Data Simulation Project

Shraddha Lanka

Thursday, May 19, 2016

## DATA SIMULATION FOR RESEARCH ON CHEMOTHERAPY VERSUS SELUMETINIB TREATMENTS

### VARIABLES

There are different variables that were studied during the course of this trial. Each of those has been simulated here.

1. Age: A normally distributed vector. However, the range of age group is only between 60 to 80 for both groups.

ageC<-rnorm(n = 51,mean = 62,sd=10)  
ageS<-rnorm(n=50,mean=62,sd=10)  
chemo<-as.data.frame(ageC)  
selum<-as.data.frame(ageS)

1. Sex: Pretty much an even distribution between Males and Females.

M<-rep("M",times=31)  
Fe<-rep("F",times=20)  
SexC<-c(M,Fe)  
M<-rep("M",times=26)  
Fe<-rep("F",times=24)  
SexS<-c(M,Fe)  
sample(SexC)

## [1] "F" "M" "F" "M" "M" "F" "F" "F" "M" "F" "F" "M" "F" "M" "F" "F" "M"  
## [18] "M" "M" "M" "F" "F" "M" "M" "M" "M" "M" "M" "F" "M" "M" "F" "F" "F"  
## [35] "M" "M" "M" "F" "F" "F" "M" "M" "M" "M" "M" "M" "F" "M" "M" "M" "M"

sample(SexS)

## [1] "M" "M" "F" "F" "F" "M" "F" "M" "M" "F" "M" "M" "M" "F" "F" "F" "M"  
## [18] "M" "M" "F" "F" "F" "M" "F" "M" "F" "M" "M" "F" "M" "F" "F" "M" "M"  
## [35] "M" "F" "F" "M" "M" "F" "M" "M" "M" "F" "M" "F" "M" "F" "F" "F"

chemo$sex<-SexC  
selum$sex<-SexS  
  
chemo$sex<-as.factor(chemo$sex)  
selum$sex<-as.factor(selum$sex)

3.Performance Status, a binary variable.

PerfStatC<-sample(c(0,1),51,replace=TRUE)  
PerfStatS<-sample(c(0,1),50,replace=TRUE)  
  
chemo$PerfStat<-PerfStatC  
selum$PerfStat<-PerfStatS

4.Cancer Stage No. % (2 types of stages)

M1ab<-rep("M1a/b",times=3)  
M1c<-rep("M1c",times=48)  
CancerStagePercentC<-c(M1ab,M1c)  
sample(CancerStagePercentC)

## [1] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [9] "M1a/b" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [17] "M1c" "M1c" "M1c" "M1c" "M1c" "M1a/b" "M1c" "M1c"   
## [25] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [33] "M1a/b" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [41] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [49] "M1c" "M1c" "M1c"

chemo$CancStage<-CancerStagePercentC  
  
M1ab<-rep("M1a/b",times=2)  
M1c<-rep("M1c",times=48)  
CancerStagePercentS<-c(M1ab,M1c)  
sample(CancerStagePercentS)

## [1] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [9] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [17] "M1c" "M1c" "M1c" "M1c" "M1a/b" "M1c" "M1c" "M1c"   
## [25] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [33] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1a/b" "M1c"   
## [41] "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c" "M1c"   
## [49] "M1c" "M1c"

selum$CancStage<-CancerStagePercentS

5.The number of Systemic therapies each patient has undergone.

SystemicTherapiesC<-sample(c(0,1,2),51,replace=TRUE)  
SystemicTherapiesS<-sample(c(0,1,2),50,replace=TRUE)  
  
chemo$Systemic<-SystemicTherapiesC  
selum$Systemic<-SystemicTherapiesS

1. The number of Liver Therapies each patient has undergone.

LiverTherapiesC<-sample(c(0,1,2),51,replace=TRUE)  
LiverTherapiesS<-sample(c(0,1,2),50,replace=TRUE)  
  
chemo$Liver<-LiverTherapiesC  
selum$Liver<-LiverTherapiesS

1. Tumor Mutation %

GNAQC<-rbinom(51,1,0.4)  
GNA11C<-rbinom(51,1,0.5)  
WildC<-rbinom(51,1,.14)  
  
chemo$GNAQ<-GNAQC  
chemo$GNA11C<-GNA11C  
chemo$WildC<-WildC  
  
GNAQS<-rbinom(50,1,0.4)  
GNA11S<-rbinom(50,1,0.4)  
WildS<-rbinom(50,1,0.18)  
   
selum$GNAQ<-GNAQS  
selum$GNA11C<-GNA11S  
selum$WildC<-WildS

1. Hematologic disorders binary variables, bernoulli distribution.

chemo$Anemia<- rbinom(51,1,0.16)  
chemo$Leukopenia<- rbinom(51,1,0.18)  
chemo$Lymphopenia<-rbinom(51,1,0.08)  
chemo$Neutropenia<-rbinom(51,1,0.08)  
chemo$Thrombocytopenia<-rbinom(51,8,0.16)  
   
  
selum$Anemia<-rbinom(50,1,0.16)  
selum$Leukopenia<-rbinom(50,1,0.18)  
selum$Lymphopenia<-rbinom(50,1,0.08)  
selum$Neutropenia<-rbinom(50,1,0.08)  
selum$Thrombocytopenia<- rbinom(50,1,0.16)

1. Fatigue and Nausea, binary variables, bernoulli distributions.

library(plyr)

## Warning: package 'plyr' was built under R version 3.1.3

VomitingC<-rep(0,51)  
#Insert a 1 in all the index values where Nausea is a 1, with a probability of getting a 1 as 0.6  
VomitingC[chemo$Nausea==1]<-rbinom(length(VomitingC[chemo$Nausea==1]),1,0.6)  
  
chemo$Vomiting<-VomitingC  
  
VomitingS<-rep(0,50)  
VomitingS[selum$Nausea==1]<-rbinom(length(VomitingS[selum$Nausea==1]),1,0.6)  
  
selum$Vomiting<-VomitingS

1. Constipation.

chemo$Constipation<-rbinom(51,1,0.3)  
selum$Constipation<-rbinom(50,1,0.06)

1. Anorexia is related to age. Simulating a binary variable, from a continuous variable.

#Chemo  
n <- 51  
beta0 <- -1.6  
beta1 <- 0.03  
  
  
pi\_x <- exp(beta0 + beta1 \* ageC) / (1 + exp(beta0 + beta1 \* ageC))  
chemo$Anorexia <- rbinom(n=length(ageC), size=1, prob=pi\_x)  
  
#selemutinib  
  
n <- 50  
beta0 <- -1.6  
beta1 <- 0.03  
  
  
pi\_x <- exp(beta0 + beta1 \* ageS) / (1 + exp(beta0 + beta1 \* ageS))  
selum$Anorexia <- rbinom(n=length(ageS), size=1, prob=pi\_x)

1. Survival Rates

chemo$survival<-rnorm(n=51,mean=7,sd=2)  
selum$survival<-rnorm(n=50,mean=15.9,sd=3)

Using SVM to predict the survival rates.

#install.packages("e1071")  
library(e1071)

## Warning: package 'e1071' was built under R version 3.1.3

chemoB<-chemo  
chemo<-chemo[-2]  
chemo<-chemo[-3]  
x<-as.matrix(chemo[-18])  
y<-as.matrix(chemo$survival)  
  
n <- nrow(y)   
# x is a vector of numeric values   
# y is a vector of labels containing the classification of the corresponding xvalue   
ntrain <- round(n\*0.8) # number of training examples   
tindex <- sample(ntrain, size=n, replace=TRUE) # indices of training samples '  
xtrain <- x[tindex,]   
xtest <- x[-tindex,]   
ytrain <- y[tindex,]   
ytest <- y[-tindex,]   
istrain=rep(0,n)   
istrain[tindex]=1   
# load the kernlab package  
  
  
  
library(kernlab)

## Warning: package 'kernlab' was built under R version 3.1.3

svp <- svm(x = xtrain,y=ytrain,kernel = "linear",gamma=1,cost=2)

## Warning in svm.default(x = xtrain, y = ytrain, kernel = "linear", gamma =  
## 1, : Variable(s) 'Vomiting' constant. Cannot scale data.

summary(svp)

##   
## Call:  
## svm.default(x = xtrain, y = ytrain, kernel = "linear", gamma = 1,   
## cost = 2)  
##   
##   
## Parameters:  
## SVM-Type: eps-regression   
## SVM-Kernel: linear   
## cost: 2   
## gamma: 1   
## epsilon: 0.1   
##   
##   
## Number of Support Vectors: 13

ypred1 = predict(svp,xtest)   
  
  
  
  
SelumB<-selum  
selum<-selum[-2]  
selum<-selum[-3]  
x<-as.matrix(selum[-18])  
y<-as.matrix(selum$survival)  
  
n <- nrow(y)   
# x is a vector of numeric values   
# y is a vector of labels containing the classification of the corresponding xvalue   
ntrain <- round(n\*0.8) # number of training examples   
tindex <- sample(ntrain, size=n, replace=TRUE) # indices of training samples '  
xtrain <- x[tindex,]   
xtest <- x[-tindex,]   
ytrain <- y[tindex,]   
ytest <- y[-tindex,]   
istrain=rep(0,n)   
istrain[tindex]=1   
  
svp <- svm(x = xtrain,y=ytrain,kernel = "linear",gamma=1,cost=2)

## Warning in svm.default(x = xtrain, y = ytrain, kernel = "linear", gamma =  
## 1, : Variable(s) 'Vomiting' constant. Cannot scale data.

summary(svp)

##   
## Call:  
## svm.default(x = xtrain, y = ytrain, kernel = "linear", gamma = 1,   
## cost = 2)  
##   
##   
## Parameters:  
## SVM-Type: eps-regression   
## SVM-Kernel: linear   
## cost: 2   
## gamma: 1   
## epsilon: 0.1   
##   
##   
## Number of Support Vectors: 14

ypred2 = predict(svp,xtest)

### DATA EXPLORATION AND PLOTS

library(corrplot)

## Warning: package 'corrplot' was built under R version 3.1.3

C<-cor(chemo)

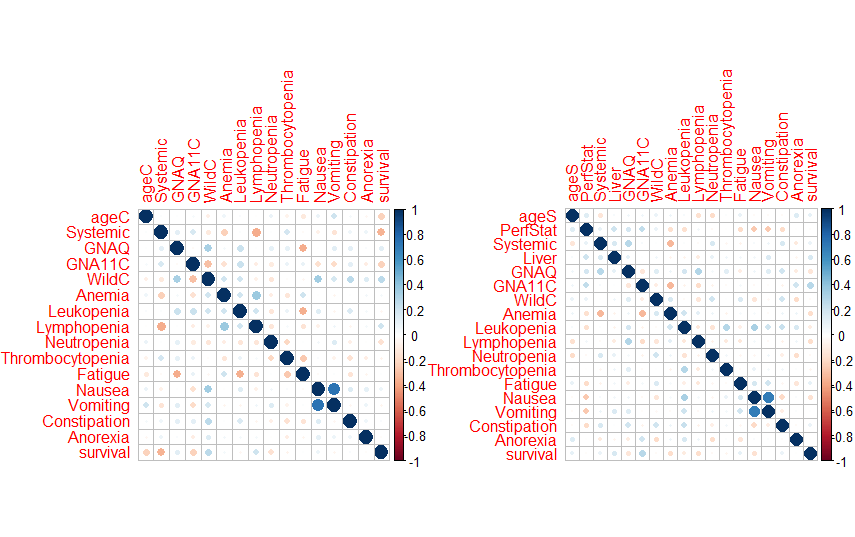
## Warning in cor(chemo): the standard deviation is zero

corrplot(C)

S<-cor(selum)

## Warning in cor(selum): the standard deviation is zero

corrplot(S)



#library(Hmisc)  
#hist.data.frame(chemoB)  
#hist.data.frame(SelumB)

Building a function to display multiple plots at once.

# Multiple plot function  
#  
# ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)  
# - cols: Number of columns in layout  
# - layout: A matrix specifying the layout. If present, 'cols' is ignored.  
#  
# If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),  
# then plot 1 will go in the upper left, 2 will go in the upper right, and  
# 3 will go all the way across the bottom.  
#  
multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {  
 library(grid)  
   
 # Make a list from the ... arguments and plotlist  
 plots <- c(list(...), plotlist)  
   
 numPlots = length(plots)  
   
 # If layout is NULL, then use 'cols' to determine layout  
 if (is.null(layout)) {  
 # Make the panel  
 # ncol: Number of columns of plots  
 # nrow: Number of rows needed, calculated from # of cols  
 layout <- matrix(seq(1, cols \* ceiling(numPlots/cols)),  
 ncol = cols, nrow = ceiling(numPlots/cols))  
 }  
   
 if (numPlots==1) {  
 print(plots[[1]])  
   
 } else {  
 # Set up the page  
 grid.newpage()  
 pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))  
   
 # Make each plot, in the correct location  
 for (i in 1:numPlots) {  
 # Get the i,j matrix positions of the regions that contain this subplot  
 matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))  
   
 print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,  
 layout.pos.col = matchidx$col))  
 }  
 }  
}

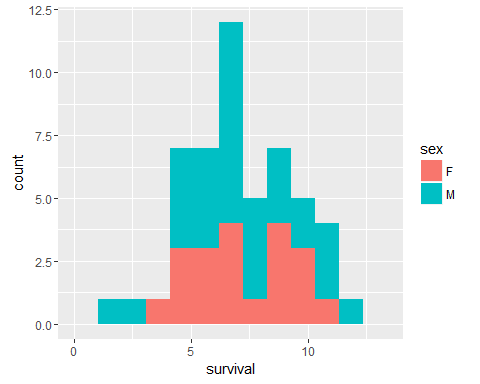
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.1.3

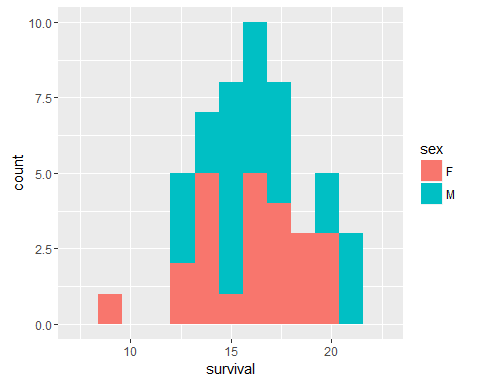
##   
## Attaching package: 'ggplot2'

## The following object is masked from 'package:kernlab':  
##   
## alpha

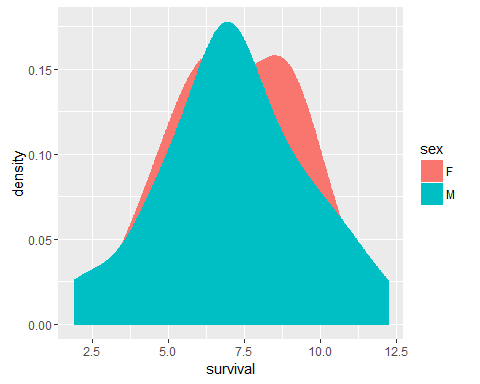
g=ggplot(chemoB)  
g+geom\_histogram(aes(x=survival,fill=sex),bins = 10)



g1=ggplot(SelumB)  
g1+geom\_histogram(aes(x=survival,fill=sex),bins=10)



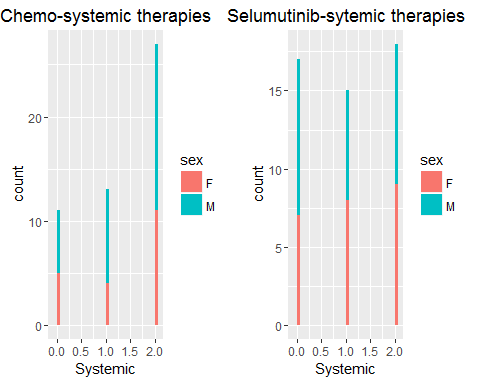
ggplot(chemoB,aes(x=survival))+geom\_density(aes(color=sex,fill=sex))



p1<-ggplot(chemoB)+geom\_histogram(aes(x=Systemic,fill=sex))+ggtitle("Chemo-systemic therapies")  
  
p2<-ggplot(SelumB)+geom\_histogram(aes(x=Systemic,fill=sex))+ggtitle("Selumutinib-sytemic therapies")  
  
multiplot(p1,p2,cols=2)

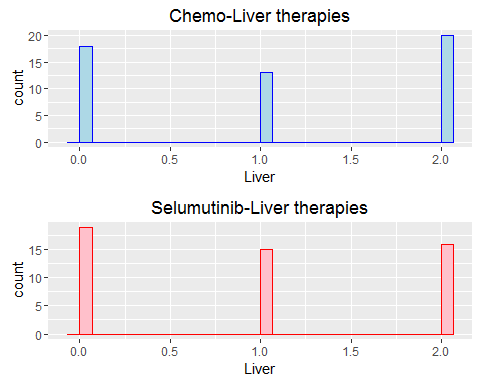
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



p1<-ggplot(chemoB)+geom\_histogram(aes(x=Liver),color="blue",fill="light blue")+ggtitle("Chemo-Liver therapies")  
  
p2<-ggplot(SelumB)+geom\_histogram(aes(x=Liver),color="red",fill="pink")+ggtitle("Selumutinib-Liver therapies")  
  
multiplot(p1,p2)

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#p1<-ggplot(chemoB)+geom\_point(aes(x=ageC,y=Fatigue),color="blue",data = chemoB)+ggtitle("Age versus Fatigue Chemotherapy")  
#p2<-ggplot(SelumB)+geom\_point(aes(x=ageS,y=Fatigue),color="purple",data = SelumB)+ggtitle("Age versus Fatigue Selumutinib")  
  
#multiplot(p1,p2)  
  
p1<-ggplot(chemoB)+geom\_line(aes(x=ageC,y=survival),color="blue")+ggtitle("Age versus survival Chemotherapy")  
p2<-ggplot(SelumB)+geom\_line(aes(x=ageS,y=survival),color="purple")+ggtitle("Age versus survival Selumutinib")  
  
multiplot(p1,p2)

