Statistics 243: *class notes*

William J. De Meo

October 20, 1997

1 Determinants

Useful facts:
If A is $n \times n$ and λ is a scalar, then $\det(\lambda A) = \lambda^n \det(A)$
Given a square psd, symmetric matrix A, let U^tU be its Cholesky decomposition.

$$\det(U^tU) = \det(U)^2 = \left[\prod_i u_{ii}\right]^2$$

We want to solve $\Theta = A^{-1}x$ Decompose $A = U^tU$ and write $U^tU\Theta = x$. Let $\lambda = U\Theta$, then

$$U^t\lambda = x$$

$$U\Theta = \lambda$$

a lower and upper triangular system, respectively.

If $X = QR$, then $X^tX = R^tQ^tQR = R^tR$, and R is the Cholesky factor of X^tX.

2 Debugging Methods

A symbolic debugger is a program which lets you progress through your program line by line to make it easier to fix it. The gdb is the GNU FSF debugger. The dbx is the standard UNIX debugger.

Recompile your program with the -g flag. With a makefile, you can rm * all your object files and put the -g option in your CFLAGS macro in your makefile. Recompiling will then rebuild all the object files using the -g option. Next, invoke the debugger

```
dbx programname
(dbx) run <argument>
```

You'll then see a message that says something like

```
segmentation violation at line ...
(dbx)
```

Now run `trace` then type things like `print i` or `print x`. If x is a pointer, there usually isn't much you can tell from it *except* when it says `x = (nil)` which means that you forgot to ask memory to `x`. Then edit your file and type make at the (dbx) prompt. Then again do a run.

2.1 Setting Breakpoints

`stop at linenumber` or
`stop at ''sourcefilename'':linenumber` or
`stop in functionname`
After you stop at a breakpoint,

`continue` resumes execution.
`step` executes one line of code at a time.
`next` is like step, but doesn't go through fucntions.
To get rid of a breakpoint, you have to delete it's breakpoint number:
`status` shows current breakpoints.
`delete` removes breakpoints.