Module Four

Chol Stephen Jok

2022-09-14

#Examining Outliers #Categorical data #Outlier is <10%

#Quantitative data #See outlier in Boxplot

```
#require("dataset")
#?rivers
#data(rivers)  #Lengths of Major North American Rivers

#hist(rivers)
#boxplot(rivers, horizontal = TRUE)
#boxplot.stats(rivers)
#rivers.low<-rivers[rivers<1210]    #REMOVE OUTLIERS
#boxplot(rivers.low, horizontal = TRUE)   #HAS NEW OUTLIERS
#boxplot(rivers.low)
#rivers.low2<-rivers[rivers<1055] #Remove again
#boxplot(rivers.low2)   #Still one outlier
```

#TRANSFORMING VARIABLES #Load data

```
#require("datasets")
#?islands
```

#The areas in thousands of square miles of the landmasses which exceed 10,000 square miles.

```
#hist(islands, breaks = 16)
#boxplot(islands)
```

#z-scores

```
#islands.z<-scale(islands)  #M=0, sd=1
#islands.z   #make matrix with attribute information
#hist(islands.z, breaks = 16) #histogram of z-scores
#boxplot(islands.z)  #boxplot of z-scores
#mean(islands.z)  # mean close to 0
#round(mean(islands.z), 2)   # round off to see 0
#sd(islands.z) # sd=1
#attr(islands.z, "scaled:center")    #show original mean
#attr(islands.z, "scaled:scale")   # original sd
#islands.z<-as.numeric(islands.z)   #convert from matrix back to numeric
#islands.z
```

#Logarithmic transformaation

```
#islands.in<-log(islands)   #natural log (base=e)
```

#island.log10<-log10(islands) #common log (base=10) #island.log12<-log2(islands) #binary log (base=2)

```
#hist(islands.in)
#boxplot(islands.in)
 # Note: Add log1 to avoid undefined log when x=0
```

#x.in<-log(x+1) #squaring #For negatively skewed variables #distribution may need to be recentreds so that all values are positive

#ranking

```
#islands.rank1<-rank(islands)
#hist(islands.rank1)
#boxplot(islands.rank1)
```

#Ties.method= c(“avarage”, “first”, “random”, “max”,“min”)

```
#islands.rank2<-rank(islands, ties.method="random")
#hist(islands.rank2)
#boxplot(islands.rank
```

#dechotomizing #Use wisely and purposely #Split at 100 (=1,000,000 square miles) #ifelse in conditional element selection

```
#continent<-ifelse(rivers>1000,1,0)
#continent
```

#COMPUTING COMPOSITE VARIABLES #Creating variable rn1 with 1 million random normal values #Will vary from one to another

```
#rn1<-rnorm(1000000)
#hist(rn1)
#summary(rn1)
```

#Creating variable rn1 with 1 million random normal values

```
#rn2<-rnorm(1000000)
#hist(rn2)
#summary(rn2)
```

#Average scores across two variables

```
#rn.mean<-(rn1+rn2)/2
#hist(rn.mean)
```

#Mutiply scores across two variables

```
#rn.product<-rn1*rn2
#hist(rn.product)
#summary(rn.product)
```

#Kurtosis comparison #The package “moment” gives kurtosis where metokurtosis, normal distribution has a value of 3 #The package “psych” recenters the kurtosis value around 0 which is more common now

```
#require("psych")
#kurtosi(rn1)
#kurtosi(rn2)
#kurtosi(rn.mean)

#kurtosi(rn.product)   # Similar to cauchy distribution
```

#CODING MISSING DATA #NA=“Not available” #Make certain calculation impossible

```
#x1<- c(1,2,3,NA,5)
#fix('x1')
#Summary(x1)  # work with NA
#mean(x1)
```

#To find missing number

```
#which(is.na(x1))  #give index number
```

#Ignoring missing values with na.rm=T

```
#mean(x1,na.rm=T)   # T for TRUE
```

#Replacing missing value with 0 (or any other value #Option 1 “is.na”

```
#x2<-x1
#x2[is.na(x2)]<-0
#x2
```

#Option 2, Using “ifelse”

```
#x3<-ifelse(is.na(x1),0,x1)
#x3
```

#for dataframe, r has many packages to deal intelligently with missing data via imputation #These are just three #mi: missing data imputation and model checking #mice: multivariate imputation for chained equations #imputation