:/Users/kjhealy/Documents/data/misc/dsquared/surface.r

```
library(lattice)

# Read in the data.
data <- read.csv("surface.csv",header=TRUE,row.names=1,check.names=FALSE)

# Look at the top few rows.
head(data)

# Dimensions of the dataframe.
dim(data)

# I know there's a simpler way of doing the axis labeling
# but I can't remember what it is.

# Manually generate a scale for the y-axis labels.
y.scale <- seq(0.02,1,by=0.02)
y.ind.ticks <- seq(50, 1,by=-5) # we don't want to display all the tick marks

# Manually generate a scale for the x-axis labels that skips in increments of 10.
x.ind.ticks <- seq(5,nrow(data),by=10)


# Flip around the column-ordering of the matrix, so the plot
# replicates DD's output. (R's default is opposite direction.)
# There's almost certainly a more elegant way to do this, too.
ind <- seq(50,1,by=-1)
tmp <- data[,ind]

# This is the working dataframe we'll use below
head(tmp)


# Create the PDF
pdf("surface.pdf", width=8, height=8)

# wireframe() does all the work.
# see help(wireframe) for details on the various options.
# leave out the scales = list( ... ) argument to see the default
# scale (basic labels, with arrows).
wireframe(as.matrix(tmp),drape = TRUE, colorkey=TRUE,
          scales = list(arrows=FALSE,
            x=list(labels=rownames(tmp)[x.ind.ticks],at=x.ind.ticks),   # This is a bit messy.
            y=list(labels=y.scale[y.ind.ticks],at=ind[y.ind.ticks])),  #
          zlab="Deviation\nfrom\nMarket\nPrice",
          xlab="Time",
          ylab="Black-Scholes Volatility",
          zoom=0.75,                    # zoom out a bit
          col.regions=topo.colors(124), # use topographical colors rather than spectrum
          screen = list(z=60,x=-70))    # set the angle of view
title(main="Overvaluation of the DEM04 Contract on the Iowa Electronic Markets\n",
      sub="See http://www.crookedtimber.org/archives/002460.html for discussion.")

dev.off()
```

-:-- surface.r Bot (25,0) (ESS[S] [R])

```
R version 2.5.1 (2007-06-27)
Copyright (C) 2007 The R Foundation for Statistical Computing
ISBN 3-900051-07-0

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> if(!exists("baseenv", mode="function")) baseenv <- function() NULL
options(STERM='iESS', editor='emacsclient')
> > library(lattice)
>
> # Read in the data.
> data <- read.csv("surface.csv",header=TRUE,row.names=1,check.names=FALSE)
>
> # Look at the top few rows.
> head(data)
           0.02       0.04       0.06       0.08       0.1        0.12
6/1/04 -0.4284262 -0.2914485 -0.2025420 -0.1512160 -0.1193105 -0.09815777
6/2/04 -0.4486465 -0.4049660 -0.3239507 -0.2584229 -0.2113409 -0.17744332
6/3/04 -0.4489749 -0.4282796 -0.3639943 -0.3003321 -0.2504616 -0.21288516
6/4/04 -0.4538281 -0.4182294 -0.3416629 -0.2761801 -0.2279997 -0.19286934
6/5/04 -0.4453742 -0.3780623 -0.2888591 -0.2247766 -0.1809802 -0.15028215
6/6/04 -0.4699799 -0.4505367 -0.3876223 -0.3242473 -0.2742313 -0.23639478
            0.14        0.16        0.18         0.2        0.22        0.24
6/1/04 -0.08347476 -0.07295777 -0.06527092 -0.05959028 -0.05538161 -0.05228389
6/2/04 -0.15248744 -0.13368812 -0.11924880 -0.10798716 -0.09910289 -0.09203874
6/3/04 -0.18442533 -0.16254601 -0.14546158 -0.13193758 -0.12111205 -0.11237196
6/4/04 -0.16679573 -0.14703592 -0.13178144 -0.11982696 -0.11034997 -0.10277456
6/5/04 -0.12806980 -0.11155499 -0.09901320 -0.08933753 -0.08179083 -0.07586561
6/6/04 -0.20766506 -0.18553674 -0.16823066 -0.15451122 -0.14351317 -0.13462004
            0.26        0.28         0.3        0.32        0.34        0.36
6/1/04 -0.05004467 -0.04848222 -0.04746248 -0.04688440 -0.04667036 -0.04675983
6/2/04 -0.08639644 -0.08188455 -0.07828529 -0.07543289 -0.07319903 -0.07148291
6/3/04 -0.10527250 -0.09948450 -0.09475987 -0.09090847 -0.08778234 -0.08526476
6/4/04 -0.09668745 -0.09178533 -0.08784103 -0.08468131 -0.08217189 -0.08020715
6/5/04 -0.07120208 -0.06753858 -0.06468051 -0.06248027 -0.06082397 -0.05962233
6/6/04 -0.12738386 -0.12147282 -0.11663666 -0.11268346 -0.10946379 -0.10685974
            0.38         0.4        0.42        0.44        0.46        0.48
6/1/04 -0.04710483 -0.04766690 -0.04841481 -0.04932298 -0.05037019 -0.05153875
6/2/04 -0.07020429 -0.06929852 -0.06871295 -0.06840427 -0.06833652 -0.06847963
6/3/04 -0.08326246 -0.08170007 -0.08051607 -0.07965973 -0.07908888 -0.07876817
6/4/04 -0.07870294 -0.07759141 -0.07681725 -0.07633496 -0.07610673 -0.07610094
```

-:** *R* Top (22,4) (iESS [R]: run)

Finished evaluation