Typesetting Mathematics

Five environments:

1. displaymath - unnumbered equations

   Example:

   $$\widehat{\beta} = (X'V^{-1}X)^{-1}X'V^{-1}Y$$

2. equation - numbered equations

   Example:

   $$\widehat{\beta} = (X'V^{-1}X)^{-1}X'V^{-1}Y \tag{1}$$

3. In-line formulas - equations in a paragraph of text.

   Example:

   The lin-log model, $Y_i = \beta_1 + \beta_2 \ln X_i + u_i$, is useful when we wish to...

4. eqnarray - multiple lines of formulas that are automatically justified

   Example:

$$
\begin{eqnarray}
\widehat{\beta} & = & (X'V^{-1}X)^{-1}X'V^{-1}Y, \text{ or} \nonumber \\
& = & (X'X)^{-1}X'Y, \text{ when} \tag{2} \\
V & = & \sigma^2 I_N. \tag{3}
\end{eqnarray}
$$

5. array - the catch-all for anything you can't do with the above.

Displaymath Environment

Example was generated with:

```
\[
  \hat{\beta} = (X^{\prime}V^{-1}X)^{-1}
                  X^{\prime}V^{-1}Y
\]
```

- Environment is invoked with `\[ \]`, or `\begin{displaymath} \end{displaymath}`.
- Equation is centered on the line.
- LaTeX will not break long equations. For long equations, adjust the text size with `\small` or `\footnotesize`, or use an array.

Equation Environment

Example was generated with:

```
\begin{equation}
  \hat{\beta} = (X^{\prime}V^{-1}X)^{-1}
                 X^{\prime}V^{-1}Y
  \label{example}
\end{equation}
```

- Environment is invoked with `\begin{equation}` `\end{equation}`.

- You can label equations for reference in the text. For example, we refer to equation 1 with the command `\ref{example}`.

- LaTeX will keep track of the equation numbers.

In-Line Formulas

Example was generated with:

```
The lin-log model,
$Y_i = \beta_1 + \beta_2 \ln X_i + u_i$,
is useful when we wish to...
```

- Environment starts and ends with a $.

- Font is small, so complicated formulas should be typeset with displaymath.

- Full range of math commands can be used.

Eqnarray Environment

Example was generated with:

```
\begin{eqnarray}
\hat{\beta}  & = & (X^{\prime}V^{-1}X)^{-1}
                    X^{\prime}V^{-1}Y, \,
                    \mbox{or} \nonumber \\
             & = & (X^{\prime}X)^{-1}
                    X^{\prime}Y, \,
                    \mbox{when} \label{beta} \\
V            & = & \sigma^2 I_N.
\end{eqnarray}
```

- Environment is invoked with `\begin{eqnarray}` `\end{eqnarray}`.

- Three columns are pre-defined. The first is right-justified, the second is centered, and the third is left-justified.

- The columns are separated by &.

- The lines each end with \\, except the last line, which does not end with \\.

- You can label any or all of the lines. The label goes before the \\.

- If you want to skip the number on a line, use the `\nonumber` command before the \\.

- If you want to skip the numbers for all of the lines, use the `\begin{eqnarray*}` `\end{eqnarray*}` invocation.

Array Environment

Template:

```
\begin{equation}
  \begin{array}{<see below>}
    <your equations here>
  \end{array}
\end{equation}
```

- In an array, you define the number of columns and the justification of each column in the `{}` after the `\begin{array}` command.

- Example: To create three columns, the first right-justified, the second centered, and the third left-justified, use the following template:

```
\begin{equation}
  \begin{array}{rcl}
    <your equations here>
  \end{array}
\end{equation}
```

- Each column of formulas is separated by &.

- Each line ends with \\, except the last line.

- The equation number is vertically centered on the array.

- You can put vertical and horizontal lines in an array.

  - Vertical lines are defined along with the columns. Example: To put vertical lines before and after every column in our example above:

    ```
    \begin{equation}
      \begin{array}{|r|c|l|}
        <your equations here>
      \end{array}
    \end{equation}
    ```

  - The command `\hline` puts a horizontal line across the row.

  - The command `\cline{x-y}` puts a horizontal line across columns x through y.

Example:

$$\Omega = \left[\begin{array}{c|ccc} \sigma_{x_1}^2 & \sigma_{x_1,x_2} & \cdots & \sigma_{x_1,x_N} \\ \hline \sigma_{x_2,x_1} & \sigma_{x_2}^2 & \cdots & \sigma_{x_2,x_N} \\ \vdots & \vdots & \ddots & \vdots \\ \cline{2-4} \sigma_{x_N,x_1} & \sigma_{x_N,x_2} & \cdots & \sigma_{x_N}^2 \end{array} \right] \tag{4}$$

Markup:

```
\begin{equation}
\Omega =
\left[
\begin{array}{c|ccc}
\sigma_{x_1}^2   & \sigma_{x_1,x_2} & \cdots & \sigma_{x_1,x_N} \\
\hline
\sigma_{x_2,x_1} & \sigma_{x_2}^2   & \cdots & \sigma_{x_2,x_N} \\
\vdots           & \vdots           & \ddots & \vdots           \\
\cline{2-4}
\sigma_{x_N,x_1} & \sigma_{x_N,x_2} & \cdots & \sigma_{x_N}^2
\end{array}
\right]
\end{equation}
```

Other Useful Stuff

Fractions are done with `\frac{}{}`, where the numerator goes in the first set of brackets, and the denominator goes in the second.

Example:

$$Y_i = \beta_1 + \beta_2 \left(\frac{1}{X_i}\right) + u_i$$

Markup:

```
\[
  Y_i = \beta_1 +
        \beta_2 \left(\frac{1}{X_i}\right) +
        u_i
\]
```

Summations are done with `\sum`, and the limits are done with `\limits` and `_` and `^`.

Example:

$$\hat{\beta} = \frac{\sum\limits_{i=1}^N x_i y_i}{\sum\limits_{i=1}^N x_i^2}$$

Markup:

```
\[
  \hat{\beta} = \frac{\sum\limits_{i=1}^N x_i y_i}
                     {\sum\limits_{i=1}^N x_i^2}
\]
```

- Integrals are done just like sums, but with the `\int` command.

- Normal text can be place in a formula with the `\mbox{}` command. You'll probably want to add a little space before the text using `\,`.

- Hats are done with `\hat{}`.

Typesetting Tables

Tables are composed of three main pieces:

1. Invocation and positioning.

2. Information the table displays.

3. Caption and label (optional).

Invocation and Positioning

Template:

```
\begin{table}[<see below>]
 <information>
\end{table}
```

Tables are "floating objects," which means they don't have a fixed position in the document.

We prod LaTeX to put a table in the right place with "positioning options" in the [] after the `\begin{table}` command.

Positioning options consist of:

- `t` top of page
- `b` bottom of page
- `h` here
- `p` on a "float page"

LaTeX reads your positioning "hints" and then decides where it will place the table.

Tips for making LaTeX put the table in the right place:

- Use the `\small`, `\footnotesize`, and, if need be, `\tiny`, commands to rescale the table so it fits on the page.

  Example:

```
{\small
\begin{table}[hhtb]
  <information>
\end{table}
} % end small
```

- Use a `\clearpage` right before or after the table to clear away text.

- Reset the internal rules LaTeX uses to format the page. Reset the `\topfraction`, `\bottomfraction`, and `\textfraction` parameters using `\renewcommand`

- To center the stuff in the table, use the `\centerline{}` command.

  Example:

```
\begin{table}[hhtb]
  \centerline{
    <the stuff>
  } % end centerline
\end{table}
```

Information the Table Displays

- Usually the information will be formatted with the tabular environment.

- The tabular environment is a lot like array, except that it's for text (although you certainly can typeset mathematics in a tabular environment).

- You can use Perl scripts as code generators to make your tables for you.

Example:

Year	$\hat{\beta}$	Price Index
1960		1.0000
1961	-0.1398	0.8695
1962	-0.4891	0.6132
1963	-0.5938	0.5522

Markup:

```
\begin{tabular}{|c|c|c|}\hline
Year & $\hat{\beta}$ & Price Index \\ \hline
1960 &               &   1.0000    \\
1961 & -0.1398       &   0.8695    \\
1962 & -0.4891       &   0.6132    \\
1963 & -0.5938       &   0.5522    \\ \hline
\end{tabular}
```

Compare `tabular` and `array`:

- The number of columns, justification, and vertical lines are specified in the same way.
- Horizontal lines specified in the same way.
- Lines end with \\.
- Columns separated with `&`.
- Only differences are invocation: `\begin{tabular}` and `\end{tabular}`, and the last line of the table ends with \\.

To finish the table, insert the tabular environment into a table environment:

```
\begin{table}[hhtb]
\centerline{
\begin{tabular}{|c|c|c|}\hline
Year & $\hat{\beta}$ & Price Index \\ \hline
1960 & -0.0035       &   0.9724    \\
1961 & -0.1398       &   0.8695    \\
1962 & -0.4891       &   0.6132    \\
1963 & -0.5938       &   0.5522    \\ \hline
\end{tabular}
}
\end{table}
```

Use the `\multicolumn{}{}{}` command to produce column headers or to change the justification of a particular cell.

Example:

```
\begin{table}[hhtb]
\centerline{
\begin{tabular}{|c|c|c|}
\multicolumn{3}{c}{Quality-Adjusted Indices}        \\
\hline
     & \multicolumn{2}{c|}{Estimates}                \\
Year & $\hat{\beta}$ & Price Index                   \\
\hline
1960 & -0.0035      &   0.9724                      \\
1961 & -0.1398      & \multicolumn{1}{|r|}{0.8695} \\
1962 & -0.4891      &   0.6132                      \\
1963 & -0.5938      &   0.5522                      \\
\hline
\end{tabular}
}
\end{table}
```

Using Perl scripts to generate tables:

- Have your statistical package output the data in columns, like:

```
1960 -0.0035 0.9724
1961 -0.1398 0.8695
1962 -0.4891 0.6132
1963 -0.5938 0.5522
```

- Write a Perl script to read the data file and format the data with LaTeX markup.

Example Perl script (to read data of previous slide):

```
#!/usr/local/bin/perl
$i=0;
while(<>){
    chop;
    ($year[$i],$beta[$i],$index[$i])=split(‘‘ ‘‘);
    $i++;
}
print ‘‘\\begin{table}\n’’;
print ‘‘\\centerline{\n’’;
print ‘‘\\begin{tabular}{|c|c|c|}\\hline\n’’;
print ‘‘Year & \$\\hat{\\beta}\$ & Index \\\\\n’’;
print ‘‘$year[0] & $beta[0] & $index[0] \\\\\n’’;
print ‘‘$year[1] & $beta[1] & $index[1] \\\\\n’’;
print ‘‘$year[2] & $beta[2] & $index[2] \\\\\n’’;
print ‘‘$year[3] & $beta[3] & $index[3] \\\\\\hline\n’’;
print ‘‘\\end{tabular}\n’’;
print ‘‘} % end centerline\n’’;
print ‘‘\\end{table}\n’’;
```

Using Perl and $\LaTeX$ together, you can quickly format enormous amounts of data.

Tips for using Perl with $\LaTeX$:

- Write your Perl scripts as filters (read from standard input, write to standard output), so you can use them in pipelines.

- Generalize your Perl scripts as much as possible, so if the table grows or shrinks you don't have to change the Perl code (the previous example is poor in this regard).

Captions and Labels

Captions provide information about the table to the reader, and make an entry in the List of Tables.

Although not useful for academic papers, you can have LaTeX automatically create a list of tables.

Labels allow you to refer to tables in the text, much like equations.

Example:

```
\begin{table}[hhtb]
\centerline{
\begin{tabular}{|c|c|c|} \hline
Year & $\hat{\beta}$ & Price Index \\ \hline
1960 & -0.0035       &   0.9724    \\
1961 & -0.1398       &   0.8695    \\
1962 & -0.4891       &   0.6132    \\
1963 & -0.5938       &   0.5522    \\ \hline
\end{tabular}
}
\caption{Estimates and Indices}
\label{estind}
\end{table}
```