R Code Sample

Braden Mailloux

October 20, 2018

## Frivolties

How many Magic booster packs would I have to buy to collect the entire set? That is a coupon collectors problem with a pretty simple approximation. Where n is the number of items in the entire collection.

coupon <- function(n) {  
 gamma <- 0.5772156649  
 return(n\*log(n) + gamma\*n + 1/2)  
}  
  
ceiling(coupon(280)/10)

## [1] 174

Thats interesting. I would have to buy 172 ten card booster packs. Those are roughly $7 a piece. $1218?!

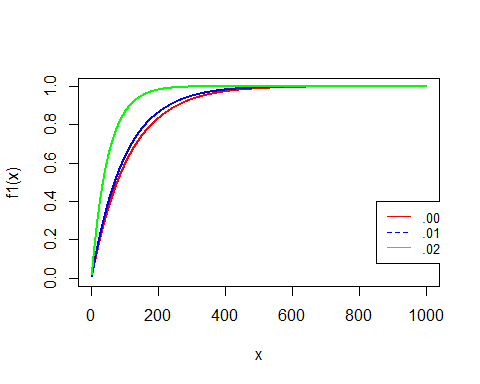
## Important stuff

How many trials do I need to repeat in order to find one false positive?

$$P(At \space Least\space 1\space Error) = 1 - (1 - e)^{m}$$

I’ll write a closure so it returns a function where we can tune the error-rate.

p.test <- function(e) {  
 function(m) {  
 1 - (1-e)^m  
 }  
}  
  
f1 <- p.test(.009)  
f2 <- p.test(.01)  
f3 <- p.test(.02)  
  
x <- seq(1,1000)  
  
plot(x,f1(x),col='red', type='l', lwd='2', ylim=c(0,1))  
lines(x, f2(x), col='blue', type='l', lwd='2')  
lines(x, f3(x), col='green', type='l', lwd='2')  
legend(850,0.4, legend=c('.009','.01','.02'),col=c('red','blue','green'), lty=1:2, cex=0.8)

 The lower the error rate the longer it takes. That makes sense.

which(f1(x) >= .99)[1]

## [1] 510

which(f2(x) >= .99)[1]

## [1] 459

which(f3(x) >= .99)[1]

## [1] 228

For the respective error rates, .009, .01, .02, we should expect (I’m rounding up here) 459, 228, and 152 patients before our first error occurs.

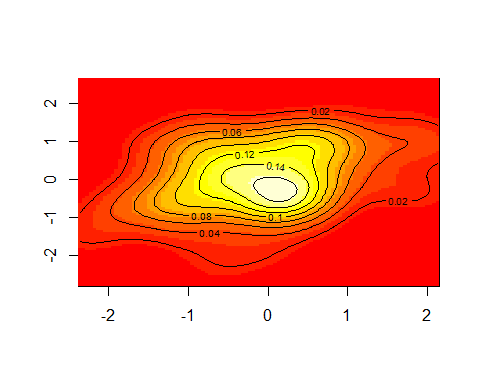
R does not have a multivariate normal in the standard library. That makes sense because its super simple to implement without too much fuss.

I’ll do it here.

mnorm <- function(mu, sigma)  
{  
 function(n) {  
 mu1 <- mu[1]  
   
 mu2 <- mu[2]  
   
 sigma1 <- Sigma[1,1]  
   
 sigma2 <- Sigma[2,2]  
   
 C <- Sigma[1,2]  
   
 rho <- C/(sigma1\*sigma2)  
   
 Z.1 <- rnorm(n)  
   
 Z.2 <- rnorm(n)  
   
 X1 <- sigma1\*Z.1 + mu1  
   
 X2 <- rho\*sigma2\*Z.1 + sigma2\*sqrt(1-rho^2)\*Z.2 + mu2  
   
 return(cbind(X1,X2))  
 }  
}

Again, this is a closure. I setup the function with a given mean and standard deviation/correlation matrix. Then we get a function back and tell it how many random variates we want. The result is returned as a two column matrix.

library(MASS)  
  
mu = c(0,0)  
  
Sigma = matrix(c(1,.5,.5,1),2)  
  
my.mvnorm <- mnorm(mu,Sigma)  
  
Z <- my.mvnorm(100)  
  
den <- kde2d(x=Z[,1],y=Z[,2], n=100)  
  
image(den)  
  
contour(den, add=TRUE)

 That looks pretty good.

## Survival Times

As per our discussion what follows is a demonstration of my coding and presentation skills using R. I decided to execute some simple analysis of survival data of breast cancer patients. The data used contained cases from a study conducted between 1958 and 1970 at the University of Chicago’s Billings Hospital on the survival of patients who had undergone surgery for breast cancer.

#Show header to explain data.  
library(survival)  
library(survminer)

## Loading required package: ggplot2

## Loading required package: ggpubr

## Loading required package: magrittr

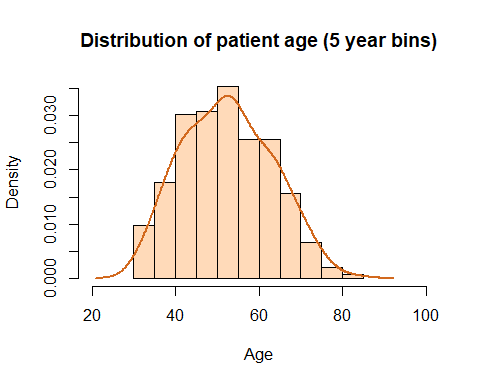
df <- read.csv('haberman.data', col.names = c('age', 'operation', 'nodes', 'status'))  
  
head(df)

## age operation nodes status  
## 1 30 62 3 1  
## 2 30 65 0 1  
## 3 31 59 2 1  
## 4 31 65 4 1  
## 5 33 58 10 1  
## 6 33 60 0 1

Columns are labeled the patient age, year of operation, the number of nodes (cancerous/benign) and whether or not the patient survived for up to 5 years following surgery. If the patient passed then the label changes to 2. The study contains 306 patients from various age groups ranging from 20 to 80.

Lets take a quick look at the distribution the ages in our population.

hist(df$age,  
 main = 'Distribution of patient age (5 year bins)',  
 prob = TRUE,  
 xlim=c(20,100),  
 xlab = 'Age',  
 col='peachpuff')  
  
lines(density(df$age),  
 col='chocolate',  
 lwd=2)



Lets fit a proportional hazard model and see what the hazard rate look like across a couple covariates.

my.coxph <- coxph(Surv(age, status) ~ operation + nodes, data=df)  
  
summary(my.coxph)

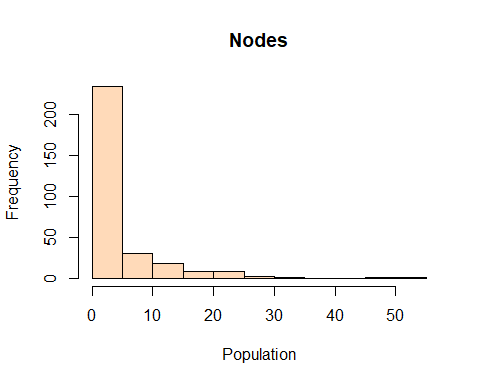
## Call:  
## coxph(formula = Surv(age, status) ~ operation + nodes, data = df)  
##   
## n= 305, number of events= 81   
##   
## coef exp(coef) se(coef) z Pr(>|z|)   
## operation -0.02801 0.97238 0.03474 -0.806 0.42   
## nodes 0.05517 1.05672 0.01009 5.469 4.52e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## exp(coef) exp(-coef) lower .95 upper .95  
## operation 0.9724 1.0284 0.9084 1.041  
## nodes 1.0567 0.9463 1.0360 1.078  
##   
## Concordance= 0.631 (se = 0.037 )  
## Rsquare= 0.069 (max possible= 0.917 )  
## Likelihood ratio test= 21.81 on 2 df, p=1.833e-05  
## Wald test = 30.08 on 2 df, p=2.932e-07  
## Score (logrank) test = 32.88 on 2 df, p=7.237e-08

exp(coef(my.coxph))

## operation nodes   
## 0.9723756 1.0567217

When they had the operation appears to matter not. However, nodes do appear to have an effect. That follows simply because it seemed as though some patients had quite a few nodes while the majority of them had none. Patients with a higher number of nodes increased there hazard rate by 1.05. This means for every one unit increase in the count of nodes causes a multiple of 1.05 of hazard rate, or simply a 5% increase. Operations, however, decrease the hazard rate by 0.97 times.

hist(df$nodes, col='peachpuff', xlab = 'Population', main='Nodes')

 The majority of the mass is placed around 0-5 nodes.

I’m curious what an accelerated failure time model looks like.

my.surv <- survreg(Surv(age, status) ~ operation + nodes, data=df, dist='loglog')  
  
summary(my.surv)

##   
## Call:  
## survreg(formula = Surv(age, status) ~ operation + nodes, data = df,   
## dist = "loglog")  
## Value Std. Error z p  
## (Intercept) 4.13491 0.38514 10.736 6.88e-27  
## operation 0.00195 0.00610 0.319 7.50e-01  
## nodes -0.01025 0.00226 -4.524 6.07e-06  
## Log(scale) -1.97283 0.08590 -22.966 1.02e-116  
##   
## Scale= 0.139   
##   
## Log logistic distribution  
## Loglik(model)= -394.8 Loglik(intercept only)= -405.2  
## Chisq= 20.71 on 2 degrees of freedom, p= 3.2e-05   
## Number of Newton-Raphson Iterations: 4   
## n= 305

exp(coef(my.surv))

## (Intercept) operation nodes   
## 62.4842684 1.0019473 0.9898068

Nodes cause an increase in failure rate and surgeries cause a decrease. This is consistent. Surgery increases survival time by 1.001 times. Whereas the presences of nodes shortens survival time by 0.98.

## References

Dua, D. and Karra Taniskidou, E. (2017). UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.

Data.gov. (2018). Accidents. Version 1. [Coal Miner Accidents]. Location: <https://catalog.data.gov/dataset/accident-injuries/resource/37ce36bb-b38c-480f-893f-ca33fa5589d9>. Department of Labor.