Why Details Matter in Statistical Graphics

Paul Murrell

The University of Auckland

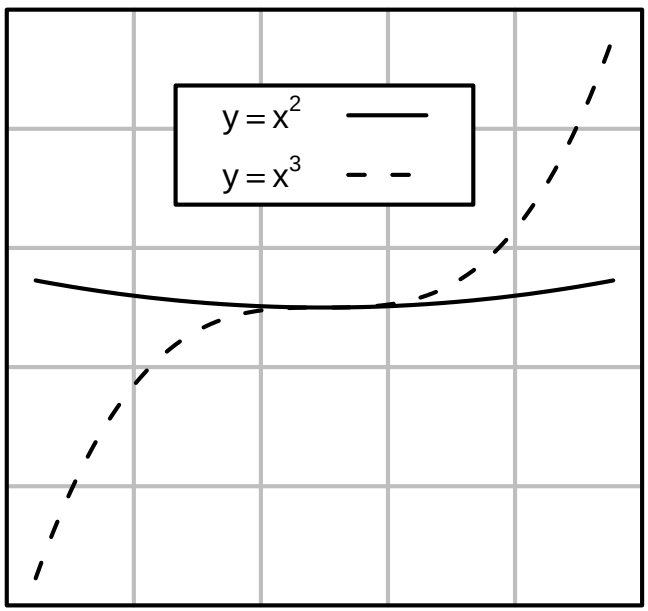
November 14 2007

Introduction

> *For the highest level graphics (elegant, custom, expensive), enter the crunched data or the graphical output into Adobe Illustrator. Or have all your graphical templates designed and set up in Illustrator.*
>
> *...*
>
> *This program gives complete control over typography, line weight, color, grids, layout–just what we need for doing graphical work.*

Edward Tufte, April 27, 2001;
http://www.edwardtufte.com/bboard/

Image Processing



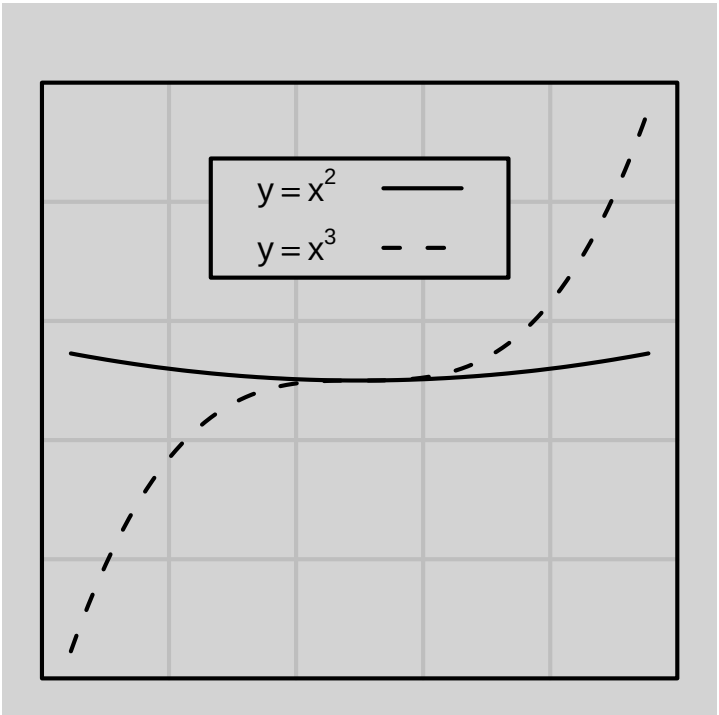
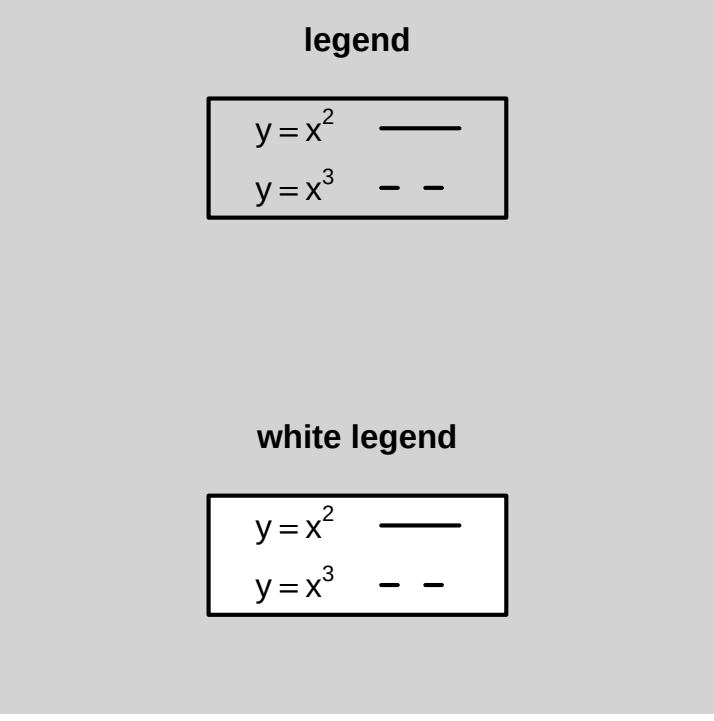


Image Processing



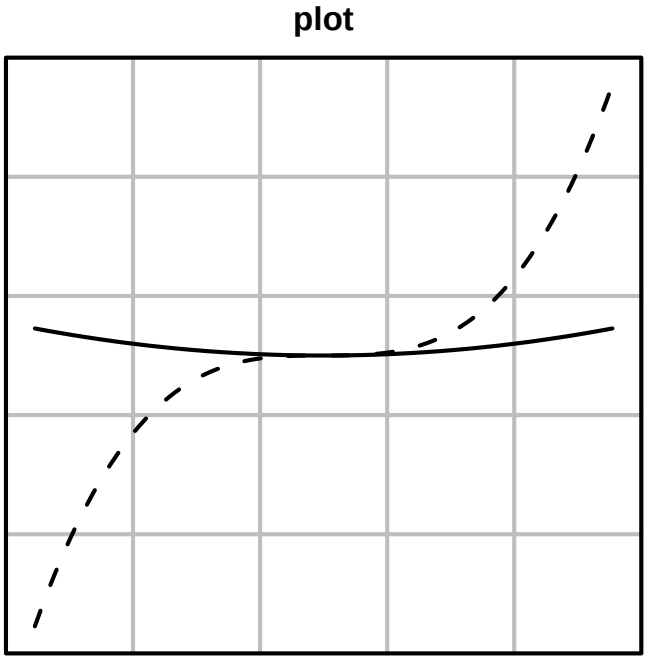
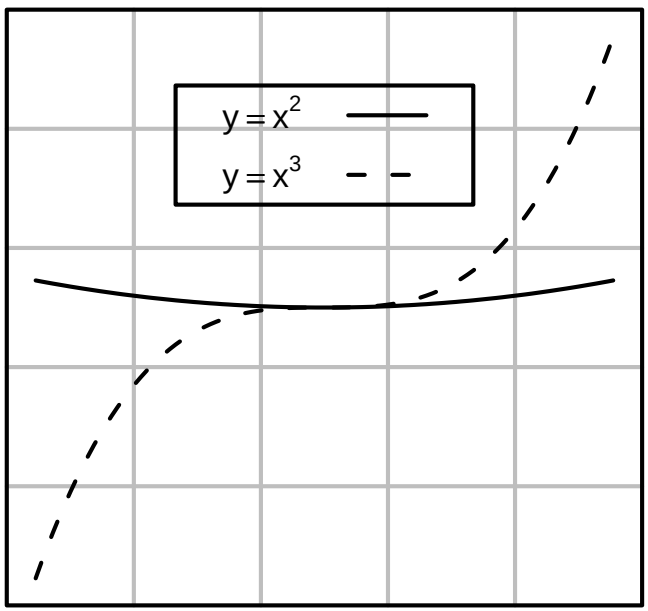


Image Processing

legend OVER plot



white legend OVER plot

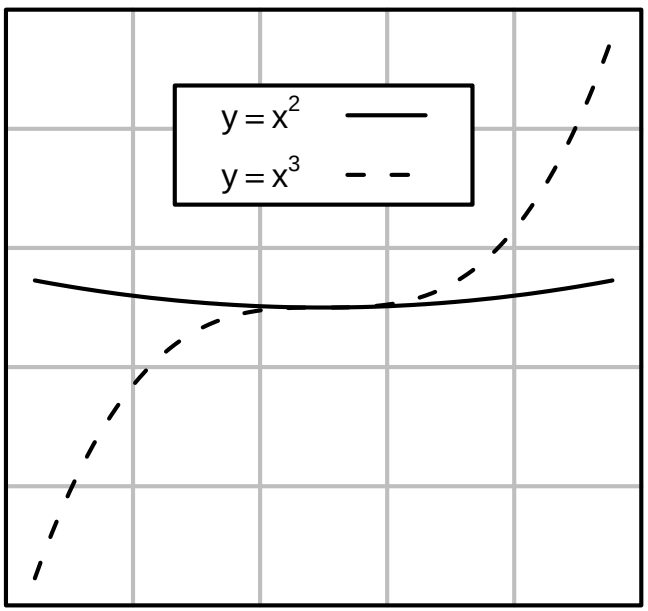
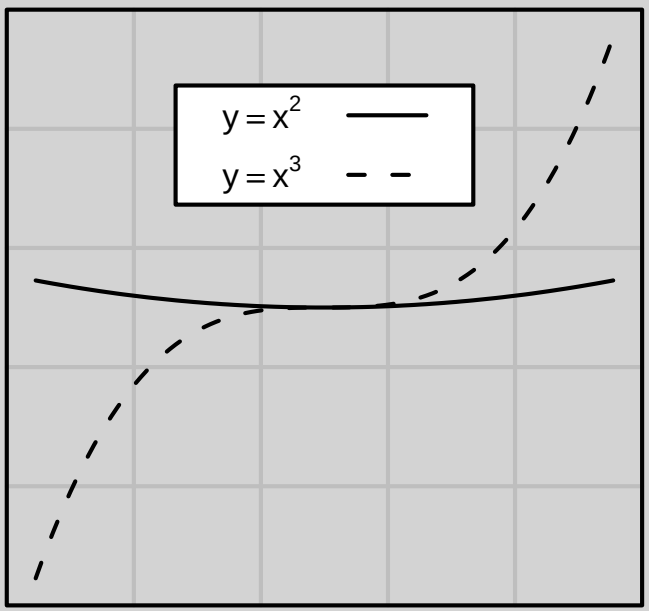


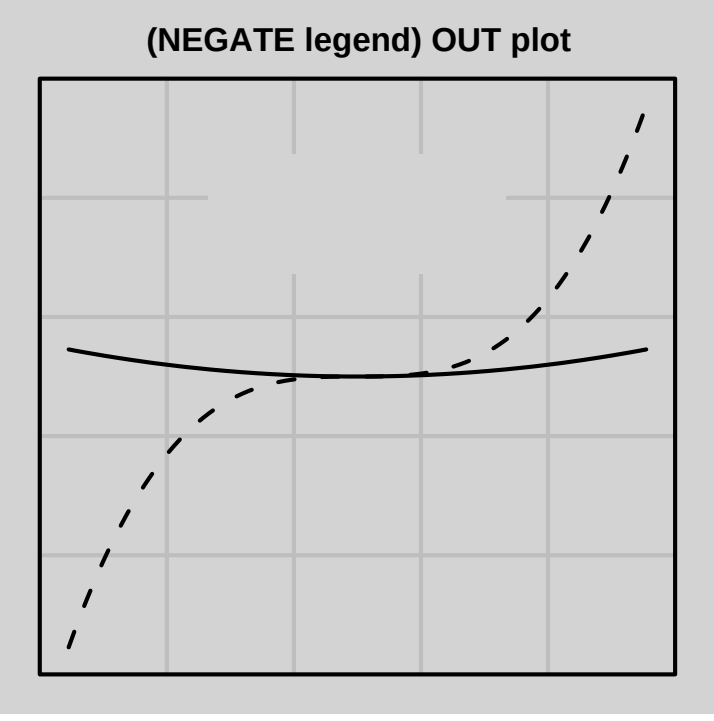
Image Processing

white legend OVER plot



NEGATE legend

Image Processing



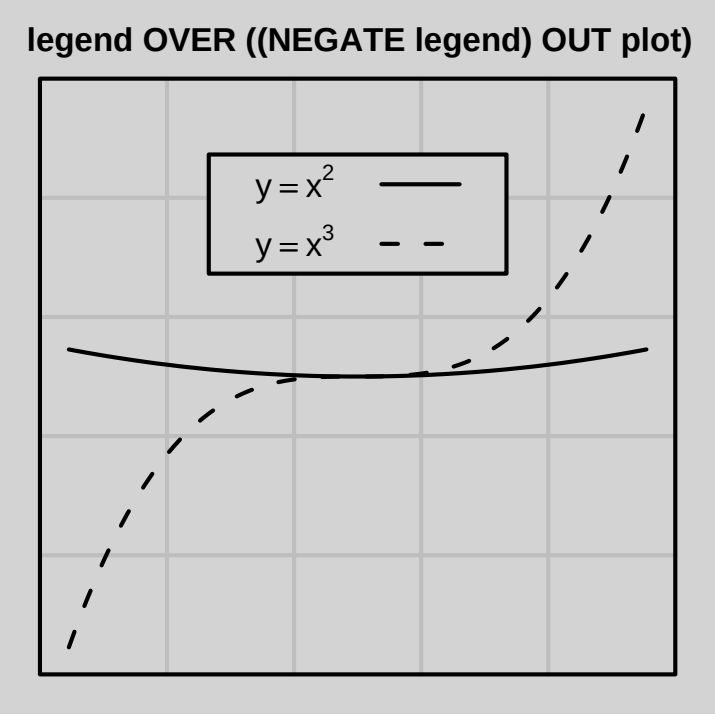
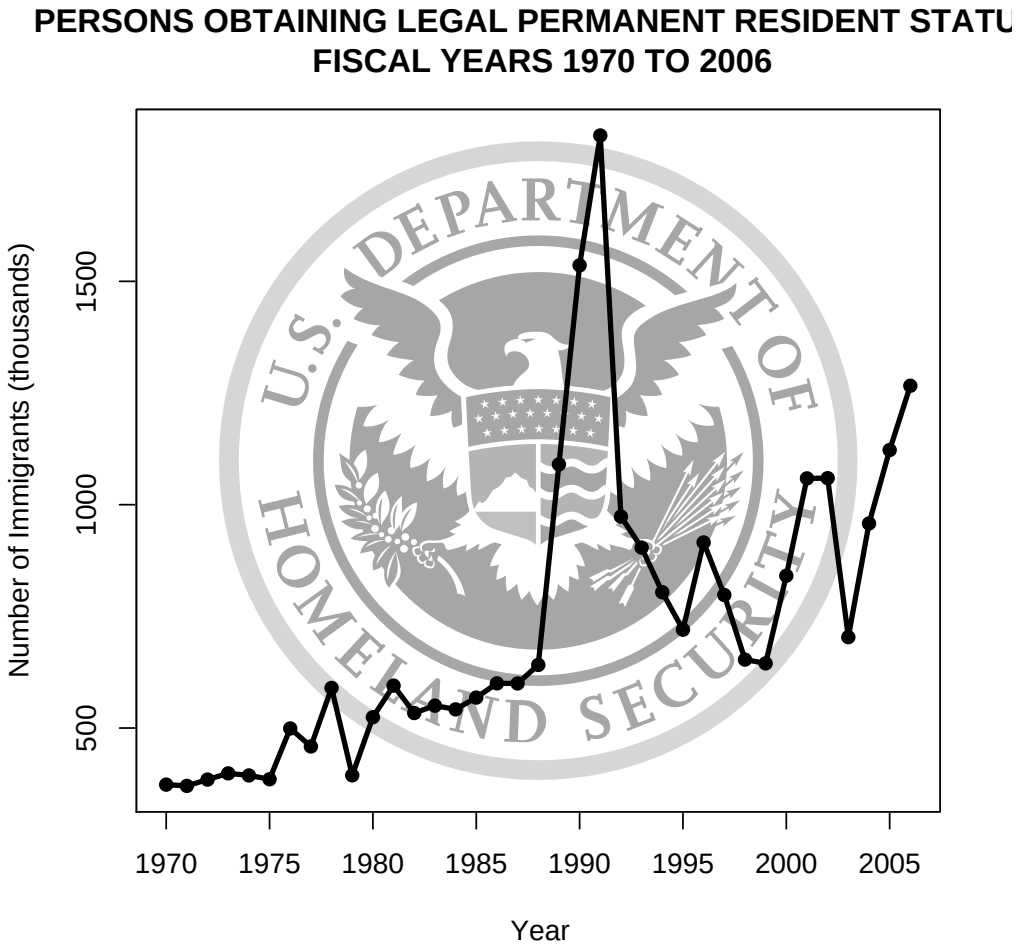


Image Processing

- ImageMagick
  `http://www.imagemagick.org/`
- EBImage package for R
  `http://www.ebi.ac.uk/~osklyar/EBImage/`

Converting SVG to greyscale

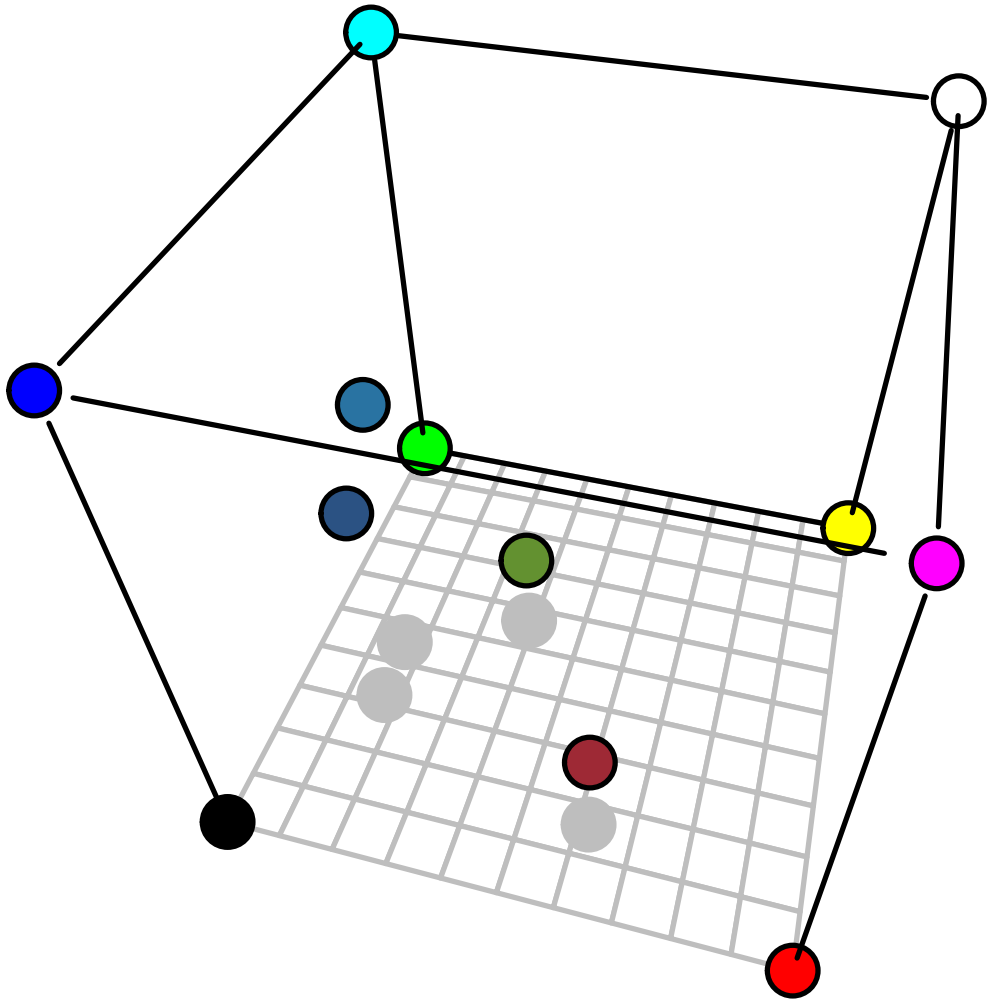
Converting SVG to greyscale



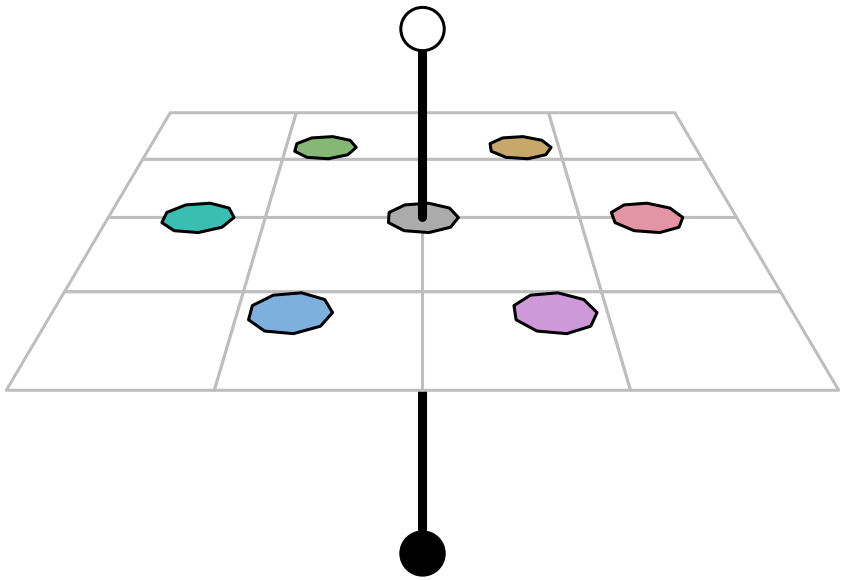
Converting SVG to greyscale

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Generator: Adobe Illustrator 12.0.0, SVG Export Plug-In  -->
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
          "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd" [
<!ENTITY ns_flows "http://ns.adobe.com/Flows/1.0/">
<!ENTITY ns_svg "http://www.w3.org/2000/svg">
<!ENTITY ns_xlink "http://www.w3.org/1999/xlink">
]>
<svg  version="1.1" xmlns="&ns_svg;" xmlns:xlink="&ns_xlink;"
      xmlns:a="http://ns.adobe.com/AdobeSVGViewerExtensions/3.0/"
      width="360" height="359" viewBox="-0.469 -0.067 360 359"
      enable-background="new -0.469 -0.067 360 359" xml:space="preserve">
<defs>
</defs>
<g>
    <g>
        <circle fill="#FFFFFF" cx="178.755" cy="179.102" r="176.553"/>
        <path fill="#2B5283" d="M293.496,194.116l-8.784,2.705l1.987,
                                0.811l-3.076,1.948c0.327-1.792,0.645
        ...
```

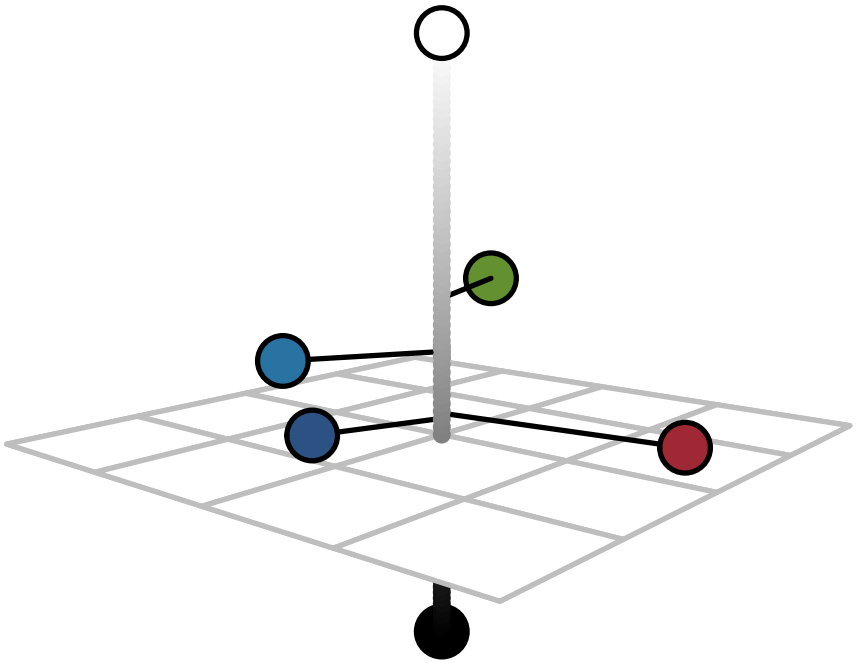
Converting SVG to greyscale



Converting SVG to greyscale



Converting SVG to greyscale



Converting SVG to greyscale

```
USDHSlogo <- readLines("US_Department_of_Homeland_Security_Seal.svg")

colourPattern <- "#[0-9ABCDEF]6"
colourLocns <- regexpr(colourPattern, USDHSlogo)
keep <- which(colourLocns > 0)

library(colorspace)
greyscale <- function(colour) {
    rgb <- hex2RGB(colour)
    lch <- as(rgb, "polarLUV")
    hex(polarLUV(lch@coords[1], 0, 0))
}

for (i in keep) {
    replacementColour <-
        greyscale(substr(USDHSlogo[i], colourLocns[i],
                         colourLocns[i] + attr(colourLocns,
                                               "match.length")[i] - 1))
    USDHSlogo[i] <- gsub(colourPattern, replacementColour,
                         USDHSlogo[i])
}

writeLines(USDHSlogo, "USDHSsealgray.svg")
```

Converting SVG to greyscale

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Generator: Adobe Illustrator 12.0.0, SVG Export Plug-In  -->
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
          "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd" [
    <!ENTITY ns_flows "http://ns.adobe.com/Flows/1.0/">
    <!ENTITY ns_svg "http://www.w3.org/2000/svg">
    <!ENTITY ns_xlink "http://www.w3.org/1999/xlink">
]>
<svg  version="1.1" xmlns="&ns_svg;" xmlns:xlink="&ns_xlink;"
      xmlns:a="http://ns.adobe.com/AdobeSVGViewerExtensions/3.0/"
      width="360" height="359" viewBox="-0.469 -0.067 360 359"
      enable-background="new -0.469 -0.067 360 359" xml:space="preserve">
<defs>
</defs>
<g>
    <g>
        <circle fill="#FFFFFF" cx="178.755" cy="179.102" r="176.553"/>
        <path fill="#505050" d="M293.496,194.116l-8.784,2.705l1.987,
                                0.811l-3.076,1.948c0.327-1.792,0.645
        ...
```

Converting SVG to greyscale

Colorspace

- colorspace package for R
  `http://r-forge.r-project.org/projects/colorspace/`

Scavenging Type 1 Fonts

- LaTeX:

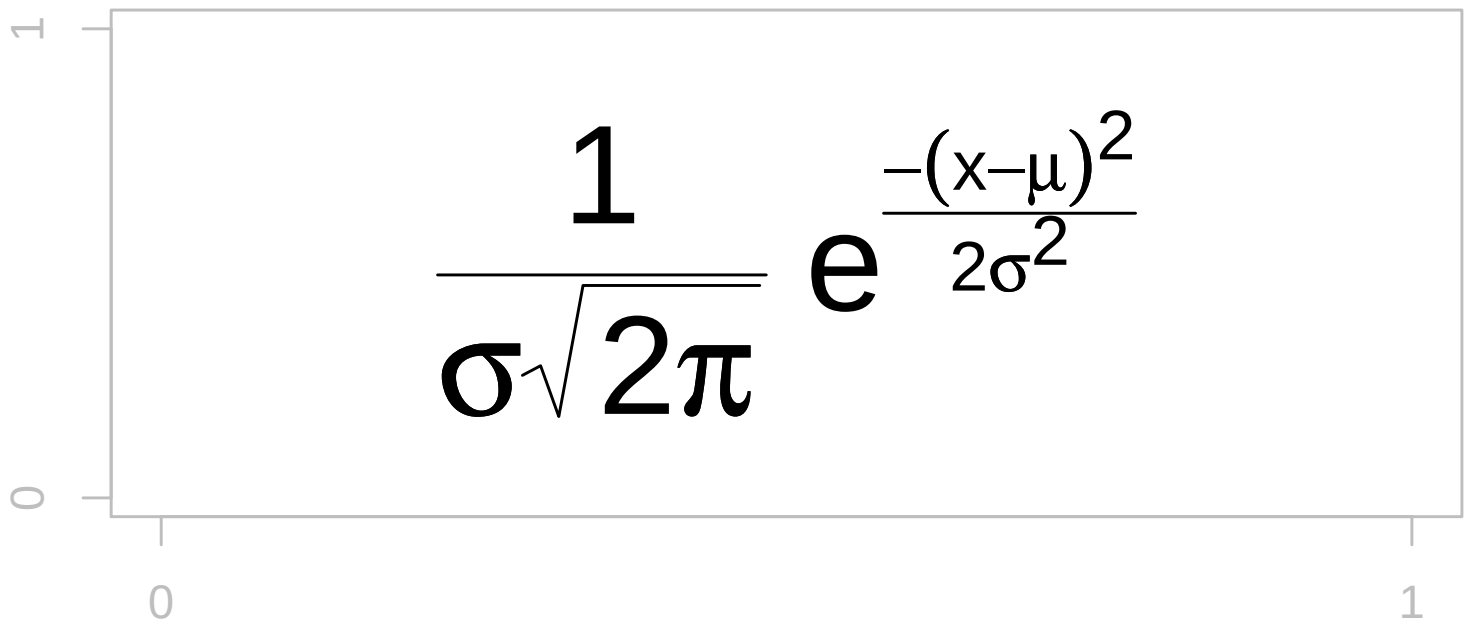
```
$$\frac{1}{\sigma \sqrt{2\pi}}
  e^{\frac{-(x-\mu)^2}{2 \sigma^2}}$$
```

- R plotmath:

```
text(0.5, 0.5,
     expression(paste(frac(1, sigma*sqrt(2*pi)),
                      " ",
                      e^{frac(-(x-mu)^2,
                              2*sigma^2)})),
```

Scavenging Type 1 Fonts

$$\frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

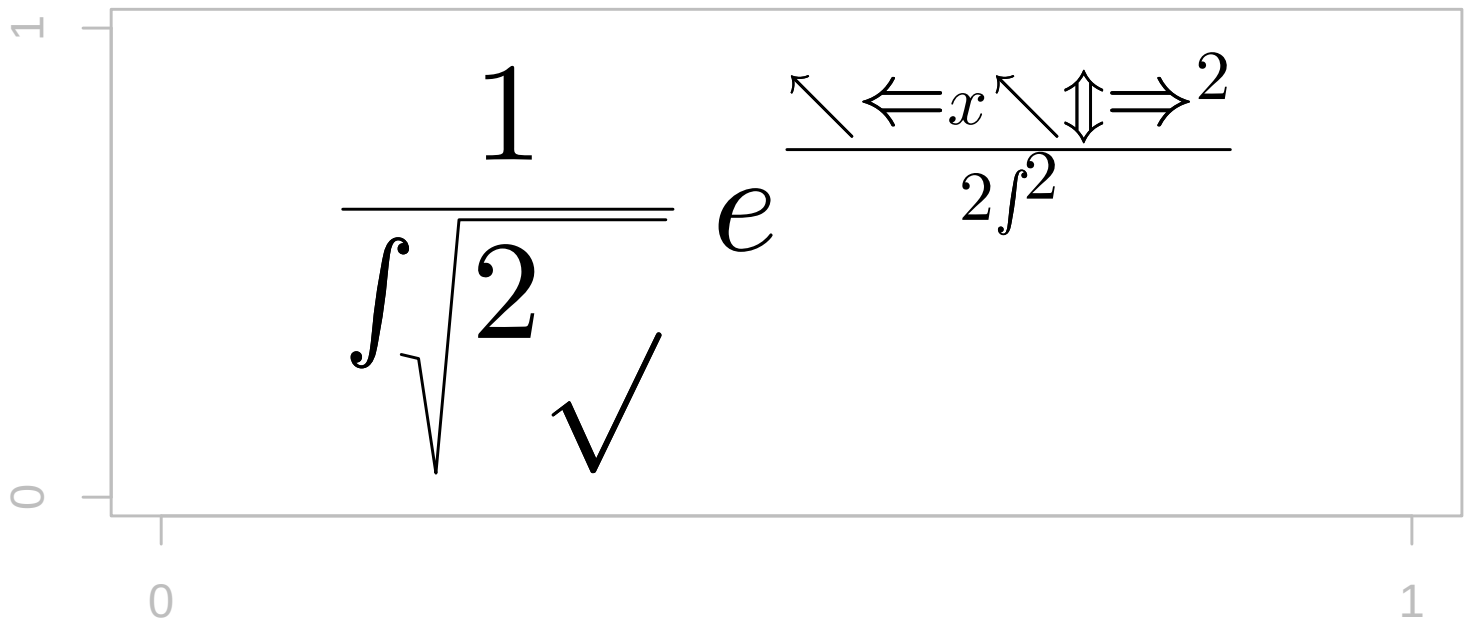


Scavenging Type 1 Fonts

```
pdfFonts(cmmi=Type1Font("CMMI",
            c("fonts/fcmr8a.afm",
              "fonts/fcmb8a.afm",
              "fonts/cmmi10.afm",
              "fonts/fcmbi8a.afm",
              "cmpsfont/afm/cmsy10.afm")))
```

Scavenging Type 1 Fonts

$$\frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$



Scavenging Type 1 Fonts

	Adobe		**CMMI**	
32	space		psi	ψ
33	exclam	!	omega	ω
34	universal	$\forall$	epsilon	ϵ
35	numbersign	#	theta1	θ
36	existential	$\exists$	pi1	π
37	percent	%	rho1	ρ
38	ampersand	&	sigma1	σ
39	suchthat	$\ni$	phi1	ϕ

Scavenging Type 1 Fonts

Type 1 = .pfb + .afm

```
4000  :  0f 5d 82 b6 9c dd f8 ff 4d ac fa 9c 54 be d5 a3  |  .]......M...T...
4016  :  aa 3e a5 b1 29 fe 96 be 63 28 43 b9 b6 bc 91 b6  |  .>..)...c(C.....
4032  :  15 58 1a 98 5d b5 6b 1e 01 ca 60 ee 69 ca 92 cf  |  .X..].k...`.i...
4048  :  5c 08 82 ec e6 2e da d3 e1 06 d8 35 34 88 22 40  |  \..........54."@
4064  :  0f 0b 66 af 65 8f 2a e5 6e d0 8f 8b 00 10 57 18  |  ..f.e.*.n.....W.
4080  :  07 00 9b 73 ab 12 a8 cf 14 ca 6c 71 f0 3c 2a 48  |  ...s......lq.<*H
4096  :  c5 00 f9 d6 22 66 af 15 4a 63 75 ff 60 0d 9b ac  |  ...."f..Jcu.`...
4112  :  3f 05 ce 34 14 2d 68 67 a7 95 81 c5 33 17 6b b2  |  ?..4.-hg....3.k.
4128  :  f3 11 73 36 67 1e 2e 44 63 8a 97 16 7e 2e a9 64  |  ..s6g..Dc...~..d
4144  :  4e 31 ea 16 c2 ad 29 90 ea 33 c5 40 01 e0 c8 15  |  N1....)..3.@....
4160  :  6e 6d e8 ab 6a 4d 40 a7 13 7b a2 75 f3 95 89 fe  |  nm..jM@..{.u....
4176  :  a2 e2 db 82 56 ad c1 03 d6 f9 cc 03 80 37 a4 7e  |  ....V........7.~
4192  :  8f d4 69 c5 f9 8a 5e 3c                          |  ..i...^<
```

```
t1utils-1.32/t1disasm cmr10.pfb cmr10.raw
```

Scavenging Type 1 Fonts

```
/Delta {
        47 2500 3 div
        hsbw
        0 76 hstem
        696 20 hstem
        0 738 vstem
        395 698 rmoveto
        -7 13 -2 5 -17 0 rrcurveto
        -17 0 -2 -5 -7 -13 rrcurveto
        -338 -678 rlineto
        -5 -9 0 -2 0 -1 rrcurveto
        -8 6 0 16 vhcurveto
        694 hlineto
        16 6 0 8 hvcurveto
        0 1 0 2 -5 9 rrcurveto
        closepath
        -396 596 rmoveto
        269 -540 rlineto
        -539 hlineto
        closepath
        endchar
        } ND
```

Scavenging Type 1 Fonts

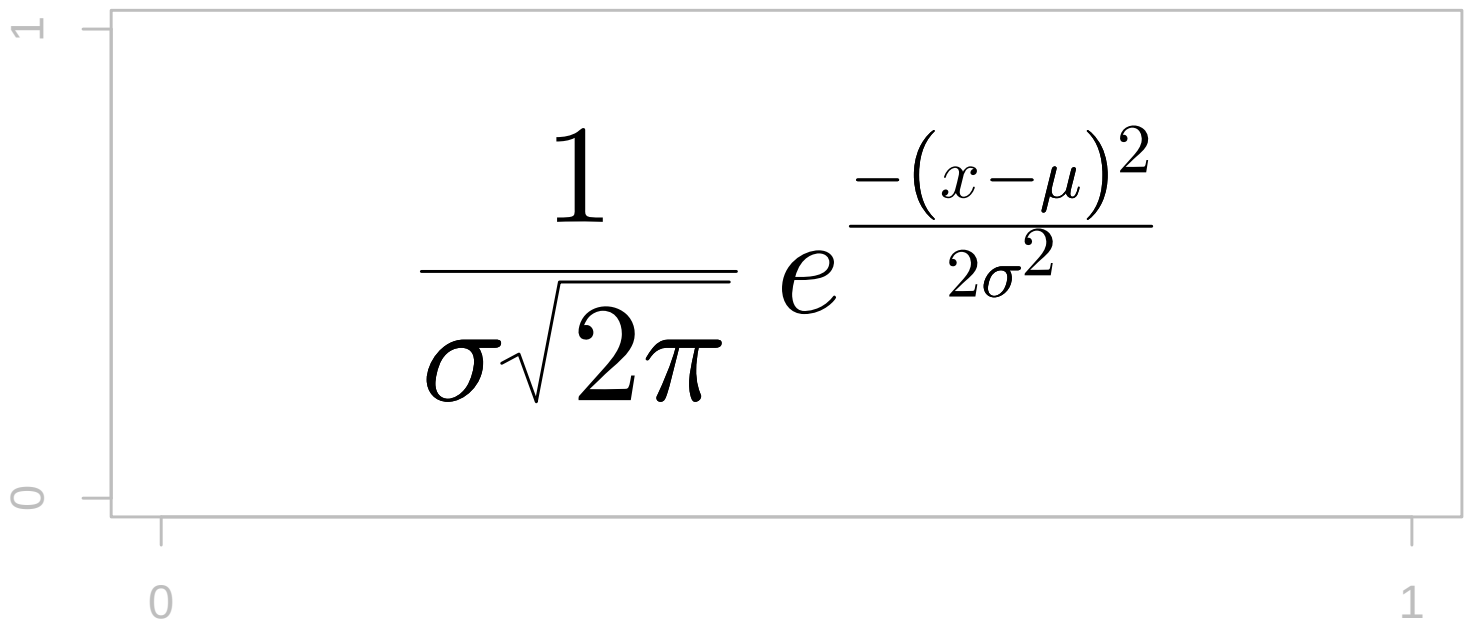
```
C 0 ; WX 625 ; N Gamma ; B 33 0 582 680 ;
C 1 ; WX 833.333 ; N Delta ; B 47 0 785 716 ;
C 2 ; WX 777.778 ; N Theta ; B 56 -22 721 705 ;
C 3 ; WX 694.444 ; N Lambda ; B 32 0 661 716 ;
C 4 ; WX 666.667 ; N Xi ; B 42 0 624 677 ;
C 5 ; WX 750 ; N Pi ; B 33 0 716 680 ;
C 6 ; WX 722.222 ; N Sigma ; B 56 0 665 683 ;
C 7 ; WX 777.778 ; N Upsilon ; B 56 0 721 705 ;
C 8 ; WX 722.222 ; N Phi ; B 56 0 665 683 ;
C 9 ; WX 777.778 ; N Psi ; B 57 0 720 683 ;
C 10 ; WX 722.222 ; N Omega ; B 44 0 677 705 ;
C 11 ; WX 583.333 ; N ff ; B 27 0 628 705 ; L i ffi ; L l ffl ;
C 12 ; WX 555.556 ; N fi ; B 27 0 527 705 ;
C 13 ; WX 555.556 ; N fl ; B 27 0 527 705 ;
C 14 ; WX 833.333 ; N ffi ; B 27 0 804 705 ;
C 15 ; WX 833.333 ; N ffl ; B 27 0 804 705 ;
C 16 ; WX 277.778 ; N dotlessi ; B 33 0 247 442 ;
C 17 ; WX 305.556 ; N dotlessj ; B -40 -205 210 442 ;
C 18 ; WX 500 ; N grave ; B 107 510 293 698 ;
C 19 ; WX 500 ; N acute ; B 206 510 392 698 ;
C 20 ; WX 500 ; N caron ; B 118 516 381 638 ;
```

Scavenging Type 1 Fonts

```
pdfFonts(cmsyase=Type1Font("CMSYASE",
            c("fonts/fcmr8a.afm",
              "fonts/fcmb8a.afm",
              "fonts/cmmi10.afm",
              "fonts/fcmbi8a.afm",
              "fonts/cmsyase.afm")))
```

Scavenging Type 1 Fonts

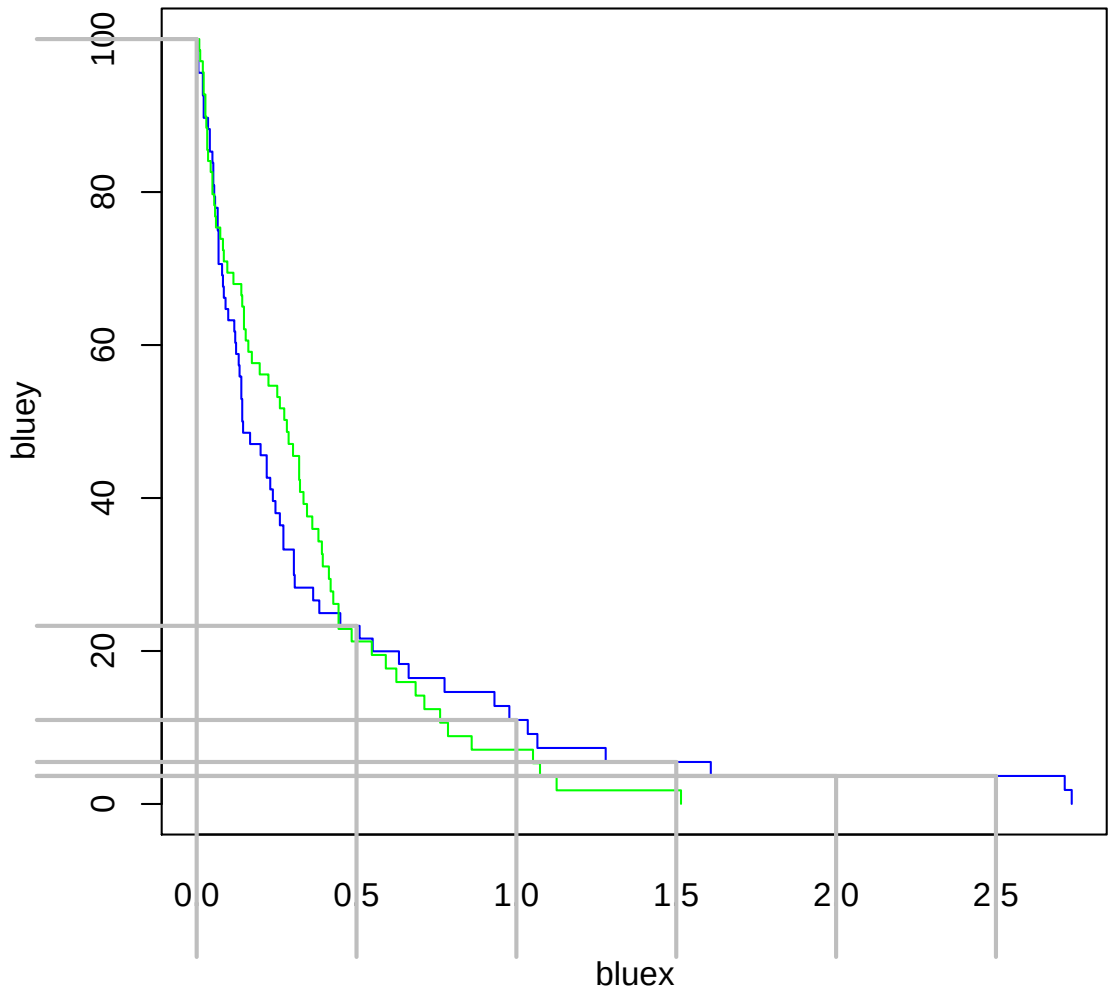
$$\frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$



Computer Modern font in Adobe Encoding

- The cmsyase font
  `http://www.stat.auckland.ac.nz/~paul/R/CM/CMR.html`

Extracting Data from Plots



Extracting Data from Plots

R News

The Newsletter of the R Project Volume 4/1, June 2004

Editorial

by Thomas Lumley

R has been well accepted in the academic world for some time, but this issue of R News begins with a description of the use of R in a business setting. Marc Schwartz describes how his consulting company, MedAnalytics, came to use R instead of commercial statistical software, and has since provided both code and financial support to the R Project.

The growth of the R community was also evident at the very successful useR! 2004, the first conference for R users. While R aims to blur the distinctions between users and programmers, useR! was definitely different in aims and audience from the previous DSC conferences held in 1999, 2001, and 2003. John Fox provides a summary of the conference in this issue for those who could not make it to Vienna.

A free registration to useR! was the prize in the competition for a new graphic for http://www.r-project.org. You will have seen the winning graphic, from Eric Lecoutre, on the way to downloading this newsletter. The graphic is unedited R output (click on it to see the code), showing the power of R for creating complex graphs that are completely reproducible. I would also encourage R users to visit the Journal of Statistical Software (http:///www.jstatsoft.org), which publishes peer-reviewed code and documentation. R is heavily represented in recent issues of the journal.

The success of R in penetrating so many statistical niches is due in large part to the wide variety of packages produced by the R community. We have articles describing contributed packages for principal component analysis (ade4, used in creating Eric Lecoutre's winning graphic) and statistical process control (qcc) and the recommended survival package. Jianhua Zhang and Robert Gentlement describe tools from the Bioconductor project that make it easier to explore the resulting variety of code and documentation.

The Programmers' Niche and Help Desk columns both come from guest contributors this time. Gabor Grothendieck,known as frequent poster of answers on the mailing list r-help, has been invited to contribute to the R Help Desk. He presents "Date and Time Classes in R". For the Programmers' Niche, I have written a simple example of classes and methods using the old and new class systems in R. This example arose from a discussion at an R programming course.

Programmers will also find useful Doug Bates' article on least squares calculations. Prompted by discussions on the r-devel list, he describes why the simple matrix formulae from so many textbooks are not the best approach either for speed or for accuracy.

Thomas Lumley
Department of Biostatistics
University of Washington, Seattle
thomas.lumley@R-project.org

Contents of this issue:

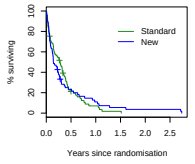


Figure 1: Survival distributions for two lung cancer treatments

Proportional hazards models

The mainstay of survival analysis in the medical world is the Cox proportional hazards model and its extensions. This expresses the hazard (or rate) of events as an unspecified baseline hazard function multiplied by a function of the predictor variables.

Writing $h(t;z)$ for the hazard at time t with predictor variables $Z = z$ the Cox model specifies

$$\log h(t,z) = \log h_0(t) e^{\beta z}.$$

Somewhat unusually for a semiparametric model, there is very little loss of efficiency by leaving $h_0(t)$ unspecified, and computation is, if anything, easier than for parametric models.

A standard example of the Cox model is one constructed at the Mayo Clinic to predict survival in patients with primary biliary cirrhosis, a rare liver disease. This disease is now treated by liver transplantation, but at the same there was no effective treatment. The model is based on data from 312 patients in a randomised trial.

```
> data(pbc)
> mayomodel<-coxph(Surv(time,status)~edtrt+
                   log(bili)+log(protime)+
                   age+platelet,
          data=pbc, subset=trt>0)
> mayomodel
Call:
coxph(formula = Surv(time, status) ~ edtrt +
    log(bili) + log(protime) +
    age + platelet, data = pbc,
    subset = trt > 0)

                coef exp(coef)
edtrt        1.02980     2.800
log(bili)    0.95100     2.588
log(protime) 2.88544    17.911
age          0.03544     1.036
platelet    -0.00128     0.999
             se(coef)     z       p
edtrt        0.300321  3.43 0.00061
log(bili)    0.097771  9.73 0.00000
log(protime) 1.031908  2.80 0.00520
age          0.008489  4.18 0.00003
platelet     0.000927 -1.38 0.17000

Likelihood ratio test=185  on 5 df, p=0  n= 312
```

The survexp function can be used to compare predictions from a proportional hazards model to actual survival. Here the comparison is for 106 patients who did not participate in the randomised trial. They are divided into two groups based on whether they had edema (fluid accumulation in tissues), an important risk factor.

```
> plot(survfit(Surv(time,status)~edtrt,
       data=pbc,subset=trt==-9))
> lines(survexp(~edtrt+
                ratetable(edtrt=edtrt,bili=bili,
                      platelet=platelet,age=age,
                      protime=protime),
                data=pbc,
                subset=trt==-9,
                ratetable=mayomodel,
                cohort=TRUE),
          col="purple")
```

The ratetable function in the model formula wraps the variables that are used to match the new sample to the old model.

Figure 2 shows the comparison of predicted survival (purple) and observed survival (black) in these 106 patients. The fit is quite good, especially as people who do and do not participate in a clinical trial are often quite different in many ways.

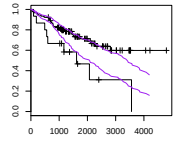


Figure 2: Observed and predicted survival

The main assumption of the proportional hazards model is that hazards for different groups are in fact proportional, *i.e.* that β is constant over time. The

 ISSN 1609-3631

Extracting Data from Plots

```
%!PS-Adobe-3.0
%%Pages: (atend)
%%BoundingBox: 54 22 551 798
%%HiResBoundingBox: 54.900000 22.600000 550.700000 797.100000
%.....................................
%%Creator: ESP Ghostscript 81503 (pswrite)
%%CreationDate: 2007/10/04 10:06:15
%%DocumentData: Clean7Bit
%%LanguageLevel: 2
%%EndComments
%%BeginProlog
% This copyright applies to everything between here and the %%EndProlog:
% Copyright (C) 2004 artofcode LLC, Benicia, CA.  All rights reserved.
%%BeginResource: procset GS_pswrite_2_0_1001
/GS_pswrite_2_0_1001 80 dict dup begin
/PageSize 2 array def/setpagesize{ PageSize aload pop 3 index eq exch

...

918.92 6272.74 1090.81 0 S
918.92 6272.74 0 -45 S
1137.1 6272.74 0 -45 S
1355.23 6272.74 0 -45 S
```

Extracting Data from Plots

```
%!PS-Adobe-2.0 EPSF-1.2
%%BeginProcSet:convertToR 0 0

...

/mycurve {
  (curve ) print
  str cvs print ( ) print
  str cvs print ( ) print
  } def

...

/stroke
  flattenpath {mymove} {myline} {mycurve} {myclose}
  mystroke
  newpath
 def

...
(./page27.ps) run
```

Extracting Data from Plots

```
<?xml version='1.0'?>

<picture xmlns:rgml = 'http://r-project.org/RGML'>

<path type='stroke' id='1'>
<context>
<rgb r='0' g='0' b='0'/>
<style lwd='5.2934'/>
</context>

<move y='7860.82' x='549.918'/>
<line y='7860.82' x='5368.82'/>
</path>

<path type='stroke' id='2'>
<context>
<rgb r='0' g='0' b='0'/>
<style lwd='6.23438'/>
</context>

<move y='6272.74' x='918.918'/>
<line y='6272.74' x='2009.73'/>
</path>
```

Extracting Data from Plots

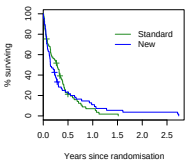


Figure 1: Survival distributions for two lung cancer treatments

Proportional hazards models

The mainstay of survival analysis in the medical world is the Cox proportional hazards model and its extensions. This expresses the hazard (or rate) of events as an unspecified baseline hazard function multiplied by a function of the predictor variables.

Writing $h(t;z)$ for the hazard at time t with predictor variables $Z = z$ the Cox model specifies

$$\log h(t,z) = \log h_0(t) e^{\beta z}.$$

Somewhat unusually for a semiparametric model, there is very little loss of efficiency by leaving $h_0(t)$ unspecified, and computation is, if anything, easier than for parametric models.

A standard example of the Cox model is one constructed at the Mayo Clinic to predict survival in patients with primary biliary cirrhosis, a rare liver disease. This disease is now treated by liver transplantation, but at the same there was no effective treatment. The model is based on data from 312 patients in a randomised trial.

```
> data(pbc)
> mayomodel<-coxph(Surv(time,status)~edtrt+
                   log(bili)+log(protime)+
                   age+platelet,
                 data=pbc, subset=trt>0)
> mayomodel
Call:
coxph(formula = Surv(time, status) ~ edtrt +
          log(bili) + log(protime) +
           age + platelet, data = pbc,
           subset = trt > 0)


                coef exp(coef)
edtrt        1.02980     2.800
log(bili)    0.95100     2.588
log(protime) 2.88544    17.911
age          0.03544     1.036
platelet    -0.00128     0.999
                se(coef)     z       p
edtrt        0.300321  3.43 0.00061
log(bili)    0.097771  9.73 0.00000
log(protime) 1.031908  2.80 0.00520
age          0.008489  4.18 0.00003
platelet     0.000927 -1.38 0.17000

Likelihood ratio test=185  on 5 df, p=0  n= 312
```

The survexp function can be used to compare predictions from a proportional hazards model to actual survival. Here the comparison is for 106 patients who did not participate in the randomised trial. They are divided into two groups based on whether they had edema (fluid accumulation in tissues), an important risk factor.

```
> plot(survfit(Surv(time,status)~edtrt,
       data=pbc,subset=trt==-9))
> lines(survexp(~edtrt+
                ratetable(edtrt=edtrt,bili=bili,
                      platelet=platelet,age=age,
                      protime=protime),
                data=pbc,
                subset=trt==-9,
                ratetable=mayomodel,
                cohort=TRUE),
         col="purple")
```

The ratetable function in the model formula wraps the variables that are used to match the new sample to the old model.

Figure 2 shows the comparison of predicted survival (purple) and observed survival (black) in these 106 patients. The fit is quite good, especially as people who do and do not participate in a clinical trial are often quite different in many ways.

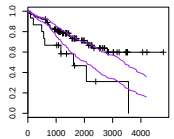
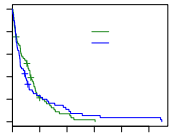
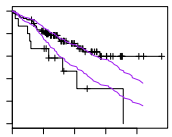


Figure 2: Observed and predicted survival

The main assumption of the proportional hazards model is that hazards for different groups are in fact proportional, *i.e.* that β is constant over time. The

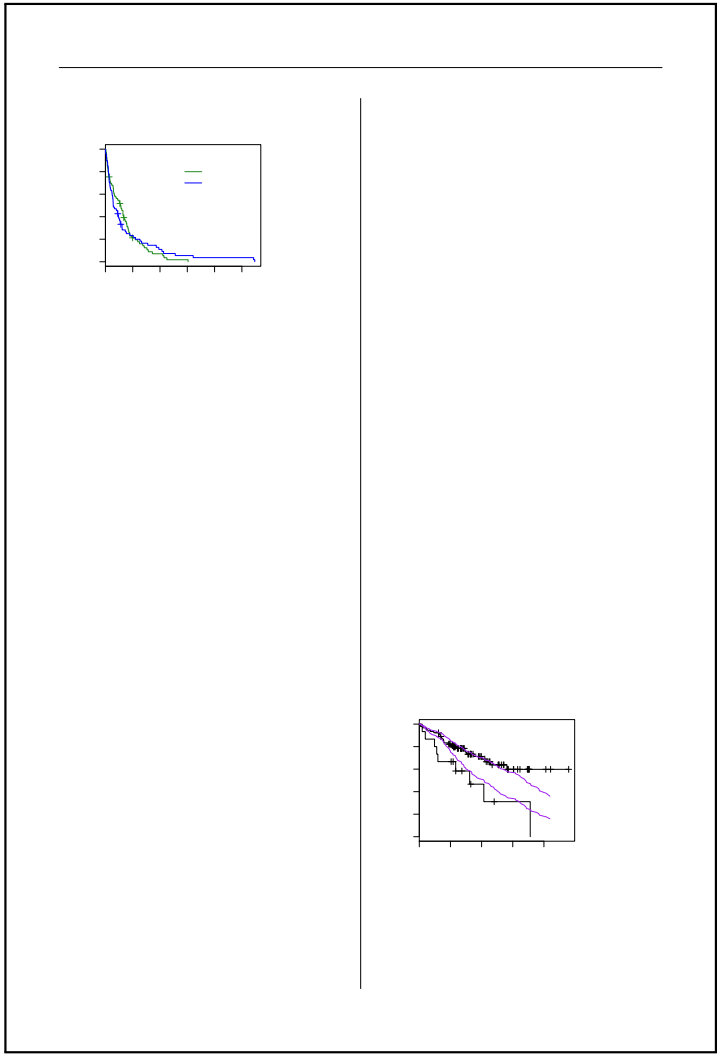


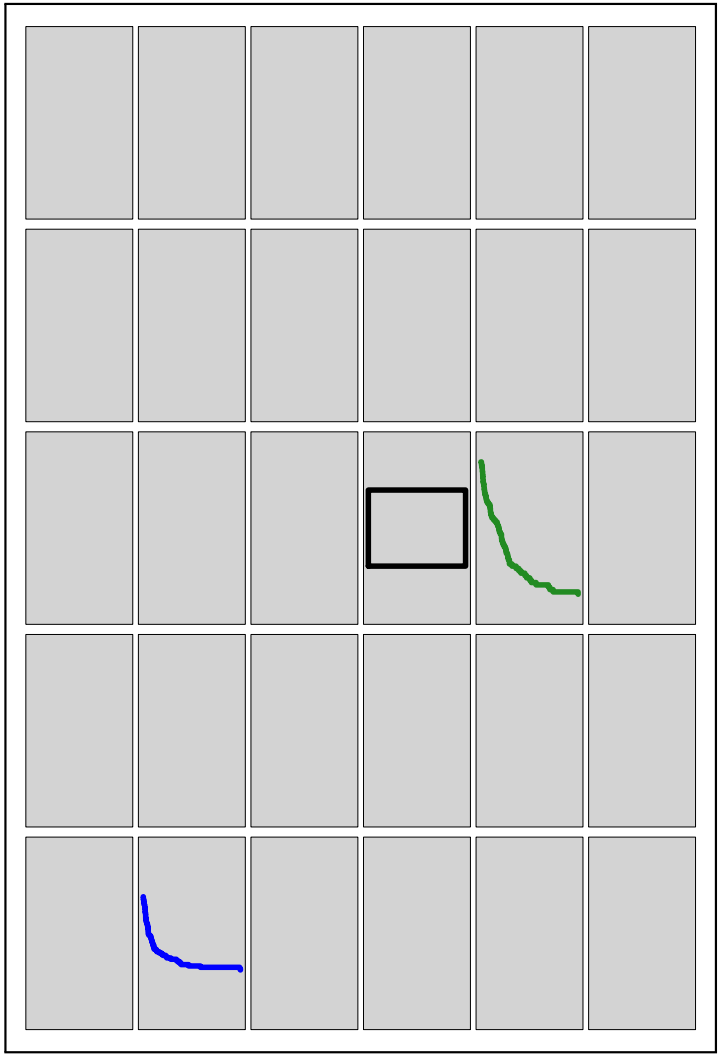


Extracting Data from Plots

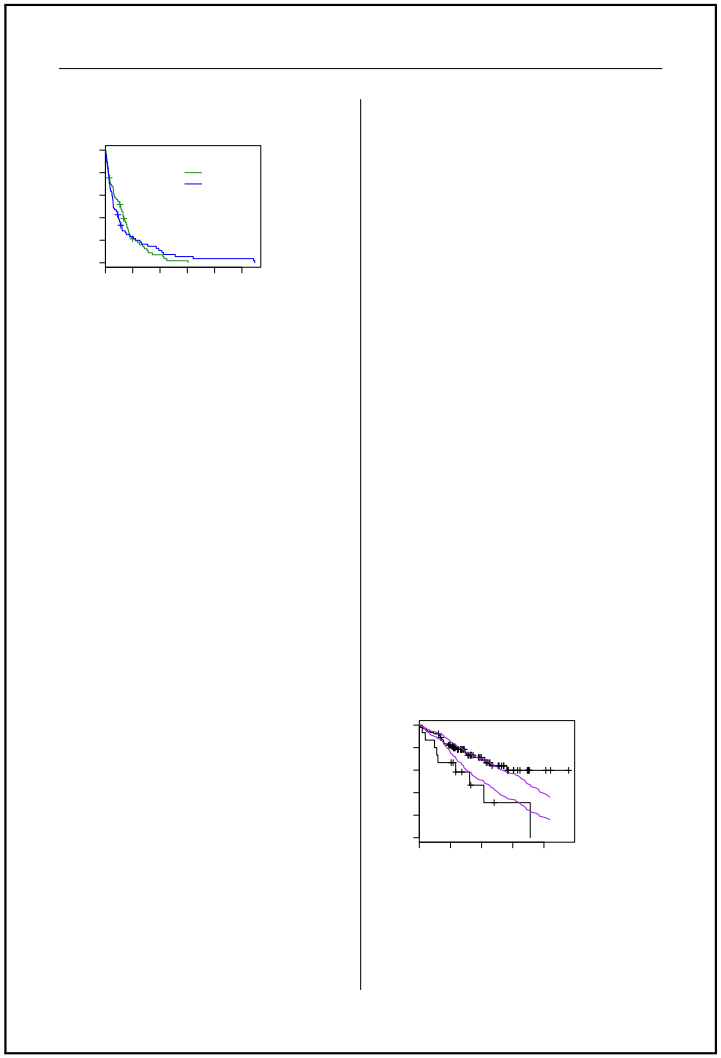
```
> page27 <- readPicture("page27.ps.xml")

> page27@summary

An object of class "PictureSummary"
Slot "numPaths":
count
  191

Slot "xscale":
    xmin     xmax
 549.918 5368.820

Slot "yscale":
    ymin     ymax
 499.039 7860.820
```

Extracting Data from Plots

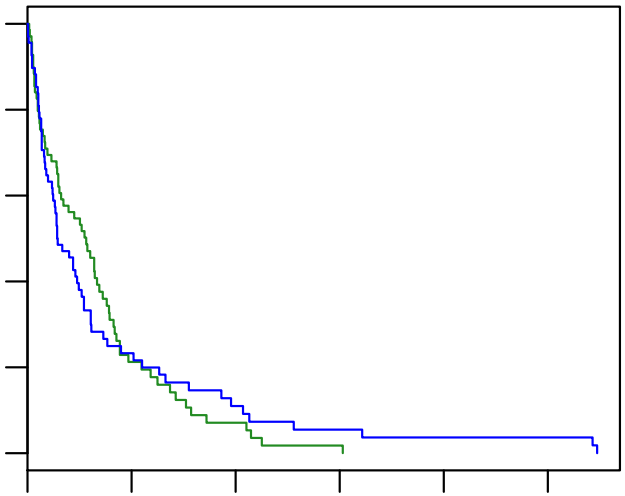




Extracting Data from Plots



```
barePlot <-
  page27[c(3:16, 17, 26)]
```



Extracting Data from Plots

```
> barePlot@paths[[1]]

An object of class "PictureStroke"
Slot "x":
move line
 919  919

Slot "y":
move line
6273 6228

Slot "rgb":
[1] "#000000"

Slot "lwd":
[1] 6.23
```

Extracting Data from Plots

```
yunit <- (page27@paths[[15]]@y[1] -
          page27@paths[[10]]@y[1])/100
```

```
          x     y
1   0.00000 100.0
2   0.00270 100.0
4   0.00543  97.1
6   0.01918  95.6
8   0.02188  92.6
10  0.03563  89.7
...
```

Importing Graphics Files

- grImport package for R
  `http://cran.r-project.org/src/contrib/Descriptions/grImport.html`

Summary

- Image compositing operators
- SVG file format
- Colourspaces
- Type 1 font file structure and format
- PostScript file format and language
- *Text processing*

Acknowledgements

- The seal of the United States Department of Homeland Security was obtained from Wikipedia.
- The immigration data are from the Homeland Secutiry web site `http://www.dhs.gov/ximgtn/statistics/publications/LPR06.shtm`
- **R News** is a publication of the **R Foundation for Statistical Computing**. The article used is **The survival Package** by **Thomas Lumley** R News, 4(1), pp. 26–28.
- The application for measuring survival curves was suggested by **Dan Jackson**.