criteria are briefly described in Section 5.4. Since R^2_{adj} and MSE are equivalent for model comparisons, only MSE will be calculated in this example.

MSE is defined as:

$$MSE = \frac{SSE}{df}$$

where SSE is the sum of the squared differences between the observed value, Y_i, and the predicted value, $\hat{Y}_i$. That is,

$$SSE = \sum_{i=1}^{n} (Y_i = \hat{Y}_i)^2$$

With `PROC NLIN`, the model $Y = 3597.21(1.66^X)$ was fitted, and the MSE of the model is 94 190 423. This is given in the SAS output ANOVA table.

With `PROC REG`, the model $\ln(Y) = 7.05 + 0.60X$ was fitted. Because the SSE calculated by SAS is for the linearized model only, we must calculate the SSE for the transformed model $Y = 1150(1.93^X)$ using equation 1. To avoid rounding error, SSE should be calculated using the results directly from `PROC REG`, as shown in the following SAS code.

SAS code to compute SSE:

```
PROC REG;
  ID POPN;
  MODEL  LOGP = DECADE;
  OUTPUT  OUT=PRED  P=PRED;

DATA PRED;
  SET PRED;
  PHAT = EXP(PRED);
  SSE + (POPN-PHAT)**2;

PROC PRINT;
  VAR POPN SSE;

RUN;
```

SAS output:

POPN	SSE
731	175490.10
2300	182611.94
2636	2849115.37
16159	65793951.37
17700	69223801.47
39578	150994331.41
74361	392077957.20
147995	1591719792.04
203341	1818966177.48
334387	9292094235.98
573224	65756264558.98

Cumulative SSE's are calculated for the observations. The SSE of the model is the cumulative SSE of the last observation — that is, $SSE = 65\ 756\ 264\ 558.98$ and $MSE = \frac{SSE}{9} = 7\ 306\ 251\ 617.66$. The MSE of the fitted model from `PROC NLIN` is smaller than the MSE of the fitted model from `PROC REG`; therefore, based on the MSE criteria,

$$Y = 3600.21\,(1.66^X)$$

is a better model.

5.6.4 Subjective consideration for a "better" model

Figure 3 is a plot of the two models and the observed data. Both models are quite good for the early decades (decades ≤6), but for the later decades, the model from PROC NLIN provides a better fit for the data.
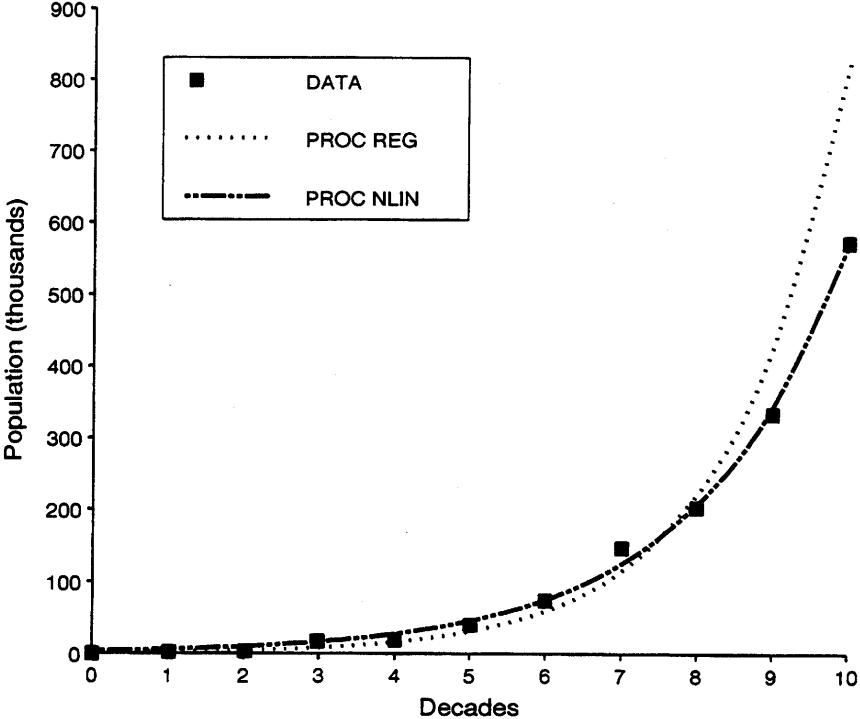

FIGURE 3. Fitted models and the observed data.

6 BASIC ATTRIBUTES OF CURVES

This section describes some basic attributes of curves that have been used throughout this handbook.

A function is expressed as:

$$Y = f(X)$$

where Y is a function of X. The following symbols are used in this chapter.

- Y_c = the value of Y at $X = c$

- $\dfrac{dY}{dX}$ = the first derivative of Y with respect to X

- $\dfrac{d^2Y}{dX^2}$ = the second derivative of Y with respect to X

A function $Y = f(X)$ is **defined** at a point, say $X = a$, if it has a finite value at $X = a$. Some possible causes of undefined functions are:

- Division by zero, e.g., $Y = \dfrac{1}{X-1}$ is undefined at $X = 1$.

- Logarithm of a non-positive number, e.g., $Y = \log(X)$ is undefined at $X \leq 0$.

- Even roots of a negative number, e.g., $Y = \sqrt{X}$ is undefined at $X < 0$.

6.1 Increasing and Decreasing Functions

A function is **increasing** if the Y-values get bigger as the X-values get bigger. A function is **decreasing** if the Y-values get smaller as the X-values get bigger. Figures 4a and 4b show an example of each type of function.
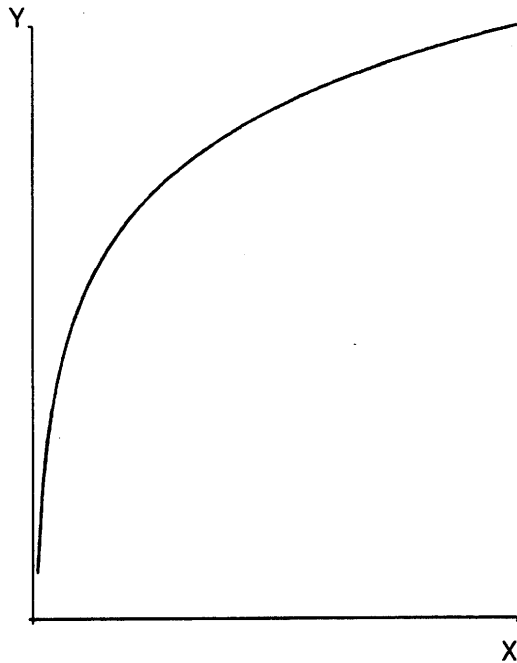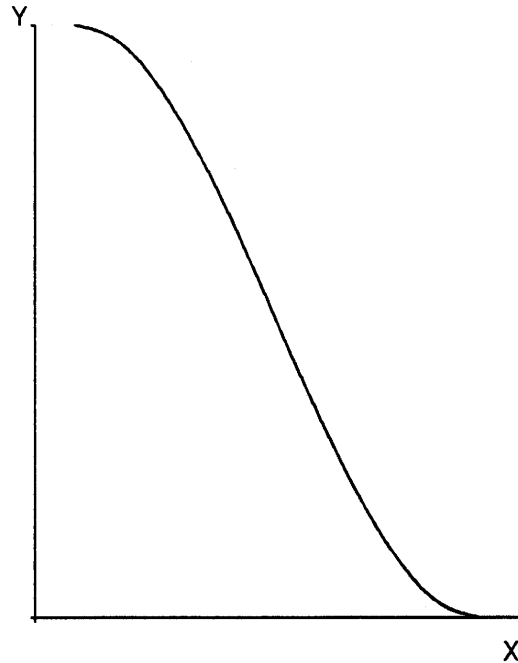

FIGURE 4a. An increasing function.

FIGURE 4b. A decreasing function.

The first derivative can be used to test whether a function is increasing or decreasing at a point, say at $X = a$:

1. A function is increasing at $X = a$ if $\frac{dY}{dX} > 0$ at $X = a$.

2. A function is decreasing at $X = a$ if $\frac{dY}{dX} < 0$ at $X = a$.

6.2 Symmetric Functions

A function is **symmetric** about a certain line if its graph on one side of the line is a mirror image of the graph on the other side of the line (Figure 5). For example, a quadratic function is a symmetric function.
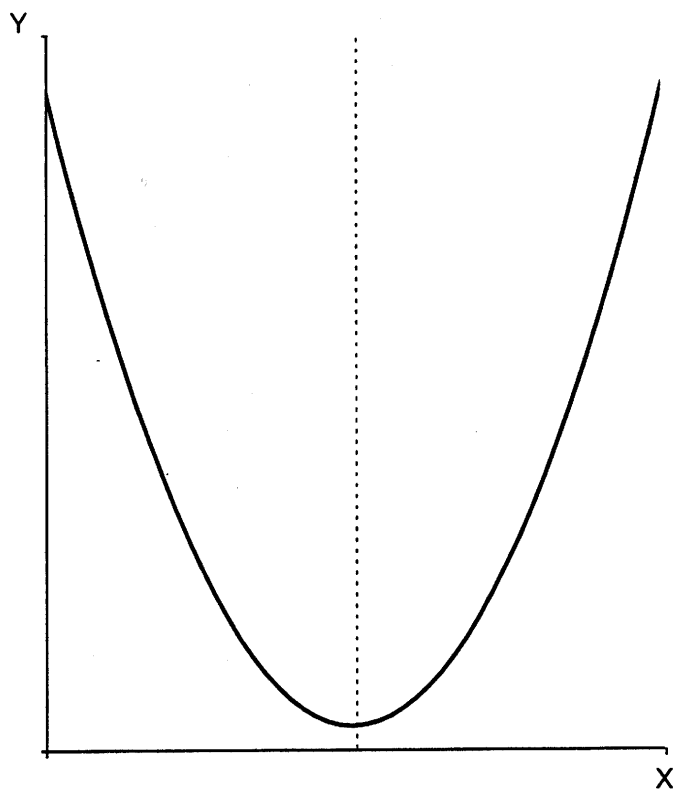

FIGURE 5. A symmetric function.

6.3 Asymptotes

An Asymptote to a curve is a straight line that the curve approaches but never reaches.

The line X = **a** is a **vertical asymptote** for the curve Y= f(X) if Y approaches positive infinity or negative infinity as X approaches **a**.

The line Y = **b** is a **horizontal asymptote** for the curve Y = f(X) if Y approaches **b** when X approaches positive infinity or negative infinity.

Figure 6 is a plot of the function Y $= \dfrac{1}{X-1}$. The dotted lines are the asymptotes: a horizontal asymptote at Y = 0, and a vertical asymptote at X = 1.
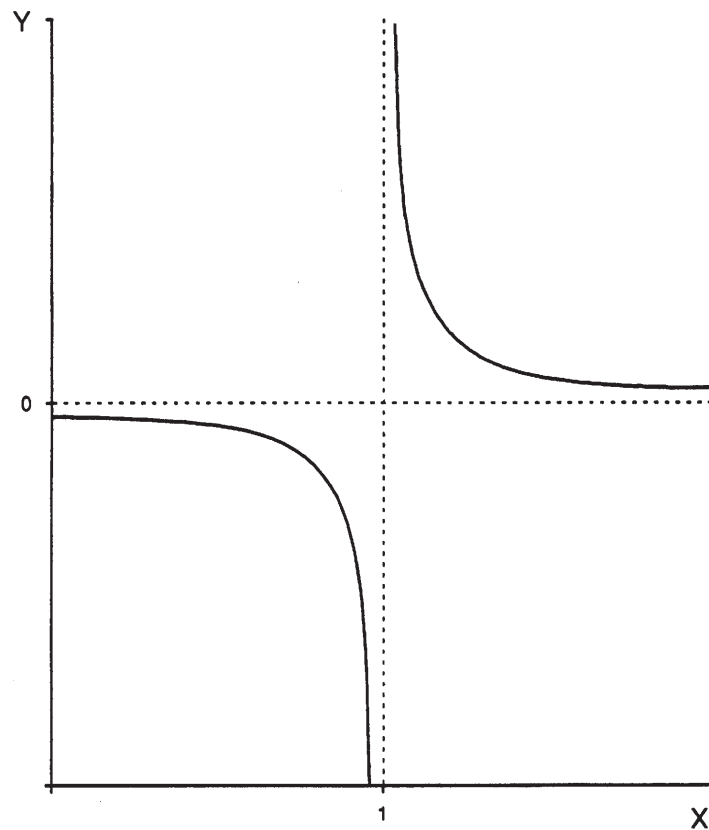

FIGURE 6. A function with asymptotes.

6.4 Concavity of a Function

A curve that is **concave upward** has an apparent U-shape. A curve that is **concave downward** has an upside down U-shape. Concavity of a function can be tested with the second derivative (provided that it is defined):

1. A function $Y = f(X)$ is concave upward at $X = a$ if $\dfrac{d^2Y}{dX^2} > 0$ at $X = a$.

2. A function $Y = f(X)$ is concave downward at $X = a$ if $\dfrac{d^2Y}{dX^2} < 0$ at $X = a$.

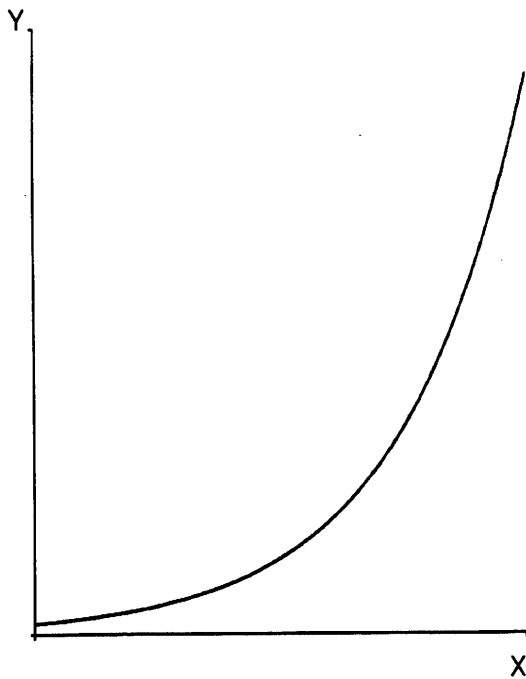Figures 7a and 7b show an example of each type of function.
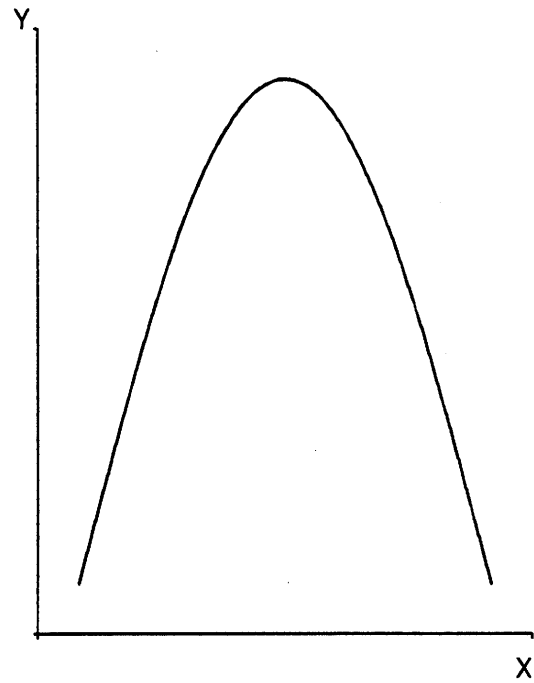

FIGURE 7a. A concave upward function.

FIGURE 7b. A concave downward function.

6.5 Maximum and Minimum

The value Y_c is a **local maximum** of the function $Y = f(X)$, if Y_c is the largest value of Y in the neighbourhood of $X = c$. Similarly, the value Y_c is a **local minimum** of the function $Y = f(X)$, if Y_c is the smallest value of Y in the neighbourhood of $X = c$. A local **extremum** is a value of Y that is either a local minimum or a local maximum.

Local extremums occur at the **critical points** of f — that is, when $\frac{dY}{dX} = 0$ or is undefined. If $X = c$ is a critical point of the function $Y = f(X)$ then:

Y_c is a local maximum if $\frac{d^2Y}{dX^2} < 0$ at $X = c$; and

Y_c is a local minimum if $\frac{d^2Y}{dX^2} > 0$ at $X = c$.
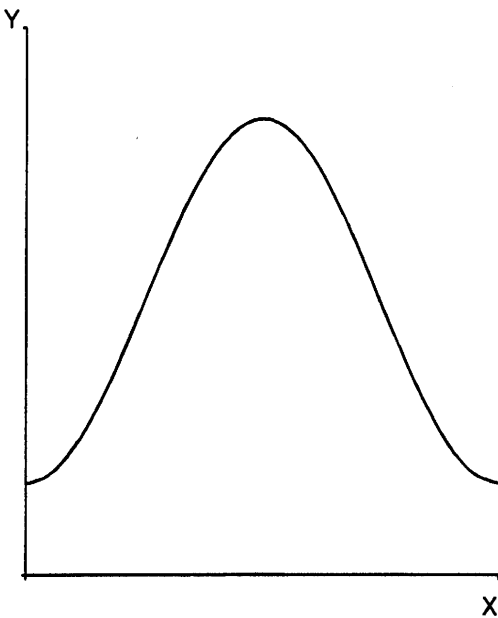
Figures 8a and 8b show example functions with extremums.

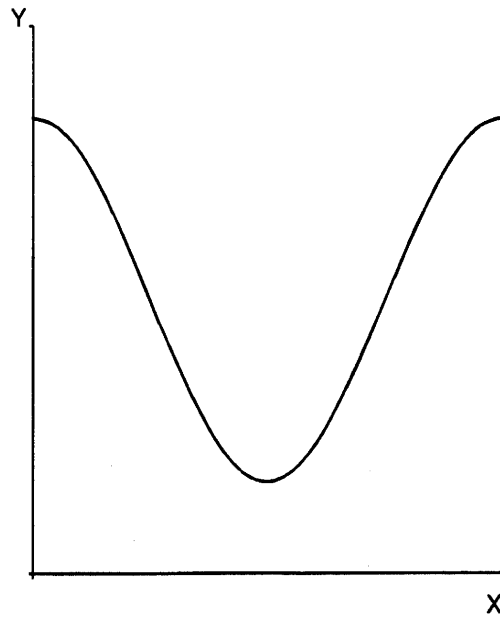FIGURE 8a. A function with a local maximum.

FIGURE 8b. A function with a local minimum.

Y_c is a **global maximum** if it is the largest value of Y for all possible values of X. Conversely, Y_c is a **global minimum** if it is the smallest value of Y for all possible values of X.

6.6 Point of Inflection

The **point of inflection** is where a function changes concavity from concave upward on one side to concave downward on the other side.

The point (c, Y_c) is a point of inflection if $\dfrac{d^2Y}{dX^2} = 0$ or is undefined.

Figure 9 represents the curve $Y = 8X^5 - 5X^4 - 20X^3$ with the local extremums and inflection points identified. The global maximum is positive infinity and the global minimum is negative infinity.
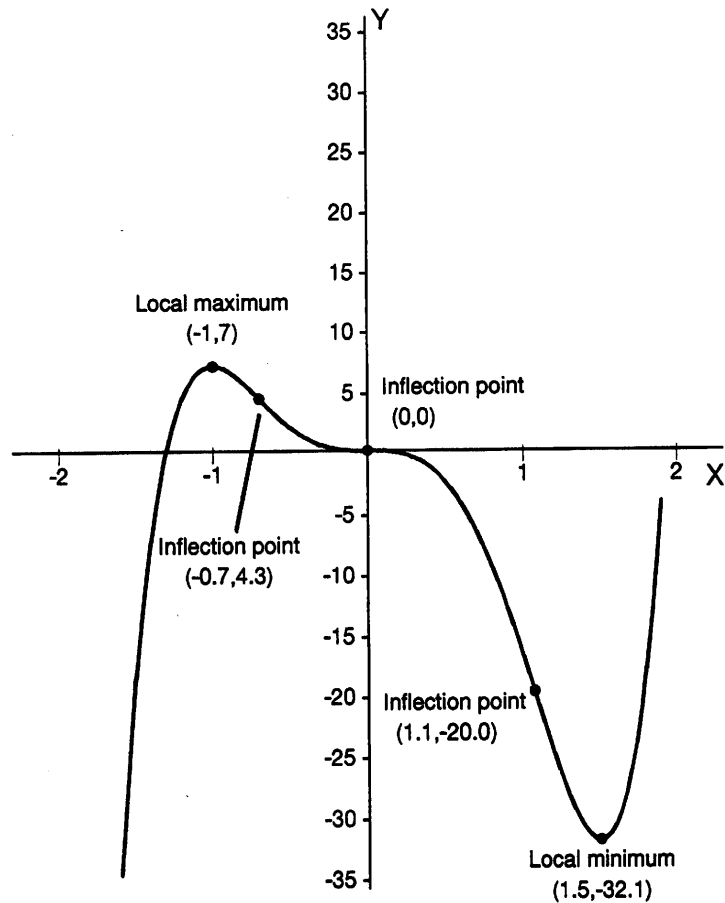

FIGURE 9. The graph of $Y = 8X^5 - 5X^4 - 20X^3$ showing the local extremums and inflection points.

APPENDIX 1: GAMMA FUNCTION

The Gamma Function has the form:

$$\Gamma(a) = \int_0^\infty Y^{a-1} e^{-Y} dY \qquad a > 0$$

The Gamma function has the following special properties that can be useful in evaluating $\Gamma(a)$:

1. $\Gamma(a+1) = a\Gamma(a)$

2. $\Gamma(a+1) = a!$ if a is an integer.

3. $\Gamma\left(\dfrac{1}{2}\right) = \sqrt{\pi}$

4. $\Gamma(1) = 1$

Example:

For $a = \dfrac{5}{2}$,
$$\Gamma(a) = \Gamma\left(\frac{5}{2}\right)$$

$$= \Gamma\left(\frac{3}{2} + 1\right)$$

$$= \left(\frac{3}{2}\right)\Gamma\left(\frac{3}{2}\right) \qquad\qquad \text{Property 1}$$

$$= \left(\frac{3}{2}\right)\Gamma\left(\frac{1}{2} + 1\right)$$

$$= \left(\frac{3}{2}\right)\left(\frac{1}{2}\right)\Gamma\left(\frac{1}{2}\right) \qquad\qquad \text{Property 1}$$

$$= \left(\frac{3}{2}\right)\left(\frac{1}{2}\right)\sqrt{\pi} = \frac{3\sqrt{\pi}}{4} \qquad\qquad \text{Property 3}$$

Evaluating the Gamma function in SAS:

$\Gamma(a)$ can be evaluated in SAS by the function GAMMA. For example, to compute $G = \Gamma(9)$, one would use the command `G = GAMMA(9);`.

BIBLIOGRAPHY

Bard, Y. 1974. Nonlinear parameter estimation. Academic Press, New York, N.Y.

Bergerud, W. 1991. Handbook No. 1: Pictures of linear models. B.C. Min. of For., Res. Br. Victoria, B.C. Biometrics Information Handb. Ser.

Beyer, W.H. 1984. CRC Standard mathematical tables. 27th ed. CRC Press, Boca Raton, Fla.

Bredenkamp, B.V. and T.G. Gregoire. 1988. A forestry application of Schnute's generalized growth function. For. Sci. 34 (3): 790-797.

Campbell, H.E. and P.F. Dierker. 1982. Calculus with analytic geometry. 3rd ed. Prindle, Weber, and Schmidt, Boston, Mass.

Daniel, C. and F. Woods. 1980. Fitting equations to data. 2nd ed. John Wiley and Sons, New York, N.Y.

Edwards, C.H. Jr. and D.E. Penney. 1982. Calculus and analytic geometry. Prentice-Hall, Englewood Cliffs, N.J.

Fletcher, R.I. 1975. A general solution for the complete Richards function. Mathematical Biosciences 27:349–360.

Freedman, D., R. Pisani, and R. Purves. 1978. Statistics. W.W. Norton Company, New York, N.Y.

Freund, J.E. and R.E. Walpole. 1987. Mathematical statistics, 4th ed. Prentice-Hall, Englewood Cliffs, N.J.

Gallant, A.R. 1987. Nonlinear statistical models. John Wiley and Sons, New York, N.Y.

Hoerl, A.E. 1954. Fitting curves to data. Chemical business handbook. J.H. Perry (editor). McGraw-Hill, New York, N.Y.

Hunt, R. 1982. Plant growth curves: the functional approach to plant growth analysis. Edward Arnold, London.

Jenson, C.E. 1964. Algebraic description of forms in space, USDA, FS Central States Forest Experiment Station, Columbus, Ohio.

Jenson, C.E. and J.W. Homeyer. 1970. Matchacurve-1 for algebraic transforms to describe sigmoid- or bell-shaped curves. U.S. Dep. of Agric. For. Serv., Intermountain For. Range Exp. Sta., Ogden, Utah.

_____. 1971. Matchacurve-2 for algebraic transforms to describe curves of the class X^n. U.S. Dep. of Agric. For. Serv., Intermountain For. Range Exp. Sta., Ogden, Utah.

Jenson, C.E. 1973. Matchacurve-3: multiple-component and multidimensional mathematical models for natural resource studies. U.S. Dep. of Agric. For. Serv., Intermountain For. Range Exp. Sta., Ogden, Utah.

_____. 1976. Matchacurve-4: segmented mathematical descriptors for asymmetric curve forms. U.S. Dep. of Agric. For. Serv., Intermountain For. Range Exp. Sta., Ogden, Utah.

Kozak, A. Forestry 466 course notes. Faculty For., Univ. B.C. Vancouver, B.C.

Leech, J.W. and I.S. Ferguson. 1981. Comparison of yield models for unthinned stands of radiata pine. Aust. For. Res. 11:231–245.

Little, T.M. and F.J. Hills. 1978. Agricultural experimentation design and analysis. John Wiley and Sons, New York, N.Y.

Nokoe, S. 1980. Nonlinear models fitted to stand volume-age data compare favourably with British Columbia Forest Service hand-drawn volume-age curves. Can. J. For. Res. 10:304–307.

Nelder, J.A. 1966. Inverse polynomials. Biometrics 22:128–141.

Parton, W.J. and G.S. Innis. 1972. Some graphs and their functional forms. Nat. Resource Ecology Lab. Colo. State Univ. Colo., Tech. Rep. No. 153.

Ratkowsky, D.A. 1983. Nonlinear regression modeling. Marcel Dekker, New York, N.Y.

_____. 1990. Handbook of nonlinear regression models. Marcel Dekker, New York, N.Y.

Rawlings, J.O. 1988. Applied regression analysis: a research tool. Wadsworth and Brooks, Belmont, Calif.

Richards, F.J. 1959. A flexible growth function for empirical use. J. Exp. Bot. Vol. 10 (29):290–300.

SAS Institute Inc. 1988a. SAS/STAT User's Guide. Release 6.03 ed., SAS Institute Inc., Cary, N.C.

_____. 1988b. SAS/Graph User's Guide. 1988. Release 6.03 ed., SAS Institute Inc., Cary, N.C.

Schlotzhauer, S.D. and R.C. Littell. 1987. SAS system for elementary statistical analysis. SAS Institute, Cary, N.C.

Schnute, J. 1981. A versatile growth model with statistically stable parameters. Can. J. Fish. Aquat. Sci. 38:1128–1140.

Schumacher, F.X. 1939. A new growth curve and its application to timber-yield studies. J. For. 37:819–820.

Sit, V. 1992. A repeated measures example. B.C. Min. For. Res. Br., Victoria, B.C., Biom. Infor. Pamp. No. 39.

Spain, J.D. 1982. BASIC microcomputer models in biology. Addison-Wesley Publishing Company, Mass.

Sweda, T. 1984. Theoretical growth equations and their applications in forestry. Nagoya Univ. For. Nagoya, Japan, Bull. Nagoya Univ. For. No. 7.

Weibull, W. 1951. A statistical distribution function of wide applicability. J. Appl. Mech. 18:293–296.

Weisburg, S. 1985. Applied Linear Regression, 2nd ed. John Wiley and Sons, New York, N.Y.

Yang, R.C., A. Kozak, and J.H.G. Smith. 1978. The potential of Weibull-type functions as flexible growth curves. Can. J. For. Res. 8:424–431.

Zeide, B. 1993. Analysis of growth equations. For. Sci., Vol. 39, No. 3:594–616.