Good Modeling Practices

Pei Huang

Department of Agricultural Economics
Texas A&M University

Based on materials written by Gillig & McCarl and improved upon by many previous lab instructors

Special thanks to Mario Andres Fernandez
Why worry and what can be done for modeling?

A user-friendly model should include the following features:

- Using longer names or descriptions
- Including comments on nature and source of data
- Including as much raw data as possible as opposed to externally calculated data
- Less * as a specification for input data
- Using sets to aid in readability
- A readable format
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features:
  - Using longer names or descriptions
  - Including comments on nature and source of data
  - Including as much raw data as possible as opposed to externally calculated data
  - Less as a set specification for input data
  - Using sets to aid in readability
  - A readable format
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features
  - Using longer names or descriptions
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features
  - Using longer names or descriptions
  - Including comments on nature and source of data

Using longer names or descriptions includes comments on nature and source of data.
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features
  - Using longer names or descriptions
  - Including comments on nature and source of data
  - Including as much raw data as possible as opposed to externally calculated data
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features
  - Using longer names or descriptions
  - Including comments on nature and source of data
  - Including as much raw data as possible as opposed to externally calculated data
  - Less “*” as a set specification for input data
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features
  - Using longer names or descriptions
  - Including comments on nature and source of data
  - Including as much raw data as possible as opposed to externally calculated data
  - Less “*” as a set specification for input data
  - Using sets to aid in readability
Why worry and what can be done for modeling?

- How easy is it to reuse or modify a model at a later time for you?
- How easy a colleague can work with your code?
- A user-friendly model should include the following features
  - Using longer names or descriptions
  - Including comments on nature and source of data
  - Including as much raw data as possible as opposed to externally calculated data
  - Less “*” as a set specification for input data
  - Using sets to aid in readability
  - A readable format
GAMS allows no more than 63 character long names and 200 characters of explanatory text to define SETS, PARAMETERS, TABLES, SCALARS, VARIABLES, EQUATIONS, MODELS.
Naming Conventions

- GAMS allows no more than 63 character long names and 200 characters of explanatory text to define SETS, PARAMETERS, TABLES, SCALARS, VARIABLES, EQUATIONS, MODELS.
- Avoid using simple letters as names.
Naming Conventions

- GAMS allows no more than 63 character long names and 200 characters of explanatory text to define Sets, Parameters, Tables, Scalars, Variables, Equations, Models.
- Avoid using simple letters as names.
- Use descriptive character names and explanatory text.
Naming Conventions

- GAMS allows no more than 63 character long names and 200 characters of explanatory text to define SETS, PARAMETERS, TABLES, SCALARS, VARIABLES, EQUATIONS, MODELS.
- Avoid using simple letters as names.
- Use descriptive character names and explanatory text.
- Enter units, sources and descriptions for variables and parameters.
GAMS allows no more than 63 character long names and 200 characters of explanatory text to define \texttt{SETS}, \texttt{PARAMETERS}, \texttt{TABLES}, \texttt{SCALARS}, \texttt{VARIABLES}, \texttt{EQUATIONS}, \texttt{MODELS}.

- Avoid using simple letters as names.
- Use descriptive character names and explanatory text.
- Enter units, sources and descriptions for variables and parameters.
- Check for completeness of defined items with \texttt{$\text{ONSYMLIST}$}.
Type `$ONSYMLIST` at the beginning of a *.gms file.

`$ONSYMLIST` controls the complete listing of all symbols and their explanatory text that have been defined.

```plaintext
$ONSYMLIST

SETS
  Ingredient Feeding ingredients / Corn, Hay, Wheat /
  Nutrient  Formula characteristics / Calories, Protein /

Symbol Listing

FUNCTIONS

SETS
  Ingredient Feeding ingredients
  Nutrient  Formula characteristics

PARAMETERS
  Cost            The ingredient cost per unit
  MinNutrient    The minimum nutrient content requirement
  NutrientContent Nutrient content in ingredients

VARIABLES
  Quantity       Each ingredient's quantity used
  TotCost        The total feeding cost

EQUATIONS
  NutrientReq    Minimum nutrient requirement
  TotalDietCost  Minimize total diet cost
  TotalVolume    Total volume of diet

MODELS
  DietProblem
```
Using Longer Names

- Names in the following example are not informative.

```plaintext
cs ..  z  =E=  SUM((i,j), c(i,j)*x(i,j)) ;

su(i) ..  SUM(j, x(i,j))  =L=  a(i) ;

dc(j) ..  SUM(i, x(i,j))  =G=  b(j) ;
```

- Same algebra but different names

```plaintext
Costsum ..
   TotalCost
    =E=  SUM((Source, Destination),
            Trancost(Source, Destination)*Transport(Source, Destination)) ;

Supplybal(Source) ..
    SUM(Destination, Transport(Source, Destination))
    =L=  Supply(Source) ;

Demandbal(Destination) ..
    SUM(Source, Transport(Source, Destination))
    =G=  Need(Destination) ;
```
Including Comments

- A good programmer never hesitate to comment every part, even every block of her code.
Including Comments

- A good programmer never hesitate to comment every part, even every block of her code.

- For data used in a model, questions often asked are:
  - Where did the data come from, including sources, page numbers, table number, years, units, etc?
  - What characteristics such as units, and year of applicability do those data possess?
Including Comments

- A good programmer never hesitate to comment every part, even every block of her code.

- For data used in a model, questions often asked are:
  - Where did the data come from, including sources, page numbers, table number, years, units, etc?
  - What characteristics such as units, and year of applicability do those data possess?

- Comments can be made by
  - an asterisk * in the first character position for a single line comments
  - $ONTEXT and $OFFTEXT for multiple line comments
  - $EOLCOM # or $inlinecom{ } for end of line or in line comments
Single-line & Multi-line Comments

* This is an example of using * asterisk

VARIABLE
TotCost    The total feeding cost;

$ONTEXT
    This is a $ontext and $offtext example.
    Lines are in between are ignored.
$OFFTEXT

POSITIVE VARIABLES
Quantity(Ingredient)    Ingredient Quantity;
Single-line & Multi-line Comments

- Asterisk

```plaintext
* This is an example of using * asterisk

VARIABLE
TotCost The total feeding cost;

$ONTEXT
This is a $oncontext and $offtext example.
Lines are in between are ignored.
$OFFTEXT

POSITIVE VARIABLES
Quantity(Ingredient) Ingredient Quantity;
```
Single-line & Multi-line Comments

- Asterisk
- $\text{ONTEXT}$ and $\text{OFFTEXT}$

```plaintext
* This is an example of using * asterisk

\text{VARIABLE}
TotCost  The total feeding cost;

$\text{ONTEXT}$
This is a $\text{oncontext}$ and $\text{offtext}$ example
Lines are in between are ignored.
$\text{OFFTEXT}$

\text{POSITIVE VARIABLES}
Quantity(Ingredient)  Ingredient Quantity;
```
End-of-line Comments

$EOLCOM #

EQUATIONS

TotalDietCost Minimize total diet cost
NutrientReq(Nutrient) Minimum nutrient requirement
TotalVolume Total volume of diet; #this is an end of line comment
End-of-line Comments

- Authorize end-of-line comments with a $EOLCOM statement.

```
$EOLCOM #
EQUATIONS
    TotalDietCost     Minimize total diet cost
    NutrientReq(Nutrient) Minimum nutrient requirement
    TotalVolume      Total volume of diet;   #this is an end of line comment
```
End-of-line Comments

- Authorize end-of-line comments with a $EOLCOM statement.
- Put a designator # at the end of a line you want to comment.

```plaintext
$EOLCOM #

EQUATIONS

TotalDietCost    Minimize total diet cost
NutrientReq(Nutrient)  Minimum nutrient requirement
TotalVolume    Total volume of diet;  #this is an end of line comment
```
Raw Data v.s. Calculated Data

Which one is better?

- Externally process data entering the final results in GAMS (e.g. from a spreadsheet where the data are previously manipulated).
- Enter raw data into GAMS and transform it to the extent needed inside GAMS.
Raw Data v.s. Calculated Data

Which one is better?

- Externally process data entering the final results in GAMS (e.g. from a spreadsheet where the data are previously manipulated).
- Enter raw data into GAMS and transform it to the extent needed inside GAMS.
Raw Data v.s. Calculated Data

- Which one is better?
  - Externally process data entering the final results in GAMS (e.g. from a spreadsheet where the data are previously manipulated).
  - Enter raw data into GAMS and transform it to the extent needed inside GAMS.

- Recommendation: put data in as close to the form as it was collected into GAMS and then manipulate the data with incorporated functions in GAMS.

- Justification: we might change spreadsheets over time or lose it.
Example

- Directly enter the transportation cost

```
<table>
<thead>
<tr>
<th>Source</th>
<th>New York</th>
<th>Chicago</th>
<th>Topeka</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seattle</td>
<td>250.00</td>
<td>178.00</td>
<td>187.00</td>
</tr>
<tr>
<td>San Diego</td>
<td>250.00</td>
<td>187.00</td>
<td>151.00</td>
</tr>
</tbody>
</table>
```

- Calculate transportation cost from raw data

```
<table>
<thead>
<tr>
<th>Source</th>
<th>Distance (Source, Destination)</th>
<th>Distance in thousands of miles</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&quot;New York&quot;</td>
<td>&quot;Chicago&quot;</td>
</tr>
<tr>
<td></td>
<td>Seattle</td>
<td>2.5</td>
</tr>
<tr>
<td></td>
<td>&quot;San Diego&quot;</td>
<td>2.5</td>
</tr>
</tbody>
</table>

PARAMETER

```
TRANCost(Source, Destination) = Transport cost in dollars per case
TRANCost(Source, Destination) = Loadcost + PrMileCst * Distance(Source, Destination)
```
No * in Set Specification

- * in Table definition.

```
TABLE Miscdata(*,RawMaterial) miscellaneous production data
  scrap new
  max-stock  400  275
  store-cost .5  2
  endinv-value 15  25
```

Profitacct.. profit =E=
```
SUM(quarter,
  SUM(products, expectprof(products,quarter)
  *production(products,quarter))
- SUM(RawMaterial, miscdata("store-cost",RawMaterial)
  *openstock(RawMaterial,quarter))
+ SUM(RawMaterial, miscdata("endinv-valu",RawMaterial)
  *openstock(RawMaterial,"winter");
```
No * in Set Specification

- * in Table definition.

```
TABLE Miscdata(*,RawMaterial) miscellaneous production data
    scrap new
max-stock   400   275
store-cost  .5    2
endinv-value 15   25
```

```
Profitacct.. profit = \=E= 
    SUM(quarter, 
        SUM(products, expectprof(products,quarter) 
            *production(products,quarter)) 
        - SUM(RawMaterial, miscdata("store-cost",RawMaterial) 
            *openstock(RawMaterial,quarter))) 
    + SUM(RawMaterial, miscdata("endinv-valu",RawMaterial) 
            *openstock(RawMaterial,"winter"));
```
No * in Set Specification

- * in Table definition.

```
TABLE Miscdata(*,RawMaterial) miscellaneous production data
     scrap new
     max-stock  400  275
     store-cost .5  2
     endinv-value 15  25 ;
```

- * can be replaced by anything even that is not defined.

```
Profitacct.. profit =E=
    SUM(quarter,
        SUM(products, expectprof(products,quarter)
            *production(products,quarter))
        - SUM(RawMaterial, miscdata("store-cost",RawMaterial)
            *openstock(RawMaterial,quarter)))
    + SUM(RawMaterial, miscdata("endinv-valu",RawMaterial)
            *openstock(RawMaterial, "winter");
```
No * in Set Specification

- Here we replace * with \texttt{InputItem} that is defined before.

\begin{verbatim}
\begin{table}
\centering
\begin{tabular}{c c c}
\hline
\textit{InputItem} & \textit{RawMaterial} & \textit{miscellaneous production data} \\
\hline
\textit{scrap} & 400 & 275 \\
\textit{new} & .5 & 2 \\
\textit{endinv-value} & 15 & 25 \\
\hline
\end{tabular}
\end{table}
\end{verbatim}

- GAMS reports an error.

\begin{verbatim}
56  Profitacct.. profit =E= \\
57      SUM(quarter,
58      SUM(products, expectprof(products, quarter)
59      *production(products, quarter))
60      - SUM(RawMaterial, miscdata("store-cost",RawMaterial)
61      *openstock(RawMaterial,quarter))
62      + SUM(RawMaterial, miscdata("endinv-valu",RawMaterial)
63 *** 170  Domain violation for element
64      *openstock(RawMaterial,"winter"));
\end{verbatim}
No * in Set Specification

- Here we replace * with `InputItem` that is defined before.

```
Table MiscData(InputItem, RawMaterial) miscellaneous production data
     scrap  new
max-stock    400  275
store-cost   0.5  2
endinv-value 15   25 ;
```

- GAMS reports an error.

```
56  Profitacct.. profit =E=
57    SUM(quarter,
58        SUM(products, expectprof(products, quarter)
59            *production(products, quarter))
60    - SUM(RawMaterial, miscdata("store-cost",RawMaterial)
61            *openstock(RawMaterial, quarter))
62    + SUM(RawMaterial, miscdata("endinv-valu",RawMaterial)
63          $170
64  ****
65  **** 170  Domain violation for element
66          *openstock(RawMaterial, "winter") ;
```
No * in Set Specification

- Here we replace * with \texttt{InputItem} that is defined before.

\begin{verbatim}
TABLE Miscdata(InputItem,RawMaterial) miscellaneous production data
        scrap new
max-stock  400  275
store-cost  .5  2
\textbf{endinv-value}  15  25
;
\end{verbatim}

- GAMS reports an error.

\begin{verbatim}
  56  Profitacct.. profit =E=
  57      SUM(quarter,
  58          SUM(products, expectprof(products,quarter)
  59           *production(products,quarter))
  60          - SUM(RawMaterial, miscdata("store-cost",RawMaterial)
  61           *openstock(RawMaterial,quarter)))
  62          + SUM(RawMaterial, \textbf{miscdata("endinv-valu",RawMaterial)}
  \textbf{****} $170
  \textbf{****} 170  Domain violation for element
  63      *openstock(RawMaterial, "winter");
\end{verbatim}
No * in Set Specification

- Here we replace * with **InputItem** that is defined before.

```
TABLE Miscdata(InputItem,RawMaterial) miscellaneous production data
   scrap new
max-stock  400  275
store-cost .5  2
endinv-value 15  25
```

- GAMS reports an error.

```
56  Profitacct.. profit =E=
57   SUM(quarter,
58   SUM(products, expectprof(products,quarter)
59   *production(products,quarter))
60   - SUM(RawMaterial, miscdata("store-cost",RawMaterial)
61     *openstock(RawMaterial,quarter))
62   + SUM(RawMaterial, miscdata("endinv-valu",RawMaterial)
    $170
****
*** 170 Domain violation for element
63   *openstock(RawMaterial,"winter"));
```
Consequences of Mistyping

- The solution from solving the model with mistyping on “endinv-value” as “endinv-valu”

```
SOLVE SUMMARY
MODEL  robert
TYPE   LP
SOLVER BDMLP

OBJECTIVE     profit
DIRECTION     MAXIMIZE
FROM LINE     72

**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS   1 OPTIMAL
**** OBJECTIVE VALUE 6741.6667
```

- The solution from solving the model with correction on “endinv-value”

```
SOLVE SUMMARY
MODEL  robert
TYPE   LP
SOLVER BDMLP

OBJECTIVE     profit
DIRECTION     MAXIMIZE
FROM LINE     72

**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS   1 OPTIMAL
**** OBJECTIVE VALUE 11025.0000
```
Consequences of Mistyping

- The solution from solving the model with mistyping on “endinv-value” as “endinv-valu”

```
SOLVE SUMMARY
MODEL    robert
TYPE     LP
SOLVER   BDMLP
OBJECTIVE profit
DIRECTION MAXIMIZE
FROM LINE 72

**** SOLVER STATUS = 1 NORMAL COMPLETION
**** MODEL STATUS = 1 OPTIMAL
**** OBJECTIVE VALUE = 6741.6667
```

- The solution from solving the model with correction on “endinv-value”

```
SOLVE SUMMARY
MODEL    robert
TYPE     LP
SOLVER   BDMLP
OBJECTIVE profit
DIRECTION MAXIMIZE
FROM LINE 72

**** SOLVER STATUS = 1 NORMAL COMPLETION
**** MODEL STATUS = 1 OPTIMAL
**** OBJECTIVE VALUE = 11025.0000
```
Consequences of Mistyping

- The solution from solving the model with mistyping on “endinv-value” as “endinv-valu”

```
SOLVE SUMMARY
MODEL  robert          OBJECTIVE  profit
TYPE   LP               DIRECTION  MAXIMIZE
SOLVER BDMLP            FROM LINE 72

**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 1 OPTIMAL
**** OBJECTIVE VALUE 6741.6667
```

- The solution from solving the model with correction on “endinv-value”

```
SOLVE SUMMARY
MODEL  robert          OBJECTIVE  profit
TYPE   LP               DIRECTION  MAXIMIZE
SOLVER BDMLP            FROM LINE 72

**** SOLVER STATUS 1 NORMAL COMPLETION
**** MODEL STATUS 1 OPTIMAL
**** OBJECTIVE VALUE 11025.0000
```
Sets v.s. Subsets

- In some cases, it is desirable to treat items simultaneously in specific places, but separately elsewhere.

```
SETs
ALLI               ALL BUDGET ITEMS
  /Corn , Soybeans
  Cropland , Pasture
  Labor , Water
  Nitrogen , Potassium
  TranCost
/
Primary(ALLI) Primary products
  /Corn , Soybeans /
Input(ALLI) National inputs
  /Nitrogen , Potassium /
LandType(ALLI) Land types
  /CropLand , Pasture /
Resource(ALLI) Resource use
  /Labor , Water /
TCost(ALLI) Transportation cost
  /TranCost /
```
In some cases, it is desirable to treat items simultaneously in specific places, but separately elsewhere.

Define a set, \textit{ALLI} that contains all items.
Sets v.s. Subsets

- In some cases, it is desirable to treat items simultaneously in specific places, but separately elsewhere.

- Define a set, `ALLI` that contains all items.

- Select relevant items to form different subsets, Primary, Input, Landtype, Resource, and Tcost.
Improve Readability

- Format the code for readability using spacing and indents.
  - Align item names, descriptions, and definitions
  - Indent in sums, loops, and ifs to delineate terms
  - Use blank lines to set things off
  - Do not split variables between lines in equations, but rather keep them together with all their index positions
Which format do you prefer?

- **Format 1**

```
Variables  production(products,Quarters)  production and sales
           openstock(rawmaterial,Quarters)  opening stocks, profit;
Positive variables production, openstock;
Equations   capacity(quarter)  capacity constraint,
           stockbalan(rawmaterial,Quarters)  stock balance,
           profitacct  profit definition;
           capacity(quarter). sum(products, production(products,quarter))
           =l=  mxcapacity;
           stockbalan(rawmaterial,Quarters+1). openstock(rawmaterial,
           Quarters+1) =e=  openstock(rawmaterial,Quarters)-
           sum(products, usage(rawmaterial,products) *production(products,Quarters));
           profitacct.. profit =e=  sum(quarter, sum(products, expectprof(
           products,quarter) *production(products,quarter)))-sum(
           rawmaterial,miscdata("store-cost",rawmaterial)*openstock(rawmaterial,
           ,quarter))+ sum(rawmaterial, miscdata("endinv-value",rawmaterial) *
           openstock(rawmaterial,"winter"));
           openstock.up(rawmaterial,"spring")=miscdata("max-stock",rawmaterial);
```
Which format do you prefer?

- **Format 2**

```
Equations  capacity(quarter)  capacity constraint
stockbalan(rawmaterial,Quarters) stock balance
profitacct profit definition ;

capacity(quarter)..
   sum(products, production(products,quarter))
   =L=  mxcapacity ;
stockbalan(rawmaterial,Quarters+1)..
   openstock(rawmaterial,Quarters+1)
   =E=  openstock(rawmaterial,Quarters)
       - sum(products, usage(rawmaterial,products)
               *production(products,Quarters) ) ;
profitacct..
   profit
   =E=
   sum(quarter, sum(products, expectprof(products,quarter)
                      *production(products,quarter) )
        -sum(rawmaterial,misdata("store-cost",rawmaterial)
               * openstock(rawmaterial,quarter) ) )
   + sum(rawmaterial,misdata("endinv-value",rawmaterial)
          *openstock(rawmaterial,"winter") ) ;
```