Class Notes For STT 3850

Alan T. Arnholt

Last compiled Tuesday, April 01, 2014 - 13:28:43.

Most of the material in the notes is taken from the class text *Mathematical Statistics with Resampling and R* by Laura Chihara and Tim Hesterberg with slight modifications. Some material in the notes is also taken from the first chapter of *Practicing Statistics: Guided Investigations for the Second Course* by Shonda Kuiper and Jeffrey Sklar. There are a number of places to get help with R. The class text has some material online at: https://sites.google.com/site/chiharahesterberg/rtutorials.

Exploratory Data Analysis

Reading *.csv Data

```
site <- "http://www1.appstate.edu/~arnholta/Data/FlightDelays.csv"
FlightDelays <- read.csv(file = url(site))
head(FlightDelays)  # shows first 6 rows of data frame
```

```
  ID Carrier FlightNo Destination DepartTime Day Month
1  1      UA      403         DEN      4-8am Fri   May
2  2      UA      405         DEN     8-Noon Fri   May
3  3      UA      409         DEN      4-8pm Fri   May
4  4      UA      511         ORD     8-Noon Fri   May
5  5      UA      667         ORD      4-8am Fri   May
6  6      UA      669         ORD      4-8am Fri   May
  FlightLength Delay Delayed30
1          281    -1        No
2          277   102       Yes
3          279     4        No
4          158    -2        No
5          143    -3        No
6          150     0        No
```

```
str(FlightDelays)  # shows structure of data frame
```

```
'data.frame':   4029 obs. of  10 variables:
 $ ID          : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Carrier     : Factor w/ 2 levels "AA","UA": 2 2 2 2 2 2 2 2 2 2 ...
 $ FlightNo    : int  403 405 409 511 667 669 673 677 679 681 ...
 $ Destination : Factor w/ 7 levels "BNA","DEN","DFW",..: 2 2 2 6 6 6 6 6 6 6 ...
 $ DepartTime  : Factor w/ 5 levels "4-8am","4-8pm",..: 1 4 2 4 1 1 4 4 5 5 ...
 $ Day         : Factor w/ 7 levels "Fri","Mon","Sat",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Month       : Factor w/ 2 levels "June","May": 2 2 2 2 2 2 2 2 2 2 ...
 $ FlightLength: int  281 277 279 158 143 150 158 160 160 163 ...
 $ Delay       : int  -1 102 4 -2 -3 0 -5 0 10 60 ...
 $ Delayed30   : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 1 1 2 ...
```

```r
levels(FlightDelays$Month)
```

```
[1] "June" "May"
```

```r
FlightDelays$Month <- factor(FlightDelays$Month, levels = c("May",
    "June"))
levels(FlightDelays$Month)
```

```
[1] "May"  "June"
```

Creating Tables

```r
table(FlightDelays$Carrier)
```

```

  AA   UA 
2906 1123 
```

```r
xtabs(~Carrier, data = FlightDelays)
```

```
Carrier
  AA   UA 
2906 1123 
```

Creating Barplots

```r
barplot(table(FlightDelays$Carrier))
```

```r
require(ggplot2)
ggplot(data = FlightDelays, aes(x = Carrier)) +
  geom_bar()
```

```r
ggplot(data = FlightDelays, aes(x = Carrier, fill= Month)) +
  geom_bar()
```

```r
ggplot(data = FlightDelays, aes(x = Carrier, fill= Month)) +
  geom_bar() +
  guides(fill = guide_legend(reverse = TRUE))
```

```r
ggplot(data = FlightDelays, aes(x = Carrier, fill= Month)) +
  geom_bar(position="dodge") +
  guides(fill = guide_legend(reverse = TRUE))
```

```r
xtabs(~ Carrier + (Delay > 30), data = FlightDelays)
```

```
       Delay > 30
Carrier FALSE TRUE
     AA  2513  393
     UA   919  204
```

```r
addmargins(xtabs(~ Carrier + (Delay > 30), data = FlightDelays))
```

```
       Delay > 30
Carrier FALSE TRUE  Sum
    AA   2513  393 2906
    UA    919  204 1123
    Sum  3432  597 4029
```

```r
ggplot(data = FlightDelays, aes(x = Carrier, fill= Delayed30)) +
  geom_bar(position="dodge")
```

```r
ggplot(data = FlightDelays, aes(fill = Carrier, x= Delayed30)) +
  geom_bar(position="dodge")
```

Histograms of `Delay` values.

```r
hist(FlightDelays$Delay)  # Ugly with Defaults...you change
```

```r
library(lattice)
histogram(~Delay, data = FlightDelays)
```

```r
histogram(~Delay, data = FlightDelays, type = "density")
```

```r
histogram(~Delay, data = FlightDelays, type = "density",
          panel = function(...){
            panel.histogram(col = "peru",...)
            panel.densityplot(col = "red", lwd = 2, ...)
          }
          )
```

```r
ggplot(data = FlightDelays, aes(x = Delay)) +
  geom_histogram()
```

```r
ggplot(data = FlightDelays, aes(x = Delay, y = ..density..)) +
  geom_histogram(binwidth = 10, color = "blue") +
  geom_density(color = "red")
```

```r
ggplot(data = FlightDelays, aes(x = Delay)) +
  geom_density(fill = "blue")
```

Numeric Summaries

```r
summary(FlightDelays)
```

```
       ID          Carrier        FlightNo      Destination
 Min.   :   1   AA:2906   Min.   :  71   BNA: 172
 1st Qu.:1008   UA:1123   1st Qu.: 371   DEN: 264
 Median :2015             Median : 691   DFW: 918
 Mean   :2015             Mean   : 827   IAD:  55
 3rd Qu.:3022             3rd Qu.: 787   MIA: 610
 Max.   :4029             Max.   :2255   ORD:1785
                                         STL: 225
    DepartTime      Day         Month       FlightLength
 4-8am    : 699   Fri:637   May :1999   Min.   : 68
 4-8pm    : 972   Mon:630   June:2030   1st Qu.:155
 8-Mid    : 257   Sat:453               Median :163
 8-Noon   :1053   Sun:551               Mean   :185
 Noon-4pm:1048   Thu:566               3rd Qu.:228
                  Tue:628               Max.   :295
                  Wed:564
     Delay         Delayed30
 Min.   :-19.0   No :3432
 1st Qu.: -6.0   Yes: 597
 Median : -3.0
 Mean   : 11.7
 3rd Qu.:  5.0
 Max.   :693.0
```

```r
sd(FlightDelays$Delay)
```

```
[1] 41.63
```

```r
sd(FlightDelays$Delay)^2
```

```
[1] 1733
```

```r
var(FlightDelays$Delay)
```

```
[1] 1733
```

```r
IQR(FlightDelays$Delay)
```

```
[1] 11
```

```r
quantile(FlightDelays$Delay)
```

```
  0%  25%  50%  75% 100%
 -19   -6   -3    5  693
```

Boxplots

```
boxplot(Delay ~ Carrier, data = FlightDelays)
```

```
bwplot(Delay ~ Carrier, data = FlightDelays)
```

```
bwplot(Delay ~ Carrier | Month, data = FlightDelays, as.table = TRUE,
    layout = c(2, 1))
```

```
ggplot(data = FlightDelays, aes(x = Carrier, y = Delay)) + geom_boxplot()
```

```
ggplot(data = FlightDelays, aes(x = Carrier, y = Delay)) + geom_boxplot() +
    facet_grid(. ~ Month)
```

```
site <- "http://www1.appstate.edu/~arnholta/Data/NCBirths2004.csv"
NCBirths <- read.csv(file=url(site))
head(NCBirths)
```

```
  ID MothersAge Tobacco Alcohol Gender Weight Gestation
1  1      30-34      No      No   Male   3827        40
2  2      30-34      No      No   Male   3629        38
3  3      35-39      No      No Female   3062        37
4  4      20-24      No      No Female   3430        39
5  5      25-29      No      No   Male   3827        38
6  6      35-39      No      No Female   3119        39
```

```
boxplot(Weight ~ Gender, data = NCBirths, col = c("pink", "blue"))
```

```
bwplot(Weight ~ Gender, data = NCBirths)
```

```
p <- ggplot(data = NCBirths, aes(x = Gender, y = Weight, fill = Gender))
p + geom_boxplot()
```

```
p + geom_boxplot() +
  guides(fill = FALSE) +
  labs( x = "Newborn Gender", y = "Weight in ounces", title = "You Put Something Here")
```

```
p + geom_boxplot() +
  guides(fill = FALSE) +
  labs( x = "Newborn Gender", y = "Weight in ounces", title = "You Put Something Here") +
  scale_fill_manual(values = c('pink', 'blue'))
```

```
p + geom_boxplot() + guides(fill = FALSE) +
  labs( x = "Newborn Gender", y = "Weight in ounces", title = "You Put Something Here") +
  scale_fill_brewer() + theme_bw()
```

Density Plots

```r
curve(dnorm(x), -4, 4, ylab = "", xlab = "")
x.region <- seq(from = 1, to = 4, length.out = 200)
y.region <- dnorm(x.region)
region.x <- c(x.region[1], x.region, x.region[200])
region.y <- c(0, y.region, 0)
polygon(region.x, region.y, col = "red")
abline(h = 0, lwd = 2)
```

```r
# Same now with ggplot2
p <- ggplot(data = data.frame(x = c(-4, 4)), aes(x = x))
dnorm_func <- function(x){
  y <- dnorm(x)
  y[x<1] <- NA
  return(y)
}
p1 <- p + stat_function(fun = dnorm_func, geom = 'area', fill = 'blue', alpha = 0.2) +
  geom_hline(yintercept = 0) +
  stat_function(fun = dnorm)
p1 + theme_bw()
```

```r
p1 + theme_bw() +
  labs(x = '', y = '', title = expression(integral(frac(1, sqrt(2*pi))*e^{-x^2/2}*dx, 1, infinity)==0.15
```

Example 2.11

Note this is not how `qqnorm` computes the quantiles! The left graph of Figure 2.9 in the book is not quite correct…it does not use the data in the table…the first value 17.7 should be 21.7.

```r
x <- c(21.7, 22.6, 26.1, 28.3, 30, 31.2, 31.5, 33.5, 34.7, 36)
n <- length(x)
p <- (1:10)/(n + 1)
q <- qnorm(p)
rbind(x, p, q)
```

```
      [,1]    [,2]    [,3]    [,4]    [,5]    [,6]    [,7]
x 21.70000 22.6000 26.1000 28.3000 30.0000 31.2000 31.5000
p  0.09091  0.1818  0.2727  0.3636  0.4545  0.5455  0.6364
q -1.33518 -0.9085 -0.6046 -0.3488 -0.1142  0.1142  0.3488
     [,8]    [,9]   [,10]
x 33.5000 34.7000 36.0000
p  0.7273  0.8182  0.9091
q  0.6046  0.9085  1.3352
```

```r
plot(q, x)
XS <- quantile(q, prob = c(0.25, 0.75))
YS <- quantile(x, prob = c(0.25, 0.75))
slopeA <- (YS[2] - YS[1])/(XS[2] - XS[1])
slopeB <- diff(YS)/diff(XS)
slopeA
```

```
  75%
5.873
```

```
slopeB
```

```
  75%
5.873
```

```
Intercept <- YS[1] - slopeA * XS[1]
Intercept
```

```
  25%
29.83
```

```
abline(a = Intercept, b = slopeA)
```

```
#
pc <- (1:10 - 3/8)/n
qc <- qnorm(pc)
rbind(x, pc, qc)
```

```
      [,1]    [,2]    [,3]    [,4]     [,5]    [,6]    [,7]
x  21.7000 22.6000 26.1000 28.3000 30.00000 31.2000 31.5000
pc  0.0625  0.1625  0.2625  0.3625  0.46250  0.5625  0.6625
qc -1.5341 -0.9842 -0.6357 -0.3518 -0.09414  0.1573  0.4193
      [,8]    [,9]   [,10]
x  33.5000 34.7000 36.0000
pc  0.7625  0.8625  0.9625
qc  0.7144  1.0916  1.7805
```

```
xs <- quantile(qc, prob = c(0.25, 0.75))
ys <- quantile(x, prob = c(0.25, 0.75))
slope <- diff(ys)/diff(xs)
intercept <- ys[1] - slope * xs[1]
c(intercept, slope)
```

```
   25%    75%
29.625  5.268
```

Consider using the `R` functions `qqnorm()` and `qqline()`.

```
qqnorm(x)
qqline(x)
```

```
# ggplot
ggplot(data = data.frame(x), aes(sample=x)) +
  stat_qq() +
  geom_abline(intercept = intercept, slope = slope)
```

Empirical Cumulative Distribution Function

The *empirical cumulative distribution function* (ecdf) is an estimate of the underlying cumulative distribution function for a sample. The empirical cdf, denoted by $\hat{F}$, is a step function

$$\hat{F}(x) = \frac{1}{n}(\text{number of values} \leq x),$$

where n is the sample size.

```
y <- c(3, 6, 15, 15, 17, 19, 24)
plot.ecdf(y)
```

```
set.seed(1)  # set seed for reproducibility
rxs <- rnorm(25)
plot.ecdf(rxs, xlim = c(-4, 4))
curve(pnorm(x), col = "blue", add = TRUE, lwd = 2)
```

An alternative approach to the book's Figure 2.12 is provided using `ggplot2` after first creating Figure 2.12

```
site <- "http://www1.appstate.edu/~arnholta/Data/Beerwings.csv"
Beerwings <- read.csv(file=url(site))
head(Beerwings) # shows first 6 rows of data frame
```

```
  ID Hotwings Beer Gender
1  1        4   24      F
2  2        5    0      F
3  3        5   12      F
4  4        6   12      F
5  5        7   12      F
6  6        7   12      F
```

```
str(Beerwings)  # shows structure of data frame
```

```
'data.frame':   30 obs. of  4 variables:
 $ ID      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Hotwings: int  4 5 5 6 7 7 7 8 8 8 ...
 $ Beer    : int  24 0 12 12 12 12 24 24 0 12 ...
 $ Gender  : Factor w/ 2 levels "F","M": 1 1 1 1 1 1 2 1 2 2 ...
```

```
beerM <- subset(Beerwings, select = Beer, subset = Gender == "M", drop = TRUE)
beerF <- subset(Beerwings, select = Beer, subset = Gender == "F", drop = TRUE)
plot.ecdf(beerM, xlab = "ounces", col = "blue", pch = 19)
plot.ecdf(beerF, col = "pink", pch = 19, add = TRUE)
abline(v = 25, lty = "dashed")
legend("topleft", legend = c("Males", "Females"), pch = 19, col = c("blue", "pink"))
```

```
# Using ggplot2 now
ggplot(data = Beerwings, aes(x = Beer, colour = Gender)) +
  stat_ecdf() +
  labs(x = "Beer in ounces", y ="", title = 'ECDF')
```

Scatter Plots

```r
with(data = Beerwings, plot(Hotwings, Beer, xlab = "Hot wings eaten", ylab = "Beer consumed",
                            pch = 19, col = "blue"))
```

```r
p <- ggplot(data = Beerwings, aes(x = Hotwings, y = Beer)) +
  geom_point() +
  labs(x = "Hot wings eaten", y = "Beer consumed in ounces")
p
```

```r
p + geom_smooth()
```

```r
p + geom_smooth(method = lm) + theme_bw()
```

Integrating with R

```r
f <- function(x) {
    (x - 1)^3 * exp(-x)
}
ans <- integrate(f, lower = 0, upper = Inf)$value
ans
```

```
[1] 2
```

```
{r child='../Children/Schistosomiasis.Rmd'} #

{r child='../Children/Verizon.Rmd'} #

{r child='../Children/CocaineForPDF.Rmd'} #

{r child='../Children/PRTMPD.Rmd'} #

{r child='../Children/CTables.Rmd'} #

{r child='../Children/SamplingDistributions.Rmd'} #

{r child='../Children/BootStrapEXP.Rmd'} #

{r child='../Children/InClass.Rmd'} #

{r child='../Children/InClassSol.Rmd'} #

{r child='../Children/ConfidenceIntervals.Rmd'} #
```