Stat 6390
Environmental Statistics

Syllabus

Stat 6390 Environmental Statistics Course Information

Instructor: Dr. Larry P. Ammann
Office hours: Wed, 2:00-3:30 pm, others by appt.
Email: ammann@utdallas.edu
Office: FO 2.410C
Phone: (972) 883-2164
Text: Course notes and web resources

Topics

• One-sample estimation and hypothesis tests
• Two-sample estimation and hypothesis tests
• Regression and ANOVA
• Generalized linear models

Important components of these topics include power, sample size determination, and observable differences for methods based on normality assumptions as well as alternatives to those assumptions. The statistics programming language R will be used heavily in this course. An extension to R called Rstudio also is recommended.

Grading Policy
Course grade will be based on quizzes, homework projects and the final project.
Quizzes: 25%  Homework projects: 50%
Final project: 25%

Note: the complete syllabus is available here:
This course uses the statistical programming language and environment \texttt{R}. This is freely available software with binaries for Linux, MacIntosh, Windows that can be obtained from: \url{http://cran.r-project.org}

A very useful extension for \texttt{R} is another freely available open-source package, \texttt{RStudio}. There are versions available for Windows, Mac, and Linux which can be downloaded from: \url{https://www.rstudio.com/products/rstudio/download/}

This package provides an intelligent editor for script files, it allows specific projects to be assigned to their own directories to aid in organization of your work, \texttt{RStudio} includes an interactive debugger to help identify and correct errors, and it provides an interactive GUI for easily exporting graphics to image files for incorporation into documents.

\section*{Class Notes}

\section*{Background}

Environment regulation had its roots in public health. The U.S. Public Health Service was formed in 1912 from several other organizations. Its initial focus was on water-borne diseases such as typhoid. Later the PHS developed standards for air quality in the industrial workplace in addition to those for water quality. However, it was the tremendous industrial growth after WWII, and all the highly visible pollution associated with that growth, that caused some to call for federal regulatory action. The large scale of factories in the chemical, plastics, petroleum, and automotive industries meant that problems associated with their waste products no longer were confined to a single town, county, or even state.

I was born in grew up in South Florida after WWII and witnessed first-hand some significant environmental problems. Once home air conditioning became widely available, people began migrating from northern states to Florida. As a result its coastal cities grew rapidly. The attitude among many of those involved with building public infrastructure to support this growth was that the ocean was so vast it never could be polluted, and so cities built large pipes into the ocean to discharge raw sewage. But then under certain weather conditions beaches had to be closed due to bacterial contamination from those sewage outfalls.

Another major problem came from the expansion of farms below Lake Okeechobee. Water from much of the interior of central and south Florida drains into the lake causing it to expand and contract depending on seasonal rainfall. Lake Okeechobee does not discharge into a well-defined river but instead flows slowly through a 50 mile wide area called the Everglades. As a result the soil south of the lake is very rich and crops could be grown without fertilizer. The first large farms there grew sugar cane. Later large vegetable farms were added that supply most of the winter vegetables for the cities of the northeastern and midwestern U.S. But there were two problems that needed to be addressed for these farms to be successful. The periodic flooding of the lake needed to be controlled and so did the
insects that flourished in this semi-tropical environment.

To solve the first problem levees were built around the lake. This had a significant impact on the Everglades by seriously restricting the amount of water that flowed through the Everglades into Florida Bay. In the 1950s and early 1960s the glades were viewed as a worthless swamp fit only for alligators and a few crazy people. It was felt that this water was better used for farming and as drinking water for the increasingly large numbers of new residents of south Florida. But Florida is mostly limestone. As the Everglades dried and more subsurface water along the coast was pumped to water all those new lawns in ever-expanding coastal cities, these areas began to experience saltwater intrusion which made those wells unfit for use. At the same time we began to understand the critical role of Florida Bay as a nursery for fish and shellfish that inhabit the eastern Gulf of Mexico and the Atlantic Ocean south and east of Florida. This rich nursery was threatened by the reduced flow of fresh water into the bay that supported the dense mangrove forests along the coast.

The insect problem was exacerbated by mosquitoes that feasted on all those new residents. Not only are mosquitoes a nuisance, but they also carry diseases some of which are deadly. To combat the insect problem, DDT was applied heavily in farming areas. Another pesticide, malathion, was routinely sprayed in residential areas to temporarily reduce adult mosquito populations, although that relief only lasted about two days at most. Pesticides also were applied to wetlands to kill mosquito larvae in the water before they could morph into adults. Although DDT helped eradicate malaria in some areas of the world, not much was known about its negative impacts until Rachael Carson published her ground-breaking book *Silent Spring* in 1962. In this book she argued that DDT and other pesticides were harmful to wildlife and humans in ways we did not understand. One very influential reader of her book was President John F. Kennedy who ordered his scientific advisory committee to investigate those claims. This was an important milestone. JFK asked for science-based decisions rather than simply listening to other politicians and the opinions of lobbyists who had significant financial interests in the outcome.

At the same time Floridians began to notice a disturbing trend - birds that depended on fish were disappearing. These included Osprey, Bald Eagles, herons, egrets, and Florida’s iconic pelicans. As studies motivated by *Silent Spring* were undertaken, we learned that long-lasting compounds in DDT were being absorbed by aquatic insects which then were consumed by fish. Those compounds remained intact in the fish and accumulated, a process referred to as bioaccumulation. Birds that consumed fish were then exposed to those compounds the effect of which was thinning of their eggshells. As a result the eggs were not strong enough to support the weight of an adult bird and the eggs ended up getting crushed in the nest.

Other highly visible environmental disasters such as the Cayahoga River in Cleveland catching fire brought greater awareness that damage to the environment by human activity has dangerous consequences. This awareness created strong support for the formation of the Environmental Protection Agency to oversee development, application, and enforcement of federal standards for air and water quality. The broad public support for creating such an
agency resulted in President Nixon signing the bill creating the EPA on December 2, 1970. This did not cause the end of environmental disasters in this country and so it became clear to many that the EPA needed to be proactive, not just reactive, to sources of pollution. This led to what are referred to as the Clean Water Act and the Clean Air Act. The first CWA was enacted in 1972 (over Nixon’s veto). This landmark legislation prohibits the release of toxic substances in toxic amounts into the waters of the U.S. The CAA first passed in 1970 and is designed to regulate emissions of hazardous air pollutants with a similar goal of prohibiting the release into the atmosphere of toxic substances in toxic amounts.

Examples

- **Love Canal, NY.** In the late 1950s about 100 homes and a school were built on property that once was a chemical dumpsite. Then in 1978 it was discovered that carcinogenic compounds from this dumpsite had been leaching into those homes and school producing birth defects, miscarriages, and leukemia among the residents.

- **Hudson River PCB contamination.** PCB is a chemical that was used in electrical transformers until it was banned in 1976 as a highly toxic carcinogen. Tragically over 1 million pounds of PCB were discharged into the Hudson River by G.E. and absorbed by river sediments. Even today an advisory to not eat fish from the Hudson remains in effect due to this PCB contamination. This part of the Hudson River is an EPA Superfund site and delicate dredging is being done to remove PCB from the river sediment. The challenge is to remove contaminated sediment without stirring up the sediment and causing PCB to flow downstream. This will take a long time. Also, hundreds of thousands of pounds of PCB remains in the shale beneath G.E.’s plant next to the river and it continues to leach into the Hudson.

- Environmental disasters have occurred in other countries as well. A Union Carbide plant in **Bhopal, India** released a highly toxic gas, methyl isocyanate, as well as other toxic gases that immediately killed over 2,000 people who lived around the plant. Many others, especially children, experienced long-term health effects.

Incredibly, similar events continue today.

- **West, Texas.** On April 17, 2013 a massive explosion of improperly stored fertilizer, ammonium nitrate, killed 15 people, injured more than 160, and destroyed or damaged over 150 buildings.

- **Beijing, China.** Rapid industrialization and use of coal-fired power plants and heating stoves have created dangerous levels of smog in Beijing and other cities such as Shanghai. At its worst, levels of smog have been recorded that are over seven times greater than safe levels. This smog contains particulates that penetrate deep into lung tissues when inhaled causing permanent damage. These events cause schools and factories to close and residents are advised to stay indoors. China is working now to replace coal with oil and gas for heating and power.
• **Flint, Michigan.** A decision to reduce costs associated with drinking water by using water from nearby Flint River instead of water from Lake Huron. However, higher salinity of the Flint River caused lead in pipes carrying water into homes to be leached into the water. Eventually it was discovered that blood lead levels in children were dangerously high as a result. It has been known for a long time that plumbing in older dwellings contained lead and so were susceptible to leaching. Proactive testing for potential contamination before the switch to the Flint River would have detected this in which case this tragedy would have been avoided.

**Ban of leaded gasoline.** One very important example of the hurdles faced by environmental regulation is the story of how leaded gasoline was banned in the U.S. In the 1920s tetraethyl lead (TEL) was added to gasoline to improve engine performance. Although lead was known to be toxic to human health, instruments sensitive enough to measure atmospheric lead from automobile exhausts did not exist at that time. Scientists employed by the company that produced TEL testified to Congress that low levels of lead had no effect on human health. In spite of the obvious conflict of interest, Congress accepted the opinion of that company’s chief scientist, Robert Kehoe, instead of demanding scientific proof of his assertions like JFK did regarding DDT. So TEL continued to be used as an additive and the producers of TEL continued to profit from it.

In the late 1940s a geology graduate student named Clair Patterson was given a dissertation topic to measure the age of the Earth using the decay of uranium into lead in meteorites. This work required measuring extremely small levels of lead using the most advanced mass spectrometers available at that time. However, a major problem Patterson had to overcome was contamination of his samples by lead in the environment. Patterson eventually realized that he needed to perform his work in a room that was completely isolated from the outside that had been scrubbed thoroughly and he needed to be covered in protective clothing. This was the first **clean room.** This was successful and Patterson determined that the Earth’s age is $4.55 \pm 0.07$ billion years (current estimate is $4.54 \pm 0.05$ billion years).

After he completed this work, Dr. Patterson wanted to determine the sources of contamination he had found in his initial efforts to measure lead in meteorites. He examined ocean water and found significantly higher lead levels near the surface compared to the ocean floor. He then examined ice cores from Greenland and found that levels of atmospheric lead began to increase significantly at the same time that TEL began to be added to gasoline. To Patterson this was the smoking gun that proved leaded gasoline was a major source of atmospheric lead. Unfortunately, lobbyists and scientists with financial connections to TEL engaged in a campaign to discredit Dr. Patterson and his research. As catalytic converters were introduced as a way to remove smog-producing compounds from engine exhaust, it was found that leaded gasoline corroded them and so these two effects finally led to a ban on leaded gasoline was approved. In the meantime, many more children were exposed to this toxic substance. We know now that there is no safe level of lead and that it causes irreversible damage to the development of children’s brains resulting in behavioral problems and reduced IQs. The story of Dr. Patterson and the ban of leaded gasoline is told in an episode of **Cosmos - A Spacetime Odyssey** that appeared on PBS in 2014.
Sampling

The first step in any research problem is to define the population – the set of all individuals about which you have questions to answer. The next step is to express the main question in terms of observable characteristics of individuals in the population. The types of questions we will discuss in this course involve estimation, model-building, and decision-making (hypothesis testing) when the entire population cannot be measured.

Estimation refers to determining a particular attribute of the population. Model-building refers to determining how multiple attributes are related. For decision-making we must formulate the question into a set of actions that will be taken based on the value of some measure derived from the observable characteristics.

A sample is any subset of the population. If the entire population cannot be measured, then we are forced to answer our questions using a sample. This produces error or uncertainty in our estimates, models, and decisions. These only have meaning if we can quantify this error and that can be done only if the samples are obtained randomly. Suppose for example we want to compare species population densities among different zones of a river where zones are defined in terms of depth, flow, and vegetable type on the banks of the river. Suppose there are 3 zones and a survey of the river gives the following.

<table>
<thead>
<tr>
<th>Distance from start</th>
<th>Zone</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 3</td>
<td>1</td>
</tr>
<tr>
<td>3-5.5</td>
<td>2</td>
</tr>
<tr>
<td>5.5-7</td>
<td>1</td>
</tr>
<tr>
<td>7-10</td>
<td>3</td>
</tr>
<tr>
<td>10-14</td>
<td>2</td>
</tr>
<tr>
<td>14-18</td>
<td>3</td>
</tr>
</tbody>
</table>

Then the total length of zone 1 is 5.5 and so we would generate \( n_1 \) i.i.d. r.v.’s from \( \text{Unif}(0,5.5) \). These r.v.’s would give the locations of our sampling sites within zone 1.

In all cases we must determine the appropriate sample sizes that will result in acceptable levels of error. For estimation we must specify the precision of the estimate along with the confidence expressed as a probability that the actual error is within the stated precision. For hypothesis testing we must specify the level of significance of the test (probability of making Type I error) and the power (probability of rejecting null hypothesis) at a specified distance from the null hypothesis. To accomplish this we must make assumptions about the underlying distribution of the data, or we must apply distribution-free methods. Any assumptions we make must be verified.

Data frames

R had a special data type called a data frame. This structure was introduced to unify modeling functions in R. A matrix in R has the restriction that all elements must be the same type. However, data sets that include a mix of numeric and categorical variables and
so cannot be represented by a matrix in R. A data frame is a structure designed for such cases. It is like a matrix in that a data frame can be considered as a two-dimensional object with columns representing variables and rows representing sampling units. Each column of a data frame must contain the same type of data but different columns may be different types. The column names can be thought of as the names of the variables in the data set. A data frame also is like a list in which the columns are its components. The column names correspond to the names of those components and can be referenced by those names.

The most common way in which a set of tabular data is entered into R is by using one of its functions specifically designed for tabular data. These functions include `read.table()` and `read.csv()`. `read.table()` is the general function and `read.csv()` is designed for comma-separated-values data. Since data sets often are extracted from spreadsheets or database systems, the easiest way to enter such data into R is to save the data as file type CSV from the spreadsheet or database and then use `read.csv()` to bring it into R. These functions return the data as a data frame. Typically the first row of the file contains names for the variables and the remaining rows contain the data. In such cases, the optional argument `header` in `read.table()` should be set to `TRUE`. That is the default in `read.csv()` so that argument does not need to be set in that function.

If a column contains strings instead of numbers then these functions automatically convert those columns to factors. If a categorical variable is coded numerically, then it would need to be explicitly converted to a factor using the `factor()` function after reading the data. If one of the columns is not data but contains names of the rows, then that column should be specified in the `row.names` argument. These functions check to see if each line of the file contains the same number of fields and returns an error if not. The function `count.fields()` can help locate the offending lines. This can happen when extracting data from Excel which often puts in extraneous commas and extra lines into the CSV file it creates. Another problem that may occur is when a column of string data includes some values with apostrophes. By default apostrophes are assumed to be string delimiters in addition to the double quote symbol, so the `quote` argument must be used.

The file

http://www.utdallas.edu/~ammann/stat6390scripts/cars.csv

is a data set that gives several measurements for a set of cars. It is a CSV file with column names in the first line and the first row contains vehicle names. Note that rows for diesel cars begin with the # symbol. That is the comment symbol and indicates those rows should be ignored. That is the default for the `comment=` argument in `read.table()` but not in `read.csv()`, so it must be specified for that function. The cars data can be read into R by

```r
Cars = read.csv("http://www.utdallas.edu/~ammann/stat6390scripts/cars.csv", comment="#", row.names=1)
```

The plot and modeling functions in R have a formula interface that encourages users to think of the data in terms of dependent variables and independent variables. These formulas
can contain the names of variables in a data frame in which case the `data=` argument must give the name of the data frame object.

```r
plot(mpg ~ weight, data=Cars) # scatterplot since x and y are numeric
plot(mpg ~ origin, data=Cars) # box plots since y is numeric and x is a factor
```

These plots need to be improved visually. For the first plot let’s color points differently for the different levels of `origin`, add an informative title and a legend. In the second plot use the same colors for the boxes as were used for the points in the first plot.

```r
ocols = c("blue","gold","red")
names(ocols) = levels(Cars$origin)
plot(mpg ~ weight, data=Cars, pch=19, col=ocols[Cars$origin])
legend(max(Cars$weight),max(Cars$mpg),legend=levels(Cars$origin),pch=19,col=ocols,xjust=1,
title="EPA Mileage vs Weight")
plot(mpg ~ origin, data=Cars, col=ocols)
title("EPA Mileage by Vehicle Origin")
```

These plots can be saved into graphic files to be included in documents. The graphic file format most often used for \LaTeX{} is `png`. The graphics function `png()` can be given at the beginning of the commands for a graphic or it can be saved interactively if you are using RStudio. If you are preparing graphics for slides, then it usually is more effective to use a black background and white for annotation. This requires some graphical color parameters to be set explicitly to white or some other non-black color.

```r
ocols = c("SkyBlue","gold","red")
names(ocols) = levels(Cars$origin)
png("Cars_wgt_mileage.png",bg="black",width=720,height=720)
plot(mpg ~ weight, data=Cars, pch=19, col=ocols[Cars$origin],
     fg="white",col.axis="white",col.lab="white",col.main="white",col.sub="white")
legend(max(Cars$weight),max(Cars$mpg), legend=levels(Cars$origin),pch=19,
     col=ocols,xjust=1, bty="n", title="Origin",
     text.col="white", title.col="white")
title("EPA Mileage vs Weight",col.main="white")
```

Quantile-Quantile Plots and Probability Distributions

A Quantile-Quantile plot gives a graphical comparison of a data set to a specified probability distribution or another data set. If two distributions are the same, then their quantiles should
be the same. If a r.v. is a linear transformation of another, \( y = a + bX \), then their quantiles should exhibit the same linear relationship. Suppose for example that \( Y \sim N(\mu, \sigma) \) and we observe a random sample from that distribution. A quantile-quantile plot of the data versus the standard normal distribution should fall on a line with intercept \( \mu \) and slope \( \sigma \).

Since the mean and standard deviation are easily distorted by an outlier, \( \text{R} \) uses a robust method for its \texttt{qqline} function to estimate the slope and intercept of a comparison line. By default, \texttt{qqline} estimates the intercept and slope from the 25th and 75th percentiles of the two distributions. Suppose for example \( y \) contains the ordered values of the data and we want to compare its distribution to the standard normal distribution. Let \( n \) denote the sample size. Then the sample quantile of \( y[k] \) is \( k/n \). Instead of comparing \( y \) to those quantiles of the standard normal distribution, an offset is used, by default .5.

```r
n = 50
mu = 100
sig = 12
y = rnorm(n,mu,sig)
### mimic qqnorm(y)
n = length(y)
a = .5 # offset; qqline uses 3/8 if n <= 10
probs = (seq(n) - a)/n
x = qnorm(probs)
plot(x,sort(y)) #equivalent to qqnorm(y)
# mimic qqline(y)
p = c(.25,.75)
yq = quantile(y,p)
xq = qnorm(p)
slope = diff(yq)/diff(xq)
abline(intercept,slope) # equivalent to qqline(y)
```

An alternative to Q-Q plots is Tukey’s Mean Difference Q-Q plot. For this plot the y-axis contains the difference between the sample and reference quantiles, the x-axis contains the average of those values.

```r
### mimic tmdnorm(y)
n = length(y)
a = .5 # offset; qqline uses 3/8 if n <= 10
probs = (seq(n) - a)/n
x = qnorm(probs)
y = sort(y)
y1 = y-x
x1 = (x+y)/2
plot(x1,y1)
```
A script for TMD plots is here:
http://www.utdallas.edu/~ammann/stat6390scripts/tmdplot.r
This script defines functions for Q-Q and M-D plots. The function \( tmdplot(x,y) \) is a basic function with arguments \( x,y \) where \( x \) represents the target quantiles and \( y \) represents the sample. The function \( tmdnorm(y) \) generates a TMD plot with normal distribution target. Other functions are defined that obtain Q-Q and TMD plots for several other common distributions.

A TMD plot is interpreted differently than a standard QQ plot. By default a TMD plot with the normal distribution as target uses the sample mean and s.d. to estimate parameters of the normal distribution from which target quantiles are obtained. In this case if the sample is approximately normal, then the TMD plot will be close to a horizontal line at 0.

source("http://www.utdallas.edu/~ammann/stat6390scripts/tmdplot.r")
n = 100
mu = 50
sig = 8
y = rnorm(n, mu, sig)
par(mfrow=c(1,2),cex.main=.95)
qqnorm(y)
qqline(y)
mtext("Normal data",side=3,line=.5,cex=.9)
tmdnorm(y)
mtext("Normal data",side=3,line=.5,cex=.9)

A commonly encountered distribution with environmental data is the lognormal distribution. This is defined to be the distribution of a r.v. \( X \) such that \( Y = \log(X) \) has a normal distribution. It’s density function is given by

\[
f(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-[\log(x) - \mu]^2/(2\sigma^2)\right), \quad x > 0,
\]

where \( \mu, \sigma^2 \) are the mean and variance, respectively, of \( \log(X) \). Then

\[
E(X) = \theta = \exp(\mu + \sigma^2/2),
\]

\[
\text{median}(X) = \exp(\mu),
\]

\[
\text{Var}(X) = \exp(2\mu + \sigma^2)[\exp(\sigma^2) - 1] = \theta^2[\exp(\sigma^2) - 1],
\]

\[
CV(X) = \tau = \sqrt{\exp(\sigma^2) - 1}.
\]

It can be seen that

\[
\sigma = \sqrt{\log(\tau^2 + 1)},
\]

\[
\mu = \log\left(\frac{\theta}{\tau^2 + 1}\right),
\]

\[
\log(\text{median}(X)) = \mu,
\]

\[
\log(E(X)) = \mu + \sigma^2/2.
\]
Note that the mean of $\log(X)$ corresponds to $\log(\text{median}(X))$. Parameters of the lognormal distribution can be estimated by the sample mean and s.d. of the log-transformed data.

```r
mu = 5
sig = 2
ly = rlnorm(n, log(mu), log(sig))
qqnorm(ly)
qqline(ly)
mtext("Lognormal data",side=3,line=.5,cex=.9)
tmdnorm(ly)
mtext("Lognormal data",side=3,line=.5,cex=.9)
#
1ly = log(ly)
qqnorm(1ly)
qqline(1ly)
mtext("Lognormal data, log-transformed",side=3,line=.5,cex=.9)
tmdnorm(1ly)
mtext("Lognormal data, log-transformed",side=3,line=.5,cex=.9)
```

The lognormal is a skewed distribution. Another commonly encountered skewed distribution is the gamma distribution which has density

$$f(x) = \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-x/\beta}, \quad x > 0, \quad \alpha > 0, \quad \beta > 0.$$ 

This density is derived from the gamma integral,

$$\beta^\alpha \Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x/\beta} \, dx.$$ 

If $X \sim \text{Gamma}(\alpha, \beta)$, then

$$E(X) = \alpha \beta$$

$$\text{Var}(X) = \alpha \beta^2.$$ 

Basic method of moments estimation of these parameters can be derived from the mean and variance. If $m, s$ represent the sample mean and s.d., respectively, of a random sample, then the method of moments estimators are given by

$$\hat{\beta} = \frac{s^2}{m}$$

$$\hat{\alpha} = \frac{m^2}{s^2}.$$ 

A Q-Q plot of gamma distributed data has a similar bowl shape like is seen with skewed data such as the lognormal, so sometimes it can be difficult to distinguish between them. Q-Q and M-D plots using lognormal and gamma targets can help. The file [http://www.utdallas.edu/~ammann/stat6390scripts/lnormEX.r](http://www.utdallas.edu/~ammann/stat6390scripts/lnormEX.r)
gives an example with the standard lognormal distribution and the gamma with same mean and s.d.

The package **EnvStats** includes some data sets taken from EPA guidance documents. One of them is *EPA.94b.tccb.df* which contains measurements of a pollutant TcCB from soil samples in a reference site and a cleanup site. Since these measurements are all positive and their distributions within each site are skewed, we will compare them to the lognormal and gamma distributions.

```r
library(EnvStats)
attach(EPA.94b.tccb.df)
table(Area)
TcCBref = TcCB[Area=="Reference"]
TcCBcleanup = TcCB[Area=="Cleanup"]
source("http://www.utdallas.edu/~ammann/stat6390scripts/tmdplot.r")
par(mfrow=c(2,2))
qqnorm(TcCBref)
qqline(TcCBref)
mtext("Reference Area",line=.25,cex=.8)
tmdnorm(TcCBref)
mtext("Reference Area",line=.25,cex=.8)
#
qqnorm(TcCBcleanup)
qqline(TcCBcleanup)
mtext("Cleanup Area",line=.25,cex=.8)
tmdnorm(TcCBcleanup)
mtext("Cleanup Area",line=.25,cex=.8)
###
lnxy = lnormPlot(TcCBref)
mtext("Reference area",line=.25,cex=.8)
title(sub=paste("Estimated parameters: meanlog = ",round(lnxy$meanlog,2),", sdlog = ",round(lnxy$sdlog,2),"))
gxy = gammaPlot(TcCBref)
mtext("Reference area",line=.25,cex=.8)
title(sub=paste("Estimated parameters: shape = ",round(gxy$shape,2),", scale = ",round(gxy$scale,2),"))
#
lnxyc = lnormPlot(TcCBcleanup)
mtext("Cleanup area",line=.25,cex=.8)
title(sub=paste("Estimated parameters: meanlog = ",round(lnxyc$meanlog,2),", sdlog = ",round(lnxyc$sdlog,2),"))
gxyyc = gammaPlot(TcCBcleanup)
mtext("Cleanup area",line=.25,cex=.8)
```

These plots show that both lognormal and gamma distributions are reasonable models for the reference area, but the gamma distribution gives a better model for the cleanup area. However, note that the estimated shape parameter for the gamma fit is very small. This implies that the cleanup data has mostly very small values with a relatively few larger values.

Now suppose the data represents species counts at some location. A common model for count data is the Poisson distribution. One important property of the Poisson distribution is that it has dispersion equal to 1, where dispersion is the variance divided by the mean. However, if count data is over-dispersed, then the Poisson distribution would not be appropriate. In such cases the negative binomial distribution may be used instead since its dispersion is not 1. The standard definition of the negative binomial is the number of failures before the \( r \)-th success, where \( r \) is a positive integer. The negative binomial has probability mass function,

\[
P(N = k) = \binom{k + r - 1}{k} p^r (1 - p)^k, \quad k \geq 0.
\]

Then \( E(N) = r(1-p)/p \), \( \text{Var}(N) = r(1-p)/p^2 \), and so its dispersion is \( 1/p \). This shows that the negative binomial is over-dispersed. A more general definition of the negative binomial is the distribution of a Poisson r.v. with a random mean that has a gamma distribution. That is,

\[
P(N = k| \Lambda = x) = \frac{x^k}{k!} e^{-x} e^{-\frac{x}{\beta}} x^{\alpha - 1} e^{-x/\beta} dx
\]

where the density of \( \Lambda \) is

\[
f(x) = \frac{1}{\beta^\alpha \Gamma(\alpha)} e^{-x/\beta}, \quad x > 0, \quad \alpha > 0, \quad \beta > 0.
\]

Then the marginal distribution of \( N \) is

\[
P(N = k) = \frac{1}{k! \beta^\alpha \Gamma(\alpha)} \int_0^{\infty} x^{k+\alpha-1} e^{-x(1+\beta)/\beta} dx
\]

\[
= \frac{1}{k! \beta^\alpha \Gamma(\alpha)} \frac{\beta^{k+\alpha}}{(1+\beta)^{k+\alpha}} \Gamma(k+\alpha)
\]

\[
= \frac{\Gamma(k+\alpha)}{k! \Gamma(\alpha)} p^\alpha (1 - p)^k,
\]
where
\[ p = \frac{1}{1 + \beta}. \]

Note that if \( \alpha = r \), then this corresponds to the probability mass function for the negative binomial given above since in that case
\[ \frac{\Gamma(k + r)}{k!\Gamma(r)} = \binom{k + r - 1}{k}. \]

The \texttt{R} function \texttt{rnbinom} generates random samples based on this representation of the negative binomial. It has arguments \texttt{size} and \texttt{prob} corresponding to \( \alpha \) and \( p \), respectively. That parameterization is equivalent to
\[
\begin{align*}
\alpha &= \text{size} \\
\beta &= \frac{\mu}{\text{size}} \\
p &= \frac{\text{size}}{\mu + \text{size}}.
\end{align*}
\]

Let \( E(N) = \mu = \alpha(1 - p)/p \). Then
\[ \text{Var}(N) = \frac{\mu}{p}. \]

If \( m, s \) represent the sample mean and s.d. from a random sample of negative binomial observations, then method of moments estimators of \( p, \alpha \) can be derived. Let
\[ d = s^2/m. \]

Then
\[
\begin{align*}
\hat{p} &= 1/d \\
\hat{\alpha} &= m\hat{p}/(1 - \hat{p}).
\end{align*}
\]

In addition to these diagnostic plots, the \texttt{EnvStats} package includes a function \texttt{gofTest} to perform goodness of fit tests, by default the Shapiro-Wilk test for continuous distributions. Discrete distributions can be tested by a chi-square GOF test, but that test requires additional arguments that make this less useful. In practice one should not rely solely on a GOF test to check assumptions.

The file \texttt{http://www.utdallas.edu/~ammann/stat6390scripts/negbinEX.r} gives an example with \( \mu = 45, \text{size}=5 \) which corresponds to \( p = .1 \) for the negative binomial. This script also examines the \texttt{quine} data frame from the \texttt{MASS} library.
Estimation

Many questions in environmental studies involve estimation of population characteristics. Problems associated with these questions include identification and verification of appropriate probability models for parameter estimation or application of appropriate nonparametric methods if parametric models are not warranted; determination of sample sizes; precision of estimates; dealing with observations that were below detection thresholds (censored data). We will discuss estimation of proportions and quantiles, location (mean and median), and s.d.’s. Other characteristics such as regression and ANOVA parameters will be discussed later in the course.

Estimation of a population proportion

Here we are interested in estimating the proportion $\theta$ of a population that is contained in one particular level of a factor (categorical) variable using a random sample. This problem also provides one method for estimation of quantiles. The basic theory for this problem is the central limit theorem for the binomial distribution. If $X_1, \ldots, X_n$ are i.i.d. Bernoulli r.v.’s with success probability $\theta$, then

$$
\frac{\hat{\theta} - \theta}{\sqrt{\theta(1 - \theta)/n}} \xrightarrow{d} N(0, 1),
$$

where $\hat{\theta}$ is the sample proportion,

$$
\hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} X_i,
$$

The problem with this CLT is that the s.d. is a function of $\theta$. The Law of Large Numbers implies that

$$
\sqrt{\hat{\theta}(1 - \hat{\theta})} \rightarrow \sqrt{\theta(1 - \theta)} \text{ wp 1.}
$$

Slutsky’s Theorem then implies that

$$
\frac{\hat{\theta} - \theta}{\sqrt{\hat{\theta}(1 - \hat{\theta})/n}} \xrightarrow{d} N(0, 1)
$$

We can use this result to obtain a large sample $1 - \alpha$ confidence interval for $\theta$,

$$
\hat{\theta} \pm z\sqrt{\hat{\theta}(1 - \hat{\theta})/n},
$$

where $z$ is the appropriate quantile from the standard normal distribution, $z_{1-\alpha/2}$.

The performance of this confidence interval degrades for $\theta$ near 0 or 1. A more accurate approximation can be obtained by solving the quadratic equation

$$
\frac{\hat{\theta} - \theta}{\sqrt{\theta(1 - \theta)/n}} = z.
$$
This was first proposed by Wilson in 1927 and gives the confidence interval
\[
\frac{1}{1 + \frac{z^2}{n}} \left[ \left( \hat{\theta} + \frac{z^2}{2n} \right) \pm z \sqrt{\frac{\hat{\theta}(1 - \hat{\theta})}{n} + \frac{z^2}{4n^2}} \right].
\]

Another confidence interval uses the binomial distribution directly rather than an approximation. For this interval we must find

\[
\theta_l = \max(\theta : P(N_{n, \theta} \leq k) > \alpha/2)
\]
\[
\theta_u = \min(\theta : P(N_{n, \theta} \geq k) > \alpha/2),
\]

where \( k \) is the number of successes and

\[N_{n, \theta} \sim \text{Binomial}(n, \theta).\]

Although this is straightforward in R we also can use the relationship between the Beta distribution and the binomial,

\[P(N_{n, \theta} \geq k) = P(X_{k, n-k-1} \leq \theta),\]

where

\[X_{\alpha, \beta} \sim \text{Beta}(\alpha, \beta).\]

This gives

\[
\theta_l = \text{qbeta}(\alpha/2; k, n - k + 1)
\]
\[
\theta_u = \text{qbeta}(1 - \alpha/2; k + 1, n - k),
\]

where \( \text{qbeta} \) is the quantile function for the Beta distribution. This is referred to as the Clopper-Pearson confidence interval for a population proportion. It can be expressed in R as follows. Let \( n, k \) denote the number of trials and number of successes, respectively. Then a \( 1 - \alpha \) C-P confidence interval for \( \theta \) is given by

\[
\text{cp.lower} = \text{qbeta}(\alpha/2, k, n-k+1)
\]
\[
\text{cp.upper} = \text{qbeta}(1-\alpha/2, k+1, n-k)
\]

We can use the simulation capabilities of R to compare these three confidence intervals in terms of their coverage probabilities and widths.

**Example.** Suppose the sample size is 100 and the observed number of successes is 6, so the sample proportion is 0.06. The following R code obtains 95% confidence intervals using these three methods.
\begin{verbatim}
n = 400
k = 24
a = .05
phat = k/n
z = qnorm(1-a/2)
shat = phat*(1-phat)/n
p.conf = matrix(0,3,2)
dimnames(p.conf) = list(c("CLT","Wilson","C-P"),c("Lower","Upper"))
p.conf["CLT",] = phat + c(-1,1)*z*sqrt(shat)
b = z^2/n
p.conf["Wilson",] = (phat + b/2 + c(-1,1)*z*sqrt(shat + b/(4*n)))/(1 + b)
p.conf["C-P",] = c(qbeta(a/2, k, n-k+1), qbeta(1-a/2, k+1, n-k))
print(p.conf)
# now generate 4000 samples and compare coverage and width
n = 400
N = 4000
p = .06
a = .05
z = qnorm(1-a/2)
b = z^2/n
X = rbinom(N,n,p)
phat = X/n
shat = phat*(1-phat)/n
CLT.ci = Wilson.ci = CP.ci = matrix(0,2,N)
CLT.ci[1,] = phat - z*sqrt(shat)
CLT.ci[2,] = phat + z*sqrt(shat)
Wilson.ci[1,] = (phat + b/2 - z*sqrt(shat + b/(4*n)))/(1 + b)
Wilson.ci[2,] = (phat + b/2 + z*sqrt(shat + b/(4*n)))/(1 + b)
CP.ci[1,] = qbeta(a/2, X, n-X+1)
CP.ci[2,] = qbeta(1-a/2, X+1, n-X)
CLT.cov = mean(CLT.ci[1,] <= p & p <= CLT.ci[2,])
CLT.w = mean(CLT.ci[2,] - CLT.ci[1,])
Wilson.w = mean(Wilson.ci[2,] - Wilson.ci[1,])
CP.cov = mean(CP.ci[1,] <= p & p <= CP.ci[2,])
CP.w = mean(CP.ci[2,] - CP.ci[1,])

Sample size determination for estimating a population proportion. The half-width of a confidence interval gives the precision of the estimate and is a function of the sample size and level of confidence for a particular method. When possible it is best to plan ahead and determine what sample size would be required to attain specific goals for precision and level of confidence. We can get two estimates of sample size from the CLT confidence interval.
\end{verbatim}
interval,
\[ \hat{\theta} \pm z\sqrt{\theta(1-\theta)/n}. \]

A conservative sample size can be obtained using the inequality
\[ \sqrt{\theta(1-\theta)} \leq 0.5, \ 0 \leq \theta \leq 1. \]

This implies that the half-width is bounded by
\[ \frac{z}{2\sqrt{n}}. \]

Let \( e \) denote the desired precision for the confidence interval and let \( z \) denote the quantile from the standard normal distribution corresponding to the desired level of confidence. Then the required sample size would be
\[ n = \left( \frac{z}{2e} \right)^2. \]

This conservative sample size gives reasonable values when \( \theta \) is moderate, but is too conservative when \( \theta \) is close to 0 or 1. In those cases it is better to use a bound on \( \theta \) if available, or invest in a small preliminary sample to obtain a rough estimate of \( \theta \). If \( \theta_0 \) denotes this prior bound (or preliminary estimate), then we have
\[ \sqrt{\theta(1-\theta)} \leq (\approx)\sqrt{\theta_0(1-\theta_0)} \]

The corresponding sample size is then
\[ n = \left( \frac{z}{e} \right)^2 \theta_0(1-\theta_0). \]

For example, if we wish to estimate a population proportion to within 0.02 with 98% confidence with no prior bound on \( \theta \), then the required sample size would be
\[ n = \left( \frac{qnorm(.99)}{2(.02)} \right)^2 = 3382. \]

If we expect this proportion to be no more than 0.1, then the corresponding sample size would be
\[ n = \left( \frac{qnorm(.99)}{.02} \right)^2 (.1)(.9) = 1218. \]

Of course if the population is much larger than we expect, then this sample size will be too small to attain the stated goals.
Estimation of a population mean

The first case considered is for data that is approximately normally distributed. It is assumed that a quantile-quantile plot of the data has been examined and shows this assumption to be reasonable. Since the population variance never is known in actual applications, it must be treated as an unknown nuisance parameter. A confidence interval for this case is motivated by the linearity property of the normal distribution. If $X_1, \cdots, X_n$ are i.i.d. $N(\mu, \sigma)$ r.v.’s, then

$$\frac{\bar{X} - \mu}{\sigma/n} \sim N(0, 1).$$

We cannot construct a confidence interval directly from this result since the s.d. is unknown. William Gosset solved this problem by replacing $\sigma$ with the sample s.d. and then deriving the distribution of

$$t = \frac{\bar{X} - \mu}{s/n}$$

where $s^2$ is the sample variance,

$$s^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \bar{X})^2.$$

This distribution is referred to as the $t$-distribution with $n-1$ degrees of freedom. The proof of this result is based on the representation of $t$ as the ratio

$$\frac{Z}{\sqrt{Q/d}},$$

where $Z, Q$ are independent r.v.’s, $Z \sim N(0, 1)$ and $Q$ has a chi-square distribution with $d$ degrees of freedom. The $t$-distribution is symmetric about 0, has infinite mean for 1 df (this is the Cauchy distribution), has finite mean but infinite variance for 2 df, and has finite variance for df $\geq 3$. It has heavier tails than the standard normal distribution. The law of large numbers and Slutsky’s Theorem imply that

$$t_{n-1} \xrightarrow{d} N(0, 1).$$

The Central Limit Theorem implies that this limit also holds for any set of i.i.d. r.v.’s with finite variance.

A confidence interval for $\mu$ is

$$\bar{X} \pm ts/\sqrt{n},$$

where $t$ is the appropriate quantile from the $t_{n-1}$ distribution. Note that this also can be used as a large sample confidence interval for the mean, but then we must address the question
of how large is large for the coverage probability to be approximately the same as the level of confidence.

Determination of an appropriate sample size for this confidence interval requires a prior bound or preliminary estimate of the population standard deviation. Denote this by \( \sigma_0 \). An initial estimate for the sample size can be obtained by using \( z \) instead of \( t \) in the confidence interval, setting the resulting precision equal to the required half-width, and then solving for \( n \). This gives

\[
    n_0 = \left( \frac{z\sigma_0}{e} \right)^2,
\]

where \( e \) is the goal for precision. Since \( t_{n-1} \geq z \) for the same level of confidence, then \( n_0 \) will be smaller than the required sample size. So we can iterate the following steps, 

\[
    n_1 = (qt(1 - \alpha/2,n0-1)*s0/e)^2 \\
n0 = ceiling(n1)
\]

until \( n_1 \leq n_0 \). Note that the 2nd step rounds up the intermediate sample size. The following defines a function in \( \textbf{R} \) that implements this iterative process and returns the final sample size.

```r
normN = function(e, s0, alpha=.05, maxN=5000) {
    prob = 1 - alpha/2
    n0 = (qnorm(prob)*s0/e)^2
    n0 = ceiling(n0)
    n1 = maxN - 1
    checkN = TRUE #ensure loop iterates at least once
    while(checkN & n1 < maxN) {
        n1 = (qt(prob,n0-1)*s0/e)^2
        n1 = ceiling(n1)
        if(n1 <= n0) {
            checkN = FALSE
        }
    }
    n0 = ceiling(n1)
    return(n0)
}
```

The \( maxN \) argument is a maximum for the sample size to prevent this function from iterating indefinitely under some degenerate conditions.

A weakness of the mean and s.d. is their sensitivity to outliers. This can be managed by using robust alternatives to estimate location and scale. The median is an example of a location estimator that is not sensitive to outliers, but its efficiency relative to the mean is fairly low, so other estimators have been developed. Likewise, the MAD is a robust estimator of scale also with low efficiency. The \textit{MASS} package in \textbf{R} includes a function \textit{hubers} which
provides an alternative to these estimators that is robust and more efficient than the median and MAD. A more robust confidence interval for the mean can be obtained by replacing the sample mean and sample s.d. by the respective estimates returned by `hubers`.

```r
library(MASS)
alpha = .05
n = 50
tval = qt(1-alpha/2,n-1)
x = rnorm(n)
hx = hubers(x)
m = mean(x)
s = sd(x)
mrob = hx$mu
srob = hx$s
ci = m + c(-1,1)*tval*s/sqrt(n-1)
ci.rob = mrob + c(-1,1)*tval*srob/sqrt(n-1)
# replace 2 values with outliers
xout = c(10,15)
x1 = x
x1[seq(length(xout))] = xout
hx1 = hubers(x1)
m1 = mean(x1)
s1 = sd(x1)
mrob1 = hx1$mu
srob1 = hx1$s
ci1 = m1 + c(-1,1)*tval*s1/sqrt(n-1)
ci1.rob = mrob1 + c(-1,1)*tval*srob1/sqrt(n-1)
```

Note that the outliers have inflated the sample s.d. which makes the standard confidence interval much wider than it would have been without those outliers.

**Estimation of a population variance**

In this section it is again assumed that the assumption of normality is reasonable. The basic theory for this case is given by the following theorem. If $X_1, \cdots, X_n$ are i.i.d. $N(\mu, \sigma)$, then $\bar{X}$ and $s^2$ are independent r.v.'s and

\[
\frac{(n-1)s^2}{\sigma^2} \sim \chi_{n-1}^2.
\]

We can use this result to derive a confidence interval for $\sigma^2$. Let $C_L, C_U$ satisfy

\[
C_L = qchisq(\alpha/2, n-1) \\
C_U = qchisq(1 - \alpha/2, n-1).
\]
Then
\[
\alpha/2 = P \left[ \frac{(n-1)s^2}{\sigma^2} \leq C_L \right]
\]
\[
= P \left[ \frac{(n-1)s^2}{C_L} \leq \sigma^2 \right],
\]
\[
\alpha/2 = P \left[ \frac{(n-1)s^2}{\sigma^2} > C_U \right]
\]
\[
= P \left[ \sigma^2 < \frac{(n-1)s^2}{C_U} \right],
\]
and so a 1 - \( \alpha \) confidence interval for \( \sigma^2 \) is
\[
P \left[ \frac{(n-1)s^2}{C_U} \leq \sigma^2 \leq \frac{(n-1)s^2}{C_L} \right] = 1 - \alpha.
\]
This interval is referred to as an equal probability interval, but it is not necessarily the interval with minimum width. Derivation of a minimum width confidence interval for the variance requires numerical techniques that often produce only modest reductions in width and so is rarely worth it.

There are situations in which a one-sided confidence interval is required, typically in the form of an upper bound on the variance or s.d. This can be obtained by setting
\[
C_L = qchisq(\alpha, n-1)
\]
to give
\[
P \left[ \sigma^2 \leq \frac{(n-1)s^2}{C_L} \right] = 1 - \alpha.
\]
Corresponding confidence intervals for the s.d. are obtained by taking the square root of the intervals for the variance.

**Estimation of parameters for other distributions**

Two other distributions often encountered in environmental data, the lognormal and the gamma will be discussed in this section.

**Lognormal distribution.** Suppose \( Y_1, \ldots, Y_n \) are i.i.d. lognormal with parameters \( \mu, \sigma \), and let \( X_i = \log(Y_i) \). Since the log-transformed data has a normal distribution with mean and s.d. \( \mu, \sigma \), we can use the confidence intervals given above for those parameters. This gives
\[
P(\bar{X} - t_{n-1}s_x/\sqrt{n} \leq \mu \leq \bar{X} + t_{n-1}s_x/\sqrt{n}) = 1 - \alpha.
\]
However, if we transform this interval back to the original scale, we have
\[
P(\exp\{\bar{X} - t_{n-1}s_x/\sqrt{n}\} \leq \exp\{\mu\} \leq \exp\{\bar{X} + t_{n-1}s_x/\sqrt{n}\}) = 1 - \alpha.
\]
Since \( \exp\{\mu\} = \text{median}(X) \), then this is a confidence interval for the median of the population, not the mean. If a confidence interval for the mean is required, then we must use a different method.

Let

\[ \theta = e^\beta, \]

where

\[ \beta = \mu + \sigma^2/2. \]

Then the minimum variance unbiased estimator (MVUE) of \( \beta \) is

\[ \hat{\beta}_{mvue} = \bar{X} + \frac{s^2_x}{2} \]

and the maximum likelihood estimator (MLE) of \( \beta \) is

\[ \hat{\beta}_{mle} = \bar{X} + \frac{\hat{s}^2_{mle}}{2}, \]

where

\[ \hat{s}^2_{mle} = \frac{1}{n} \sum_{i=1}^{n} (X_i - \bar{X})^2 = \frac{n-1}{n} s^2_x. \]

The corresponding estimators for \( \theta \) are

\[ \hat{\theta}_{mvue} = \exp(\hat{\beta}_{mvue}) \]
\[ \hat{\theta}_{mle} = \exp(\hat{\beta}_{mle}). \]

This shows that estimation of the mean of a lognormal distribution requires simultaneous estimation of the mean and variance of the log-transformed variables. Several authors have proposed methods to obtain confidence intervals for the mean.

A confidence interval for \( \beta \) proposed by Land (1971) is

\[ \left[ \hat{\beta}_{mvue} + s_x \frac{C_{\alpha/2}}{\sqrt{n-1}}, \hat{\beta}_{mvue} + s_x \frac{C_{1-\alpha/2}}{\sqrt{n-1}} \right] \]

where \( C_p \) are critical values derived by Land. The method proposed by Zou, et al (2009) is based on normal approximations to these statistics. Their confidence interval has lower and upper limits given by

\[ LL = \hat{\theta}_{mle} \exp\left\{ - \left[ \frac{z^2_{1-\alpha/2} s^2_x}{n} + \left( \frac{s^2_x}{2} - \frac{(n-1)s^2}{2\chi^2_{1-\alpha/2,n-1}} \right)^2 \right]^{1/2} \right\} \]
\[ UL = \hat{\theta}_{mle} \exp\left\{ - \left[ \frac{z^2_{1-\alpha/2} s^2_x}{n} - \left( \frac{s^2_x}{2} + \frac{(n-1)s^2}{2\chi^2_{1-\alpha/2,n-1}} \right)^2 \right]^{1/2} \right\} \]
Cox constructed confidence limits based on the normal approximations for $\beta$. These confidence intervals are given by

$$
\left[ \exp(\hat{\beta} - t_{1-\alpha/2,n-1}\hat{\sigma}_\beta), \exp(\hat{\beta} + t_{1-\alpha/2,n-1}\hat{\sigma}_\beta) \right]
$$

where

$$
\hat{\sigma}_\beta^2 = \frac{s_x^2}{n} + \frac{s^4}{2(n+1)}.
$$

Details and additional methods can be found in the help page for function \texttt{elnormAlt} in package \textit{EnvStats}.

There are several problems with these confidence intervals. Some data may produce lower confidence limits for $\theta$ that are negative, but the main problem is due to the sensitivity of these estimators to outlying observations. Since the lognormal distribution can have very heavy tails, some approximations used to derive these confidence limits may have very high upper limits. To address the lack of robustness, Serfling (2002) introduced a family of estimators for $\theta$ based on generalized-median estimation. These estimators are generated from a kernel function

$$
h(x_1, \cdots, x_k)
$$

which is a median unbiased estimator of a parameter $\theta$. That is, the median of the distribution of $h(X_1, \cdots, X_m)$ equals $\theta$. A generalized median estimator of $\theta$ is the median of $h$ over all subsets of size $m$ of the data. In the case of the lognormal distribution we need estimators for the mean and variance, so we can use the MLE estimators for those kernel functions. The kernel size, $m$, provides a trade-off between asymptotic relative efficiency (ARE) and break-down point (BP). Higher value for $m$ increases ARE but decreases BP. An \textsf{R} function to obtain these estimates is defined in

\url{http://www.utdallas.edu/~ammann/stat6390scripts/GMest.r}

\textbf{GMlnormEst:} Generalized Median estimation of lognormal parameters

\textbf{Usage:}

\texttt{GMlnormEst(y, k=5, m=NULL, alpha=.05, Nsamp=10000)}

\textbf{Arguments}

- \textit{y} vector of data assumed to be lognormally distributed
- \textit{k} size of kernel for the mean, default is 5, minimum value is 2
- \textit{m} size of kernel for variance, default is same as \textit{k}, minimum value is 2
- \textit{alpha} level of confidence interval is 1-\textit{alpha}, default is 0.05
- \textit{Nsamp} number of samples for kernel functions, default is 10000

\textbf{Value:} a list with components

- \textit{ci} vector containing lower and upper 1-\textit{alpha} confidence limits for the mean of a lognormal distribution
- \textit{mu.hat} GM-estimate of the mean of the log-transformed data
- \textit{sig2.hat} GM-estimate of the variance of the log-transformed data
- \textit{theta.hat} GM-estimate of the mean, $\theta = \exp(\mu + \sigma^2/2)$
- \textit{cv.hat} GM-estimate of the CV, $CV = \sqrt{\exp(\sigma^2) - 1}$
Since resampling is used for the kernel functions instead of all possible subsamples, estimates will vary slightly if the function is applied to the same data more than once.

Reference
http://www.utdallas.edu/~serfling/papers/naaj02.pdf

Examples

```r
source("http://www.utdallas.edu/~ammann/stat6390scripts/GMest.r")
require(EnvStats)
n = 100
mu = 4
sig = 1
theta = exp(mu + sig^2/2)
y = rlnorm(n,mu,sig)
cat(paste("theta =",round(theta,3)),"\n")
elnormAlt(y, ci=TRUE)
GM1normEst(y)
# add some outliers
ymax = max(y)
yc = c(y,c(5,10,15)*ymax)
elnormAlt(yc, ci=TRUE)
GM1normEst(yc, k=3)
# repeat with larger sample size
n = 250
y = rlnorm(n,mu,sig)
ymax = max(y)
yc = c(y,c(5,10,15)*ymax)
elnormAlt(yc, ci=TRUE)
GM1normEst(yc, m=3)
```

**Gamma distribution.**

As we saw earlier, simple method of moments estimators exist for the parameters of the gamma distribution,

\[
\hat{\beta} = \frac{s^2}{m} \\
\hat{\alpha} = \frac{m^2}{s^2}.
\]

The likelihood equations for MLE give

\[
\beta_{mle} = \frac{1}{\hat{\alpha}} \bar{X},
\]

but the likelihood equation for \(\alpha\) does not have a closed form solution. Therefore numerical optimization algorithms must be used. Confidence intervals can be obtained using a normal approximation for a fractional power of the original data. Functions `egamma` and `egammaAlt`
in package *EnvStats* implement these methods to obtain estimates and confidence intervals for $\alpha, \beta$ (*egamma*) and the mean, CV (*egammaAlt*). The value returned by these functions is a list that includes the normal transformation power $p$ used to obtain the confidence intervals. A diagnostic `qqnorm` plot of $x^p$ can show the appropriateness of this approximation. Here is an example.

```r
library(EnvStats)
n = 100
shp = .4
c1 = 1
X = rgamma(n, shape=shp, scale=c1) # E(X) = .5, Var(X) = .5
Xme = egammaAlt(X, ci=TRUE)
tpwr = round(Xme$interval$normal.transform.power,4)
qqnorm(X^tpwr,main=expression(paste("Normal Quantile-Quantile Plot of ", X^p)))
mtext(paste("where p =", tpwr), line=.25)
qqline(X^tpwr)
cat(paste("Actual mean =", round(shp*c1,3)),"\n")
cat(paste("95% confidence interval for the mean: (",
       paste(round(Xme$interval$limits,3), collapse=" ", ")",sep=""),"\n")
### repeat with shape > 1
shp = 3
X = rgamma(n, shape=shp, scale=c1) # E(X) = .5, Var(X) = .5
Xme = egammaAlt(X, ci=TRUE)
tpwr = round(Xme$interval$normal.transform.power,4)
qqnorm(X^tpwr,main=expression(paste("Normal Quantile-Quantile Plot of ", X^p)))
mtext(paste("where p =", tpwr), line=.25)
qqline(X^tpwr)
cat(paste("Actual mean =", round(shp*c1,3)),"\n")
cat(paste("95% confidence interval for the mean: (",
       paste(round(Xme$interval$limits,3), collapse=" ", ")",sep=""),"\n")
```

**Simulation.**

To illustrate how simulation can be used for comparison of methods, the following steps describe a process to set up and perform a simulation to compare four methods that give confidence intervals for the mean of lognormal populations. The R code developed here will be recycled for comparisons in other problems.

1. **What measures will be used for the comparisons?** *Answer.* Comparisons will be based on the proportion of confidence intervals that contain the actual population parameter and the widths of the confidence intervals.

2. **How many replications will be performed.** *Answer.* The goal for estimation of coverage probabilities for these confidence intervals is to have a precision of .005 with 99%
confidence. The minimum sample size needed to attain this goal is

\[ N_1 = (2.576/0.005)^2(0.95)(0.05) = 7299. \]

This will be rounded up to 10,000 replications. Determination of an appropriate sample size for the widths requires a preliminary estimate of the variability associated with the widths. Therefore an initial simulation will be performed using 1000 replications to estimate the s.d. of widths for each method. The largest of those s.d.'s will be used to determine sampling error associated with the widths if 10,000 replications are used. If that sampling error is not sufficiently small, then the number of replications will be based on estimation of a mean using the preliminary estimates of standard error for widths.

3. All of the simulated data will be stored in a \( n \times N \) matrix so that the \texttt{R} function \textit{apply()} can be used to efficiently compute confidence intervals for each column of the simulated data matrix. This requires defining functions that just return a confidence interval for a sample since the \textit{EnvStats} function \textit{elnormAlt} returns a list one component of which is the confidence interval.

\begin{verbatim}
library(EnvStats)
Land.ci = function(x, alpha=.05) {
  ci = elnormAlt(x, ci=TRUE, ci.method="land", conf.level=1-alpha)$interval$limits
  return(ci)
}
Zou.ci = function(x, alpha=.05) {
  ci = elnormAlt(x, ci=TRUE, ci.method="zou", conf.level=1-alpha)$interval$limits
  return(ci)
}
Cox.ci = function(x, alpha=.05) {
  ci = elnormAlt(x, ci=TRUE, ci.method="cox", conf.level=1-alpha)$interval$limits
  return(ci)
}
source("http://www.utdallas.edu/~ammann/stat6390scripts/GMest.r")
GM.ci = function(x, k=5, alpha=.05, Nsamp=4000) {
  ci = GMlnormEst(x, k=k, alpha=alpha, Nsamp=Nsamp)$ci
  return(ci)
}
\end{verbatim}

Since \textit{GMlnormEst} uses subsampling, it will be slow when used with \textit{apply}. The \textit{elnormAlt} methods will be faster or slower depending on how the methods are coded internally. Nevertheless, there will be significant time saved using \textit{apply} compared to a for loop over replications.
4. Generate lognormal data with sample size 50, obtain confidence intervals for each method, obtain proportions of intervals that cover actual parameter, obtain mean widths of the intervals.

\[
\begin{align*}
n &= 50 \\
\alpha &= .05 \\
N &= 10000 \\
lmu &= 3 \\
l_{sig} &= 1 \\
\theta &= exp(lmu + l_{sig}^2/2) \\
X &= matrix(rlnorm(n*N, lmu, l_{sig}), n, N) \\
\text{Xland.ci} &= apply(X, 2, Land.ci) \\
\text{Xzou.ci} &= apply(X, 2, Zou.ci) \\
\text{Xcox.ci} &= apply(X, 2, Cox.ci) \\
\text{XGM.ci} &= apply(X, 2, GM.ci, Nsamp=1000) \quad \# \text{use smaller number of subsamples to reduce time} \\
\end{align*}
\]

\[
\begin{align*}
\text{Xout} &= matrix(0, 4, 3) \\
\text{dimnames}(\text{Xout}) &= list(c("Land", "Zou", "Cox", "GM"), c("Coverage", "Width", "se(Width)")) \\
\text{Xout}["Land", "Coverage"] &= mean(\text{Xland.ci}[1,] < \theta & \theta < \text{Xland.ci}[2,]) \\
\text{Xland.w} &= \text{Xland.ci}[2,] - \text{Xland.ci}[1,] \\
\text{Xout}["Land", "Width"] &= mean(\text{Xland.w}) \\
\text{Xout}["Land", "se(Width)"] &= sd(\text{Xland.w})/sqrt(N) \\
\text{Xout}["Zou", "Coverage"] &= mean(\text{Xzou.ci}[1,] < \theta & \theta < \text{Xzou.ci}[2,]) \\
\text{Xzou.w} &= \text{Xzou.ci}[2,] - \text{Xzou.ci}[1,] \\
\text{Xout}["Zou", "Width"] &= mean(\text{Xzou.w}) \\
\text{Xout}["Zou", "se(Width)"] &= sd(\text{Xzou.w})/sqrt(N) \\
\text{Xout}["Cox", "Coverage"] &= mean(\text{Xcox.ci}[1,] < \theta & \theta < \text{Xcox.ci}[2,]) \\
\text{Xcox.w} &= \text{Xcox.ci}[2,] - \text{Xcox.ci}[1,] \\
\text{Xout}["Cox", "Width"] &= mean(\text{Xcox.w}) \\
\text{Xout}["Cox", "se(Width)"] &= sd(\text{Xcox.w})/sqrt(N) \\
\text{Xout}["GM", "Coverage"] &= mean(\text{XGM.ci}[1,] < \theta & \theta < \text{XGM.ci}[2,]) \\
\text{XGM.w} &= \text{XGM.ci}[2,] - \text{XGM.ci}[1,] \\
\text{Xout}["GM", "Width"] &= mean(\text{XGM.w}) \\
\text{Xout}["GM", "se(Width)"] &= sd(\text{XGM.w})/sqrt(N) \\
\end{align*}
\]

```r
print(\text{Xout}, digits=4)
```

Now repeat for contaminated data.

\[
\begin{align*}
n &= 50 \\
\alpha &= .05 \\
N &= 10000 \\
lmu &= 3 \\
l_{sig} &= 1 \\
\theta &= exp(lmu + l_{sig}^2/2) \\
X &= matrix(rlnorm(n*N, lmu, l_{sig}), n, N) \\
\text{Xland.ci} &= apply(X, 2, Land.ci) \\
\text{Xzou.ci} &= apply(X, 2, Zou.ci) \\
\text{Xcox.ci} &= apply(X, 2, Cox.ci) \\
\text{XGM.ci} &= apply(X, 2, GM.ci, Nsamp=1000) \quad \# \text{use smaller number of subsamples to reduce time} \\
\end{align*}
\]
\theta = \exp(lmu + lsig^2/2)
X = \text{matrix}(rlnorm(n*N, lmu, lsig), n, N)
Xc = \text{apply}(X, 2, \text{sort})
ncontam = 2
contam.fac = 5
xcontam = Xc[seq(n-ncontam+1,n),]*contam.fac
Xc[seq(n-1,n),] = xcontam
Xlandc.ci = \text{apply}(Xc, 2, \text{Land}.ci)
Xzouc.ci = \text{apply}(Xc, 2, \text{Zou}.ci)
Xcoxc.ci = \text{apply}(Xc, 2, \text{Cox}.ci)
XGMc.ci = \text{apply}(Xc, 2, \text{GM}.ci, \text{Nsamp}=2000) # use smaller number of subsamples to reduce time
Xoutc = \text{matrix}(0, 4, 3)
dimnames(Xoutc) = \text{list}(c("Land", "Zou", "Cox", "GM"), c("Coverage", "Width", "\text{se}(\text{Width})")
Xoutc["Land", "\text{Coverage}"] = \text{mean}(Xlandc.ci[1,] < \theta & \theta < Xlandc.ci[2,])
Xlandc.w = Xlandc.ci[2,] - Xlandc.ci[1,]
Xoutc["Land", "\text{Width}"] = \text{mean}(Xlandc.w)
Xoutc["Land", "\text{se}(\text{Width})"] = \text{sd}(Xlandc.w)/sqrt(N)
Xoutc["Zou", "\text{Coverage}"] = \text{mean}(Xzouc.ci[1,] < \theta & \theta < Xzouc.ci[2,])
Xzouc.w = Xzouc.ci[2,] - Xzouc.ci[1,]
Xoutc["Zou", "\text{Width}"] = \text{mean}(Xzouc.w)
Xoutc["Zou", "\text{se}(\text{Width})"] = \text{sd}(Xzouc.w)/sqrt(N)
Xoutc["Cox", "\text{Coverage}"] = \text{mean}(Xcoxc.ci[1,] < \theta & \theta < Xcoxc.ci[2,])
Xcoxc.w = Xcoxc.ci[2,] - Xcoxc.ci[1,]
Xoutc["Cox", "\text{Width}"] = \text{mean}(Xcoxc.w)
Xoutc["Cox", "\text{se}(\text{Width})"] = \text{sd}(Xcoxc.w)/sqrt(N)
Xoutc["GM", "\text{Coverage}"] = \text{mean}(XGMc.ci[1,] < \theta & \theta < XGMc.ci[2,])
XGMc.w = XGMc.ci[2,] - XGMc.ci[1,]
Xoutc["GM", "\text{Width}"] = \text{mean}(XGMc.w)
Xoutc["GM", "\text{se}(\text{Width})"] = \text{sd}(XGMc.w)/sqrt(N)
print(\text{round}(Xoutc, 3))

Examples of how this code can be organized into files that generate the results and then incorporate them into a pdf document using Sweave are given in the following files.
Generate data, apply results, organize results:
http://www.utdallas.edu/~ammann/stat6390scripts/compare_ci.r
http://www.utdallas.edu/~ammann/stat6390scripts/compare_ci.Rnw
The pdf output is created by the following commands in a unix shell:
R --no-save < compare_ci.r
R CMD Sweave compare_ci.Rnw
pdflatex compare_ci.tex
Here is the output: http://www.utdallas.edu/~ammann/stat6390scripts/compare_ci.pdf

If you are using **Rstudio** you can import a `.Rnw` file into **Rstudio** and then use the **Compile PDF** option in the **File** menu to create the pdf.

**Estimation of Quantiles**

Some regulations involve restrictions on quantiles, for example a requirement that concentrations of a chemical can exceed some threshold no more than 5% of the time. Other situations may exist in which parametric models are not appropriate and we must use nonparametric measures such as the median. One problem with quantile estimation is that larger sample sizes are required, especially for more extreme quantiles. We will consider this problem for normal, lognormal, and gamma distributions, and for nonparametric estimates.

Suppose $X_1, \ldots, X_n$ denote i.i.d. r.v.’s with density $f$ and let $X_{(1)}, \ldots, X_{(n)}$ denote their ordered values. For $0 < p < 1$ let $q_p$ denote the $p^{th}$ quantile of $f$,

$$p = \int_{-\infty}^{q_p} f(x) dx.$$ 

The sample quantile $Q_n(p)$ is given by

$$Q_n(p) = (1 - \gamma)X_{([np])} + \gamma X_{([np])},$$

where

$$[np] = \text{floor}(np), \quad [np] = \text{ceiling}(np),$$

and $0 \leq \gamma \leq 1$. Note that this definition results in a function that is discontinuous in $p$. This definition can be modified to produce quantiles that are continuous in $p$ by defining $\gamma$ to be a linear interpolator between the ordered values.

A CLT exists for sample quantiles,

$$\frac{\sqrt{n}(Q_n(p) - q_p)}{\sigma_p} \xrightarrow{d} N(0, 1),$$

where

$$\sigma_p = \sqrt{p(1 - p)f(q_p)}.$$ 

Therefore, a large sample confidence interval for $q_p$ is

$$Q_n(p) \pm \frac{z_{1-\alpha/2}}{\hat{f}(q_p)} \sqrt{p(1 - p)/n}.$$

The difficulty here is obtaining an estimate of $f(q_p)$. The problem is straightforward if $p = .5$ and the data are normally distributed. In that case,

$$f(q_{.5}) = \frac{1}{\sigma \sqrt{2\pi}},$$
and so we could use 

\[ \hat{f}(q) = \frac{1}{s\sqrt{2\pi}} \]

and use t-values rather than z-values,

\[ Q_n(.5) \pm t_{1-\alpha/2,n-1}s\sqrt{\pi/2n}. \]

Note that the half-width of this confidence interval is \( \sqrt{\pi/2} = 1.253 \) times the half-width of a confidence interval for the mean in this case.

Otherwise, estimation of the density function at the quantile is difficult and other methods have been proposed. One alternative is based on the equivalence between a confidence interval for a quantile and a tolerance interval for a population. Suppose for example that we wish to find \( C_u \) such that

\[ P(X_{\lfloor np \rfloor}) \leq C_u) = 1 - \alpha. \]

In this case \((-\infty, C_u)\) is referred to as an upper tolerance interval with coverage probability \( p \) and confidence \( 1 - \alpha \). Note that

\[ X_{\lfloor np \rfloor} \]

is the sample \( p^{th} \) quantile and so we are \( 100(1 - \alpha)\% \) confident that \( 100p\% \) of the sample will be no more than \( C_u \). It can be shown that if the data follows a normal distribution then

\[ C_u = \bar{X} + ks, \]

where \( s \) is the sample standard deviation and \( k \) is a function of the sample size given below. Let

\[ \delta = z_p\sqrt{n}, \ t = qt(1 - \alpha, n - 1, \delta), \]

then \( k = t/\sqrt{n} \). The value of \( k \) for two-sided tolerance intervals is similar but involves the chisquare distribution instead of the t-distribution. The \texttt{EnvStats} library contains a function \texttt{eqnorm} that uses this approach to obtain confidence intervals for quantiles of normal populations. Obviously the assumption of normality for the data must be checked before using this function. These results can be extended immediately to the lognormal distribution since the logarithm of a quantile from the lognormal distribution is the same as the quantile from the normal distribution. The function \texttt{eqlnorm} provides quantile confidence intervals for the lognormal in this way.

**Quantile estimation for gamma distributions.** One approach for gamma distributed data is to follow the same approach used for estimation of parameters by a power transformation to make the data approximately normal. We can use quantile estimates for the normal as above and then transform back to the original scale since the power transformation is monotone. This is the approach taken by the \texttt{eqgamma} function in \texttt{EnvStats}.

**Examples.**
1. Simulate normally distributed data and obtain upper 95% confidence limits for the 0.9 quantile.
source("http://www.utdallas.edu/~ammann/stat6390scripts/tmdplot.r")
require(EnvStats)

n = 80
mu = 10
sig = 2
p = .9
alpha = .05

cat("Actual quantile for normal distribution\n")
qn = qnorm(p,mu,sig)
cat(paste("Actual", p, "quantile =", round(qn,4)),"\n")

X = rnorm(n,mu,sig)  # simulate sample from normal distribution

# check assumption of normality
qqnorm(X)
qqline(X)

# obtain upper confidence interval for .9 quantile
qXe = eqnorm(X, p=p, ci=TRUE, ci.type="upper", conf.level=1-alpha)
cat(paste("Upper confidence limit for",p, "quantile =",round(qXe$interval$limits[2],4)),")

# Perform simulation to generate 4000 samples from this normal distribution
# Then find the proportion of samples for which the actual .9 quantile <= upper limits
N = 4000
X = matrix(rnorm(n*N,mu,sig),n,N)  # simulated samples are columns of X

upL = function(x,p,alpha) {
  xe = eqnorm(x, p=p, ci=TRUE, ci.type="upper", conf.level=1-alpha)
  return(xe$interval$limits[2])
}

Xu = apply(X,2,upL,p,alpha)
Xp = mean(qn <= Xu)
cat(paste("Proportion of upper confidence limits greater than ",p," quantile: ",round(Xp,3), sep=""))

2. Obtain a 90% confidence interval for the median of TcCB in the reference area of EPA.94b.tccb.df. Assume this data has a lognormal distribution.

source("http://www.utdallas.edu/~ammann/stat6390scripts/tmdplot.r")
require(EnvStats)

alpha = .1
p = .5
attach(EPA.94b.tccb.df)
TcCBref = TcCB[Area="Reference"]

lnxy = lnormPlot(TcCBref)
mtext("Reference area",line=.25,cex=.8)
title(sub=paste("Estimated parameters: meanlog = ",round(lnxy$meanlog,2),", sdlog = ",round(lnxy$sdlog,2),"\n")}
3. Simulate samples from a gamma distribution and obtain upper confidence limits for the lower quartile. Compare with upper confidence limits using quantile estimate for lognormal.

```r
require(EnvStats)

n = 80 # sample size
N = 4000 # number of samples
p = .25 # lower quartile
alpha = .05
shp = .5 # shape parameter for gamma
sc = 2 # scale parameter for gamma
qn = qgamma(p, shape=shp, scale=sc)
X = matrix(rgamma(n*N, shape=shp, scale=sc),n,N) # matrix of simulated data

upL = function(x,p,alpha) {
  xe = eqgamma(x, p=p, ci=TRUE, ci.type="upper", conf.level=1-alpha)
  return(xe$interval$limits[2])
}
upLln = function(x,p,alpha) {
  xe = eqlnorm(x, p=p, ci=TRUE, ci.type="upper", conf.level=1-alpha)
  return(xe$interval$limits[2])
}
XupL = apply(X,2,upL,p,alpha)
XupLln = apply(X,2,upLln,p,alpha)
qpL = mean(qn <= XupL)
qpLln = mean(qn <= XupLln)
qp = c(qpL,qpLln)
names(qp) = c("eqgamma","eqlnorm")
cat(paste("Proportion of upper confidence limits greater than ",p," quantile:"),"
")
print(qp)

# repeat using lognormal simulated data
lmu = 3
lsig = 1
qln = qlnorm(p, lmu, lsig)
Xl = matrix(rlnorm(n*N, lmu, lsig),n,N) # matrix of simulated data
XlupL = apply(Xl,2,upL,p,alpha)
```

Nonparametric quantile estimators.

Some situations cannot be represented very well by any parametric model. In such situations we can estimate quantiles based solely on the order statistics. Let \( X_{(1)}, \ldots, X_{(n)} \) denote the ordered values of i.i.d. r.v.'s \( X_1, \ldots, X_n \) with d.f. \( F \) and let \( q_p \) denote the \( p \)th quantile of \( F \). Then

\[
Y_i = I(X_i \leq q_p), \quad 1 \leq i \leq n
\]

are i.i.d. Bernoulli(\( p \)) r.v.'s, and so

\[
P(X_{(r)} < q_p < X_{(s)}) = \sum_{i=r}^{s} \binom{n}{i} p^i (1-p)^{n-i}.
\]

A 1 - \( \alpha \) confidence interval for \( q_p \) can be obtained by finding

\[
r = \max\{k : \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i} \leq \alpha/2,
\]

\[
s = \min\{k : \sum_{i=k+1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \leq \alpha/2.
\]

A function to obtain this confidence interval is contained in the file http://www.utdallas.edu/~ammann/stat6390scripts/nonpar.ci.r

Note that if one of the limits in a two-sided confidence interval is NA, then we can't obtain a two-sided interval for the specified \( \alpha \). In that case we could switch to a 1-sided confidence interval.

Example. First generate a sample from \( N(100,20) \) and estimate the 90th percentile. Then replace 10% of the sample with Gamma r.v.'s with mean 200 and s.d. 100. Compare quantile estimates using normal, gamma, and nonparametric confidence intervals.
sig = 20
X = rnorm(n, mu, sig)
qn = qnorm(p,mu,sig)
norm.ci = eqnorm(X, p=p, conf.level=1-alpha, ci=TRUE)
npar.ci = nonpar.ci(X, p, alpha)
cat(paste("Actual quantile is:", round(qn,3)),"
")
cat(paste("Confidence interval based on normal dist:",
    paste(round(norm.ci$interval$limits,3),collapse=" "," ")),"
")
cat(paste("Nonparametric confidence interval:",
    paste(round(npar.ci,3),collapse=" "," ")),"
")
# now contaminate 10% with gamma
gmu = 200
gsig = 100
pcontam = .1
a = (gmu/gsig)^2
b = gsig^2/gmu
ng = round(pcontam*n)
Xg = rgamma(ng, shape=a, scale=b)
X[1:ng] = Xg
# obtain quantile for contaminated distribution
N = 1000
X0 = seq(min(X),max(X),length=N)
H = pcontam*pgamma(X0, shape=a, scale=b) + (1-pcontam)*pnorm(X0,mu,sig)
qncontam = X0[max(seq(N)[H <= p])]
# show Q-Q plots
qqnorm(X)
qqline(X)
tmdnorm(X)
lnormPlot(X)
lnormPlot(X, plot.type="T")
gammaPlot(X)
gammaPlot(X, plot.type="T")
norm.ci = eqnorm(X, p=p, conf.level=1-alpha, ci=TRUE)
lnorm.ci = eqlnorm(X, p=p, conf.level=1-alpha, ci=TRUE)
gamma.ci = eqgamma(X, p=p, conf.level=1-alpha, ci=TRUE)
npar.ci = nonpar.ci(X, p, alpha)
cat(paste("Actual quantile for contaminated dist is:", round(qncontam,3)),"
")
cat(paste("Confidence interval based on normal dist:",
    paste(round(norm.ci$interval$limits,3),collapse=" "," ")),"
")
cat(paste("Confidence interval based on lognormal dist:",
    paste(round(lnorm.ci$interval$limits,3),collapse=" "," ")),"
")
cat(paste("Confidence interval based on gamma dist:",
...
Now repeat this for lognormal and gamma distributions.

n = 100
p = .9
alpha = .05
# lognormal data
lmu = 5
lsig = 2
X = rlnorm(n, lmu, lsig)
qn = qlnorm(p, lmu, lsig)
# show Q-Q plots
qqnorm(X)
qqline(X)
tmdnorm(X)
lnormPlot(X)
lnormPlot(X, plot.type="T")
gammaPlot(X)
gammaPlot(X, plot.type="T")
norm.ci = eqnorm(X, p=p, conf.level=1-alpha, ci=TRUE)
lnorm.ci = eqlnorm(X, p=p, conf.level=1-alpha, ci=TRUE)
gamma.ci = eqgamma(X, p=p, conf.level=1-alpha, ci=TRUE)
npar.ci = nonpar.ci(X, p, alpha)
cat(paste("Actual quantile for lognormal dist is: ", round(qn,3)),"\n")
cat(paste("Confidence interval based on normal dist: ",
          paste(round(norm.ci$interval$limits,3),collapse=" ", ")),"\n")
cat(paste("Confidence interval based on lognormal dist: ",
          paste(round(lnorm.ci$interval$limits,3),collapse=" ", ")),"\n")
cat(paste("Confidence interval based on gamma dist: ",
          paste(round(gamma.ci$interval$limits,3),collapse=" ", ")),"\n")
cat(paste("Nonparametric confidence interval: ",
          paste(round(npar.ci,3),collapse=" ", ")),"\n")

# gamma data
shp = 2
sc = 5
X = rgamma(n, shape=shp, scale=sc)
qn = qgamma(p, shape=shp, scale=sc)
# show Q-Q plots
qqnorm(X)
qqline(X)
\text{tmdnorm}(X) \\
\text{gammaPlot}(X) \\
\text{gammaPlot}(X, \text{plot.type}="T") \\
\text{lnormPlot}(X) \\
\text{lnormPlot}(X, \text{plot.type}="T") \\
norm\_ci = \text{eqnorm}(X, p=p, \text{conf.level}=1-\alpha, \text{ci}=\text{TRUE}) \\
lnorm\_ci = \text{eqlnorm}(X, p=p, \text{conf.level}=1-\alpha, \text{ci}=\text{TRUE}) \\
gamma\_ci = \text{eqgamma}(X, p=p, \text{conf.level}=1-\alpha, \text{ci}=\text{TRUE}) \\
npar\_ci = \text{nonpar.ci}(X, p, \alpha) \\
cat(paste("Actual quantile for gamma dist is:", round(qn,3)),"\n") \\
cat(paste("Confidence interval based on normal dist:", \\
\phantom{\text{\text{paste}}}
\text{paste}(\text{round}(\text{norm.ci}\$\text{interval}\$\text{limits},3),\text{collapse}="","\"),"\n") \\
cat(paste("Confidence interval based on lognormal dist:", \\
\phantom{\text{\text{paste}}}
\text{paste}(\text{round}(\text{lnorm.ci}\$\text{interval}\$\text{limits},3),\text{collapse}="","\"),"\n") \\
cat(paste("Confidence interval based on gamma dist:", \\
\phantom{\text{\text{paste}}}
\text{paste}(\text{round}(\text{gamma.ci}\$\text{interval}\$\text{limits},3),\text{collapse}="","\"),"\n") \\
cat(paste("Nonparametric confidence interval:", \\
\phantom{\text{\text{paste}}}
\text{paste}(\text{round}(\text{npar.ci},3),\text{collapse}="","\"),"\n") \\

A more complete comparison of these three confidence intervals could be made by simulating a large number of samples from various distributions and comparing the coverage probabilities and widths of the intervals.

\textbf{Prediction and Tolerance Intervals}

Prediction and tolerance intervals are closely related to confidence intervals. These intervals are appropriate, for example, whenever a site must be monitored to compare its status to standard or background levels. A prediction interval is an interval constructed so that it will contain \( k \) future observations, or the mean of those observations, with specified confidence \( 1 - \alpha \). A future observation or sample mean that falls outside the prediction interval is evidence that conditions at the site have changed. A tolerance interval is constructed for a population of values so that the interval contains proportion \( \beta \) of the values. Since this interval is estimated from a random sample, then there also is a level of confidence \( 1 - \alpha \) associated with a tolerance interval that specifies the probability that the interval contains proportion \( \beta \) of the population.

\textbf{Prediction intervals for normal distributions.}

Let \( X_1, \ldots, X_n \) be i.i.d. r.v.'s with distribution \( N(\mu, \sigma) \). Suppose the parameters are known and \( Y_1, \ldots, Y_k \) are i.i.d. future observations from the same distribution. Let \( z_p \) be the \( p^{th} \) quantile from the standard normal distribution where \( p \) is to be determined.

\[
P(\bigcap_{i=1}^{k}\{Y_i \in [\mu - z_p \sigma, \mu + z_p \sigma]\}) = \left[ P(Y_1 \in [\mu - z_p \sigma, \mu + z_p \sigma]) \right]^k = [1 - 2(1 - p)]^k = [2p - 1]^k,
\]
Therefore, setting
\[ 1 - \alpha = [2p - 1]^k \]
and solving for \( p \) gives
\[ p = [1 + (1 - \alpha)^{1/k}]/2. \]
In the case \( k = 1 \) this reduces to \( p = 1 - \alpha/2 \). However, if the parameters must be estimated, then adjustments associated with the t-distribution must be made. In addition, events in the intersection are not independent if \( k > 1 \).

First consider the case \( k = 1 \) and replace \( \mu, \sigma \) by their sample estimates, \( \bar{X}, s_X \). Let \( c \) denote a constant to be determined. Then
\[
P(\bar{X} - cs_X \leq Y_1 \leq \bar{X} + cs_X) = P\left(-c \leq \frac{Y_1 - \bar{X}}{s_X} \leq c\right).
\]
We know that \( Y_1 - \bar{X} \) and \( s_X \) are independent, that \( Y_1 - \bar{X} \) is normally distributed with
\[
E(Y_1 - \bar{X}) = 0,
\]
\[
\text{Var}(Y_1 - \bar{X}) = \sigma^2 + \sigma^2/n = \sigma^2(n + 1)/n.
\]
Therefore,
\[
\frac{Y_1 - \bar{X}}{\sigma} \sqrt{\frac{n}{n + 1}}
\]
has a standard normal distribution. This implies that
\[
\frac{Y_1 - \bar{X}}{s_X} \sqrt{\frac{n}{n + 1}}
\]
has a t-distribution with \( n - 1 \) d.f. Hence, the required prediction interval is obtained by setting
\[
c = t_{1-\alpha/2} \sqrt{(n + 1)/n},
\]
to give the interval
\[
\bar{X} \pm t_{1-\alpha/2}s_X \sqrt{(n + 1)/n}
\]
An upper prediction interval is given by
\[
(-\infty, \bar{X} + t_{1-\alpha}s_X \sqrt{(n + 1)/n}]
\]
and a lower prediction interval is
\[
(\bar{X} - t_{1-\alpha}s_X \sqrt{(n + 1)/n}, \infty)
\]
If \( k > 1 \) then we can use the Bonferroni inequality to approximate the prediction interval. Let \( A_i \) denote events with \( P(A_i) = 1 - \alpha_0 \). Then

\[
P(\bigcap_{i=1}^{k} A_i) = 1 - P(\bigcup_{i=1}^{k} A_i^c) \\
\geq 1 - \sum_{i=1}^{k} P(A_i^c) \\
= 1 - k\alpha_0.
\]

Therefore, a \( 1 - \alpha \) prediction interval for \( k \) future observations is given by

\[
\bar{X} \pm t_{1-\alpha/(2k)} s_X \sqrt{(n + 1)/n}
\]

Now suppose we need a prediction interval for the mean of \( k \) future observations,

\[
P(\bar{X} - cs_X \leq \bar{Y} \leq \bar{X} + cs_X) = P\left(-c \leq \frac{\bar{Y} - \bar{X}}{s_X} \leq c\right) = 1 - \alpha
\]

Then \( \bar{Y} - \bar{X} \) and \( s_X \) are independent, \( \bar{Y} - \bar{X} \) is normally distributed with

\[
E(\bar{Y} - \bar{X}) = 0,
\]

\[
\text{Var}(\bar{Y} - \bar{X}) = \frac{\sigma^2}{k} + \frac{\sigma^2}{n} = \sigma^2 \frac{n + k}{nk}.
\]

This implies that

\[
\frac{\bar{Y} - \bar{X}}{s_X} \sqrt{\frac{nk}{n+k}}
\]

has a t-distribution with \( n - 1 \) d.f. The corresponding prediction interval is

\[
\bar{X} \pm t_{1-\alpha/2} s_X \sqrt{(n + k)/nk}
\]

These methods are implemented in the EnvStats library by the function \textit{predIntNorm}.

If the population is not normally distributed but a monotone transformation exists such that the transformed variables are approximately normal, then the methods described above can be applied to the transformed variables and then the resulting intervals can be transformed back to the original scale. Since the lognormal (log transform) and the gamma (power transform) distributions satisfy this requirement, prediction intervals for those distributions can be derived accordingly. This approach is implemented in the EnvStats library by functions \textit{predIntLnorm}, \textit{predIntGamma}. Note however that a prediction interval for the \textbf{mean} of \( k \) future observations on the original scale is not returned by these functions. In particular, for the lognormal the interval corresponds to a prediction interval for

\[
\exp\left\{\frac{1}{k} \sum_{i=1}^{k} \log(Y_i)\right\} = \left[\prod_{i=1}^{k} Y_i\right]^{1/k}
\]
which is the geometric mean of the $k$ future observations. In the case of the gamma distribution, the prediction interval corresponds to

$$\left[ 1 \frac{1}{k} \sum_{i=1}^{k} Y_i^p \right]^{1/p}.$$  

**Simultaneous prediction intervals.**

Some site monitoring situations require taking observations at multiple locations within the site. The problem then is similar to the problem of multiple comparisons in which experimentwise Type I errors must be controlled. For example, suppose we must monitor a site for ground-water contamination using 4 different wells with 3 different pollutants measured quarterly for each well. Prediction intervals are to be obtained from a reference site for each pollutant. If 95% upper prediction intervals are obtained for each pollutant, the probability that at least 1 of the 12 comparisons would fall outside these intervals when the population is actually the same as the reference population would be much higher than 5%.

For the case of 4 wells and 3 pollutants measured quarterly at each well, a decision must be made after sampling whether the site is in compliance. Use of a Bonferroni correction with $1 - \alpha/12$ prediction intervals typically would be overly conservative and result in reduced power to detect contamination. On the other hand, use of uncorrected confidence levels for the prediction intervals would result in higher rates of Type I errors: deciding the site is contaminated when in fact it is not. One way to balance these errors is to utilize a sequential approach. For example, suppose one of the pollutants at one well is higher than its upper prediction limit. Instead of declaring the site contaminated, we wait until the next sampling time. If the same pollutant at the same well is now under the upper prediction limit, we do not declare that the site is contaminated, but if it is, then we do decide that that site is contaminated. This is referred to as a 1-out-of-2 rule. EPA guidance documents suggest using adjusted confidence levels for each sampling period determined by

$$1 - \alpha_0 = (1 - \alpha)^{1/n},$$

where $n$ is the number of comparisons (12 in this example), $1 - \alpha$ is the desired overall level of confidence, and $1 - \alpha_0$ is the level of confidence used for the individual prediction intervals. The *EnvStats* library includes a function for simultaneous prediction intervals for such situations: *predIntNormSimultaneous()*.  

Simultaneous prediction intervals for lognormal and gamma distributions are derived from normal approximations for appropriate transformations in the same way ordinary prediction intervals for those distributions were derived.

**Tolerance intervals for normal distributions.**

If the mean and s.d. are known, then a $\beta$ tolerance interval for the population would be

$$\mu \pm q_{\beta/2} \sigma.$$  

However, since these parameters ordinarily would need to be estimated from data, the corresponding tolerance intervals would be random. There are two ways this randomness can be handled.
• A $\beta$-content tolerance interval with confidence $1 - \alpha$. This interval can be interpreted as follows. The probability that the interval covers at least 100$\beta$% of the population is $1 - \alpha$.

• A $\beta$-expectation tolerance interval. This is an interval $(c_l, c_u)$ such that

$$P(c_l < X < c_u) = \beta.$$ 

Note that a $\beta$-expectation tolerance interval is equivalent to a prediction interval for one future observation with confidence level $\beta$, so we will consider here derivation of $\beta$-content tolerance intervals. Suppose $X_1, \cdots, X_n$ are i.i.d. r.v.'s with d.f. $F$. We must find functions $L = L(X_1, \cdots, X_n)$ and $U = U(X_1, \cdots, X_n)$ such that

$$P(F(U) - F(L) \geq \beta) = 1 - \alpha.$$ 

First consider an upper tolerance interval,

$$P(F(U) \geq \beta) = 1 - \alpha.$$ 

If $F$ is $N(\mu, \sigma)$, then this interval is equivalent to

$$1 - \alpha = P(U \geq F^{-1}(\beta)) = P(U \geq \mu + q_\beta \sigma),$$

where $q_\beta$ is the $\beta$-quantile from the standard normal distribution. Suppose we set

$$U = \bar{X} + Ks$$

for some $K$. Then $K$ must satisfy

$$P(\bar{X} + Ks \geq \mu + q_\beta \sigma) = 1 - \alpha.$$ 

Therefore,

$$1 - \alpha = P(\bar{X} + Ks \geq \mu + q_\beta \sigma)$$

$$= P\left(\frac{\bar{X} - \mu - q_\beta \sigma}{s/\sqrt{n}} \geq -K \sqrt{n}\right)$$

$$= P\left(\frac{Z - q_\beta \sqrt{n}}{\sqrt{V/df}} \geq -K \sqrt{n}\right),$$

where

$$Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \sim N(0, 1),$$

$$V/df = \frac{s^2}{\sigma^2} \sim \chi^2_{n-1}.$$
Since \( Z \) and \(-Z\) have the same distribution, then

\[
1 - \alpha = P\left( \frac{-Z - q_\beta \sqrt{n}}{\sqrt{V/df}} \geq -K \sqrt{n} \right)
\]

\[
= P\left( \frac{Z + q_\beta \sqrt{n}}{\sqrt{V/df}} \leq K \sqrt{n} \right)
\]

\[
= P(T \leq K \sqrt{n}),
\]

where \( T \) has a non-central t-distribution with \( n - 1 \) d.f. and non-centrality parameter \( \delta = q_\beta \sqrt{n} \).

This implies

\[
K = qt(1 - \alpha, n - 1, q_\beta \sqrt{n})/\sqrt{n}.
\]

Derivation of a lower tolerance interval for the normal distribution is similar and gives

\[
L = \bar{X} - Ks
\]

for the same \( K \).

A two-sided tolerance interval is more complicated to derive because it requires solution of an integral equation. This can be accomplished by numerical integration or approximate solutions obtained by Wald-Wolfowitz can be utilized. These methods are implemented in the EnvStats library by function tolIntNorm. The tolerance library provides a more complete package of tolerance intervals for many continuous and discrete distributions as well as nonparametric tolerance intervals.

**Functions for tolerance intervals in packages tolerance and EnvStats**
<table>
<thead>
<tr>
<th>Distribution</th>
<th>tolerance</th>
<th>EnvStats</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binomial</td>
<td>bintol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Cauchy</td>
<td>cautol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Exponential</td>
<td>exptol.int</td>
<td>tolIntGamma, tolIntGammaAlt</td>
</tr>
<tr>
<td>2- param Exponential</td>
<td>exp2tol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Weibull</td>
<td>exttol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Gamma</td>
<td>gamtol</td>
<td>tolIntGamma, tolIntGammaAlt</td>
</tr>
<tr>
<td>Hypergeometric</td>
<td>hypertol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Laplace</td>
<td>laptol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Logistic</td>
<td>logistol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Lognormal</td>
<td>normtol.int</td>
<td>tolIntNorm</td>
</tr>
<tr>
<td>Negative Binomial</td>
<td>negbintol.int</td>
<td>tolIntLnorm, tolIntLnormAlt</td>
</tr>
<tr>
<td>Normal</td>
<td>normtol.int</td>
<td>tolIntNorm</td>
</tr>
<tr>
<td>Multivariate Normal</td>
<td>mvtol.region</td>
<td>NA</td>
</tr>
<tr>
<td>Negative Hypergeometric</td>
<td>neghypertol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Nonparametric</td>
<td>nptol.int</td>
<td>tolIntNpar</td>
</tr>
<tr>
<td>Pareto</td>
<td>paretotol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Poisson</td>
<td>poistol.int</td>
<td>tolIntPois</td>
</tr>
<tr>
<td>Poisson-Lindley</td>
<td>poislindtol.int</td>
<td>NA</td>
</tr>
<tr>
<td>Uniform</td>
<td>uniftol.int</td>
<td>NA</td>
</tr>
</tbody>
</table>

In addition to those listed, `tolerance` includes functions to obtain tolerance intervals for ANOVA, regression, the difference between two independent normal r.v.’s, and Bayesian normal models. `EnvStats` includes functions for tolerance intervals with censored normal and lognormal data. Details can be found in the help pages for these packages.

**Example.** `EnvStats` contains the data frame `EPA.94b.tccb.df`. Suppose we would like to use the Reference area data to construct an upper 80% $\beta$-content tolerance limit with confidence level of 95% for $TcCB$ in the Cleanup area. Note that this is equivalent to finding an upper 95% confidence limit for the .8 quantile of the Reference area.

```r
library(EnvStats)
library(tolerance)
source("http://www.utdallas.edu/~ammann/stat6390scripts/tmdplot.r")
refndx = EPA.94b.tccb.df$Area == "Reference"
X = EPA.94b.tccb.df$TcCB[refndx]
Xname = "EPA.94b.tccb.df Reference area"
## first check Q-Q plots
par(mfrow=c(1,2))
qqnorm(X, pch=19)
qqline(X)
mtext(paste("Data:", Xname),line=.25)
tmdnorm(X, pch=19)
mtext(paste("Data:", Xname),line=.25)
lnormPlot(X, pch=19)
```
Data: Xname

lnormPlot(X, plot.type="T", pch=19)

data: Xname

gammaPlot(X, pch=19)

data: Xname

gammaPlot(X, plot.type="T", pch=19)

data: Xname

### both lognormal and gamma fit reasonably well

### compare qqnorm of log-transformed and power-transformed data

tmdnorm(log(X), pch=19)

qqnorm(log(X), pch=19)

tmdnorm(X^p, pch=19)

qqnorm(X^p, pch=19)

### both distributions fit reasonably well

### in such cases use lognormal since its tolerance intervals can be computed exactly

### results for lognormal

XLtol = tolIntLnormAlt(X, coverage=.8, ti.type="upper")

print(XLtol)

XLtol1 = normtol.int(X, P=.8, log.norm=TRUE)

print(XLtol1)

### both functions give same results since intervals are exact

### compare results with gamma

Xtol = tolIntGammaAlt(X, coverage=.8, ti.type="upper")

print(Xtol)

Xtol1 = gamtol.int(X, P=.8)

print(Xtol1)

### find proportion in Cleanup area below the upper limit using lognormal tolerance limits

clndx = EPA.94b.tccb.df$Area == "Cleanup"

Xc = EPA.94b.tccb.df$TcCB[clndx]

UTL = XLtol$interval$limits["UTL"]

pXc80 = mean(Xc <= UTL)

pX80 = mean(X <= UTL)
cat(paste("Upper 80% tolerance limit =",round(UTL,4)),"
")
cat(paste(round(100*pX80,2),"% of Reference area observations are below the limit",sep=""),
cat(paste(round(100*pXc80,2),"% of Cleanup area observations are below the limit",sep=""),
### QQ plots of Cleanup area show those observations are heavier-tailed than both lognormal and gamma models. These results show that Cleanup area still has some pollution problems.

**Hypothesis Testing**

**Background**

Hypothesis tests are the formal mechanism by which we make statistical decisions. The way in which these tests are constructed for environmental studies is critically important as will be seen. We will first consider tests that require deciding which of two possible hypotheses is true. Later we will examine multi-decision problems.

A hypothesis test involves deciding which of two or more possible actions to take for a specified population. This decision is based on a random sample from the population and therefore is subject to error. Suppose for example we must decide whether or not to grant a permit to a factory that allows the factory to discharge water used in its manufacturing process into a river. The possible actions are to issue the permit or do not issue the permit. The Clean Water Act states that no discharge of toxic substances in toxic amounts is allowed. So suppose there is an upstream reference site available and we base this decision on survival of a particular species of aquatic insect exposed to samples of upstream water and samples of the factory discharge. Details of such tests, called Whole Effluent Toxicity tests, will be discussed later.

There are three attributes associated with statistical tests for this situation.

1. Risk (or cost) of deciding to issue the permit when the discharge violates the CWA.
2. Risk (or cost) of deciding not to issue the permit when the discharge does not violate the CWA.
3. Amount (or cost) of data used to make the decision.

The fundamental problem here is that these attributes are inter-related. We can impose constraints on two of them but then the third will be a function of the other two and not controllable. For example, we could limit the amount of data and make our decision so that the risk of issuing the permit when the discharge violates the CWA is small. But then the risk of not issuing the permit when the discharge is ok might be large depending on the constraints imposed on the other attributes and the efficiency of the methods used. On the other hand, if we limit the amount of data and make our decision to keep small the risk of not issuing the permit when the discharge is ok, then the risk of issuing the permit when the discharge violates the CWA could be high depending on the constraints and efficiency of our methods. Finally, we could constrain both risks, but that only can be done if we place no constraints on the amount of data required to make our decision. In practice we
almost always have a constraint on the amount of data used, so the first step in such decision problems is to decide which of the two risks we need to constrain.

This decision problem is analogous to judicial decision problems in which a person accused of a crime is brought to trial and the jury must decide whether or not to convict the defendant. This decision problem also has three inter-related attributes.

1. Risk of deciding to convict when the defendant is innocent.
2. Risk of deciding not to convict when the defendant is guilty.
3. Time allowed to make a decision.

It is possible to make judicial decisions in such a way that a guilty defendant is always convicted, but the only way that can be done is to convict everyone. Of course, that guarantees every innocent defendant will be convicted. Likewise, decisions can be made in a way that guarantees no innocent defendant is ever convicted, but the only way that can be done is not to convict anyone. We have seen in the case of DNA evidence that technology exists today for extracting forensic information from DNA samples that was not available previously. Suppose a crime was committed 25 years ago that included DNA evidence which could have identified the perpetrator, but it couldn’t be examined with the technology of that time. If we had no constraint on the time allowed to make a decision, then we could have waited until such technology was available which then would have made both risks very small. But then we must decide what to do with the defendant while we wait for technology to advance. If the defendant is allowed to be free while we wait, then that would be equivalent to a decision not to convict. If the defendant is put in jail, then that would be equivalent to a decision to convict, especially if the time required is longer than the defendant lives. So the only reasonable decision process places a limit on the time to make a decision. But this implies that some innocent defendants will be convicted and some guilty defendants will not be convicted.

Every system for judicial decision-making must make the fundamental choice about which risk to keep small. The U.S. constitution places a limit on the time allowed to bring a defendant to trial and requires that decisions are made in such a way that the risk of convicting an innocent defendant is kept small. This implies that we cannot control the risk of not convicting a guilty defendant. That risk will depend on how low we require the first risk to be along with the efficiency with which evidence is obtained and the reliability of that evidence.

After all the evidence is presented and rebutted, jurors must make a decision. The evidence could be strong for guilt, weak for guilt, inconclusive, weak for innocent, or strong for innocent. To ensure that the risk of convicting an innocent defendant is kept low, jurors should only decide to convict if they believe the evidence is strong for guilt. If the evidence is only weak for guilt or inconclusive, the jury cannot convict since doing so would result in a higher risk of convicting an innocent defendant. This has two major implications. First, until the jury decides to convict, we cannot treat the defendant as if he/she were guilty. That is, we must presume the defendant is innocent. Only if the jury decides to convict can
we act as if the defendant is guilty. Secondly, since we must presume innocence, the strength of evidence for guilt must be evaluated under the assumption of innocence. Note that our constitution does not require that jurors be absolutely convinced that the defendant is guilty before returning a decision to convict. It only requires that this risk is reasonably small. The constitution does not define reasonably small, that is left to individual jurors. Finally, note that under this system a defendant does not need to prove innocence, and if a jury decides not to convict, that does not imply the jury believes the defendant is innocent. It simply means that their doubt about a decision to convict is not reasonably small. The prosecution has the burden of proof to convince jurors that the doubt about guilt is less than reasonably small.

<table>
<thead>
<tr>
<th>Decision</th>
<th>Innocent</th>
<th>Guilty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquit</td>
<td>✓</td>
<td>Type II error</td>
</tr>
<tr>
<td>Convict</td>
<td>Type I error</td>
<td>✓</td>
</tr>
</tbody>
</table>

Type I error refers to the error in which an innocent defendant is convicted. This is the error whose risk must be reasonably small. Type II error refers to the error in which a guilty defendant is not convicted. We don’t have direct control over the risk of that error.

The choice about which risk to control is critical. If a system of justice is setup to control the time allowed to make a decision and control the risk of not convicting a guilty defendant, then everything in the previous paragraph would be reversed. The defendant would be presumed guilty and would have the burden of proof to show innocence. The defendant would not be convicted only if the doubt concerning innocence was reasonably small. In the table above the error types would be reversed.

The judicial decision process hinges on the concept of reasonably small doubt, yet that is never defined for jurors. During the jury selection process judges and lawyers only give vague hints about what it means. It can, however, be described in statistical terms. Since the jury must presume innocence, then it must evaluate the evidence by asking how likely it would be to observe the evidence for guilt that has been presented during the trial if the defendant is innocent. Suppose a juror decides that there is just 1 chance in 1000 that the evidence would occur against an innocent defendant. If the juror believes that this chance is reasonably small, then the juror would make the decision to convict. This implies that the juror considers convicting 1 innocent defendant out of every 1000 innocent defendants to be a reasonably small rate for that error. Now suppose the juror decides that there is 1 chance in 10 that the incriminating evidence could accrue against an innocent person. If a juror believes that convicting 1 innocent defendant out of every 10 innocent defendants is not reasonably small, that that juror would vote not to convict. That vote does not imply that the juror believes the defendant is innocent, it just means the doubt is not small enough to convict. Therefore, reasonably small doubt concerning a decision to convict is equivalent to what would be considered reasonably small for the rate at which innocent defendants are convicted.

Now let’s consider the statistical decision problem of whether or not to issue a permit for discharge into a receiving stream. Almost always there will be a constraint on time and
budget for the decision process. That implies we must decide which risk must be controlled, deciding to issue the permit when the discharge violates the CWA, or deciding not to issue the permit when the discharge does not violate the CWA. That is equivalent to deciding which of two hypotheses,

1. Discharge is safe
2. Discharge is toxic

will have the burden of proof. Note that this choice implies what action should be taken if the data turns out to be inconclusive. If the burden of proof falls on the factory to prove that the discharge is safe, then no permit would be issued if the data is inconclusive. If the burden of proof falls on EPA to prove the discharge is toxic, then a permit would be issued if the data is inconclusive.

There are several components to an integrated approach for water quality-based control of toxicity. One component involves control of specific chemicals that are known hazards to human health. Chemicals that are toxic to other species also may be considered. Standards for allowable concentrations of these chemicals have been developed and the factory would need to prove that its discharge satisfies these standards. There are several limitations to this approach. It does not consider all toxins that may be present. Bioavailability of a chemical is not measured and interactions of mixtures of chemicals are not considered.

The second component is whole effluent toxicity (WET) testing. These tests are designed to measure the impact of the effluent on a few specific species that are surrogates for all organisms that depend on the receiving water. These include tracking reproduction rates of an aquatic insect, Ceriodaphnia dubia, and survival of fathead minnows that are exposed to a series of dilutions of the effluent. WET tests are designed to assess aggregate toxicity but do not provide direct human health protection. They measure bioavailability and can prevent impacts, but they do not identify causitive toxicants. Furthermore, since these tests are performed in laboratories rather than in situ, the lab effects may not scale directly to receiving water impacts. Standards for WET tests involve comparisons of reproduction and survival rates in dilutions of effluent to the rates using reference water. A WET test receives a pass unless it is shown that its rates are below reference rates. That is, the burden of proof for these tests is on the hypothesis that the effluent is toxic.

The third component is a bioassessment which measures actual receiving water effects on species that depend on the water. However, it can be difficult to separate effects of the effluent from other sources that may be available. Bioassessment also describes impacts after they have occurred.

Hypothesis tests for a population proportion

This section only considers large sample tests based on normal approximations to the sampling distributions

We only will consider fixed sample problems here which implies one of the hypotheses must be designated as the hypothesis with burden of proof. This hypothesis is referred
to as the alternative hypothesis \((H_1)\) and corresponds to the hypothesis of guilt in a trial. The other hypothesis, called the null hypothesis \((H_0)\), is assumed to be true initially. This choice for the null and alternative hypotheses is equivalent to the requirement that the risk of believing the alternative when the null hypothesis is true is kept reasonably small. This error is referred to as the Type I error. We only reject the initial assumption if the data strongly supports the alternative hypothesis. Since we must evaluate the data under the assumption that \(H_0\) is true, we make a decision by the following steps.

1. Express research question as a choice between two hypotheses.

2. Decide which hypothesis has the burden of proof. This becomes \(H_1\) and the other hypothesis becomes \(H_0\) and is assumed to be true initially.

3. Set a standard that defines reasonably small for the chance of making a Type I error. This probability is referred to as the level of significance of the test.

4. Obtain a measure of how far the data is from \(H_0\) with a test statistic whose sampling distribution under \(H_0\) is known and differs from the sampling distribution under \(H_1\).

5. Obtain the appropriate quantile from the sampling distribution such that the probability is no more than \(\alpha\) (level of significance) that a value from this distribution would be at least as far away as that quantile. This quantile is referred to as the critical value and the set of values farther away is referred to as the critical region. Note that if \(H_0\) is true, then the probability the test statistic falls in the critical region is \(\alpha\). Since \(\alpha\) is our definition for reasonably small probability of making a Type I error, if the test statistic is in the critical region, then our decision would be to reject \(H_0\). That decision will be wrong if the test statistic is in the critical region given that \(H_0\) is true, and so

\[
P(\text{Type I error}|H_0) = P(T \in \text{CR}|H_0) \leq \alpha,
\]

where \(T\) is the test statistic.

6. Additional information is contained in the p-value of the data, also referred to as the observed level of significance. The p-value represents the probability under the null hypothesis of observing values farther away from \(H_0\) than the test statistic. Therefore, an equivalent way of making the decision is to reject \(H_0\) if the p-value is no more than \(\alpha\).

Suppose we observe concentrations of a chemical at \(n\) randomly selected locations within a large region, \(X_1, \ldots, X_n\). Suppose standards for this chemical are defined so that it is considered safe if the concentration does not exceed \(C_s\). Let \(\theta\) denote the proportion of all locations at which concentration of this chemical exceeds \(C_s\). Define

\[
Y_i = I\{X_i > C_s\}, \ 1 \leq i \leq n.
\]
Then
$$\hat{\theta} = \frac{1}{n} \sum_{i=1}^{n} Y_i$$
is the sample proportion and is an estimator of $\theta$. Now suppose the goal for stage 1 of toxicity reduction of this chemical in the region is to have fewer than 20% of locations with concentrations that exceed the safe level. Also suppose that before moving to stage 2, the region must show strong evidence that this goal has been met using 10% level of significance. What decision should be made if 40 of 250 locations exceed the safe level?

1. *Express research question as a choice between two hypotheses.*
   These are $\theta < .20$ and $\theta \geq .20$.

2. *Decide which hypothesis has the burden of proof.*
   We must show strong evidence that $\theta < .20$, so this is the alternative hypothesis.
   This gives
   $$H_0: \theta \geq .20, \quad H_1: \theta < .20.$$  

3. *Specify a standard that defines reasonably small for the chance of making a Type I error.*
   Level of significance is 10%.

4. *Obtain a measure of how far the data is from $H_0$.*
   The Central Limit Theorem for proportions states that if $\theta = \theta_0$, then the distribution of
   
   $$\frac{\hat{\theta} - \theta_0}{\sqrt{\theta_0(1-\theta_0)/n}}$$

   is approximately standard normal. Therefore, under $H_0$,
   
   $$T = \frac{\hat{\theta} - .20}{\sqrt{(.20)(1-.20)/250}}$$

   is approximately standard normal. Negative values of $T$ give some evidence for the alternative hypothesis.

5. *Obtain the appropriate quantile from the sampling distribution for $\alpha = .10$.*
   Critical value is given by

   $$v = qnorm(.10)$$
so we reject \( H_0 \) if \( T \leq v = -1.282 \). Since \( \hat{\theta} = 40/250 = 0.16 \), then

\[
T = \frac{.16 - .20}{\sqrt{(.20)(1 - .20)/250}} = -1.581,
\]

so our decision at the 10\% level of significance is to reject \( H_0 \). The critical value for \( T \) is equivalent to the critical value

\[
.20 - 1.282\sqrt{(.20)(1 - .20)/250} = 0.1676
\]

for \( \hat{\theta} \).

6. p-value is given by

\[
\text{pnorm}(-1.581)
\]

This is equivalent to

\[
\text{pnorm}(.16, .20, \sqrt{.2*(1-.2)/250})
\]

The p-value is 0.057. If we make the decision to reject \( H_0 \), then the probability of making a Type I error would be 0.057. Since our definition of reasonably small for the risk of making that error is 0.10, then the p-value is reasonably small and we can reject \( H_0 \).

Even though emphasis of a hypothesis test is to control the probability of making a Type I error, we also must consider the probability of making a Type II error. To that end, we define the power function of a test to be the probability of rejecting \( H_0 \) as a function of \( \theta \),

\[
\Pi(\theta) = P_\theta(T \in \text{CR}),
\]

where \( P_\theta \) represents the sampling distribution of \( T \) when \( \theta \) is the true value of the parameter. Note that \( \Pi(\theta) \) for the example is a monotone decreasing function of \( \theta \) and

\[
\Pi(\theta) \leq \alpha, \ \theta \geq \theta_0 = .20,
\]

\[
\Pi(\theta) > \alpha, \ \theta < \theta_0.
\]

Tests that satisfy these properties are called unbiased. Note that if \( \theta_1 \in H_1 \), then \( 1 - \Pi(\theta_1) \) is the probability of making a Type II error at \( \theta_1 \).

In the example above,

\[
\Pi(\theta) = P_\theta(T \in \text{CR}) = P_\theta(\hat{\theta} \leq 0.1676) \approx P \left( Z \leq \frac{0.1676 - \theta}{\sqrt{\theta(1 - \theta)/n}} \right)
\]
In general, for alternative hypothesis $\theta < \theta_0$ and level of significance $\alpha$, $H_0$ is rejected if
\[
\hat{\theta} \leq \theta_0 - z_{1-\alpha}\sqrt{\theta_0(1-\theta_0)/n}.
\]
The p-value is given by
\[
\text{pnorm}(\hat{\theta}, \theta_0, \sqrt{\theta_0(1-\theta_0)/n})
\]
and the power function is
\[
\Pi(\theta) = P\left(Z \leq \frac{\theta_0 - \theta}{\sqrt{\theta(1-\theta)/n}} - z_{1-\alpha} \sqrt{\frac{\theta_0(1-\theta_0)}{\theta(1-\theta)}}\right).
\]

Now suppose the burden of proof is reversed, that is, the alternative hypothesis is $\theta > \theta_0$. Then $H_0$ would be rejected if
\[
\hat{\theta} \geq \theta_0 + z_{1-\alpha}\sqrt{\theta_0(1-\theta_0)/n},
\]
and the p-value would be given by
\[
1 - \text{pnorm}(\hat{\theta}, \theta_0, \sqrt{\theta_0(1-\theta_0)/n}).
\]
The power function would be
\[
\Pi(\theta) = P\left(Z \geq \frac{\theta_0 - \theta}{\sqrt{\theta(1-\theta)/n}} + z_{1-\alpha} \sqrt{\frac{\theta_0(1-\theta_0)}{\theta(1-\theta)}}\right)
\]
which is a monotone increasing function of $\theta$.

These two tests are referred to as one-sided tests because the burden of proof is to show that $\theta$ is on one particular side of $\theta_0$. One-sided tests have directional hypotheses, e.g., \textit{show that $\theta$ is less than} $\theta_0$, or, \textit{show that $\theta$ is greater than} $\theta_0$. These hypotheses correspond to a decision between two actions depending on the value of $\theta$:
take action $A_1$ if $\theta < \theta_0$
take action $A_0$ if $\theta > \theta_0$
By convention, when stating hypotheses we put the equal sign with $H_0$.

Some research questions are non-directional, such as \textit{show that $\theta$ has changed}. In this case the critical region would be values far away from $\theta_0$ in either direction. In practice such non-directional questions would be appropriate if there were three potential actions to take depending on the value of $\theta$:
take action $A_{-1}$ if $\theta < \theta_0$,
take action $A_0$ if $\theta = \theta_0$,
take action $A_1$ if $\theta > \theta_0$.
In such cases we decide which action to take by a two-step process. First test the hypotheses
\[
H_0: \theta = \theta_0, \quad H_1: \theta \neq \theta_0.
\]
If the decision is to not reject $H_0$, then take action $A_0$. If the decision is to reject $H_0$, then take action $A_{-1}$ if $\hat{\theta} < \theta_0$ and take action $A_1$ if $\hat{\theta} > \theta_0$. This is referred to as a two-sided test.

The critical region of a two-sided test for a population proportion is given by

$$\frac{|\hat{\theta} - \theta_0|}{\sqrt{\theta_0(1 - \theta_0)/n}} \geq z_{1-\alpha/2}$$

This is equivalent to

$$\{\hat{\theta} \leq \theta_0 - z_{1-\alpha/2}\sqrt{\theta_0(1 - \theta_0)/n}\} \cup \{\hat{\theta} \geq \theta_0 + z_{1-\alpha/2}\sqrt{\theta_0(1 - \theta_0)/n}\}.$$

The p-value is

$$2 \ast (1 - \text{pnorm}(abs(\hat{\theta} - \theta_0), 0, \sqrt{\theta_0(1 - \theta_0)/n})).$$

The power function is

$$\Pi(\theta) = P\left(Z \leq \frac{\theta_0 - \theta}{\sqrt{\theta(1 - \theta)/n}} - z_{1-\alpha/2}\sqrt{\theta_0(1 - \theta_0)/\theta(1 - \theta)}\right) + P\left(Z \geq \frac{\theta_0 - \theta}{\sqrt{\theta(1 - \theta)/n}} + z_{1-\alpha/2}\sqrt{\theta_0(1 - \theta_0)/\theta(1 - \theta)}\right)$$

A link to an R function that returns power for tests of population proportions is here: http://www.utdallas.edu/~ammann/stat6390scripts/prop.power.r

This function can be used to plot the power functions for these tests. By default the function uses normal approximations, but the optional argument method can specify exact binomial probabilities by:

```r
method = "exact"
```

```r
source("http://www.utdallas.edu/~ammann/stat6390scripts/prop.power.r")
p0 = .2
n1 = 250
alpha = .10
p1 = seq(.10,.3,length=500)
### H1: p < .2
pwr1 = prop.power(n1, p0, p1, alpha = .1, alt="less")
plot(p1, pwr1,type="l", xlab=expression(theta), ylab="Power", col="blue", ylim=c(0,1))
abline(h=alpha, v=p0, lty=2, col="gray50")
title(expression(paste("Power function for ", H[1], ": ", theta < .20)))
# add power function for n=500
n2 = 500
pwr2 = prop.power(n2, p0, p1, alpha = .1, alt="less")
lines(p1, pwr2, col="orange")
legend(max(p1), .5, legend=paste("n =",c(n1,n2)),
   lty=1, col=c("blue","orange"), xjust=1, yjust=.5)
```
### H1: p > .2

```r
pwr1 = prop.power(n1, p0, p1, alpha = .1, alt="greater")
plot(p1, pwr1,type="l", xlab=expression(theta), ylab="Power", col="blue", ylim=c(0,1))
abline(h=alpha, v=p0, lty=2, col="gray50")
title(expression(paste("Power function for ",H[1],": ", theta > .20)))
```

Add power function for n=500:

```r
pwr2 = prop.power(n2, p0, p1, alpha = .1, alt="greater")
lines(p1, pwr2, col="orange")
legend(min(p1), .5, legend=paste("n =",c(n1,n2)),
       lty=1, col=c("blue","orange"), xjust=0, yjust=.5)
```

### H1: p ≠ .2

```r
pwr1 = prop.power(n1, p0, p1, alpha = .1, alt="two.sided")
plot(p1, pwr1,type="l", xlab=expression(theta), ylab="Power", col="blue", ylim=c(0,1))
abline(h=alpha, v=p0, lty=2, col="gray50")
title(expression(paste("Power function for ",H[1],": ", theta != .20)))
```

Add power function for n=500:

```r
pwr2 = prop.power(n2, p0, p1, alpha = .1, alt="two.sided")
lines(p1, pwr2, col="orange")
legend(min(p1), .5, legend=paste("n =",c(n1,n2)),
       lty=1, col=c("blue","orange"), xjust=0, yjust=.5)
```

Note that this script incorporates some of the `plotmath` capabilities of R.

The power function also can be used to determine observable effects. For example, if the alternative is $H_1: \theta < .20$, we can ask how far below 0.20 $\theta$ must be so that the probability of rejecting $H_0$ is high, say 0.85. An algebraic solution can be obtained from the power function given above by solving a quadratic equation,

$$a\theta^2 + b\theta + c = 0,$$

where

$$a = n + q^2_{1-\beta}$$
$$b = -(q^2_{1-\beta} + 2 \ast n \ast v)$$
$$c = n \ast v^2$$

and $v$ is the critical value for the test. An easier method is to invert the power function numerically.

```r
source("http://www.utdallas.edu/~ammann/stat6390scripts/prop.power.r")
p0 = .2
n1 = 250
alpha = .10
p1 = seq(.10,.3,length=500)
coln = c("blue","orange")
```
### H1: \( p < .2 \)

\[
pwr1 = \text{prop.power}(n1, p0, p1, \text{alpha} = .1, \text{alt}="less")
\]

\[
\text{plot}(p1, pwr1, \text{type}="l", \text{lab}=\text{expression}(\theta), \text{ylab}=\text{"Power"}, \text{col}=\text{coln}[1], \text{ylim}=c(0,1))
\]

\[
\text{abline}(h=\text{alpha}, v=p0, \text{ty}=2, \text{col}="\text{gray50"})
\]

\[
\text{title}(\text{expression}("\text{Power function for } \text{H}_1: \theta < .20"))
\]

\[
\text{mtext}(\text{expression}("\text{Observable effects for } \beta = 0.15"), \text{line}=.25)
\]

# add power function for n=500

\[
n2 = 500
\]

\[
pwr2 = \text{prop.power}(n2, p0, p1, \text{alpha} = .1, \text{alt}="less")
\]

\[
\text{lines}(p1, pwr2, \text{col}=\text{coln}[2])
\]

\[
\text{legend}(\text{max}(p1), .5, \text{legend}="\text{paste}("n =",c(n1,n2)), lty=1, \text{col}=\text{coln}, \text{xjust}=1, \text{yjust}=.5)
\]

\[
\beta = .15
\]

\[
\text{theta1} = \text{max}(p1[pwr1 >= 1-beta])
\]

\[
\text{theta2} = \text{max}(p1[pwr2 >= 1-beta])
\]

\[
\text{abline}(h=1-beta, lty=4, \text{col}="\text{gray50"})
\]

\[
\text{abline}(v=\text{theta1}, \text{col}=\text{coln}[1], \text{ty}=4)
\]

\[
\text{abline}(v=\text{theta2}, \text{col}=\text{coln}[2], \text{ty}=4)
\]

### obtain exact values for theta1, theta2 using normal approximation

\[
\text{qa} = \text{qnorm}(1-\text{alpha})
\]

\[
\text{qb} = \text{qnorm}(1-\beta)
\]

\[
\text{v1} = p0 - \text{qa}\ast\text{sqrt}(p0\ast(1-p0)/n1)
\]

\[
\text{a1} = n1 + \text{qb}\ast2
\]

\[
\text{b1} = -(\text{qb}\ast2 + 2\ast n1\ast v1)
\]

\[
\text{c1} = n1\ast v1\ast2
\]

\[
\text{theta1e} = (-\text{b1} + c(-1,1)\ast\text{sqrt(b1}\ast2 - 4\ast a1\ast c1))/(2\ast a1)
\]

\[
\text{theta1e} = \text{theta1e}[\text{theta1e} < \text{v1}]
\]

\[
\text{v2} = p0 - \text{qa}\ast\text{sqrt}(p0\ast(1-p0)/n2)
\]

\[
\text{a2} = n2 + \text{qb}\ast2
\]

\[
\text{b2} = -(\text{qb}\ast2 + 2\ast n2\ast v2)
\]

\[
\text{c2} = n2\ast v2\ast2
\]

\[
\text{theta2e} = (-\text{b2} + c(-1,1)\ast\text{sqrt(b2}\ast2 - 4\ast a2\ast c2))/(2\ast a2)
\]

\[
\text{theta2e} = \text{theta2e}[\text{theta2e} < \text{v2}]
\]

\[
\text{obs} = \text{matrix}(0,2,2)
\]

\[
\text{dimnames} = \text{list}("\text{Numerical","Exact"), paste("n =",c(n1,n2)))
\]

\[
\text{obs}[1,] = c(\text{theta1,theta2})
\]

\[
\text{obs}[2,] = c(\text{theta1e,theta2e})
\]

\[
\text{cat}(\text{paste("Observable effects for alpha = ",alpha," beta = ",beta,sep=""),"\n")}
\]

\[
\text{print(\text{round(obs,4)})}
\]

Another application of the power function is to determine sample sizes. Suppose the hypotheses are \( H_0: \theta \geq .20, H_1: \theta < .20 \) and we wish to test these hypotheses at 10% level of significance. Also suppose we want to determine the sample size required so that the test
will reject the null hypothesis with probability 0.90 if $\theta \leq 0.17$. That is, we want

$$\Pi(0.17) = 0.90.$$ 

This is obtained by solving

$$z_{1-\beta} = \frac{\theta_0 - \theta_1}{\sqrt{\theta_1(1-\theta_1)/n}} - z_{1-\alpha} \sqrt{\frac{\theta_0(1-\theta_0)}{\theta_1(1-\theta_1)}};$$

where $\theta_0 = 0.20$, $\theta_1 = 0.17$, $\alpha = .10$, $\beta = .10$, and so

$$n = \left[ \frac{z_{1-\beta} \sqrt{\theta_1(1-\theta_1)} + z_{1-\alpha} \sqrt{\theta_0(1-\theta_0)}}{\theta_0 - \theta_1} \right]^2.$$

This gives $n = 1098$. We can check by obtaining the power at that sample size,

```r
pwr = prop.power(1098,.2,.17,alpha=.1,alt="less")
print(pwr)
```

Although it is straightforward to obtain exact binomial probabilities and quantiles for the power function, obtaining observable effects and sample sizes is more complicated than the algebraic solutions when the normal approximation is used.

**Sign test.** One of the applications of hypothesis tests for proportions is the nonparametric sign test. Suppose for example we wish to test the hypotheses

$$H_0 : q_p \leq q_0, \quad H_1 : q_p > q_0$$

where $q_p$ is the $p^{th}$ quantile of some d.f. $F$. That is,

$$F(q_p) = p.$$ 

Let $X_1, \cdots, X_n$ denote a random sample from $F$ and set

$$Y_i = I\{X_i \leq q_0\}, \quad 1 \leq i \leq n.$$ 

Then $Y_1, \cdots, Y_n$ are i.i.d. Bernoulli r.v.'s and

$$S_n = \sum_{i=1}^{n} Y_i \sim \text{Binomial}(n, F(q_0)).$$

Let $\theta = F(q_0)$. Then under $H_0$, $p = F(q_p) \leq F(q_0) = \theta$ and under $H_1$, $p = F(q_p) > F(q_0) = \theta$, so the hypotheses are equivalent to a test for a binomial proportion:

$$H_0 : \theta \geq p, \quad H_1 : \theta < p.$$ 

Although the critical region for this test does not depend on $F$, the power function does. For small samples, the test statistic would be $S_n$ and critical values would come from the binomial distribution. For large samples we can use the sample proportion of observations
that are less than or equal to $q_0$ and use normal approximations to the power function as given above. The large sample power function of this one-sided sign test is given by

$$
\Pi(F) = P \left( Z \leq \frac{p - F(q_0)}{\sqrt{F(q_0)(1 - F(q_0))}/n} - z_{1-\alpha} \sqrt{\frac{p(1-p)}{F(q_0)(1 - F(q_0))}} \right).
$$

Suppose for example that $p = 0.5$ and $q_0 = 10$. Then $q_p$ corresponds to the median of $F$. Let $\hat{q}$ denote the sample proportion of observations that are less than or equal to 10 and suppose that $F$ is a Gamma distribution. The following R script generates a plot of power functions for a one-sided test of hypotheses

$$H_0 : \text{median} \leq 10, \ H_1 : \text{median} > 10 \iff H_0 : \theta \geq 0.5, \ H_1 : \theta < 0.5$$

when $n = 50$, $\alpha = 0.05$, and shape $= 0.5, 1, 2, 4$.

```R
source("http://www.utdallas.edu/~ammann/stat6390scripts/prop.power.r")
p = .5
n = 50
m0 = 10
alpha=.05
shp = c(.5,1,2,5)
b = seq(1,200,length=10001)
bcol = c("red","black","blue","cyan")
X.lim = c(8,30)
k = 1
med0 = qgamma(p,shape=shp[k],scale=b)
F = pgamma(m0,shape=shp[k],scale=b)
pwr = prop.power(n, p, F, alt="less", method="exact")
par(mar=c(5.1, 4.1, 4.1, 3.1))
plot(med0,pwr,type="l",xlab="Median",ylab="Power",xlim=X.lim,ylim=c(0,1),col=bcol[k])
title("Power Function of Sign Test for Gamma")
mtext(expression(paste(H[0],": \text{median} \leq 10, \ H[1],:\text{median} > 10 \iff H[0]: \theta \geq 0.5, \ H[1]: \theta < 0.5"),line=.25)
axis(4,at=alpha)
abline(h=alpha,v=m0,col="gray50",lty=3)
for(k in seq(2,length(shp))) {
    med0 = qgamma(p,shape=shp[k],scale=b)
    F = pgamma(m0,shape=shp[k],scale=b)
    pwr = prop.power(n, p, F, alt="less", method="exact")
    lines(med0,pwr,col=bcol[k])
}
legend(max(X.lim),2*alpha,legend=paste("Shape =",shp),lty=1,col=bcol,xjust=1,yjust=0,
title=paste("n =",n),cex=.8)
```

Now repeat for the lognormal distribution.
p = .5
n = 50
m0 = 10
alpha=.05
sig = c(.5,1,1.5,2)
bcol = c("red","black","blue","cyan")
med0 = seq(8,30,length=401)
X.lim = range(med0)
logm = log(med0)
k = 1
F = plnorm(m0, meanlog=logm, sdlog=sig[k])
pwr = prop.power(n, p, F, alt="less", method="exact")
par(mar=c(5.1, 4.1, 4.1, 3.1))
plot(med0,pwr,type="l",xlab="Median",ylab="Power",xlim=X.lim,ylim=c(0,1),col=bcol[k])
title("Power Function of Sign Test for Lognormal")
mtext(expression(paste(H[0],": median \leq 10, H[1],": median > 10 ")),line=.25)
axis(4,at=alpha)
abline(h=alpha,v=m0,col="gray50",lty=3)
for(k in seq(2,length(sig))) {
    F = plnorm(m0, meanlog=logm, sdlog=sig[k])
    pwr = prop.power(n, p, F, alt="less", method="exact")
    lines(med0,pwr,col=bcol[k])
}
legend(max(X.lim),2*alpha,legend=paste("sdlog =",sig),lty=1,col=bcol,xjust=1,yjust=0,
title=paste("n =",n),cex=.8)

Note that the power curves do not pass through the point (10,.05), but instead are slightly lower because exact binomial probabilities are used in the power function.

**Hypothesis tests for a population mean**

Let $\mu, \sigma$ denote the population mean and s.d. and we wish to test hypotheses about $\mu$, for example,

$$H_0 : \mu \leq \mu_0, \quad H_1 : \mu > \mu_0$$

based on i.i.d. observations $X_1, \cdots, X_n$ with $E(X_i) = \mu$, $Var(X_i) = \sigma^2$.

**Tests for the normal distribution.** The first case we will consider is the case in which observations are normally distributed. Then under the assumption that $\mu = \mu_0$ we have

$$T_n = \frac{\bar{X} - \mu_0}{s/\sqrt{n}} \sim t_{n-1}.$$ 

Therefore, we will reject $H_0$ if $T_n \geq t_{1-\alpha,n-1}$ where 

$$t_{1-\alpha,n-1} = qt(1 - \alpha, n - 1).$$
The p-value is given by

\[ 1 - pt(T_n, n - 1). \]

Critical values and p-values for the other sets of hypotheses are obtained similarly.

The power function for this test involves the non-central t-distribution. Let \( t_c \) denote the critical value of the test and let

\[ Q = (n - 1)s^2/\sigma^2. \]

Then

\[
\Pi(\mu) = P_{\mu}(T_n \geq t_c) = P\left( \frac{\sqrt{n}(\bar{X} - \mu_0)}{\sigma\sqrt{Q/(n - 1)}} \geq t_c \right) \\
= P\left( \frac{Z + \delta}{\sqrt{Q/(n - 1)}} \geq t_c \right) \\
= P(T \geq t_c),
\]

where

\[ \delta = \frac{\sqrt{n}(\mu - \mu_0)}{\sigma} \]

and \( T \) has a non-central t-distribution with non-centrality parameter \( \delta \). Therefore, for these hypotheses,

\[ \Pi(\mu) = 1 - pt(t_c, n - 1, \delta). \]

Note that this power function depends on the population s.d.

Suppose for example we wish to test

\[ H_0 : \mu \leq 50, \quad H_1 : \mu > 50 \]

based on \( n = 25 \) i.i.d observations with

\[ \bar{X} = 55, \quad s = 10. \]

The critical value for a test with 5% level of significance is

\[ tc = qt(.95, 24) \]

and the test statistic is

\[ T = \sqrt{25} \times \frac{(55 - 50)}{10} \]

Since \( tc = 1.711 \) and \( T = 2.5 \), then we reject \( H_0 \). The p-value for this test is
1 - pt(2.5,24)

If the raw data is available, then the R function `t.test` can be used.

The following code produces a plot of the power function of this test using the sample s.d. for $\sigma$ in the non-centrality parameter.

```r
n = 25
m0 = 50
sig = 10
alpha = .05
tc = qt(1-alpha,n-1)
m = seq(48,60,length=301)
delta = sqrt(n)*(m - m0)/sig
P = 1 - pt(tc,n-1,delta)
plot(P ~ m, xlab=expression(mu), ylab="Power", type="l")
abline(h=alpha, v=m0, col="red")
title(expression(paste("Power Function of T-test for ",H[1],": ",mu > 50)))
mtext(expression(paste("Assumes ",sigma == 10)), line=.25)
```

The EnvStats library contains the function `tTestPower()` that gives the power function. The following code gives an example of using this function.

```r
library(EnvStats)
n = length(X)
m0 = 50
t.test(X, alt="greater", mu=m0)
sig = 10
alpha = .05
m = seq(48,60,length=301)
delta = (m - m0)/sig
Pt = tTestPower(n, delta=delta, sample.type="one.sample", alt="greater")
plot(Pt ~ m, xlab=expression(mu), ylab="Power", type="l")
abline(h=alpha, v=m0, col="red")
title(expression(paste("Power Function of T-test for ",H[1],": ",mu > 50)))
mtext(expression(paste("Assumes ",sigma == 10)), line=.25)
```

Note that `t.test()` returns the p-value of the test along with a one-sided confidence interval since the specified alternative is one-sided. If you want a two-sided confidence interval, then you must specify two-sided for the alternative with appropriate confidence level. Also note that the `delta` argument in `tTestPower()` does not include the $\sqrt{n}$ term.

As was shown with hypothesis testing for proportions, the power function can be used to obtain observable differences and sample sizes. However, a prior bound or preliminary estimate of $\sigma$ must be available. For example, suppose we wish to find the smallest value of $\mu$ at which the power function is at least 0.80 for the previous test.
bpwr = .8
m.8 = min(m[Pt >= bpwr])

The value returned is 55.15. This implies that with a sample size of \(n = 25\) and assuming \(\sigma = 10\), the probability the test will reject \(H_0\) is at least 0.8 when the population mean is at least 55.15.

Now suppose we wish to perform a size 0.05 test of these hypotheses such that the probability of rejecting \(H_0\) is 0.95 when \(\mu\) is 54 under the assumption that \(\sigma = 10\). We know that this sample size must be greater than 25.

library(EnvStats)
n = seq(25,200)
m0 = 50
sig = 10
alpha = .05
m = 54
delta = (m - m0)/sig
Pt = tTestPower(n, delta=delta, sample.type="one.sample", alt="greater")
n1 = min(n[Pt >= .95])

This gives \(n_1 = 70\). We can check by

Pt1 = tTestPower(n1, delta=delta, sample.type="one.sample", alt="greater")

All of this requires the assumption that the population of interest has a normal distribution, so Q-Q plots must be examined to validate use of these methods.

Note that for normally distributed data, a test about the median is equivalent to a test about the mean. A comparison of the power functions of the t-test and sign test is given in the following R code to test hypotheses

\[
H_0 : \theta \leq 10,
H_1 : \theta > 10,
\]

where \(\theta\) is the mean (median) of a normally distributed population. This comparison will be based on a sample of size 50 and for s.d.'s \(\sigma = 2, 4, 10, 15\).

source("http://www.utdallas.edu/~ammann/stat6390scripts/prop.power.r")
library(EnvStats)
p = .5
n = 50
m0 = 10
alpha=.05
sig = c(2,4,10,15)
theta = seq(8,20,length=501)
bcol = c("black","red","blue","cyan")
k = 1
F = pnorm(m0,theta,sig[k])
pwr = prop.power(n, p, F, alt="less", method="exact")
del = (theta - m0)/sig[k]
pwrt = tTestPower(n, delta=del, sample.type="one.sample", alt="greater")
par(mar=c(5.1, 4.1, 4.1, 3.1))
plot(theta,pwr,type="l",xlab=expression(theta),ylab="Power",ylim=c(0,1),col=bcol[k],lty=2)
lines(theta,pwrt,col=bcol[k])
title("Power Functions of Sign Test and T-test\nfor Normal Distributions")
mtext(expression(paste(H[0],": \theta \leq 10, \ H[1],": \theta > 10")),side=1,line=-1)
axis(4,at=alpha)
abline(h=alpha,v=m0,col="gray50",lty=3)
for(k in seq(2,length(sig))) {
  F = pnorm(m0,theta,sig[k])
  pwr = prop.power(n, p, F, alt="less", method="exact")
  lines(theta,pwr,col=bcol[k],lty=2)
  del = (theta - m0)/sig[k]
  pwrt = tTestPower(n, delta=del, sample.type="one.sample", alt="greater")
  lines(theta,pwrt,col=bcol[k])
}
legend(max(theta),.5,legend=c("t-test","sign test"),lty=c(1,2),xjust=1,yjust=.5)
# build a vector of strings that will be parsed into expressions for plotmath to use as
ltxt = paste("sigma ==",sig)
legend(max(theta),alpha,legend=parse(text=ltxt),lty=1,col=bcol,xjust=1,yjust=-.1)

This script uses \texttt{plotmath} in the legend text to give values of $\sigma$. Previous applications of
\texttt{plotmath} involved a single expression like is used above in \texttt{mtext} for stating the hypotheses.
However, the \texttt{legend()} function requires that its \texttt{legend=} argument must be a vector of strings
or expressions. To use \texttt{plotmath} for a legend text that also includes parameter values, we
must first build a vector of strings whose elements contain the text of a \texttt{plotmath} expression.
Then the \texttt{R} function \texttt{parse()} converts this vector of strings into a vector of expressions that
now can be used for the \texttt{legend=} argument.

\begin{verbatim}
legend=parse(text=ltxt)
\end{verbatim}

\textbf{Tests for the lognormal distributions.} If the population follows a lognormal distribution
we cannot simply log-transform the data and then apply a t-test. Let $X_1,\ldots,X_n$ be
i.i.d. lognormal r.v.’s with mean $\theta$ and coefficient of variation $\tau$. Then $Y_i = \log(X_i)$ are i.i.d.
r.v.’s with mean $\mu$ and s.d. $\sigma$ where
\begin{align*}
\mu &= \log\left(\frac{\theta}{\sqrt{\tau^2 + 1}}\right) \\
\sigma &= \left[\log(\tau^2 + 1)\right]^{1/2}
\end{align*}
\[
\theta = \exp\{\mu + (\sigma^2/2)\}
\]
\[
\tau = [\exp\{\sigma^2\} - 1]^{1/2}.
\]
These relationships imply that if the hypotheses
\[
H_0 : \theta \leq \theta_0, \quad H_1 : \theta > \theta_0
\]
are tested by a t-test for
\[
H_0 : \mu \leq \mu_0, \quad H_1 : \mu > \mu_0
\]
using \(Y_1, \cdots, Y_n\), this would correspond to hypotheses on both \(\theta\) and \(\eta\).

One approach to dealing with this problem is to use confidence intervals designed for lognormal distributions, one-sided intervals for one-sided hypotheses, two-sided intervals for two-sided hypotheses. A similar problem exists for the gamma distribution. The functions \(elnormAlt\) and \(egammaAlt\) in the \(EnvStats\) library can be used for this. However, power functions for such tests are not available directly. Approximations for the lognormal require strong assumptions about the behavior of the coefficient of variation under alternatives. See help pages for \(tTestLnormAltPower\) and \(tTestLnormAltN\) for details.

Since these distributions are skewed, the mean may not be a good measure of location. As an alternative, we could test hypotheses about quantiles such as the median and use a nonparametric test like the sign test.

**Two-sample comparisons**

Many questions in environmental sciences involve comparison of two (or more) populations. For example, we may need to compare a remediation site to a reference site, or compare biological activity downstream of a discharger to biological activity upstream. For these problems there are two fundamental approaches to estimation and testing depending on how the populations are sample, paired samples and independent samples.

Paired samples occur when there is a direct link between an observation in one sample with one and only one observation in the other sample to form a pair. For example, one population could represent measurements at randomly selected locations before treatment followed by repeating those measurements after treatment. Another way to obtained paired samples to match by other factors or variables. Relevant information about differences between location parameters for two populations is contained in the pair differences. After obtaining pair differences, the problem and data have now been reduced to one sample and so one-sample methods can be applied to the differences.

The remainder of this section discusses two-sample comparisons with independent samples. The first case considered is testing for differences between means in normally distributed populations. Let \(X_i, 1 \leq i \leq n_x\) and \(Y_i, 1 \leq i \leq n_y\) denote independent random samples with \(X_i \sim N(\mu_x, \sigma_x)\) and \(Y_i \sim N(\mu_y, \sigma_y)\). Suppose we wish to test the null hypothesis,
\[
H_0 : \mu_x = \mu_y,
\]
versus \(H_1 : \mu_x > \mu_y\) or \(H_1 : \mu_x \neq \mu_y\).
The first solution to this problem assumes that the population variances are equal, 
\[ \sigma_x = \sigma_y = \sigma. \]
In that case, 
\[ Z = \frac{(\bar{Y} - \bar{X}) - (\mu_x - \mu_y)}{\sigma / \sqrt{n}} \sim N(0, 1), \]
\[ Q = \frac{(n_x - 1)s_x^2 + (n_y - 1)s_y^2}{\sigma^2} \sim \chi^2_{df}, \]
where \( df = n_x + n_y - 2 \), and \( Z, Q \) are independent. This implies that under the null hypothesis, 
\[ T = \frac{\bar{Y} - \bar{X}}{s_p \sqrt{1/n_x + 1/n_y}} \sim t_{df} \]
where \( s_p^2 = Q / df \), referred to as the pooled variance. Confidence intervals, critical values and p-values can be obtained from this t-distribution. The power function is given by a non-central t-distribution with d.f. \( n_x + n_y - 2 \) and non-centrality parameter
\[ \Delta = \frac{n_x n_y (\delta)}{n_x + n_y (\sigma)}, \]
where \( \delta = \mu_y - \mu_x \). Observable differences and sample sizes can be derived from this power function under assumptions about the ratio \( \delta / \sigma \). EnvStats functions \( tTestPower \), \( tTestN \), and \( tTestScaledMdd \) return power, sample sizes, and minimum detectable differences for one and two sample t-tests.

A generalization of this model is referred to as the location-shift model in which the distribution functions for the two populations are
\[ P(X \leq t) = F(t), \quad P(Y \leq t) = F(t - \delta), \]
where \( F \) is some unspecified d.f. Note that the distributions are the same if \( \delta = 0 \). However, if \( \delta > 0 \), then \( X \) tends to be smaller than \( Y \). A nonparametric test for this case is the Mann-Whitney-Wicoxon rank sum test. An R function that implements this test is \( wilcox.test() \). Power functions for this test are based on large sample normal approximations to the distribution of the rank sum statistic, but they are beyond the scope of this course. Simulation also can be used to determine power for specific situations.

Many applications involve comparisons between two populations with different s.d.’s. In that case the scales of the measurements in these populations would be different which implies that the mean may not be the best parameter to use for comparisons. If we wish to compare means of normally distributed populations without the assumption of equal variances, then we must use an approximate test. The most commonly used test for this is referred to as Welch’s approximation to the two-sample t-test. Let
\[ V_x = s_x^2 / n_x, \quad V_y = s_y^2 / n_y, \]
\[ T_w = \frac{\bar{Y} - \bar{X}}{\sqrt{V_x + V_y}}. \]

Then under \( H_0 \), \( T_w \approx t_{dw} \), where
\[ dw = \frac{(V_x + V_y)^2}{\frac{V_x^2}{n_x - 1} + \frac{V_y^2}{n_y - 1}}. \]

This test is implemented by default in the \( R \) function \( t.test() \). However, power functions for these models are very complicated and difficult to obtain directly. Specific questions regarding power can be answered using simulation.

**Linear Models**

Research questions that require comparisons among more than two populations can be examined using ANOVA and related methods. Experiments in which a continuous response variable is obtained along with continuous predictor variables can be modeled using regression methods. More generally, if the predictor variables include a mix of continuous and categorical variables, then methods associated with linear models can be used. Appropriate methods for these situations are summarized in this section. Details about derivations of test statistics and sampling distributions are left to courses in linear models, advanced statistical methods, and multivariate statistics.

1. **Continuous response, one categorical predictor**

   This situation occurs when we wish to compare means of several populations. The basic method is ANOVA. Let \( Y_{ij} \) be independent r.v.s and assume
   \[ Y_{ij} \sim N(\mu_i, \sigma^2), \; 1 \leq j \leq n_i, \; 1 \leq i \leq k, \]
   where \( k \) is the number of populations. Note that this assumption requires homogeneity of variance. The null hypothesis to be tested is
   \[ H_0 : \mu_1 = \cdots = \mu_k \]
   versus the alternative that there are some differences. ANOVA models are usually parameterized by
   \[ \mu_i = \mu + \alpha_i. \]
   Since this results in \( k + 1 \) parameters for just \( k \) groups, it is necessary to add a constraint on \( \alpha_i \). In \( R \) these constraints are referred to as contrasts. The default parameterization in \( R \) is the treatment contrast in which \( \alpha_1 = 0 \). This is the natural parameterization to use if population 1 represents a control and the other populations are experimental groups, but this can be used for other situations as well. The parameterization used for ANOVA determines how parameter estimates are interpreted.

   **Example.** Consider the crabs data set
   
   \[ \text{http://www.utdallas.edu/~ammann/stat6390scripts/crabs.txt} \]
   
   Suppose we wish to compare means of FL among the four combinations of sex and species.
Crabs = read.table("http://www.utdallas.edu/~ammann/stat6390scripts/crabs.txt", header=TRUE)
SpeciesSex = paste(Crabs$species,Crabs$sex,sep="")
SpeciesSex = factor(SpeciesSex)
Crabs.df = data.frame(FL=Crabs$FL,SpeciesSex)
FL.aov = aov(FL ~ SpeciesSex, data=Crabs.df)

The overall test of the null hypothesis of equal means is returned by the summary of FL.aov.

summary(FL.aov)

This produces the ANOVA table:

<table>
<thead>
<tr>
<th>Df</th>
<th>Sum Sq</th>
<th>Mean Sq</th>
<th>F value</th>
<th>Pr(&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SpeciesSex</td>
<td>3</td>
<td>551.6</td>
<td>183.85</td>
<td>19.17</td>
</tr>
<tr>
<td>Residuals</td>
<td>196</td>
<td>1879.7</td>
<td>9.59</td>
<td></td>
</tr>
</tbody>
</table>

---

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

The p-value of this test is $6.08\times 10^{-11}$ and so we reject the null hypothesis and conclude that there are some differences among these means. However, that p-value depends on the assumptions which must be verified. This can be done by the `plot()` function which produces standard residual plots for ANOVA.

plot(FL.aov)

Note that the Q-Q plot looks reasonable. The residuals vs fitted plot shows that within group variability is essentially the same so homogeneity of variance also is reasonable. This can be tested formally using Bartlett’s test.

bartlett.test(FL ~ SpeciesSex, data=Crabs.df)

This gives

Bartlett test of homogeneity of variances

data:  FL by SpeciesSex
Bartlett’s K-squared = 4.3219, df = 3, p-value = 0.2287

This test does not reject the null hypothesis of equal variances. Bartlett’s test is based on the ratio of the pooled variance to the geometric mean of the individual group variances and so is sensitive to outliers and departures from normality, especially asymmetry.

More robust tests for homogeneity of variance include modified versions of Levene’s test. These tests are based on ANOVA applied to absolute deviations from group medians and a more robust version in which ANOVA is applied to the ranks of the absolute deviations. A function for Levene-type tests is provided by the `lawstat` library.
library(lawstat)
levene.test(Crabs.df$FL, Crabs.df$SpeciesSex)
levene.test(Crabs.df$FL, Crabs.df$SpeciesSex, kruskal.test=TRUE)

These tests are equivalent to

\[
\begin{align*}
\text{FL} &= \text{Crabs}\$\text{FL} \\
\text{FL.med} &= \text{tapply(FL, SpeciesSex, median)} \ # \text{group medians} \\
\text{FL.d} &= \text{abs(FL - FL.med[SpeciesSex]) \ # \text{abs deviations from group medians}} \\
\text{summary(aov(FL.d \sim \text{SpeciesSex}))} \ # \text{Brown-Forsythe's modified Levene test} \\
\text{FL.rd} &= \text{rank(FL.d)} \ # \text{ranks of combined data} \\
\text{summary(aov(FL.rd \sim \text{SpeciesSex}))} \ # \text{Kruskal-Wallis ANOVA}
\end{align*}
\]

Bartlett’s test has more power if the data is normally distributed, but if not or if outliers are present, then the Kruskal-Wallis rank-based version of Levene’s test can maintain its power whereas Bartlett’s test loses power. A reasonable approach to checking for homogeneity would be to apply both Bartlett’s test and rank-based Levene’s test. If at least one test rejects the null hypothesis, then conclude that the variances are not homogeneous, but if neither test rejects, then accept the hypothesis of homogeneity.

Additional information about this model can be obtained using a summary for linear models.

summary.lm(FL.aov)

This produces

Call:
aov(formula = FL ~ SpeciesSex, data = Crabs.df)

Residuals:

\[
\begin{array}{cccccc}
\text{Min} & \text{1Q} & \text{Median} & \text{3Q} & \text{Max} \\
-7.526 & -2.309 & 0.158 & 2.188 & 6.474 \\
\end{array}
\]

Coefficients:

\[
\begin{array}{cccccc}
\text{Estimate} & \text{Std. Error} & \text{t value} & \text{Pr(>|t|)} \\
\text{(Intercept)} & 13.2700 & 0.4380 & 30.300 & < 2e-16 \ *** \\
\text{SpeciesSexBM} & 1.5720 & 0.6194 & 2.538 & 0.0119 \ * \\
\text{SpeciesSexOF} & 4.3240 & 0.6194 & 6.981 & 4.42e-11 \ *** \\
\text{SpeciesSexOM} & 3.3560 & 0.6194 & 5.418 & 1.75e-07 \ *** \\
\end{array}
\]

---

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.097 on 196 degrees of freedom
Multiple R-squared: 0.2269, Adjusted R-squared: 0.215
F-statistic: 19.17 on 3 and 196 DF, p-value: 6.084e-11

67
Since the default parameterization was used, the coefficients can be interpreted as follows. The estimated mean of BF crabs is 13.270. The estimated difference between means of BM and BF crabs is 1.572, so the estimated mean for BM is $13.270 + 1.572 = 14.842$. Estimated means for other groups are obtained similarly. Column Pr(>|t|) contains p-values for hypothesis tests that the corresponding coefficients are 0. However, these should not be used to test for differences among group means.

Pairwise comparisons among groups require corrections to control overall experimental-wise error. The best known of these methods is the Bonferroni adjustment in which the level of significance for an individual test is divided by the number of comparisons. However, more efficient methods have been developed. The R function `pairwise.t.test()` performs such comparisons. The default value of the argument `pool.sd` is TRUE so a pooled estimate of the common s.d. is used. This is the same as the Residual standard error given in the `lm` summary above.

```r
pairwise.t.test(Crabs.df$FL, Crabs.df$SpeciesSex, p.adjust.method="holm")
```

This gives

```
Pairwise comparisons using t tests with pooled SD

data:  Crabs.df$FL and Crabs.df$SpeciesSex

        BM OF
BF 0.024 -
BM 2.7e-10 5.9e-05 -
OM 8.8e-07 0.013 0.120

P value adjustment method: holm
```

The adjustment method `holm` is a more efficient modification of the Bonferroni method. This shows that at the 5% level of significance all comparisons are significant except OM-OF. If a smaller set of comparisons is needed, for example comparison of each treatment level to the control, but not with each other, then the p-values from these tests can be adjusted by the function `p.adjust()`. The p-values need to be entered as a vector and argument `method=` by default is set to `holm`.

If the assumption of homogeneity of variance is violated, then pairwise comparisons must be made with `pool.sd=FALSE`. If the assumption of normality is not reasonable, then pairwise comparisons can be made using the nonparametric rank sum test. P-values from these tests should be adjusted for multiple comparisons using the R function `p.adjust()`.

```r
lvls = levels(Crabs.df$SpeciesSex)
k = length(lvls)
nc = k*(k-1)/2
pval = rep(0,nc)
```
lmat = t(outer(lvls,lvls,paste,sep="-"))
names(pval) = lmat[lower.tri(lmat)]
r = 1
for(i in seq(k-1)) {
    ndx = Crabs.df$SpeciesSex == lvls[i]
    x = Crabs.df$FL[ndx]
    for(j in seq(i+1,k)) {
        ndy = Crabs.df$SpeciesSex == lvls[j]
        y = Crabs.df$FL[ndy]
        xy.rs = wilcox.test(x,y)
        pval[r] = xy.rs$p.value
        r = r+1
    }
}
adjPval = p.adjust(pval,method="holm")
print(round(adjPval,4))

This shows that all differences are significant at 5% except OF-OM.

Predicted values can be obtained using the predict() function. This function can return confidence intervals for group means and prediction intervals for individual within a group.

cdf = data.frame(SpeciesSex = factor(lvls))
predict.lm(FL.aov, cdf, interval="confidence")
predict.lm(FL.aov, cdf, interval="prediction")

Note that the fitted values returned are the same but the prediction intervals are wider since they must include within-group variability in addition to between-group variability.

2. Continuous response, one categorical and one continuous predictor

In this case we are interested in determining how the relationship between the continuous predictor and the response is affected by the categorical variable. This situation can be fit with three different models:
(a) one common regression model, categorical variable is not used;
(b) common slope but different intercepts for different groups;
(c) different intercepts and slopes for different groups.

Case (c) involves an interaction between the categorical and continuous predictors.

Example. Suppose we wish to include CL as a predictor for FL in addition to SpeciesSex.

Crabs2.df = data.frame(FL=Crabs$FL, CL=Crabs$CL, SpeciesSex)
FL1.lm = lm(FL ~ CL, data=Crabs2.df)
FL2.lm = lm(FL ~ CL + SpeciesSex, data=Crabs2.df)
FL3.lm = lm(FL ~ CL * SpeciesSex, data=Crabs2.df)
#check assumptions for most complex model
plot(FL3.lm)
Note that the Scale-Location plot shows a slight increase in variability as fitted values increase, but this is mainly due to just a few crabs, so homogeneity of variance is reasonable. Normal Q-Q plot shows that normality assumption also is reasonable. We can perform partial-F tests to compare these models using the `anova()` function.

```r
anova(FL1.lm, FL2.lm, FL3.lm)
```

which gives

**Analysis of Variance Table**

<table>
<thead>
<tr>
<th>Model</th>
<th>Res.Df</th>
<th>RSS</th>
<th>Df</th>
<th>Sum of Sq</th>
<th>F</th>
<th>Pr(&gt;F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1: FL ~ CL</td>
<td>1</td>
<td>198</td>
<td></td>
<td>101.793</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model 2: FL ~ CL + SpeciesSex</td>
<td>2</td>
<td>28.947</td>
<td>3</td>
<td>72.846</td>
<td>196.762</td>
<td>&lt; 2.2e-16 ***</td>
</tr>
<tr>
<td>Model 3: FL ~ CL * SpeciesSex</td>
<td>3</td>
<td>23.694</td>
<td>3</td>
<td>5.252</td>
<td>14.187</td>
<td>2.172e-08 ***</td>
</tr>
</tbody>
</table>

---

Signif. codes: 0 ‘***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

This shows that the interaction term is significant, so model `FL3.lm` should be used for this data.

```r
summary(FL3.lm)
```

The coefficients are interpreted similarly to the previous case.

- `(Intercept)`: intercept for BF
- `CL`: slope for BF
- `SpeciesSexBM`: difference between intercepts of BF and BM
- `SpeciesSexOF`: difference between intercepts of BF and OF
- `SpeciesSexOM`: difference between intercepts of BF and OM
- `CL:SpeciesSexBM`: difference between slopes of BF and BM
- `CL:SpeciesSexOF`: difference between slopes of BF and OF
- `CL:SpeciesSexOM`: difference between slopes of BF and OM

We can use the `predict()` function to construct a plot that superimposes the regression lines given by FL3.lm along with confidence and prediction intervals.

```r
np = 300
which = "BF"
CLv = seq(min(Crabs$CL), max(Crabs$CL), length=np)
Y.lim = c(5,25) # common range for y-axis
df1 = data.frame(CL=CLv, SpeciesSex=rep(which, np))
FL1.conf = predict(FL3.lm, newdata=df1, interval="confidence")
FL1.pred = predict(FL3.lm, newdata=df1, interval="prediction")
```
ndx = Crabs.df$SpeciesSex == which
plot(FL[ndx] ~ CL[ndx], data=Crabs2.df, xlab="CL", ylab="FL", ylim=Y.lim, pch=19)
title(paste("FL vs CL for", which, "Crabs"))
lines(df1$CL, FL1.conf[, "fit")
lines(df1$CL, FL1.conf[, "lwr"], col="blue")
lines(df1$CL, FL1.conf[, "upr"], col="blue")
lines(df1$CL, FL1.pred[, "lwr"], col="red")
lines(df1$CL, FL1.pred[, "upr"], col="red")
legend(max(Crabs$CL[ndx]), min(FL1.pred), legend=c("Fit", "95% Confidence", "95% Prediction")
  col=c("black", "blue", "red"), xjust=1, yjust=0, cex=.85)
FL3.coef = coef(FL3.lm)
intslp = c(FL3.coef["(Intercept)"], FL3.coef["CL"])
mtext(paste(paste(c("Intercept", "Slope"), "="), round(intslp, 3), collapse=" "), side=3, line=0)

###
which = "BM"
df1 = data.frame(CL=CLv, SpeciesSex=rep(which, npl))
FL1.conf = predict(FL3.lm, newdata=df1, interval="confidence")
FL1.pred = predict(FL3.lm, newdata=df1, interval="prediction")
ndx = Crabs.df$SpeciesSex == which
plot(FL[ndx] ~ CL[ndx], data=Crabs2.df, xlab="CL", ylab="FL", ylim=Y.lim, pch=19)
title(paste("FL vs CL for", which, "Crabs"))
lines(df1$CL, FL1.conf[, "fit")
lines(df1$CL, FL1.conf[, "lwr"], col="blue")
lines(df1$CL, FL1.conf[, "upr"], col="blue")
lines(df1$CL, FL1.pred[, "lwr"], col="red")
lines(df1$CL, FL1.pred[, "upr"], col="red")
legend(max(Crabs$CL[ndx]), min(FL1.pred), legend=c("Fit", "95% Confidence", "95% Prediction")
  col=c("black", "blue", "red"), xjust=1, yjust=0, cex=.85)
intslp = c(FL3.coef["(Intercept)"], FL3.coef["SpeciesSexBM"], FL3.coef["CL"])
mtext(paste(paste(c("Intercept", "Slope"), "="), round(intslp, 3), collapse=" "), side=3, line=0)

###
which = "OF"
df1 = data.frame(CL=CLv, SpeciesSex=rep(which, npl))
FL1.conf = predict(FL3.lm, newdata=df1, interval="confidence")
FL1.pred = predict(FL3.lm, newdata=df1, interval="prediction")
ndx = Crabs.df$SpeciesSex == which
plot(FL[ndx] ~ CL[ndx], data=Crabs2.df, xlab="CL", ylab="FL", ylim=Y.lim, pch=19)
title(paste("FL vs CL for", which, "Crabs"))
lines(df1$CL, FL1.conf[, "fit")
lines(df1$CL, FL1.conf[, "lwr"], col="blue")
lines(df1$CL, FL1.conf[, "upr"], col="blue")
lines(df1$CL, FL1.pred[, "lwr"], col="red")
lines(df1$CL, FL1.pred[, "upr"], col="red")
when multiple predictors are available, a linear model that includes all predictors can be constructed that includes interactions between categorical and continuous predictors. Large models may reduce residual error, but we must remember that the data represents a random sample from the population. Some predictor variables may appear to be contributing to reduction of residual error in the sample but are not important for prediction in the population. Inclusion of such variables in the model will increase bias of predicted values. Variable selection for large models is an important tool for balancing models between precision and bias. There have been several approaches to this problem including penalized likelihood methods such as Akaike’s Information Criterion (AIC) and Bayes Information Criterion (BIC). Shrinkage methods such as lasso also have been developed for variable selection. Although a thorough examination of these methods is beyond the scope of this course, AIC variable selection will be discussed here to illustrate these methods.

It is well known that addition of a variable to the current model will decrease residual mean square error of the model as long as the correlation between the new variable and residuals of the current model is not 0. However, that decrease may not be meaningful since the probability is essentially 0 that a randomly generated vector will have exactly 0 correlation with the current residuals. To counteract this, Akaike proposed a modified optimization criteria to replace minimization of the sum of squared residuals. Akaike added a penalty proportional to the number of parameters in the model. If the number of parameters
in the current model is $p$, then

$$ IC(p) = RSS_p + kp, $$

where $k$ is a specified constant. AIC uses $k = 2$ and BIC uses $k = \log(n)$ where $n$ is the sample size. The goal is to minimize IC. Addition of a new variable decreases $RSS_p$, but that decrease must overcome the increased penalty $kp$ in order to decrease IC. Since $\log(n) > 2$ when $n > 7$, then BIC imposes a stronger penalty on model size and so generally results in smaller models than AIC. Therefore BIC places more importance on reducing bias of predicted values whereas AIC places more importance on increasing precision, although both can reduce bias compared to ordinary least squares models.

In large data sets it is not feasible computationally to examine every possible combination of predictor variables to find the global minimum of IC. Practical implementations of variable selection based on IC involve stepwise progression through the predictors. Forward stepwise uses the following steps.

1. Begin with the null model that just includes an intercept. Residuals for that model are deviations about the sample mean of the response variable. Evaluate $IC(1)$ for that model.

2. At step $i + 1$, $i \geq 1$, find the predictor that has the highest squared correlation with the current residuals and evaluate $IC(i + 1)$.

3. If $IC(i + 1) \geq IC(i)$, then stop and use the model with $i$ variables. Otherwise repeat previous step.

4. Continue until IC increases or all variables have been added to the model.

Backward stepwise starts with all variables in the model and then at each step removes the variable that results in the minimum reduction of RSS among all current variables. This continues until removal of any remaining variable would increase IC. Stepwise regression using IC is implemented in R by the function `step()`. The following script illustrates this for the crabs data.

```r
allCrabs = data.frame(Crabs[,-(1:2)], SpeciesSex)
# fit full model with interactions between SpeciesSex and other variables
FLfull.lm = lm(FL ~ .*SpeciesSex, data=allCrabs)
plot(FLfull.lm) # check assumptions
summary(FLfull.lm)
FLbackAIC.lm = step(FLfull.lm) # backward stepwise AIC
summary(FLbackAIC.lm)
FLbackBIC.lm = step(FLfull.lm, k=log(n)) # backward stepwise BIC
summary(FLbackBIC.lm)
FL0.lm = lm(FL ~ 1, data=allCrabs) # null model
```

73
FLforwardAIC.lm = step(FLfull.lm, 
    scope=list(lower=FL0.lm, upper=FLfull.lm)) # forward AIC
summary(FLforwardAIC.lm)
FLforwardBIC.lm = step(FLfull.lm, 
    scope=list(lower=FL0.lm, upper=FLfull.lm), k=log(n)) # forward BIC
summary(FLforwardBIC.lm)
coefFull = coef(FLfull.lm)
coef.mat = matrix(0, length(coefFull), 5)
dimnames(coef.mat) = list(names(coefFull), c("Full","AIC_back","BIC_back","AIC_for","BIC_for"))
coef.mat[,1] = coefFull
coef1 = coef(FLbackAIC.lm)
coef.mat[names(coef1),2] = coef1
coef1 = coef(FLbackBIC.lm)
coef.mat[names(coef1),3] = coef1
coef1 = coef(FLforwardAIC.lm)
coef.mat[names(coef1),4] = coef1
coef1 = coef(FLforwardBIC.lm)
coef.mat[names(coef1),5] = coef1
print(round(coef.mat,3))
anova(FLfull.lm,FLforwardBIC.lm) # test for significant difference between models
# no significant difference so use BIC model

In this example forward and backward stepwise return the same model for AIC and for BIC, but that does not always happen. Also note that AIC includes an interaction between BD and SpeciesSex whereas BIC does not.

Backward stepwise requires a sufficiently large sample size to fit a model with all variables and interactions and so cannot be used if the number of variables is greater than the sample size. In those cases shrinkage methods such as lasso or least angle regression can be used.

**Generalized Linear Models**

A detailed discussion of GLM is beyond the scope of this course, but a few comments will be given here. Environmental studies may include data with a binary response and multiple predictor variables. The goal for such data is to predict the probability of the response labeled Success based on the predictor variables. This problem is referred to as logistic regression and is implemented in R by the function `glm()`. Other studies may have response variables that represent counts of species. If variability of the counts is much smaller than the means, then `lm()` typically can be used, although this should be checked by residual plots. If residual plots show that variability increases with fitted values, then Poisson regression should be used via `glm` with `family=poisson`. However, this assumes that the dispersion parameter (ratio of variance to mean) is 1. If the counts are more highly dispersed, then GLM based on the negative binomial distribution may be more appropriate. This is available in the MASS library with its function `glm.nb()`.

74
Homework and Project Assignments

Homework 1

Due Date: Feb. 16, 2016.
This assignment uses data sets from the R package EnvStats. To access them you must enter the following into R.

library(EnvStats)

Answer the following for each of the data sets listed below.

a. Determine what parametric model is most appropriate. Include any diagnostic plots used to support your answer.

b. Construct a 95% confidence interval for the mean of the data. If there is a robust method available for the confidence interval, report it as well and compare it with the standard method.

Details about these data sets can be accessed by the help function in R.

1. EPA.02d.Ex.4.mg.per.kg.vec
2. Gibbons.et.al.09.Vinyl.Chloride.vec
3. Millard.Deverel.88.df
   For this data set, extract observations for which Cu.censored is FALSE and answer the questions separately for Cu from Zone Alluvial.Fan and for Cu from Basin.Trough. Note that you also will need to remove any NA observations for Cu. Then do the same for Zn.
4. Benthic.df
   For this data set answer the questions for benthic Index in Stratum 104. In addition to those questions, use the Stratum 104 Index values as a preliminary sample to determine the sample size needed to estimate the mean Index with 95% and with precision of ±0.1.
5. For the Benthic.df data, instead of answering questions a and b, use the entire data set to construct a 95% confidence interval for the proportion of sites that have benthic index equal to 1 (the lowest). Do the same for the proportion of sites that have benthic index equal to 5 (the highest).

Homework 2

Due Date: March 10, 2016.
This assignment uses data sets from the R package EnvStats. To access them you must enter the following into R.

library(EnvStats)
1. Use the alkalinity measures in dataframe EPA.09.Ex.14.3.alkalinity.df.
   
   (a) Determine an appropriate parametric model for this data and obtain a 95% confidence interval for the population mean.

   (b) Obtain an 80%-content tolerance interval for alkalinity at this site that has 95% confidence.

   (c) Construct a one-sided upper 90% confidence interval for the lower quartile (.25 quantile).

2. Use data in Gibbons.et.al.09.Alkilinity.vec.

   (a) Determine an appropriate parametric model for this data and obtain a 95% confidence interval for the population median.

   (b) Construct a 90% prediction interval for the next 3 observations. Also construct a 90% prediction interval for the mean of the next 3 observations.

   (c) Construct a 90%-content tolerance interval that has 95% confidence.


   (a) Many of these observations were below detection limits and so are censored. Obtain a 90% confidence interval for the proportion of observations that are below detection limits.

   (b) Construct a non-parametric 90% confidence interval for the upper quartile of Ag in this dataframe. You can use all observations since the censored observations are very small.

4. Use dataframe Millard.Deverel.88.df. Construct 95% nonparametric confidence intervals for the median of Cu and the median of Zn. You can use all observations since the censored observations are very small.

Final Project

1. Let

\[ d = \frac{(u + v)^2}{\frac{u^2}{r} + \frac{v^2}{s}}. \]

Show that

\[ d \leq r + s, \ u, v, r, s > 0 \]

\[ = r + s, \ us = rv. \]

Apply this result to compare the d.f. of Welch’s two-sample t-test with the d.f. of the pooled sample t-test. \textbf{Hint:} let \( v = au \) and find the maximum of \( d(a) \).
2. The data file, 
http://www.utdallas.edu/~ammann/stat6390scripts/toxic.dat
is the result of a standard EPA test for toxicity. In this test, a sample of 10 Ceri-
adaphnia dubia females is exposed to a particular dose of a chemical. The experiment
runs until each female that survives has produced three broods (sets of hatched eggs).
The number of offspring is recorded for each brood. Note that Dose is recorded as
a concentration and therefore is a continuous variable. However, there are only five
distinct levels of dose including the control group (Dose 0). This implies that the data
could be modeled two ways, (1) Dose is treated as a continuous variable; (2) Dose is
treated as a categorical variable (factor). The first case requires regression methods
to predict total number of hatched eggs based on Dose; the second requires ANOVA
methods. Assuming that a reasonable regression model can be obtained, then this
method can provide more detail about the dose-response relationship. In particular,
regression models can return confidence and prediction intervals for intermediate dose
levels, not just the experimental dose levels. If a reasonable regression model cannot be
obtained, then ANOVA methods must be used with Dose as categorical variable. Also,
if the research questions include comparisons of control results to each experimental
dose, then ANOVA methods would be appropriate. For this project both methods will
be used. To use ANOVA, create a new data frame in which Dose is converted to a
factor.

toxic = read.table("http://www.utdallas.edu/~ammann/stat6390scripts/toxic.dat",
header=TRUE, row.names=1)
toxicA = toxic
toxicA$Dose = factor(toxicA$Dose)

Standard assumptions for ANOVA models include homogeneity of variance. If that
assumption does not hold, then multiple comparisons must be made using either t-
tests or Wilcoxon rank sum tests.

(a) (ANOVA) Fit an ANOVA model for Total. Are the assumptions reasonable?
Interpret the coefficients of this model. Determine which dose levels differ sig-
nificantly from the control dose for total number of offspring at the 5% level of
significance.

(b) (ANOVA) This data set has an unusual feature: only one female exposed to the
highest dose had offspring in Brood2 and none had offspring in Brood3. Create
a new data frame that only includes Dose and Brood23=Brood2+Brood3 from
toxic.dat, and repeat part (a) with Brood23 as the response variable. Compare
these results with the results of part (a).

(c) (Regression) Treat Dose as a continuous variable and construct a regression model
to predict Total based on dose. Are the assumptions reasonable? Construct a
95% confidence interval for the mean total number of offspring for Dose = 200
and a 95% prediction interval for the total number of offspring of a particular female exposed to $Dose = 200$.

3. Use the wasp data set,
http://www.utdallas.edu/~ammann/stat6390scripts/wasp.dat
Fit a model to predict $TL$ based all the other variables that also allows different slopes for queens (caste=Q) and workers (caste=W). Check the assumptions to make sure they are satisfied. Reduce this model to remove variables that do not contribute to the prediction of $TL$ using both AIC and BIC. Perform a formal hypothesis test to compare the full model with the reduced models. Interpret the coefficients of the reduced models.

4. The data file,
http://www.utdallas.edu/~ammann/stat6390scripts/SENIC.csv
represents a random sample of 113 hospitals selected for a study about infection risk at hospitals. Variables $MedSchool$ and $Region$ in this file have been converted to their respective category levels so read.table() will automatically convert them to factor variables. The first column is an ID number and so can be used for row names.

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stay</td>
<td>Average length of stay of all patients in the hospital</td>
</tr>
<tr>
<td>Age</td>
<td>Average age of patients</td>
</tr>
<tr>
<td>Infection</td>
<td>Average estimated probability of acquiring an infection in the hospital</td>
</tr>
<tr>
<td>Culturing</td>
<td>Ratio of number of cultures performed to number of patients (times 100)</td>
</tr>
<tr>
<td>Xray</td>
<td>Ratio of number of chest X-rays performed to number of patients (times 100)</td>
</tr>
<tr>
<td>Beds</td>
<td>Average number of beds in the hospital</td>
</tr>
<tr>
<td>MedSchool</td>
<td>Medical school affiliation (&quot;Yes&quot; or &quot;No&quot;)</td>
</tr>
<tr>
<td>Region</td>
<td>Geographic region</td>
</tr>
<tr>
<td>Census</td>
<td>Average number of patients in the hospital per day</td>
</tr>
<tr>
<td>Nurses</td>
<td>Average number of full-time equivalent nurses in the hospital</td>
</tr>
<tr>
<td>Facilities</td>
<td>Percent of 35 potential facilities and services provided by the hospital</td>
</tr>
</tbody>
</table>

(a) Fit an appropriate model to predict infection risk. Include an interaction between $MedSchool$ and each of the variables $Culturing$, $Xray$, $Nurses$. Describe how you verified the model assumptions.

(b) Reduce this model, if possible, to a set of variables that are most important for prediction of infection risk using BIC. Interpret the coefficients of this reduced model.

(c) Suppose one of the hospitals in the area has the following values for the variables in this study:
Obtain a 95% prediction interval for the infection risk of this hospital. How does the predicted risk compare to the overall mean prediction risk among all hospitals in this study?