an introduction to R for epidemiologists

the basics

Charles DiMaggio, PhD, MPH, PA-C

Professor of Surgery and and Population Health
New York University School of Medicine
Bellevue Hospital
Division of Trauma and Surgical Critical Care
462 First Avenue, New York, NY 10016

Spring 2017

- http://www.injuryepi.org/
- Charles.DiMaggio@nyumc.org

Outline

1. Statistical Functions
2. regression
   - linear
   - logistic
   - poisson
3. Epidemiology Packages
   - epitools and epicalc
4. Survival Tools for Epidemiologists
   - from risks to rates
   - Kaplan-Meir and Proportional Hazards

base R comes with many statistical tools

summary statistics

- summary(), fivenum(), stem() - examine the distribution of a data set
- qqnorm(), qqline() normal plots
- boxplots() (a, b)

test statistics

- t.test() 2-sample t test, (a, b), note R does not by default assume equality of variances, (can use an F test to examine this assumption)
- var.test() returns an F test, (a,b)
- wilcox.test() returns a two-sample non-parametric Wilcoxon (aka Mann-Whitney) or one-sample Wilcoxon (specify "paired=TRUE") test

Some statistics with R

```
myDat<-data.frame(cbind(outcome1=rnorm(1000,20,5),
outcome2=rpois(1000,5),
grp=factor(sample(c("a","b","c"), 1000, replace=T))))

summary(myDat$outcome1)
fivenum(myDat$outcome1)
stem(myDat$outcome1)

boxplot(myDat)
boxplot(outcome1˜grp, data=myDat)

myDat2<-cbind(rnorm(1000,20,5), rpois(1000,5))
boxplot(myDat2)

qqnorm(myDat$outcome1)
qqline(myDat$outcome1)

t.test(myDat$outcome1, myDat$outcome2)

wilcox.test(myDat$outcome1, myDat$outcome2)
wilcox.test(myDat$outcome1, myDat$outcome2, paired=T)
```

Outline

1. Statistical Functions
2. regression
   - linear
   - logistic
   - poisson
3. Epidemiology Packages
   - epitools and epicalc
4. Survival Tools for Epidemiologists
   - from risks to rates
   - Kaplan-Meir and Proportional Hazards

modeling functions return minimal output

this is important:

assign the function to an object to extract additional output

```
my.reg<-lm(dat, x~y)
summary(my.reg)
names(my.reg)
predict(my.reg)
```

str() - to explore the object

functions return object classes

methods return results written for those classes

- linear regression: *lm (formula, data)*

  ```
  x <- lm(y~x, data=z)
  ```

- returns object of class "lm"
  - summary(x) comprehensive summary of results
  - print(x) precise version of the object
  - deviance(x) residuals
  - plot(x) returns plots: residuals, fitted values and some diagnostics
  - coef(x) extract regression coefficients
  - predict(x, newdata=) second argument takes a vector or matrix of new data values you want predictions for
  - step() add or drop terms, model with smallest AIC is returned

linear regression

John Fox car (companion to applied regression) package

```
install.packages("car")
library(car)
?Duncan
head(Duncan)
qqnorm(Duncan$income)
duncan.model<-lm(Duncan$prestige ~ Duncan$income + Duncan$education)
duncan.model
summary(duncan.model)
confint(duncan.model)

duncan.model2<-lm(prestige ~ income, data=Duncan)
plot(Duncan$prestige, Duncan$income)
abline(duncan.model2)

newIncome<-data.frame(income=c(82,90,92))
predict(duncan.model2, newIncome, interval = "confidence")
```

the plot() command for lm objects

residual analysis

```
qqPlot(duncan.model, labels=row.names(Duncan), simulate=TRUE)
library(MASS)
hist(studres(duncan.model)) #jackknife residuals
plot(studres(duncan.model))
abline(h = c(-25,25)*3/45)
identify(1:45, studres(duncan.model), row.names(Duncan))
        # R click to stop
layout(matrix(1:4,2,2))
plot(duncan.model)
```

- residuals vs. their fitted (regression) values - expect random distribution about horizontal line
- normal q-q - like probability plot, residuals vs. standardized normal values, expect straight diagonal line
- scale-location - square root of residuals vs. fitted values, again should be no obvious trend
- leverage plot - for influential values, measure of importance (influence) on the regression, Cook's d (distance) lines superimposed

updating models

- update(old.model, ...)
- where ... can be a new formula, or some other change
- e.g. re-run the duncan model without ministers and conductors

```
which.names(c("minister", "conductor"), Duncan)
duncan.model3<- update(duncan.model, subset=-c(6, 16))
summary(duncan.model3)
```

YOUR
TURN

linear regression of dig data

- go to http://www.injuryepi.org/styled-4/styled-6/ and read the "dig" data set into R
- hints:
  - R click to either download or get the file path
  - use read.csv with options stringsAsFactors=F and header=T
  - don't forget to save it to an object!
- use str() to explore the data set
- run a linear model with heart rate (HEARTRTE) as the outcome and body mass index (BMI) and diastolic blood pressure (DIABP) as predictors
- what is the coefficient for BMI?
- what does it mean?
- plot the regression diagnostics for your model
- what is your interpretation of them?

odds and log odds

- odds - ratio of two probabilities: $\frac{p}{1-p}$
- odds of Sunday 6:1 against (vs. prob Sunday 1/7)
  - in 7 trials, "fail" 6 times, "succeed" 1 time or...
  - probability of a Sunday 1/6 that of any other day, or...
  - 6 times more likely for a day other than Sunday, or...
  - decimal odds $1/6 = 0.166$ (vs. prob $1/7 = 0.143$)
    - decimal odds is a stake, e.g. bet on day of week being Sunday, 17 cents (0.166) wins a dollar
- odds in epi because unlike probabilities, not bounded by 1, so can approximate risk ratios
- logit - log of the odds of a binary outcome
  - $prob_{succeed} = prob_{fail}$, odds=1, logit=0

logistic model

- generalized linear model - response variable not normally distributed
  - glm - $y = f(x)$
- logistic function $y = \frac{e^{\beta_0 + \beta_i}}{1+e^{\beta_0 + \beta_i}}$
- logistic transformation - $logit(y) = \beta_0 + \beta_i$
  - start with probabilities
  - convert probability (constrained to 0 to 1) to odds $(\frac{p_i}{1-p_i})$ so values now range from 0 to infinity
  - take the log of the odds to make linear on range from minus to plus infinity
- logistic regression - linear regression on the logit transformed proportion or probability of an outcome at each value of the predictor

```
(probs<-seq(0,1,.05))
(odds<-probs/(1-probs))
log(odds)
plot(probs)
plot(odds)
plot(log(odds))
```

college admission example

from UCLA IDRE

```
admit <- read.csv("http://www.ats.ucla.edu/stat/data/binary.csv")
str(admit)
admit.mod1<-glm(admit ˜ gre + gpa +as.factor(rank), family=binomial(logit), data=admit)
summary(admit.mod1)

Call:
glm(formula = admit ˜ gre + gpa + as.factor(rank), family = binomial(logit),
    data = admit)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6268  -0.8662  -0.6388   1.1490   2.0790

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -3.989979   1.139951  -3.500 0.000465 ***
gre               0.002264   0.001094   2.070 0.038465 *
gpa               0.804038   0.331819   2.423 0.015388 *
as.factor(rank)2 -0.675443   0.316490  -2.134 0.032829 *
as.factor(rank)3 -1.340204   0.345306  -3.881 0.000104 ***
as.factor(rank)4 -1.551464   0.417832  -3.713 0.000205 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 499.98  on 399  degrees of freedom
Residual deviance: 458.52  on 394  degrees of freedom
AIC: 470.52

Number of Fisher Scoring iterations: 4
```

interpreting coefficients

- coefficients all significant
    - every one unit increase gre = 0.002 increase log odds of admission
    - one unit increase gpa = 0.804 increase log odds admission
    - institution with rank of 2, versus an institution with a rank of 1, decreases log odds admission by -0.675
- *confint(admit.mod1)* for confidence intervals
- *exp(cbind(OR = coef(admit.mod1), confint(admit.mod1)))* to exponentiate for odds ratios with CI's

YOUR
TURN

- http://www.injuryepi.org/resources/R/Exercises_EPIC_R_2014_NoAnswers.pdf
- Exercise 10: logistic regression: the Titanic

Poisson model

$$y_i \sim Poisson(\theta_i)$$
$$\theta_i = exp(X_i \beta)$$

- count data
- glm, log link
- θ constrained to be positive, fit on logarithmic scale
- each unit i is a *setting*, such as a time interval or spatial location, in which y_i events have occurred,
  - e.g. traffic crashes at intersection i in a given year
  - linear predictors X e.g. continuous measure average speed, indicator for traffic light
- note: if outcome is count or number of "successes" in some number of trials, standard to use binomial/logistic
  - if no natural limit on the number of outcomes, standard to use Poisson

offset variable

- makes sense to include a measure of *exposure*, υ

$$y_i \sim Poisson(\upsilon_i \theta_i)$$

- *log* υ called the *offset* variable
- a kind of baseline predictor in the model, equivalent to a regression coefficient with coefficient value fixed to 1

predictive interpretation Poisson regression coefficients

Gelman and Hill

- traffic crash model: effect of speed and traffic lights at intersections

$$y_i \sim Pois(e^{2.8+0.012X_{i1}-0.20X_{i2}})$$

- *intercept* (2.8) - crashes when speed is zero and no light, uninterpretable
- *speed coefficient* (X_{i1}) - expected difference on log scale for each addition mph average speed,
  - expected multiplicative increase is $e^{0.0012} = 1.012$, or 1.2% increase car crash rate for each 1 mph increase
  - might make more sense to multiply this by 10, so $e^{0.012} = 1.127$ for a 12.7% increase in crash rate per ten mph increase
- *traffic light indicator coefficient* (X_{i2}) - predictive difference of having a traffic light
  - multiply crash rate by $e^{-0.20} = 0.82$, or 18% reduction

traffic fatality example

loading and exploring the data

```
install.packages("AER") #applied econometrics in R
library(AER)

data(Fatalities)
?Fatalities
str(Fatalities)

#calculate incidence per state per year, plot as time series
table.deaths<-tapply(Fatalities$fatal, list(Fatalities$state,
Fatalities$year), sum)
table.exp<-tapply(Fatalities$milestot, list(Fatalities$state,
Fatalities$year), sum)

inc.dense<-table.deaths/table.exp*100
inc.dense
plot.ts(t(inc.dense), plot.type="single") #need to transpose
```

Poisson regression of traffic fatalities

effect of law enforcement vs economic

```
model1 <- glm(fatal ~ year,
 offset = log(milestot),family = poisson, data=Fatalities)
summary(model1)
str(model1)
exp(model1$coefficients)
exp(coef(model1))
exp(confint(model1))

model2 <- glm(fatal ~ year+state,
 offset = log(milestot),family = poisson, data=Fatalities)
summary(model2)

model3 <- glm(fatal ~ year+state+jail,
 offset = log(milestot),family = poisson, data=Fatalities)
summary(model3)
exp(coef(model3))
exp(confint(model3))
```

YOUR
TURN

- run a Poisson model for the effect of a beer tax ("beertax") on traffic fatalities controlling for year and state
- include an offset variable for total miles driven
- compare the results to the those for the effect of mandatory jail sentences

overdispersion in Poisson models

- Poisson variance is equal to mean, so s.d. is square root of the mean

$$E(y_i) = \upsilon_i \theta_i$$
$$sd(y_i) = \sqrt{\upsilon_i \theta_i}$$

- standardized residuals are

$$z_i = \frac{y_i - \hat{y}_i}{sd(\hat{y}_i)}$$
$$= \frac{y_i - \upsilon \hat{\theta}_i}{\sqrt{\upsilon_i \hat{\theta}_i}}$$

- if Poisson model true, expect z_i to have mean 0 and sd=1

testing for overdispersion

- compare sum of squares of z_i (Σz_i^2) to Chi square with n-k d.f.
- χ^2_{n-k} has average value of n-k, so $\frac{\Sigma z_i^2}{n-k}$ is an estimate of overdispersion
- values above 2 considered large
- R code from Gelman and Hill
  - set n to *nrow(data)* and k to the number of predictors

```
yhat <- predict (glm.police, type="response")
z <- (stops-yhat)/sqrt(yhat)
cat("overdispersion ratio is ", sum(z^2)/(n-k), "\n")
cat("p-value of overdispersion test is",pchisq (sum(z^2),
n-k),"\n")
```

- goodness of fit chi square test based on residuals and their df's

```
1 - pchisq(summary(model.pois)$deviance,
    summary(model.pois)$df.residual)
```

adjusting for overdispersion

- can multiply all regression s.e.'s by $\sqrt{overdispersion}$
- fit "quasipoisson" family or negative binomial model

```
model4 <- glm(fatal ~ year+state+beertax,
offset = log(milestot),family = poisson, data=Fatalities)
yhat <- predict(model4, type="response")
z <- (Fatalities$fatal-yhat)/sqrt(yhat)
sum(z^2)/(nrow(Fatalities)-(48+2))
#multiply s.e.'s by sqrt(5.897498), or...

library(MASS)
mod.nb<-glm.nb(fatal ~ year+state+beertax, offset(log(milestot
yhat <- predict(mod.nb, type="response")
z <- (Fatalities$fatal-yhat)/sqrt(yhat)
sum(z^2)/(nrow(Fatalities)-(48+2))
```

Outline

1. Statistical Functions
2. regression
   - linear
   - logistic
   - poisson
3. Epidemiology Packages
   - epitools and epicalc
4. Survival Tools for Epidemiologists
   - from risks to rates
   - Kaplan-Meir and Proportional Hazards

rate ratios, relative risks and odds ratios

rate ratio

$RR = rate_1 / rate_2 = \frac{x_1/p-t_1}{x_2/p-t_2}$

se for normal approximation of the rate ratio:

$se[ln(RR)] = \sqrt{\frac{1}{x_1} + \frac{1}{x_2}}$

relative risk

$RR = risk_1 / risk_2 = \frac{x_1/n_1}{x_2/n_2}$

se for normal approximation of the relative risk:

$se[ln(RR)] = \sqrt{\frac{1}{x_1} - \frac{1}{n_1} + \frac{1}{x_2} - \frac{1}{n_2}}$

(disease) odds ratio

$OR = odds_1 / odds_2 = \frac{x_1/(n_1-x_1)}{x_2/(n_2-x_2)}$

se for normal approximation of the odds ratio:

$se[ln(RR)] = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$

epitools for 2x2 tables

epitab()

calculates risks, risk ratios, odds ratios and their associated confidence intervals

```
install.packages("epitools")
library(epitools)
?epitab
dig<-read.csv("http://www.columbia.edu/~cjd11/
charles_dimaggio/DIRE/resources/R/dig.csv",
 stringsAsFactors=F) #digitalis data
names(dig)
table(dig$TRTMT,dig$DEATH)
```

using epitab

3 ways to feed data to epitab()

- table
- factors
- cell values (row-wise...)

```
tab.1<-xtabs(~TRTMT + DEATH, data=dig)
epitab(tab.1)

epitab(dig$TRTMT,dig$DEATH)

epitab(c(2209, 1194, 2216, 1181))

epitab(tab.1, rev="rows")
```

stratified analysis

manipulating results

Assign the results of a function to an object and extract elements you need

```
tab.1<-table(dig$TRTMT[dig$AGE<50],dig$DEATH[dig$AGE<50])
tab.2<-table(dig$TRTMT[dig$AGE>=50 & dig$AGE<65],
dig$DEATH[dig$AGE>=50 & dig$AGE<65])
tab.3<-table(dig$TRTMT[dig$AGE>=65],dig$DEATH[dig$AGE>=65])

or.1<-epitab(tab.1)
or.2<-epitab(tab.2)
or.3<-epitab(tab.3)

str(or.1)

young<-or.1$tab[2,5:7]
middle<-or.2$tab[2,5:7]
old<-or.3$tab[2,5:7]

my.table<-data.frame(rbind(young, middle, old))
my.table
```

more analyses

use tools from base R or other packages, e.g. exact tests, logistic regression

```
fisher.test(tab.1)
chisq.test(tab.1)
my.model<-glm(DEATH ~ TRTMT + SEX, data=dig, family=binomial)
summary(my.model)

exp(my.model$coef)
summary(my.model)$coef

sum.coef<-summary(my.model)$coef

est<-exp(sum.coef[,1])
upper.ci<-exp(sum.coef[,1]+1.96*sum.coef[,2])
lower.ci<-exp(sum.coef[,1]-1.96*sum.coef[,2])
cbind(est,upper.ci,lower.ci)

cbind(coef(my.model),confint(my.model))
```

epicalc

cc()

equivalent to epitab(), returns exact CI by default, and a descriptive graph

```
install.packages("epicalc")
library(epicalc)
?cc
```

the births data set

Is previous pre-term birth associated with low birth weight?

```
births<-read.csv("http://www.columbia.edu/~cjd11/
charles_dimaggio/DIRE/resources/R/births.csv",
header=T, stringsAsFactors=F)
names(births)
cc(births$low, births$prev_pretrm)
```

confounding

uterine irritibility

What are some other relationships in the data?

```
cc(births$uterine_irr,births$low)
cc(births$uterine_irr,births$prev_pretrm)
```

mhor(): the mantel-haenszel odds ratio

compare the unadjusted to the adjusted estimates

```
mhor(births$low, births$prev_pretrm,births$uterine_irr)
```

Outline

1. Statistical Functions
2. regression
   - linear
   - logistic
   - poisson
3. Epidemiology Packages
   - epitools and epicalc
4. Survival Tools for Epidemiologists
   - from risks to rates
   - Kaplan-Meir and Proportional Hazards

risks vs. rates

chicken-time

$1\frac{1}{2}$ chickens laying $1\frac{1}{2}$ eggs in $1\frac{1}{2}$ days
What is the daily egg-rate per chicken?

person-time

- 100 persons
- 40 die
- risk(proportion) $= 40/100 = 0.4$
- rate $= 40/80$ person-years $= 0.5$
    - $60 + \frac{1}{2}40 = 80$

how epidemiologists tell time

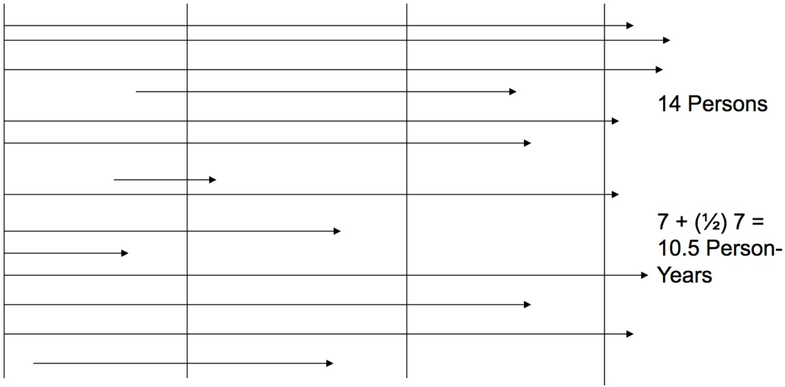
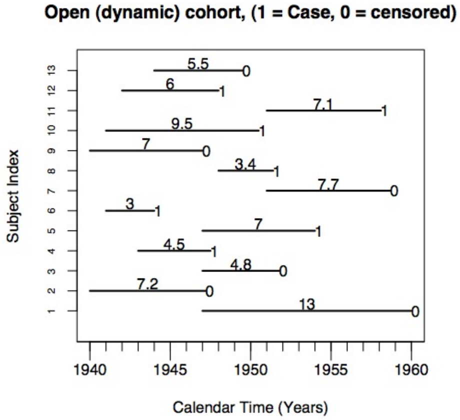


Figure: person time

how better epidemiologists tell time



Calculation of period average crude rate:

$$
\begin{aligned}
r &= \frac{\text{number of cases}}{\text{person-time at risk}} \\
&= \sum_i \frac{d}{PT_i} \\
&= \frac{\text{7 cases}}{\text{85.7 person-years}} \\
&= 0.08168028 \text{ py}^{-1} \\
&= 8.2 \text{ cases per 100 py}
\end{aligned}
$$

Figure: Source: Aragon (http://www.medepi.com)

calculating a rate from person time

$$r = \frac{\sum cases}{\sum p{-}t}$$

```
library(MASS)
data(Melanoma)
?Melanoma

mm.deaths<-sum(Melanoma$status==1)

per.time<-sum((Melanoma$time)/356)
mortality.rate<-mm.deaths/per.time
round(100*mortality.rate,1)
```

What is the *risk* of death?

mortality.risk = mm.deaths/nrow(Melanoma)
round(100*mortality.risk,1)

binomial vs exponential risk

binomial risk

- 57 malignant melanoma deaths among 205 people over 1239.67 person years 57/205 =0.278
- assumes each exposed person contributed equal amount of time

exponential risk $(1 - e^{-\lambda t})$

- risk of having become a case at the end of 5 years
- $\lambda = rate = \frac{57}{1239.67} = 0.04598$, and $t = 5$
- $risk_{5yrs} = 1 - e^{-0.04598 \cdot 5} = 0.21$
  - `1-exp(-0.04598*5)`

hazards

exponential model of risk

risk $= R(t) = 1 - e^{-\lambda t}$
where λ is the rate of an event and t is elapsed time.

- hazard - Pr[D] during a time increment $(t + \delta t)$
  - i.e. the probability of going from non-disease to disease from time(1) to time(2)
- a hazard is an individual *risk* or probability
  - at population level, hazards are essentially rates
- constant hazard = constant rate
  - if we can assume a constant hazard (and we often do) we can use exponential model

Outline

1. Statistical Functions

2. regression
   - linear
   - logistic
   - poisson

3. Epidemiology Packages
   - epitools and epicalc

4. Survival Tools for Epidemiologists
   - from risks to rates
   - Kaplan-Meir and Proportional Hazards

two survival analysis tools: exponential, Kaplan-Meier

- when it's not valid to assume equal observation periods for each person (Binomial model of risk)
- exponential
    - assume constant hazard over fixed time intervals
    - $R(T \leq t) = 1 - e^{\sum r_j h_j}$where $t = \sum h_j$ and r_j is the crude rate in the j^{th} fixed time interval
- product-limit (Kaplan-Meier)
    - accounts for "right censoring", i.e. patients drop out
    - *only interested in when an event (disease or death) occurs*
    - nonparametric
    - $S(T > t_i) = \prod \frac{n_i - d_i}{n_i}$
        - where n_i is the number at risk and d_i is the number diseased or dead at time i

How does Kaplan-Meier "Work"?

- data are divided into time intervals which vary by whether an event occurs or not
- calculate probability of survival for each time interval by dividing number survivors by number at risk, censored patients not at risk
- probability of surviving to some time is the cumulative product of the preceding probabilities
- Kaplan-Meier curve is declining series of horizontal steps that approaches the underlying survival function (if a large enough sample)

Survival Data from Breslow and Day

events occurred at 7 time periods

patient	time	status	event
1	13.00	0	
2	7.20	0	
3	4.80	0	
4	4.50	1	YES
5	7.00	1	YES
6	3.00	1	YES
7	7.70	0	
8	3.40	1	YES
9	7.00	0	
10	9.50	1	YES
11	7.10	1	YES
12	6.00	1	YES
13	5.50	0	

Kaplan-Meier (Product Limit) Approach

"condense" data to 7 time periods

i	t_i	d_i	s_i	$S(T \leq t_i)$	$R(T \leq t_i)$
1	3.0	1	13	12/13=0.92	1-.92=0.08
2	3.4	1	12	(11/12)*.92=0.85	1-0.85 =0.15
3	4.5	1	11	(10/11)*.85=0.77	1-0.77 =0.23
4	6.0	1	8	(7/8)*.77=0.67	1-0.67 =0.33
5	7.0	1	7	(6/7)*.67=0.58	1-0.58 =0.42
6	7.1	1	5	(4/5)*.57=0.46	1-0.46 =0.54
7	9.5	1	2	(1/2)*.46=0.23	1-0.23 =0.77

sorted by time to disease, d_i; survival is 1-risk, $S(T \leq t_i) = 1 - R(T \leq t_i)$

coding your own Kaplan-Meier (from Aragon)

1. prepare the population (denominator) data

enter and sort data by time

```
time <- c(13, 7.2, 4.8, 4.5, 7, 3, 7.7,
3.4, 7, 9.5, 7.1, 6, 5.5)
status <- c(0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0)
sorted.time <- sort(time)
sorted.status <- status[order(time)]
cbind(sorted.time, sorted.status)
```

define number of people in cohort at each time increment

```
nj <- length(time):1
nj <- nj[!duplicated(sorted.time)]
```

since one observation per person, initially set the number in the cohort to the length of the data

then account for period 7, when one person died and another was censored

coding your own Kaplan-Meier

2. prepare the outcome (numerator) data

sum deaths at each time increment

```
dj <- tapply(sorted.status, sorted.time, sum)
```

note that in these data there was a single death in each time interval, but this is not always the case

restrict the time data to unique levels

```
tj <- unique(sorted.time)
```

Note that this time variable is not strictly necessary for the calculations

coding your own Kaplan-Meier

3. calculate, collect, display

calculate survival (S) and risk (R)

```
Sj <- (nj - dj)/nj
cumSj <- cumprod(Sj)
cumRj <- 1 - cumSj
```

collect the results

```
results <- cbind(time = tj, n.risk = nj, n.events = dj,
condsurv = Sj, survival = cumSj, risk = cumRj)
dimnames(results)[1] <- list(NULL)
results
KM<-results[dj != 0, ] # just cases
```

display and plot the results

```
library(ggplot2)
qplot(KM[,1],KM[,5], geom="step")
```

Survival package

- *Surv()* create a survival object
- *survfit()* Kaplan Meier from a survival object
- *survdiff()* log rank test
- *coxph()* proportional hazards

```
library(survival)

library(MASS)
data(Melanoma)
names(Melanoma)

survival.object<-Surv(Melanoma$time, Melanoma$status==1)
survival.object  # + in output indicates censoring
```

Run and plot K-M

provide formula to *survfit()*, (1 means single group):

```
KM.object<-survfit(survival.object~1)
summary(KM.object)
plot(KM.object)
```

Compare two groups:

```
KM.object.ulcer<-survfit(survival.object~Melanoma$ulcer)
plot(KM.object.ulcer)
plot(KM.object.ulcer, conf.int=T, col=c("black", "red"))
```

Logrank Test

like chi square to compare two curves

$$(\Sigma(O_{ij} - Eij)^2 / var(\Sigma(O_{ij} - Eij)))$$

contingency table of event status by time points

for each group and every point in time:

- calculate observed minus expected
- square it
- divide by the variance

```
survdiff(survival.object~Melanoma$ulcer)
```

Proportional Hazards

Hazard, the opposite of survival

$$h_i(t) = h_0(t)e^{(\beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k)}$$

proportionality assumption

- non (actually semi) parametric
- assume comparing two survival curves that are parallel (proportional)
- only interested in the exponentiated beta coefficients
- don't need to know the baseline hazard, just the relative effects

- Linearity assumed on log-hazard scale
- *Allows regression-like modeling of survival times with covariates*

```
cox.object<-coxph(survival.object~Melanoma$ulcer
+ Melanoma$sex)
summary(cox.object) #hazard ratio exponentiated coeff
```

Credit where credit is due...

- Tomas Aragon, MD, DrPH
    - Applied Epidemiology Using R
    - http://www.medepi.net/
- John Fox, PhD
    - An Introduction to Statistical Computing in R
    - http://socserv.mcmaster.ca/jfox/Courses/UCLA/index.html
- Bill Venebles, PhD
    - An Introduction to R
    - cran.r-project.org/doc/manuals/R-intro.pdf
- Phil Spector, PhD
    - Data Manipulation with R