35 Applied Inferential Statistics

David Housman

## Level of Significance

The *level of significance ()* is the probability at which we say that an improbable event does not occur. More concretely, is the acceptable probability for erroneously accepting (a) that a new drug is effective when it is actually ineffective, (b) that a person is guilty when they are actually innocent, or (c) that a company is lying about the quality of its product when they are actually telling the truth. In scientific literature, often used levels of significance are 0.05, 0.01, and 0.001.

## Hypothesis Test

A *hypothesis test* consists of null and alternative hypotheses. A *test statistic* that measures how far the data are from the null hypothesis is chosen. The *p-value* is the probability of obtaining a sample from the null hypothesized population that has a test statistic at least as supportive of the alternative hypothesis as obtained from the data. The p-value could be approximated with a simulation or calculated with a theoretically derived formula. If the p-value is smaller than the level of significance , then we reject the null hypothesis and accept the alternative hypothesis. If the p-value is at least as large as the level of significance, then we fail to reject the null hypothesis and fail to accept the alternative hypothesis.

The hypothesis test process is similar to jurisprudence in most western democracies. The accused is assumed to be innocent (the null hypothesis) and is determined to be guilty (the alternative hypothesis) if and only if the evidence (summarized by the test statistic and p-value) proves guilt beyond a reasonable doubt (the p-value is less than the level of significance). Although assumed to be innocent, the accused is either proclaimed guilty or not guilty (which could be considered different from proclaiming innocence).

A hypothesis test process is also similar to a mathematician’s proof by contradiction. To prove that is not rational (the alternative hypothesis), the mathematician supposes is rational (the null hypothesis), finds a contradiction (evidence that shows the p-value is zero), and concludes that is not rational. Evidence derived from a sample almost never yields a p-value of zero, and so we are willing to say that will a small enough p-value there is sufficient evidence to accept the alternative hypothesis.

We can compare reality and our decision in the following table (with rows corresponding to reality and columns corresponding to our decision).

|  | reject  | don’t reject  |
| --- | --- | --- |
|  is true | type I error () | correct decision |
|  is false | correct decision | type II error () |

We can see that the significance level is also the probability of making a type I error: rejecting a true null hypothesis. The probability of a type II error may be a function of the possible alternative hypotheses.

The hypothesis testing process provides a quantitative way to describe the intuition we can obtain by looking at plots and more intuitive statistics. Since multiple tests have been defined, users must make choices about which test to use in any particular circumstance. This is one of the many aspects of statistics that is more art than science, and there are regular controversies and different schools of thought in the professional statistics community. Thus, it is important to state what test was used and the actual p-value obtained in any inferential statistical analysis.

## Confidence Interval

A *confidence interval* is an estimate of a population parameter that conveys the precision of the estimate. A level of confidence equal to one minus the level of significance () is typically reported as a percentage confidence. The classical (parametric) approach is to include all null hypothesized values for the parameter that would not be rejected with a hypothesis test. A resampling (nonparametric) approach is to use the actual sample data as a best estimate for the population and determine a coverage interval for the estimator. Since people use a variety of different procedures and confidence levels, these whould be reported along with the confidence interval.

## Validity

Hypothesis tests and confidence intervals are only considered valid if the significance level and procedure is chosen **before** the data are collected and analyzed. This is how the United States Food and Drug Administration deals with drug testing for safety and effectiveness. In the examples that follow, we will often consider different procedures and signifance levels as an aid to understanding the concepts. This is often what researchers do in what is often called *exploratory data analysis*. We should be careful, honest, and transparent in reporting any results obtained through exploration.

## Fair Coin Claim

An actual penny was flipped 100 times resulting in 37 heads. Is the penny a fair coin?

Statisticians organize this scenario as a *hypothesis test*. The *population* consists of heads and tails with the probability of heads being the *parameter* . The *sample* is the actual coin flips observed. The *null hypothesis ()* is that , and the *alternative hypothesis ()* is that . The *p-value* is the probability of obtaining a random sample at least as supportive of the alternative hypothesis as was actually observed given that the null hypothesis is true. In this case, the p-value is .

pbinom(37,100,0.5) + (1 - pbinom(62,100,0.5))

[1] 0.01203298

With a p-value of 0.012 and a 0.01 level of significance, we would conclude that could be , i.e., the coin could be fair. With a p-value of 0.012 and a 0.05 level of significance, we would conclude that , i.e., the coin is not fair. This illustrates the danger of choosing the level of significance after, rather than before, the data is collected.

To illustrate the danger in choosing the test design after collecting the date, perhaps we conjecture that the probability of heads is less than 1/2 (based upon the small number of heads obtained in the sample). This would suggest changing the alternative hypothesis to resulting in a change of the p-value to .

pbinom(37,100,0.5)

[1] 0.006016488

With a p-value of 0.006 and a 0.01 level of significance, we would conclude that although we earlier could not conclude (at the same level of significance) that was different from .

This example has illustrated *two-tailed* (when ) and *one-tailed* (when ) hypotheses tests and how to translate a p-value and a level of significance into statistical conclusions. Observe that we either accept the alternative hypothesis or fail to reject the null hypothesis, because there is either sufficient or insufficient evidence to reject the null hypothesis. Equally important is the importance of designing the test and choosing the significance level *before* gathering the data.

## Normal(50, 3) Concrete Block Strength Claim

Acme Concrete claims that the compressive strength of its concrete blocks is normally distributed with a mean of 50 hundred pounds per square inch and a standard deviation of 3 hundred pounds per square inch.

To test the claim, ten concrete blocks are randomly chosen and their compressive strengths (in hundred pounds per square inch) are measured.

acme = tibble(x = c(49.2,53.9,50.0,44.5,42.2,42.3,32.3,31.3,60.9,47.5))

We compare this data to the claimed distribution (the null hypothesis) in a variety of ways.

First, compare the sample mean and standard deviation with the claimed population mean and standard deviation. Observe that the sample mean is less than the claimed mean and the sample standard deviation is larger than the claimed standard deviation.

acme |> summarise(
 n = n(),
 smean = mean(x),
 cmean = 50,
 ssd = sd(x),
 csd = 3)

# A tibble: 1 × 5
 n smean cmean ssd csd
 <int> <dbl> <dbl> <dbl> <dbl>
1 10 45.4 50 9.09 3

Second, compare the sample and claimed population pdfs. Observe that the histogram of the data is shifted left from and wider than the claimed pdf. This visually indicates the sample mean is less than the claimed population mean and that the sample standard deviation is more than the claimed population standard deviation.

ggplot(acme, aes(x = x)) +
 geom\_histogram(aes(y = after\_stat(density)), binwidth = 5, center = 32.5, color = "white") +
 geom\_density(color = "blue", linewidth = 2) +
 geom\_function(fun = dnorm, args = list(mean = 50, sd = 3), color = "red", linewidth = 2)



Third, compare the sample and claimed population distributions with a qq plot. Observe that the midpoint of the qq points is at and their slope is greater than one. This visually indicates the sample mean is less than the claimed population mean and that the sample standard deviation is more than the claimed population standard deviation.

ggplot(acme, aes(sample = x)) +
 stat\_qq(distribution = function(p) qnorm(p, 50, 3)) +
 geom\_abline(slope = 1, intercept = 0, color = "red")



Fourth, compare the sample and claimed population cdfs. The sample points rise earlier (indicating a smaller center) and more slowly (indicating a larger spread) than the claimed cdf.

ggplot(acme, aes(x = x)) +
 stat\_ecdf() +
 geom\_function(fun = pnorm, args = c(50, 3), color = "red")



To turn these observations into a hypothesis test we need a way to measure how far our sample is from samples generated by the claimed population. Such a measure is called a *statistic*. One example would be the sample mean . For a one-sided test, the p-value would be .

pnorm(45.41, 50, 3/sqrt(10))

[1] 6.548219e-07

A second example would be a sample minimum . For a one-sided test, the p-value would be .

1 - (1-pnorm(31.3, 50, 3))^10

[1] 2.283066e-09

Rather than focus on a center or an end point of the data, it would be better in general to consider the overall distribution. The difference between the empirical cdf and the claimed population cdf can be summarized in a variety of ways. The Komogorov-Smirnov statistic is the maximum difference between the two cdfs. The Cramer-von Mises statistic integrates the squared difference between the two cdfs weighted by the claimed population pdf. We will use the Anderson-Darling statistic, which is similar to the Cramer-von Mises statistic but weights the tail regions more heavily than the middle region:

where the data in is sorted in ascending order in .

For Acme’s compressive strength data, the Anderson-Darling statistic is calculated as follows.

n = length(acme$x)
y = sort(acme$x)
AD = -n - sum((2\*(1:n)-1)/n\*(log(pnorm(y, 50, 3)) + log(1 - pnorm(rev(y), 50, 3)))); AD

[1] 14.4116

Is this a reasonable value for the Anderson-Darling statistic? To find out, we simulate samples from the claimed distribution, calculate their Anderson-Darling statistics, and determine the proportion of them at least as great as the Anderson-Darling statistic for the data.

set.seed(17)
runs = 1000
ad = numeric(runs)
for (i in 1:runs) {
 sample = rnorm(n, 50, 3)
 y = sort(sample)
 ad[i] = -n - sum((2\*(1:n)-1)/n\*(log(pnorm(y, 50, 3)) + log(1 - pnorm(rev(y), 50, 3))))
}
ggplot(tibble(ad = ad), aes(x = ad)) +
 geom\_density() +
 geom\_vline(xintercept = AD, color = "red")



prob = length(which(ad >= AD)) / runs; prob

[1] 0

With a p-value of essentially zero, the compressive strengths of Acme concrete blocks is significantly different from a normal random variable with mean 50 hundred pounds per square inch and standard deviation 3 hundred pounds per square inch.

The followning R function will calculate the Anderson-Darling statistic (once the goftest package is installed and required) and approximate the p-value theoretically.

goftest::ad.test(acme$x, "pnorm", 50, 3)

 Anderson-Darling test of goodness-of-fit
 Null hypothesis: Normal distribution
 Parameters assumed to be fixed

data: acme$x
An = 14.412, p-value = 6e-05

## Normal Concrete Block Strength Claim

Acme Concrete claims that the compressive strength of its concrete blocks is normally distributed, but does not specify the two parameters.

We need to compare the data to some unknown normal distribution. To do so, we need to estimate the parameters and with the data. The method of moments and the maximum likelihood estimators are the same:

 and .

To test the claim, ten concrete blocks are randomly chosen and their compressive strengths (in hundred pounds per square inch) are measured. We compare this data to the claimed distribution (using the estimators for the population parameters) in a variety of ways.

First, use sample statistics to estimate the corresponding population parameters.

acme = tibble(x = c(49.2,53.9,50.0,44.5,42.2,42.3,32.3,31.3,60.9,47.5))
n = length(acme$x); n

[1] 10

mu.hat = mean(acme$x); mu.hat

[1] 45.41

sigma.hat = sqrt((n-1)/n)\*sd(acme$x); sigma.hat

[1] 8.623161

Second, compare the sample and claimed population pdfs.

ggplot(acme, aes(x = x)) +
 geom\_histogram(aes(y = after\_stat(density)), binwidth = 5, center = 32.5, color = "white") +
 geom\_density(color = "blue", linewidth = 2) +
 geom\_function(fun = dnorm, args = list(mean = mu.hat, sd = sigma.hat), color = "red", linewidth = 2)



Third, compare the sample and claimed population distributions with a qq plot.

ggplot(acme, aes(sample = x)) +
 stat\_qq(distribution = function(p) qnorm(p, mu.hat, sigma.hat)) +
 geom\_abline(slope = 1, intercept = 0, color = "red")



Fourth, compare the sample and claimed population cdfs.

ggplot(acme, aes(x = x)) +
 stat\_ecdf() +
 geom\_function(fun = pnorm, args = c(mu.hat, sigma.hat), color = "red")



The difference between the sample and claimed population cdf can be summarized with the Anderson-Darling statistic with the normal population parameters estimated with our estimators.

n = length(acme$x)
mu.hat = mean(acme$x)
sigma.hat = sqrt((n-1)/n)\*sd(acme$x)
y = sort(acme$x)
AD = -n - sum((2\*(1:n)-1)/n\*(log(pnorm(y, mu.hat, sigma.hat)) + log(1 - pnorm(rev(y), mu.hat, sigma.hat)))); AD

[1] 0.2465171

Is this a reasonable value for the Anderson-Darling statistic? To find out, we simulate samples from the claimed distribution, calculate their Anderson-Darling statistics, and determine the proportion of them at least as great as the Anderson-Darling statistic for the data.

set.seed(17)
runs = 1000
ad = numeric(runs)
for (i in 1:runs) {
 sample = rnorm(n, mu.hat, sigma.hat)
 smu.hat = mean(sample)
 ssigma.hat = sqrt((n-1)/n)\*sd(sample)
 y = sort(sample)
 ad[i] = -n - sum((2\*(1:n)-1)/n\*(log(pnorm(y, smu.hat, ssigma.hat)) + log(1 - pnorm(rev(y), smu.hat, ssigma.hat))))
}
ggplot(tibble(ad = ad), aes(x = ad)) +
 geom\_density() +
 geom\_vline(xintercept = AD, color = "red")



prob = length(which(ad >= AD)) / runs; prob

[1] 0.795

Either (1) Acme Concrete’s claim is correct and the probability of obtaining a sample that is at least as unsupportive of Acme’s claim as the actual sample is 0.795, or (2) Acme Concrete’s claim is incorrect. With a 0.05 level of significance, statisticians would choose (1): With a p-value of 0.795, the compressive strengths of Acme concrete blocks is not significantly different from a normal random variable.

Minitab and Mathematica include black box functions that mimic the above tests as well as for many other tests. They will also calculate p-values theoretically or via simulation. For its estimate of , Minitab uses the sample standard deviation 9.090 rather than the maximum likelihood estimator 8.623, which is what Mathematica uses. So, Mathematica’s result matches the work we have done, and Minitab yields a slightly different result.

The Anderson-Darling test in goftest will take into account the fact that the parameters were estimated from the data but does so in a very different manner: the method of Braun referenced in the help documentation appears to be a general technique for handling “nuisance” parameters that relies on a Central Limit type theorem. Since we have a small sample, I would not want to use this approach with this data.

goftest::ad.test(acme$x, "pnorm", mu.hat, sigma.hat, estimated = TRUE)

 Anderson-Darling test of goodness-of-fit
 Braun's adjustment using 3 groups
 Null hypothesis: Normal distribution
 Parameters assumed to have been estimated from data

data: acme$x
Anmax = 0.66699, p-value = 0.9215

We can make use of goftest::ad.test to replace the computation of the statistic (although there is not much space savings and no time savings here).

set.seed(17)
runs = 1000
ad = numeric(runs)
for (i in 1:runs) {
 sample = rnorm(n, mu.hat, sigma.hat)
 smu.hat = mean(sample)
 ssigma.hat = sqrt((n-1)/n)\*sd(sample)
 y = sort(sample)
 ad[i] = goftest::ad.test(sample, "pnorm", smu.hat, ssigma.hat)$statistic
}
prob = length(which(ad >= AD)) / runs; prob

[1] 0.795

There is an Anderson-Darling test in the nortest package that provides a black box method that mimics our approach, but it only does so for a null hypothesized normal random variable and it uses the “divide by ”, rather than , standard deviation estimator..

nortest::ad.test(acme$x)

 Anderson-Darling normality test

data: acme$x
A = 0.2312, p-value = 0.7323

The Shapiro-Wilk normality test can only be used for testing normality, but it is often used in the literature and is built into the core of R.

shapiro.test(acme$x)

 Shapiro-Wilk normality test

data: acme$x
W = 0.96067, p-value = 0.7934

## Gamma Concrete Block Strength Claim

Acme Concrete claims that the compressive strength of its concrete blocks is distributed as a gamma random variable.

To test the claim, ten concrete blocks are randomly chosen and their compressive strengths (in hundred pounds per square inch) are measured. We compare this data to the claimed distribution in a variety of ways. First, obtain the method of moments estimates.

 and implies and

acme = tibble(x = c(49.2,53.9,50.0,44.5,42.2,42.3,32.3,31.3,60.9,47.5))
n = length(acme$x)
xbar = mean(acme$x)
v = var(acme$x)\*(n-1)/n
a.hat = xbar^2/v; a.hat

[1] 27.73129

b.hat = v/xbar; b.hat

[1] 1.637501

Compare the sample and claimed population pdfs.

ggplot(acme, aes(x = x)) +
 geom\_histogram(aes(y = after\_stat(density)), binwidth = 5, center = 32.5, color = "white") +
 geom\_density(color = "blue", linewidth = 2) +
 geom\_function(fun = dgamma, args = list(shape = a.hat, scale = b.hat), color = "red", linewidth = 2)



Compare the sample and claimed population distributions with a qq plot.

ggplot(acme, aes(sample = x)) +
 stat\_qq(distribution = function(p) qgamma(p, a.hat, 1/b.hat)) +
 geom\_abline(slope = 1, intercept = 0, color = "red")



Compare the sample and claimed population cdfs.

ggplot(acme, aes(x = x)) +
 stat\_ecdf() +
 geom\_function(fun = pgamma, args = c(a.hat, 1/b.hat), color = "red")



The difference between the sample and claimed population cdf can be summarized with the Anderson-Darling statistic.

y = sort(acme$x)
AD = -n - sum((2\*(1:n)-1)/n\*(log(pgamma(y, a.hat, 1/b.hat)) + log(1 - pgamma(rev(y), a.hat, 1/b.hat)))); AD

[1] 0.3276091

Is this a reasonable value for the Anderson-Darling statistic? To find out, we simulate samples from the claimed distribution, calculate their Anderson-Darling statistics, and determine the proportion of them at least as great as the Anderson-Darling statistic for the data. It is important to observe that we have used the data to estimate two population parameters; hence, we must do the same in our simulation.

runs = 1000
ad = numeric(runs)
for (i in 1:runs) {
 sample = rgamma(n, a.hat, 1/b.hat)
 sxbar = mean(sample)
 sv = var(sample)\*(n-1)/n
 sa.hat = sxbar^2/sv
 sb.hat = sv/sxbar
 y = sort(sample)
 ad[i] = -n - sum((2\*(1:n)-1)/n\*(log(pgamma(y, sa.hat, 1/sb.hat)) + log(1 - pgamma(rev(y), sa.hat, 1/sb.hat))))
}
ggplot(tibble(ad = ad), aes(x = ad)) +
 geom\_density() +
 geom\_vline(xintercept = AD, color = "red")



prob = length(which(ad >= AD)) / runs; prob

[1] 0.54

With a p-value of 0.54, the compressive strengths of Acme concrete blocks is not significantly different from a gamma random variable.

Either (1) Acme Concrete’s claim is correct and the probability of obtaining a sample that is at least as unsupportive of Acme’s claim as the actual sample is 0.54, or (2) Acme Concrete’s claim is incorrect. With a 0.05 level of significance, statisticians would choose (1): With a p-value of 0.54, the compressive strengths of Acme concrete blocks is not significantly different from a gamma random variable.

Observe that we have chosen to not reject two different claims: the compressive strength of Acme’s concrete blocks may be normally distributed and may be gamma distributed. That is why statisticians never say that they accept a null hypothesis. They only fail to reject a null hypothesis.

## No Answer Copying Claim

. Two students were given different 20 question multiple choice (4 per question) exams. They claim to have randomly chosen their answers. But their answers were the same on 15 of the questions. The instructor suspects the students of copying off each other. Who is right? (Based on an actual academic integrity case at a liberal arts college.)

**Answer.** Formulate as a hypothesis test.

: Answers chosen randomly.

: Answers copied.

Let be the number of matches given the null hypothesis . The p-value is . We provide four approaches to calculating the p-value.

**Approach 1:** Simulation with a margin of error of .

nruns = 100000
matches = numeric(nruns)
for(i in 1:nruns) {
 answers1 = sample(1:4, 20, replace = TRUE)
 answers2 = sample(1:4, 20, replace = TRUE)
 matches[i] = length(which(answers1 == answers2))
}
ggplot(tibble(matches), aes(x = matches))+
 geom\_bar() + geom\_vline(aes(xintercept = 15))



pvalue = length(which(matches>=15))/nruns; pvalue

[1] 0

**Approach 2:** is binomial with n = 20 and p = 1/4.

1 - pbinom(14, 20, 1/4)

[1] 3.813027e-06

**Approach 3:** where the form a sample from a Bernoulli population with . The Central Limit Theorem provides an approximation.

1 - pnorm(14.5, 20\*1/4, sqrt(20\*1/4\*(1-1/4)))

[1] 4.652865e-07

**Approach 4:** A blackbox hypothesis test. To use this approach, we need to reformulate the hypotheses using a population parameter. Let be the probability that two answers match. Then the hypotheses can be expressed

.

We have a sample of size 20 with 15 matches. From the output, we conjecture that the blackbox test makes use of our approach 3.

prop.test(15, 20, 1/4, alternative="greater")

 1-sample proportions test with continuity correction

data: 15 out of 20, null probability 1/4
X-squared = 24.067, df = 1, p-value = 4.653e-07
alternative hypothesis: true p is greater than 0.25
95 percent confidence interval:
 0.5417417 1.0000000
sample estimates:
 p
0.75

**Conclusion:** With a p-value less than 0.00001, we reject the claim that the two students chose their answers randomly.

(Note the dangers of announcing such a p-value and conclusion to quantitatively illiterate people as was done by a colleague at a different liberal arts college in a similar case of academic dishonesty.)

## Steel Bar Improvement Amounts

A steel manufacturer has been producing bars with a yield strength that is normally distributed with a mean of 900 MPa and a standard deviation of 50 MPa. The manufacturer hopes that a modified process will increase the mean yield strength of bars to 950 MPa and decrease the standard deviation. The new process is used to produce 15 bars whose yield strengths in MPa are given below.

steel = tibble(x = c(961, 962, 942, 934, 939, 920, 980, 972, 905, 985, 897, 970, 948, 888, 930))

What is the mean yield strength of bars made by the new process? What is the standard deviation yield strength made with the new process? Has the manufacturer’s hopes been fulfilled? (Based on a text book example but is typical of real-world questions.)

**Description:** Before an inferential analysis is carried out, always start with a descriptive analysis.

n = length(steel$x)
mu.hat = mean(steel$x)
sigma.hat = sd(steel$x)
tibble(n, mu.hat, sigma.hat)

# A tibble: 1 × 3
 n mu.hat sigma.hat
 <int> <dbl> <dbl>
1 15 942. 30.2

ggplot(steel, aes(x = x)) +
 geom\_histogram(aes(y = after\_stat(density)), bins = 5) +
 geom\_density(color = "red", linewidth = 2) +
 ggtitle("Steel Bars") +
 xlab("Yield Strength (MPa)")



The sample mean is close to 950 MPa, and the sample standard deviation is less than 50 MPa. So, it looks like the manufacturer’s hopes may have been fulfilled. It also looks like the yield strengths may be normally distributed. To be sure, we need to use inferential techniques.

**Mean Resampling Approach:** Use the data as the best estimate of the population.

stats = replicate(10000, mean(sample(steel$x, n, replace = TRUE)))
quantile(stats, c(0.025,0.975))

 2.5% 97.5%
927.2000 956.3333

**Mean Classical Approach:** Find the values for the null hypothesized mean yield strength that would not be rejected. For the test of vs. , the p-value if is

and the p-value if is

Since we want the p-value to be at least , it follows that

which is equivalent to

which is equivalent to

.

Unfortunately, we do not have the population standard deviation, and so we will use the sample standard deviation as an estimate.

mu.hat + qnorm(c(0.025,0.975))\*sigma.hat/sqrt(n)

[1] 926.9075 957.4925

The above assumes that is sufficiently large for the CLT to work and to allow an estimation of by . A more acceptable approach when is unknown is to use the theorem that if the population is normally distributed with mean , then has a Student-t distribution with degrees of freedom. To pursue this approach, we must first check whether the population can be assumed to be normally distributed. The following qq-plot of the actual sample versus a normal random variable with the sample mean and standard deviation indicates that the data could have come from a normal distribution.

ggplot(steel, aes(sample = x)) +
 stat\_qq(distribution = function(p) qnorm(p, mu.hat, sigma.hat)) +
 geom\_abline(slope = 1, intercept = 0, color = "red")



The following comparison of the empirical cdf with a normal cdf having the sample mean and standard deviation indicates that the data could have come from a normal distribution.

ggplot(steel, aes(x = x)) +
 stat\_ecdf() +
 geom\_function(fun = pnorm, args = c(mu.hat, sigma.hat), color = "red")



This code carries out two formal tests of the null hypothesis that the data came from a normal random variable.

nortest::ad.test(steel$x)

 Anderson-Darling normality test

data: steel$x
A = 0.2267, p-value = 0.7758

shapiro.test(steel$x)

 Shapiro-Wilk normality test

data: steel$x
W = 0.95681, p-value = 0.6372

With a p-value of 0.78 based on the Anderson-Darling normalilty test, it is reasonable to posit that the data came from a normal population.

Some statisticians argue that the qq-plot is a better approach than an inferential test because (1) for small sample sizes almost no data sets can be shown to not arise from a normal distribution, and (2) for large sample sizes data that may be statistically different from normality may nevertheless be sufficiently close to normality for other inferential tests and confidence intervals to remain reliable.

Now the previous derivation of a mean confidence interval can be repeated with replacing and replacing .

mu.hat + qt(c(0.025,0.975), df = n-1)\*sigma.hat/sqrt(n)

[1] 925.4654 958.9346

**Standard Deviation Resampling Approach:** Use the data as the best estimate of the population.

stats = replicate(10000, sd(sample(steel$x, replace = TRUE)))
quantile(stats, c(0.025,0.975))

 2.5% 97.5%
20.18429 36.39712

**Standard Deviation Classical Fail to Reject Approach:** We use the theorem that says if the population is normally distributed with variance , then

has a chi-squared distribution with degrees of freedom. For the hypotheses test

 vs.,

the p-value is

 if and if .

This is equivalent to

 if , and if .

For the p-value to be at least , we must have

 if and if .

Finally, this is equivalent to

sigma.hat\*sqrt((n-1)/qchisq(c(0.975,0.025), df=n-1))

[1] 22.12393 47.65792

**Standard Deviation Sampling Distribution from a Normal Population Approach:** An alternate argument is to let be our best estimate of for our normally distributed population. By the earlier theorem,

which is equivalent to

sigma.hat\*sqrt(qchisq(c(0.025,0.975), df=n-1)/(n-1))

[1] 19.16096 41.27528

**Summary:** With 95% confidence from a resampling approach, the mean yield strength of the steel bars is between 927 MPa and 957 MPa, and the standard deviation yield strength of the steel bars is between 20 MPa and 36 MPa. With a significance level of 0.05, we can conclude that the mean yield strength has increased from 900 MPa, the mean yield strength may now be 950 MPa, the distribution of yield strengths may be normal, and the standard deviation yield strength has decreased from 50 MPa.

Here is a comparison of the results obtained using our different approaches.

## Penny Side Flips

What is the probability of heads when a penny is balanced on its edge and then the surface is jostled? When this was done 100 times, 64 heads appeared.

**Answer:** Formulate as a hypothesis test.

: Probability of heads

: Probability of heads

**Exact Approach.** Let be the number of heads in a sample of 100 flips given . Then is binomial with and . The p-value is

For a 95% confidence interval, we need to find values of for which the p-value is at least 0.05.

pvalue = function(p0) 1 - pbinom(100\*p0 + abs(64-100\*p0), 100, p0) + pbinom(100\*p0 - abs(64-100\*p0), 100, p0)
ggplot() + geom\_function(fun = pvalue, xlim = c(0.53,0.55))



ggplot() + geom\_function(fun = pvalue, xlim = c(0.72,0.74))



With 95% confidence, the probability of heads for this type of randomizing is between 0.545 and 0.730.

**Approximation Approach:** Let be the number of heads during flips given . Since the population is discrete with probability of obtaining and probability of obtaining , the population is far from normally distributed. Nonetheless, may be sufficiently large to use the Central Limit Theorem.

n = 100; mu.hat = 64/100; sigma.hat = sqrt(0.64\*(1-0.64))
mu.hat + qnorm(c(0.025,0.975))\*sigma.hat/sqrt(n)

[1] 0.5459217 0.7340783

The confidence interval obtained is similar to the exact answer because is large and the of interest are not too close to or . The statistic assumes we know the population standard deviation, but we have actually used the sample standard deviation as an estimate for the population standard deviation. To avoid this concern, our normal recourse is to use the Student-t distribution; however, the population (1 with probablity and 0 with probability ) definitely does not have a normal distribution. On the other hand, is large enought that the Student-t is approximately normal, and so using Student-t yields a nearly identical result.

n = 100; mu.hat = 64/100; sigma.hat = sqrt(0.64\*(1-0.64))
mu.hat + qt(c(0.025,0.975), n-1)\*sigma.hat/sqrt(n)

[1] 0.5447576 0.7352424

## Teacher Differences

. Zumbrun surveyed 141 secondary mathematics teachers. She divided the teachers into two groups based on whether the teacher had ever taught a stand-alone high school statistics course. Each teacher was also asked several questions that had quantitative answers (sometimes via a 1-7 Likert-type scale). A summary of some of the results are given in the table below. For which questions were there statistically significant differences between the two groups? (Christina Zumbrun, Attitudes and Beliefs Towards Statistics: What are They and Why Do They Matter to All Teachers? Goshen College Science Speakers Presentation, April 4, 2018)

Notice that for this scenario we have not been given the data, only various statistics. This is enough to calculate confidence intervals, and based on those confidence intervals, obtain hypothesis test conclusions.

Suppose is a sample from population having mean and variance , suppose is a sample from population having mean and variance , and suppose that the two samples are independent of each other. By our basic results, we know that and . If the populations are normal, the sampling distribution of is also normal. If the sample sizes are sufficiently large, the Central Limit Theorem implies that the sampling distribution of is approximately normal. Under these conditions,

where is chosen so that where is standard normal. With some algebra, we obtain the confidence interval

.

This confidence interval can be used if the population variances are known or the sample sizes are large enough so that estimating the population standard deviations from the sample is okay.

mean.diff.ciz = function(n1, m1, s1, n2, m2, s2, alpha)
 (m1-m2) + c(-1,1)\*qnorm(1-alpha/2)\*sqrt(s1^2/n1 + s2^2/n2)
mean.diff.ciz(112, 6.33, 0.64, 27, 6.52, 0.51, 0.05)

[1] -0.41595313 0.03595313

If the sample sizes are not sufficiently large but we can assume that the populations are normal with the same variances (), then a confidence interval can be calculated via a Student-t distribution.

mean.diff.cit = function(n1, m1, s1, n2, m2, s2, alpha)
 (m1-m2) + c(-1,1)\*qt(1-alpha/2, n1+n2-2) \* sqrt((n1-1)\*s1^2 + (n2-1)\*s2^2)/sqrt(n1+n2-2)\*sqrt(1/n1+1/n2)
mean.diff.cit(112, 6.33, 0.64, 27, 6.52, 0.51, 0.05)

[1] -0.45176413 0.07176413

We can determine whether the variances are significantly different by computing a confidence interval for the ratio of the variances. This one assumes that the populations are normal.

var.ratio.cif = function(n1, s1, n2, s2, alpha)
 c(1/qf(1-alpha/2, n1-1, n2-1), qf(1-alpha/2, n2-1, n1-1)) \* s1^2/s2^2
var.ratio.cif(112, 0.64, 27, 0.51, 0.05)

[1] 0.803281 2.745446

var.ratio.cif(112, 1.68, 27, 1.67, 0.05)

[1] 0.5162184 1.7643263

var.ratio.cif(112, 0.89, 27, 0.88, 0.05)

[1] 0.5217501 1.7832324

var.ratio.cif(112, 1.10, 27, 1.33, 0.05)

[1] 0.3489233 1.1925466

var.ratio.cif(112, 1.25, 27, 1.22, 0.05)

[1] 0.5354862 1.8301793

For the Zumbrun data, none of the variances are significantly different at the 0.05 level of significance (note that a two-way test with a 0.05 level of significance uses a 95% confidence interval).

mean.diff.cit(112, 6.33, 0.64, 27, 6.52, 0.51, 0.1)

[1] -0.4092213 0.0292213

mean.diff.cit(112, 3.40, 1.68, 27, 4.63, 1.67, 0.1)

[1] -1.8258132 -0.6341868

mean.diff.cit(112, 2.38, 0.89, 27, 2.37, 0.88, 0.1)

[1] -0.3053245 0.3253245

mean.diff.cit(112, 4.73, 1.10, 27, 5.67, 1.33, 0.1)

[1] -1.3473144 -0.5326856

mean.diff.cit(112, 5.68, 1.25, 27, 6.41, 1.22, 0.1)

[1] -1.1718118 -0.2881882

For the Zumbrun data, the mean answers on three of the five questions is significantly higher for the teachers who had taught a stand-alone statistics course at the 0.05 level of significance (note that a one-way test with a 0.05 level of significance uses a 90% confidence interval or a 95% one-way confidence interval).

**Remark 1.** All of the above analysis assumes that the Likert scale used for respondents is an interval scale: respondents considered the difference between two successive numerical answers as being equally distant.

**Remark 2.** If we adopt a level of significance of and we ask two samples of respondents from the same population 100 questions, then we would expect to find significant differences between the two groups on about 5 of the 100 questions. As statisticians, our choice of tells us how often we will incorrectly reject a null hypothesis in our career. Unlike theologians and mathematicians who can imagine a career in which no errors are made, statisticians and experimental scientists know with near certainty that a certain percentage of their professional conclusions will be wrong!

If a lab tests thousands of chemical combinations for some beneficial property, we expect many to show that beneficial property even if none have that beneficial property. Labs guard against such false positives or Type I errors (rejecting a true null hypothesis) by effectively choosing much smaller levels of significance. Of course, this means that truly beneficial chemicals will not be discerned (often called Type II errors). A test that reduces the probability of a Type II error while keeping the probability of a Type I error fixed is considered to have greater power.

## Cricket Chirp Rates and Temperature Relationship

Consider the following data of the air temperature in degrees Fahrenheit (t) and cricket chirp rate in chirps per minute (c). What, if any, linear relationship exists between these two variables? How precise are predictions made by the model?

crickets = tibble(
 t = c(46, 51, 54, 57, 59, 61, 63, 66, 68, 72),
 c = c(40, 55, 72, 77, 90, 96, 99, 113, 127, 132))

**Answer:** The first decision to make is whether to think of (1) chirp rate as a function of temperature, or (2) temperature as a function of chirp rate. Since it seems more reasonable to think that temperature changes cause chirp rate changes rather than chirp rate changes cause temperature changes, (1) seems more natural. On the other hand, we might be interested in predicting temperature by listening to crickets chirp, in which case (2) would be more natural. So, let us do the analysis both ways.

We have used the following code previously to display a scatter plot with the regression line, which is the line that minimizes the sum of the squares of the residuals (the differences of the dependent variable values of the data and the corresponding predictions given by the line).

ggplot(crickets, aes(x = t, y = c)) +
 geom\_point() +
 geom\_smooth(method = "lm", se = FALSE)

`geom\_smooth()` using formula = 'y ~ x'



The graphs show us that linear equations are good models for the data. The following code finds important statistics.

model1 = lm(c ~ t, data = crickets)
model1$coefficients

(Intercept) t
-130.969276 3.703003

t0 = unname(-model1$coefficients[1]/model1$coefficients[2]); t0

[1] 35.3684

n = length(crickets$c)
sse1 = sum((model1$residuals)^2)
sst1 = sum((crickets$c - mean(crickets$c))^2)
rsq1 = 1 - sse1/sst1; rsq1

[1] 0.9878348

se1 = sqrt(sse1/(n-2)); se1

[1] 3.487187

Let be the chirp rate in chirps per minute and be the temperature in degrees Fahrenheit. The first regression line can be expressed by the equation or , has a coefficient of determination (or R-squared) of 0.988, and a standard error of 3.49. One interpretation is that crickets only begin to chirp at a temperature of 35 degrees Fahrenheit, an increase of 1 degree Fahrenheit is associated with a 3.7 chirps per minute increase in the chirp rate, the model predicts 99% of the variation in the chirp rate is based on the temperature, and model predictions deviate from true values by 3.5 chirps per minute on average.

model2 = lm(t ~ c, data = crickets)
model2$coefficients

(Intercept) c
 35.6643950 0.2667659

cmean = mean(crickets$c); cmean

[1] 90.1

tmean = mean(crickets$t); tmean

[1] 59.7

n = length(crickets$t)
sse2 = sum((model2$residuals)^2)
sst2 = sum((crickets$t - mean(crickets$t))^2)
rsq2 = 1 - sse2/sst2; rsq2

[1] 0.9878348

se2 = sqrt(sse2/(n-2)); se2

[1] 0.9359731

The second regression line can be expressed by the equation or , has a coefficient of determination (or R-squared) of 0.988, and a standard error of 0.94. One interpretation is that on average the crickets were chirping 90 chirps per minute and the temperature was 60 degrees Fahrenheit, an increase of 1 chirp per minute is associated with a 0.27 degree Fahrenheit increase in the temperature, the model predicts 99% of the variation in the temperature based on the chirp rate, and model predictions deviate from true values by 1 degree Fahrenheit on average.

**Remark.** Note that the product of the slopes equals the coefficient of determination: 3.703003 \* 0.2667659 = 0.9878348. This is not a coincidence. Recall that the regression equation is

where

can be seen to be symmetric in and . Thus, the product of the two regression line slopes is

.

This should make it clear that to find the regression line when the variables are interchanged, one cannot simply solve the first regression equation for the other variable.

**Inferential Answer.** Inferential information can also be obtained from the model. It is assumed that the chirp rate is related to a temperature via the equation , where , , and are population parameters and is a standard normal random variable. The summary function provides inferential information about the population parameters.

summary(model1)

Call:
lm(formula = c ~ t, data = crickets)

Residuals:
 Min 1Q Median 3Q Max
-3.6469 -3.0474 0.1011 2.1406 6.1651

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) -130.9693 8.7434 -14.98 3.90e-07 \*\*\*
t 3.7030 0.1453 25.49 6.02e-09 \*\*\*
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.487 on 8 degrees of freedom
Multiple R-squared: 0.9878, Adjusted R-squared: 0.9863
F-statistic: 649.6 on 1 and 8 DF, p-value: 6.018e-09

Three p-values are reported. With a p-value of , . With a p-value of , . With a p-value of , the model is significantly different from a constant function (which in this case is equivalent to ), Two confidence intervals can be obtained from the coeeficient standard errors. With 68% confidence, and . Of course, we can obtain 95% confidence intervals by multiplying the standard errors by 1.96.

The predict function provides inferential information about prediction and mean responses.

new.data = tibble(t = 60)
predict(model1, newdata = new.data, interval = 'prediction')

 fit lwr upr
1 91.2109 82.77634 99.64546

predict(model1, newdata = new.data, interval = 'confidence')

 fit lwr upr
1 91.2109 88.66598 93.75582

With 95% confidence (this is the default value, which can be changed with the level parameter), if the temperature is 60 degrees Fahrenheit, the chirp rate is between 83 and 100 chirps per minute amd the mean chirp rate is between 89 and 94 chirps per minute. The following code superimposes the prediction (in red) and mean response (in green) 95% confidence intervals on the scatter (in black) and regression line (in blue) plot. Observe that the gray shading that comes from the default s = TRUE is the 95% confidence interval for the mean response.

inferences = tibble(
 t = seq(45, 74),
 pred.lwr = predict(model1, newdata = tibble(t), interval='prediction')[,'lwr'],
 pred.upr = predict(model1, newdata = tibble(t), interval='prediction')[,'upr'],
 conf.lwr = predict(model1, newdata = tibble(t), interval='confidence')[,'lwr'],
 conf.upr = predict(model1, newdata = tibble(t), interval='confidence')[,'upr']
)
ggplot(crickets, aes(x = t, y = c)) +
 geom\_point() +
 geom\_smooth(method = "lm") +
 geom\_line(aes(x = t, y = pred.lwr), inferences, color = "red") +
 geom\_line(aes(x = t, y = pred.upr), inferences, color = "red") +
 geom\_line(aes(x = t, y = conf.lwr), inferences, color = "green") +
 geom\_line(aes(x = t, y = conf.upr), inferences, color = "green")

`geom\_smooth()` using formula = 'y ~ x'



Here is how to explore the quadratic model .

ggplot(crickets, aes(x = t + I(t^2), y = c)) +
 geom\_point() +
 geom\_smooth(method = "lm")

`geom\_smooth()` using formula = 'y ~ x'



model3 = lm(c ~ t + I(t^2), data = crickets)
summary(model3)

Call:
lm(formula = c ~ t + I(t^2), data = crickets)

Residuals:
 Min 1Q Median 3Q Max
-3.5505 -3.0044 0.1544 1.9445 6.2818

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.495e+02 6.652e+01 -2.248 0.0594 .
t 4.342e+00 2.273e+00 1.911 0.0977 .
I(t^2) -5.409e-03 1.919e-02 -0.282 0.7862
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.707 on 7 degrees of freedom
Multiple R-squared: 0.988, Adjusted R-squared: 0.9845
F-statistic: 287.5 on 2 and 7 DF, p-value: 1.909e-07

Now none of the coefficients are significantly different from zero, although the model is significantly different from a constant function. The addition of the quadratic term has not significantly improved the model.

## GapMinder

In February 2006, a Swedish physician and data advocate named Hans Rosling gave a TED talk titled [“The best stats you’ve ever seen”](https://www.ted.com/talks/hans_rosling_shows_the_best_stats_you_ve_ever_seen) where he presented global economic, health, and development data from the website [gapminder.org](http://www.gapminder.org/tools/#_locale_id=en;&chart-type=bubbles). Describe lifeExp (life expectancy in years) as a function of continent (note that “Americas” includes countries in both North and South America and that Antarctica is excluded), pop (number of people living in the country), and/or gdpPercap (gross domestic product in US dollars).

gapminder\_2007 <- gapminder %>%
 filter(year == 2007) %>%
 select(-year)
gapminder\_2007 %>%
 slice\_sample(n = 5)

# A tibble: 5 × 5
 country continent lifeExp pop gdpPercap
 <fct> <fct> <dbl> <int> <dbl>
1 Spain Europe 80.9 40448191 28821.
2 Equatorial Guinea Africa 51.6 551201 12154.
3 Haiti Americas 60.9 8502814 1202.
4 Sierra Leone Africa 42.6 6144562 863.
5 Argentina Americas 75.3 40301927 12779.

The following is a plot showing all of these variables. This is what is shown in the video but dynamically for each year.

ggplot(gapminder\_2007, aes(y = lifeExp, x = pop, size = gdpPercap, color = continent)) +
 geom\_point() +
 scale\_x\_continuous(trans='log10')



**Life expectancy as a function of continent.**

ggplot(gapminder\_2007, aes(x = continent, y = lifeExp)) +
 geom\_boxplot()



model1 = lm(lifeExp ~ continent, data = gapminder\_2007)
summary(model1)

Call:
lm(formula = lifeExp ~ continent, data = gapminder\_2007)

Residuals:
 Min 1Q Median 3Q Max
-26.9005 -4.0399 0.2565 3.3840 21.6360

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) 54.806 1.025 53.446 < 2e-16 \*\*\*
continentAmericas 18.802 1.800 10.448 < 2e-16 \*\*\*
continentAsia 15.922 1.646 9.675 < 2e-16 \*\*\*
continentEurope 22.843 1.695 13.474 < 2e-16 \*\*\*
continentOceania 25.913 5.328 4.863 3.12e-06 \*\*\*
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.395 on 137 degrees of freedom
Multiple R-squared: 0.6355, Adjusted R-squared: 0.6249
F-statistic: 59.71 on 4 and 137 DF, p-value: < 2.2e-16

The model is Life expectancy = 54.8 + 18.8IAmericas + 15.9IAsia + 22.8IEurope + 25.9IOceania. Already 63.5% of the variation in life expectancy is associated with the continent in which the country is located. Rather than having to recognize that the model says that the mean life expectancy in Africa is 54.8 years, 54.8 + 18.8 = 73.6 years in the Americas, etc., we can have R do the calculations by removing the intercept term.

model2 = lm(lifeExp ~ 0 + continent, data = gapminder\_2007)
summary(model2)

Call:
lm(formula = lifeExp ~ 0 + continent, data = gapminder\_2007)

Residuals:
 Min 1Q Median 3Q Max
-26.9005 -4.0399 0.2565 3.3840 21.6360

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
continentAfrica 54.806 1.025 53.45 <2e-16 \*\*\*
continentAmericas 73.608 1.479 49.77 <2e-16 \*\*\*
continentAsia 70.728 1.287 54.95 <2e-16 \*\*\*
continentEurope 77.649 1.350 57.52 <2e-16 \*\*\*
continentOceania 80.719 5.229 15.44 <2e-16 \*\*\*
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.395 on 137 degrees of freedom
Multiple R-squared: 0.9886, Adjusted R-squared: 0.9882
F-statistic: 2380 on 5 and 137 DF, p-value: < 2.2e-16

Recall that for models with no intercept, the errors are computed about 0 rather than the mean of the response data. So, the reported coefficient of determination is not meaningful in this context.

**Life expectancy as a function of population and continent.**

It is useful to first note that there are some countries with very large populations in comparison with most of the countries.

ggplot(gapminder\_2007, aes(x = pop)) + geom\_histogram()

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



This suggests that the logarithm of the population be plotted on the horizontal axis, and perhaps even the variable itself should be the logarithm of the population.

ggplot(gapminder\_2007, aes(x = pop, y = lifeExp, color = continent)) +
 geom\_point() +
 geom\_smooth(method = "lm", formula = y ~ log10(x)) +
 scale\_x\_continuous(trans='log10')

Warning in qt((1 - level)/2, df): NaNs produced

Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
-Inf



Notice that in the following model, all of the lines have the same slope.

model3 = lm(lifeExp ~ continent + log10(pop), data = gapminder\_2007)
summary(model3)

Call:
lm(formula = lifeExp ~ continent + log10(pop), data = gapminder\_2007)

Residuals:
 Min 1Q Median 3Q Max
-26.7945 -4.3175 0.3774 3.5919 20.5331

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) 62.5396 6.8715 9.101 9.85e-16 \*\*\*
continentAmericas 19.0599 1.8119 10.519 < 2e-16 \*\*\*
continentAsia 16.5130 1.7239 9.579 < 2e-16 \*\*\*
continentEurope 22.9552 1.6964 13.532 < 2e-16 \*\*\*
continentOceania 26.0019 5.3232 4.885 2.87e-06 \*\*\*
log10(pop) -1.1235 0.9871 -1.138 0.257
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.387 on 136 degrees of freedom
Multiple R-squared: 0.6389, Adjusted R-squared: 0.6257
F-statistic: 48.13 on 5 and 136 DF, p-value: < 2.2e-16

The model used by geom\_smooth follows.

model4 = lm(lifeExp ~ continent \* log10(pop), data = gapminder\_2007)
summary(model4)

Call:
lm(formula = lifeExp ~ continent \* log10(pop), data = gapminder\_2007)

Residuals:
 Min 1Q Median 3Q Max
-26.6603 -3.7753 0.2446 3.2017 20.1320

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) 65.353 11.336 5.765 5.49e-08 \*\*\*
continentAmericas -2.872 21.678 -0.133 0.895
continentAsia 24.246 17.418 1.392 0.166
continentEurope 12.266 20.463 0.599 0.550
continentOceania 5.052 106.019 0.048 0.962
log10(pop) -1.532 1.640 -0.934 0.352
continentAmericas:log10(pop) 3.097 3.065 1.010 0.314
continentAsia:log10(pop) -1.015 2.418 -0.420 0.675
continentEurope:log10(pop) 1.536 2.933 0.524 0.601
continentOceania:log10(pop) 3.014 15.210 0.198 0.843
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.441 on 132 degrees of freedom
Multiple R-squared: 0.6444, Adjusted R-squared: 0.6202
F-statistic: 26.58 on 9 and 132 DF, p-value: < 2.2e-16

The addition of population to continent provides no discernible increase in the explanatory power of a model to explain life expectancy.

**Life expectancy as a function of GDP per capita and continent.**

ggplot(gapminder\_2007, aes(y = lifeExp, x = pop, size = gdpPercap, color = continent)) + geom\_point() + scale\_x\_continuous(trans=‘log10’)

ggplot(gapminder\_2007, aes(y = lifeExp, x = gdpPercap, color = continent)) +
 geom\_point() +
 geom\_smooth(method = "lm", formula = y ~ log10(x)) +
 scale\_x\_continuous(trans='log10')

Warning in qt((1 - level)/2, df): NaNs produced

Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
-Inf



GDP per capita adds some explanatory power to the model.

model5 = lm(lifeExp ~ continent + gdpPercap, data = gapminder\_2007)
summary(model5)

Call:
lm(formula = lifeExp ~ continent + gdpPercap, data = gapminder\_2007)

Residuals:
 Min 1Q Median 3Q Max
-22.9145 -2.8518 0.1407 2.8881 20.0479

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) 5.374e+01 9.284e-01 57.881 < 2e-16 \*\*\*
continentAmericas 1.606e+01 1.662e+00 9.663 < 2e-16 \*\*\*
continentAsia 1.267e+01 1.557e+00 8.137 2.27e-13 \*\*\*
continentEurope 1.523e+01 1.956e+00 7.786 1.57e-12 \*\*\*
continentOceania 1.665e+01 4.974e+00 3.348 0.00105 \*\*
gdpPercap 3.466e-04 5.674e-05 6.109 9.89e-09 \*\*\*
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.574 on 136 degrees of freedom
Multiple R-squared: 0.714, Adjusted R-squared: 0.7035
F-statistic: 67.9 on 5 and 136 DF, p-value: < 2.2e-16

The interaction terms do not add much explanatory power to the model.

model6 = lm(lifeExp ~ continent \* gdpPercap, data = gapminder\_2007)
summary(model6)

Call:
lm(formula = lifeExp ~ continent \* gdpPercap, data = gapminder\_2007)

Residuals:
 Min 1Q Median 3Q Max
-22.4409 -2.6339 0.0137 2.8867 16.9448

Coefficients:
 Estimate Std. Error t value Pr(>|t|)
(Intercept) 51.6427770 1.1791430 43.797 < 2e-16 \*\*\*
continentAmericas 18.9923912 2.2949884 8.276 1.23e-13 \*\*\*
continentAsia 14.2481564 1.9114403 7.454 1.07e-11 \*\*\*
continentEurope 20.6282785 3.0364687 6.794 3.40e-10 \*\*\*
continentOceania 25.7542242 29.7251403 0.866 0.38784
gdpPercap 0.0010240 0.0002492 4.109 6.95e-05 \*\*\*
continentAmericas:gdpPercap -0.0007538 0.0002836 -2.658 0.00883 \*\*
continentAsia:gdpPercap -0.0006362 0.0002619 -2.429 0.01648 \*
continentEurope:gdpPercap -0.0008094 0.0002691 -3.008 0.00315 \*\*
continentOceania:gdpPercap -0.0009126 0.0010156 -0.899 0.37054
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.44 on 132 degrees of freedom
Multiple R-squared: 0.7336, Adjusted R-squared: 0.7154
F-statistic: 40.39 on 9 and 132 DF, p-value: < 2.2e-16

## World Swim Records

Consider the data in swim.csv: world record times in the 100 meter freestyle long course swimming race. The variable year is the year in which the record was set, time is the record time in seconds, and sex is whether this is the women’s or the men’s record. Only the last instance of a record breaking time is recorded in each year. The information was gathered from https://en.wikipedia.org/wiki/World\_record\_progression\_100\_metres\_freestyle. Find a reasonable model for this data.

The following code creates a scatter plot grouped by sex, and superimposes regression lines.

swim = read\_csv("swim.csv")

Rows: 68 Columns: 3
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (1): sex
dbl (2): year, time

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

ggplot(swim, aes(x = year, y = time, color = sex)) +
 geom\_point() +
 geom\_smooth(method = "lm", se = FALSE)

`geom\_smooth()` using formula = 'y ~ x'



The trends of the data are clearly nonlinear, and linear models do not make sense theoretically (e.g., record times could never become negative). A simple model that is decreasing and has a positive lower bound is of the form

where is the year and is the time. The parameter corresponds to the limiting lower bound and would be a vertical asymptote occuring before the start of our data. Observe that the parameters are not linear in the formula, and so the nls (nonlinear least squared) function must be used instead of the lm (linear model) function. While linear models have explicit formulas for finding the parameter values (think of generalizing what we did at the beginning of the semester), nonlinear models must use numerical approximations which involve educated guesses and checks (think of Newton’s method for finding the roots of a function). The nls function requires the user to provide an initial guess for the start argument.

f.model = nls(time ~ a/(year-b) + c, data = swim |> filter(sex == "F"), start = list(a = 450, b = 1890, c = 45)); f.model

Nonlinear regression model
 model: time ~ a/(year - b) + c
 data: filter(swim, sex == "F")
 a b c
 945.5 1886.9 45.7
 residual sum-of-squares: 83.52

Number of iterations to convergence: 6
Achieved convergence tolerance: 6.242e-06

m.model = nls(time ~ a/(year-b) + c, data = swim |> filter(sex == "M"), start = list(a = 200, b = 1890, c = 45)); m.model

Nonlinear regression model
 model: time ~ a/(year - b) + c
 data: filter(swim, sex == "M")
 a b c
5341.53 1772.56 24.35
 residual sum-of-squares: 23.76

Number of iterations to convergence: 9
Achieved convergence tolerance: 2.455e-06

pred = tibble(
 year = seq(1900, 2020),
 f.time = predict(f.model, newdata = tibble(year)),
 m.time = predict(m.model, newdata = tibble(year))
)
ggplot(swim, aes(x = year, y = time, color = sex)) +
 geom\_point() +
 geom\_line(aes(x = year, y = f.time), data = pred, color = "red") +
 geom\_line(aes(x = year, y = m.time), data = pred, color = "blue")



The model curves look reasonable where the data are located; however, the limiting record of 24.35 seconds for males seems totally unreasonable while the limiting record of 45.7 seconds for females seems only somewhat incredible. We should look for other more reasonable models. We will not pursue this further in this course but note that confidence intervals can be found for the population parameters:

summary(f.model)

Formula: time ~ a/(year - b) + c

Parameters:
 Estimate Std. Error t value Pr(>|t|)
a 945.489 111.021 8.516 7.68e-10 \*\*\*
b 1886.930 2.321 812.934 < 2e-16 \*\*\*
c 45.704 1.110 41.173 < 2e-16 \*\*\*
---
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.591 on 33 degrees of freedom

Number of iterations to convergence: 6
Achieved convergence tolerance: 6.242e-06

Predictions can be made (but without confidence intervals):

predict(f.model, newdata = tibble(year = 2024))

[1] 52.60234

The coefficient of determination can be calculated (alse SSE and model standard error which were also provided above):

f.swim = swim |> filter(sex == "F")
n = length(f.swim$year); n

[1] 36

sse = sum(residuals(f.model)^2); sse

[1] 83.51749

sst = sum((f.swim$time - mean(f.swim$time))^2)
rsq = 1 - sse/sst; rsq

[1] 0.9799725

se = sqrt(sse/(n-3)); se

[1] 1.590859

Notice that in computing the model standard error, we have divided by because three model parameters were estimated from the data.

## Coal Differences

Random samples of coal were taken from two mines and the heat-producing capacity (in millions of calories per ton) for each specimen was obtained. (This data is from Richard A. Johnson, *Miller & Freund’s Probability and Statistics for Engineers*.)

mine1 = c(8260, 8130, 8350, 8070, 8340)
mine2 = c(7950, 7890, 7900, 8140, 7920, 7840)

What is the difference between the mean heat-producing capacities of the two mines?

**Data Wrangling.** The tidyverse recommends that the data always be placed in a tibble, and here a case would be a coal chunk.

coal = tibble(
 mine = c(rep(1, length(mine1)), rep(2, length(mine2))),
 heat = c(mine1, mine2)
)

**Description Answer.** Certainly the difference in the sample means would be a best estimate for the difference in the population means, and a good visual for such a small data set is side-by-side boxplots.

coal |> group\_by(mine) |> summarise(mean = mean(heat)) |>summarise(diff = diff(mean))

# A tibble: 1 × 1
 diff
 <dbl>
1 -290

ggplot(coal, aes(x = as.factor(mine), y = heat)) + geom\_boxplot()



**Resampling Inferential Answer.** Use the data as the best estimate of the population.

stats = replicate(10000, mean(sample(mine1, length(mine1), replace=TRUE)) - mean(sample(mine2, length(mine2), replace=TRUE)))
quantile(stats, c(0.025,0.975))

 2.5% 97.5%
159.0000 408.3333

With 95% confidence, the difference between the mean heat-producing capacities of mines 1 and 2 is between 160 and 410 millions of calories per ton.

**Classical Inferential Answer.** Assume the populations are normal and use a Student-t statistic.

We first ask whether it is reasonable to posit the distributions are normal. The following calculations shows that the answer is “yes.”

shapiro.test(mine1)

 Shapiro-Wilk normality test

data: mine1
W = 0.88995, p-value = 0.3569

shapiro.test(mine2)

 Shapiro-Wilk normality test

data: mine2
W = 0.81927, p-value = 0.08697

We next examine whether the standard deviations can be considered equal.

var.ratio.fci = function(n1, s1, n2, s2, alpha)
 c(1/qf(1-alpha/2, n1-1, n2-1), qf(1-alpha/2, n2-1, n1-1)) \* s1^2/s2^2
n1 = length(mine1)
s1 = sd(mine1)
n2 = length(mine2)
s2 = sd(mine2)
var.ratio.fci(n1, s1, n2, s2, 0.05)

[1] 0.195226 13.506448

Here is the R blackbox function.

var.test(mine1, mine2)

 F test to compare two variances

data: mine1 and mine2
F = 1.4423, num df = 4, denom df = 5, p-value = 0.6872
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.195226 13.506448
sample estimates:
ratio of variances
 1.442308

Since the distributions are not significantly different from normal and the variances are not significantly different, we may be justified in using the same confidence interval used in the *Zumbrun scenario*Teacher Differences\* section.

mean.diff.cit = function(n1, m1, s1, n2, m2, s2, alpha)
 (m1-m2) + c(-1,1)\*qt(1-alpha/2, n1+n2-2) \* sqrt((n1-1)\*s1^2 + (n2-1)\*s2^2)/sqrt(n1+n2-2)\*sqrt(1/n1+1/n2)
n1 = length(mine1)
m1 = mean(mine1)
s1 = sd(mine1)
n2 = length(mine2)
m2 = mean(mine2)
s2 = sd(mine2)
mean.diff.cit(n1, m1, s1, n2, m2, s2, 0.05)

[1] 133.4183 446.5817

R has a blackbox test and confidence interval for the difference of two means.

t.test(mine1, mine2)

 Welch Two Sample t-test

data: mine1 and mine2
t = 4.1136, df = 7.8588, p-value = 0.003505
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 126.9208 453.0792
sample estimates:
mean of x mean of y
 8230 7940

Notice that the confidence interval is wider because the default behavior is to not assume the variances are equal and use a more sophisticated approach. We can obtain the previous confidence interval by adding the parameter var.equal=TRUE.

With 95% confidence, the difference between the mean heat-producing capacities of mines 1 and 2 is between 130 and 450 millions of calories per ton.

## Facility Usage Differences

A student who used the college recreational facilities was interested in whether there is a difference between the facilities used by men and women. A randomly selected sample of users is summarized in the table. Note that I seem to have forgotten the labels for the table (men vs. women, and facility A vs. facility B), but this actually does not matter for a test of independence.

**Answer.** The table with row and column totals is as follows:

The row and column sums determine the expected number in each cell under the assumption that the two variables are independent. For example, the expected number of EC students would be .

The observed and expected frequencies are different in each cell. The inferential question is whether such a difference cuold have been due to chance alone. We need a statistic that measures the difference between the expectation () and observation () in the th cell. The classical statistic is

$*{(r-1)(c-1)}^2 =* {i=1}^{rc} $

where there are rows and columns. If the expected value of each cell is at least 5, then the chi-square statistic has approximately a chi-square distribution with degrees of freedom.

For our example,

observed = matrix(c(51, 30, 43, 48), nrow=2, byrow=TRUE); observed

 [,1] [,2]
[1,] 51 30
[2,] 43 48

expected = rowSums(observed) %o% colSums(observed) / sum(observed); expected

 [,1] [,2]
[1,] 44.26744 36.73256
[2,] 49.73256 41.26744

chisquare = sum((observed-expected)^2/expected); chisquare

[1] 4.267728

1 - pchisq(chisquare, 1)

[1] 0.03884284

With a p-value of 0.04, sex and college recreational facilities used are dependent variables.

**Simulation Approach.** .

nruns = 10000
stats = numeric(nruns)
for (i in 1:nruns) {
 scol = sample(3:4, 172, prob=c(94,78), replace=TRUE)
 srow = sample(5:6, 172, prob=c(81,91), replace=TRUE)
 obs = table(data.frame(srow,scol))
 exp = rowSums(obs) %o% colSums(obs) / sum(obs)
 stats[i] = sum((obs-exp)^2/exp)
}
pvalue = sum(stats >= chisquare) / nruns; pvalue

[1] 0.0421

**R Blackbox.** The first function mimics our approach above. The second applies the Yates continuity correction for 2 by 2 tables.

chisq.test(observed, correct=FALSE)

 Pearson's Chi-squared test

data: observed
X-squared = 4.2677, df = 1, p-value = 0.03884

chisq.test(observed)

 Pearson's Chi-squared test with Yates' continuity correction

data: observed
X-squared = 3.6574, df = 1, p-value = 0.05582

## MIT Suicides

The number of days between MIT student suicides from 10/8/64 to 6/15/91: 38, 336, 151, 444, 868, 269, 978, 27, 30, 9, 428, 366, 138, 52, 622, 169, 1772, 295, 204, 696, 139, 16, 347, 1, 19, 169, 7, 61, 0, 852, 243, 0. (data obtained from Elaine Chew and Philip Greenspun, Is Suicide at MIT a Poisson Process?, document found at http://philip.greenspun.com/research/suicide-at-mit.pdf.) Is the population exponentially distributed?

**Answer.** Estimate the mean number of suicides per day.

MIT = tibble(suicides = c(38, 336, 151, 444, 868, 269, 978, 27, 30, 9, 428, 366, 138, 52, 622, 169, 1772, 295, 204, 696, 139, 16, 347, 1, 19, 169, 7, 61, 0, 852, 243, 0))
n = length(MIT$suicides); n

[1] 32

lambda.hat = 1/mean(MIT$suicides); lambda.hat

[1] 0.003283398

Compare the data with an exponential distribution via a qq-plot.

ggplot(MIT, aes(sample = suicides)) +
 stat\_qq(distribution = function(p) qexp(p, lambda.hat)) +
 geom\_abline(slope = 1, intercept = 0, color = "red")



Compare the sample and claimed population cdfs.

ggplot(MIT, aes(x = suicides)) +
 stat\_ecdf() +
 geom\_function(fun = function(x) pexp(x, lambda.hat), color = "red")



The difference between the sample and claimed population cdf can be summarized with the Anderson-Darling statistic with the exponential parameter estimated from the data. Since there are some 0 data (which results in an infinite Anderson-Darling statistic), I added 1 to each time interval–not sure that this is totally valid, but it was an attempt to increase the p-value. So, the fact that the p-value was still small bolsters the inferential conclusion.

n = length(MIT$suicides)
lambda.hat = 1/mean(MIT$suicides+1)
y = sort(MIT$suicides+1)
AD = -n - sum((2\*(1:n)-1)/n\*(log(pexp(y, lambda.hat)) + log(1 - pexp(rev(y), lambda.hat)))); AD

[1] 2.429117

Is this a reasonable value for the Anderson-Darling statistic? To find out, we simulate samples from the claimed distribution, calculate their Anderson-Darling statistics, and determine the proportion of them at least as great as the Anderson-Darling statistic for the data.

set.seed(17)
runs = 1000
ad = numeric(runs)
for (i in 1:runs) {
 sample = rexp(n, lambda.hat)
 slambda.hat = 1/mean(sample+1)
 y = sort(sample+1)
 ad[i] = -n - sum((2\*(1:n)-1)/n\*(log(pexp(y, slambda.hat)) + log(1 - pexp(rev(y), slambda.hat))))
}
ggplot(tibble(ad = ad), aes(x = ad)) +
 geom\_density() +
 geom\_vline(xintercept = AD, color = "red")



prob = length(which(ad >= AD)) / runs; prob

[1] 0.004

goftest::ad.test(MIT$suicides, "pexp", lambda.hat, estimated = TRUE)

 Anderson-Darling test of goodness-of-fit
 Braun's adjustment using 6 groups
 Null hypothesis: exponential distribution
 Parameters assumed to have been estimated from data

data: MIT$suicides
Anmax = Inf, p-value = 0.0007198

goftest::ad.test(MIT$suicides+1, "pexp", 1/(1/lambda.hat+1), estimated = TRUE)

 Anderson-Darling test of goodness-of-fit
 Braun's adjustment using 6 groups
 Null hypothesis: exponential distribution
 Parameters assumed to have been estimated from data

data: MIT$suicides + 1
Anmax = 4.1206, p-value = 0.04924

With a p-value of 0.004 using the Anderson-Darling statistic, the number of days between MIT student suicides is not exponential. Based on the graphs, short time intervals between suicides occur disproportionately to short and long time intervals between suicides. Perhaps a recent suicide tends to encourage another suicide (but remember that statistics only suggests association, not causation).