Workshop on Quantitative Methods in Linguistics (WoQuMeL)

School of Languages and Linguistics | Jadavpur University | 22nd July - 24th July, 2024

Outline

- Objectives
- To learn statistical methods for quantitative analyses of linguistic data
- Approach empirical questions in linguistics from a model-theoretic approach
- Models, similar to theories, make predictions, but not always.
- We tweak them till we arrive at a model whose unpredictable aspects are within acceptable bounds

What we will be following

- Quantitative methods in Linguistics (Johnson 2008)
- Code and datsets related to Johnson (2008)
  - Code and data
- Statistics for Linguistics: An Introduction Using R (Winter 2020)

Topics that we will cover

- Descriptive statistics
  - mean
- Distributions
- Models
- Data visualization
- Summary Stats
- Linear Models
- Correlations
- Multiple Regressions

Working with R

- Basic R functions and packages
- Designing and building the statistical components of experiments
- Writing code and debugging

This document

- We are writing R code and associated content in Quarto
- Markdown flavor syntax
- Weaving r code and text in the same document

What statistic are and what they are not

- Statistical analyses lend validity
- We perform tests that allow us to either accept or reject the null hypothesis
- They give us a means to uncover causal relationships
- They are, however, not magic wands
- Each test and set of analyses are specific to the conditions, variables, nature and distribution of the data; so we decide first before we conduct the experiment what tests to perform NOT after

Statistical environment

- R because it is:

    1. a powerful statistics package, good at reading data, wide range of statistical tests and techniques, good graphics, very flexible
    2. a usable package available for many platforms (PC, Mac, Unix, Linux....) programmable user community for support 3.it is noncommercial - distributed under the GNU "copyleft", maintained by a community of users, upgrades happen because the users need improvements, not because the company needs more money.

- Where: R project page
- How:

    1. Go to the R project page,
    2. click the CRAN link to see the download servers on the Comprehensive R Archive Network,
    3. choose a download server near your location,
    4. choose your platform (Windows, Linux, Mac)

Describing data

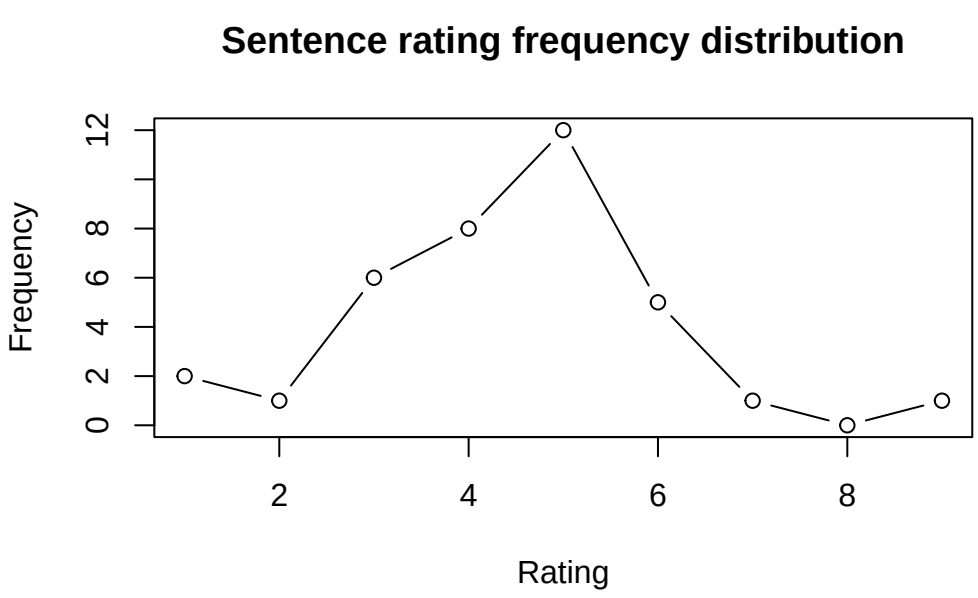
- Let's say we ask 36 people to score a sentence on a grammaticality scale So that a score of 1 means that it sounds pretty ungrammatical, and 10 sounds perfectly OK. A simple way of generating data in R

```
x=round(rnorm(36,4.5,2))
```

- *rnorm* needs some arguments: N, mean and the SD
- How many people gave the sentence a rating of "1"?
- How many rated it a "2"? When we answer these questions for all of the possible ratings we have the values that make up the *frequency distribution* of our sentence grammaticality ratings

Getting the frequency distribution

```
data = c(2,1,6,8,12,5,1,0,1)#c function to catenate individual values together
rating = c(1,2,3,4,5,6,7,8,9)
plot(rating,data,type = "b", main="Sentence rating frequency distribution",
     xlab = "Rating", ylab = "Frequency")
```



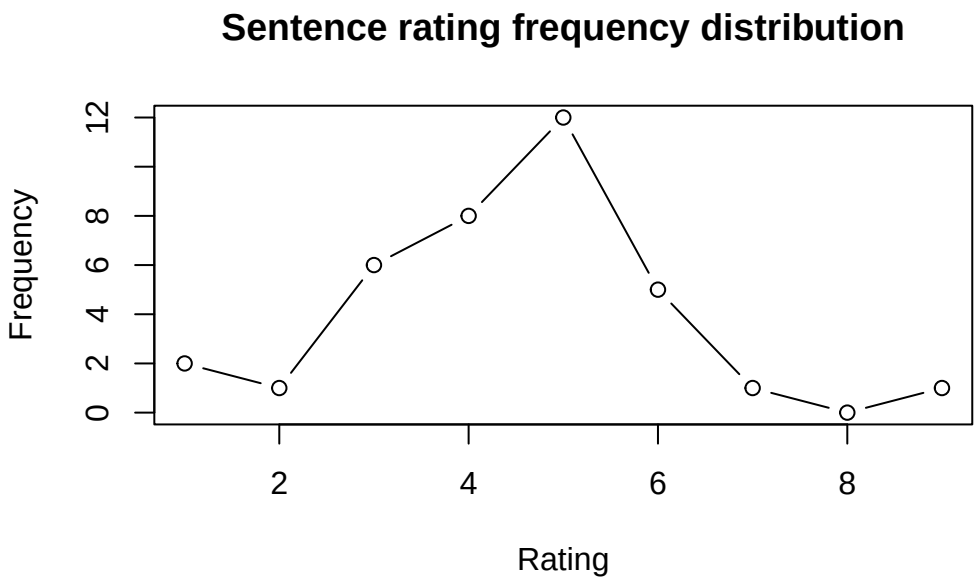
- Here we defined two vectors, data and rating, data is the frequency data of ratings, and rating refers to a vector of the rating scale
- How many people gave a particular sentence the rating of 5? Or how frequently was the rating 5 given?

What is a vector?

- Container vector
  - Ordered collection of numbers with no other structure
  - The length of a vector is the number of elements in the container.
- Operations are applied componentwise.
  - Given two vectors x and y of equal length, x*y would be the vector whose nth component is the product of the nth components of x and y.
  - log(x) would be the vector whose nth component is the logarithm of the nth component of x.

How informative are frequency distributions?

```
plot(rating,data,type = "b", main="Sentence rating frequency distribution",
     xlab = "Rating", ylab = "Frequency")
```

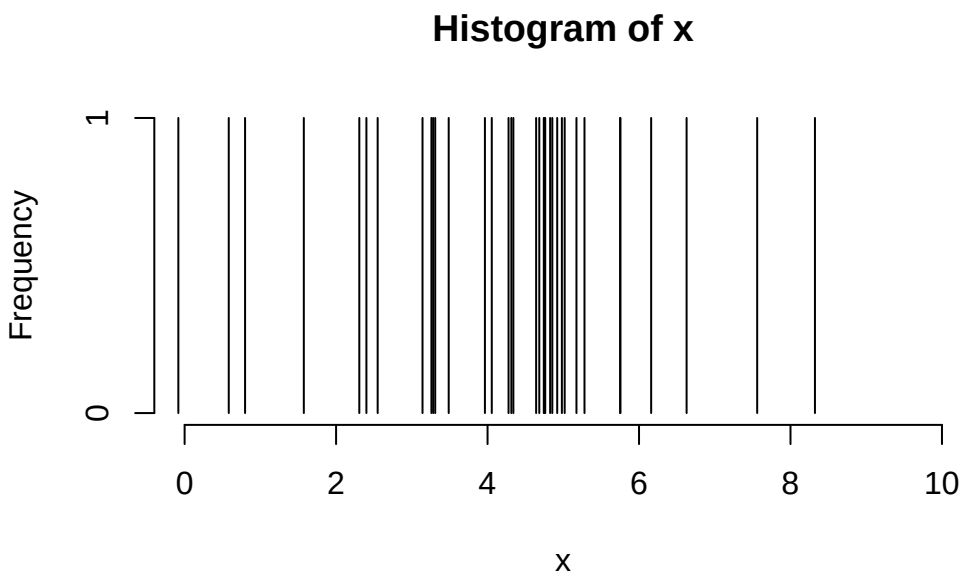


- Plotting rating and data gives us the frequency distribution
- Majority of subjects (12) rated the sentence to be 5 on the scale
- Few people rated the sentence to be absolutely ungrammatical rating of 1 (2) and absolutely grammatical rating of 9 (1)
- A lot many subjects rated the sentence to be 5 than 1 or 9
- This suggests that the frequency of ratings is crowded around the average rating of 4.5

Changing the granularity of the rating scale

- The rating scale we used forces the subject to rate in integers
- Imagine a situation were subjects are given the freedom to use decimals to rate
- If so, then: no two ratings are ever going to be the same; each subject will have a rating that is different from the other, and will have a frequency of 1

```
x=rnorm(36,4.5,2)
hist(x, breaks=300000,xlim=c(0,10))
```



- If we quantize this difference and put individual ratings in intervals, say between 0 and 1, 1 and 2, and 2 and 3, again we will get a distribution similar to the first one

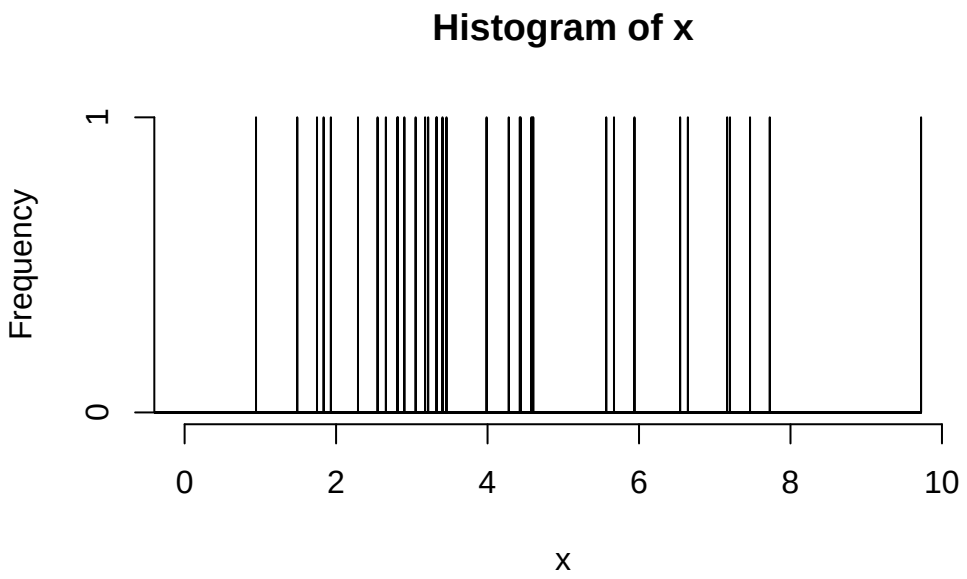
Frequency distribution in R

- How did we generate these plots and distributions
- First we defined a vector using the function, rnorm

```
x = rnorm(36, 4.5, 2)
#notice that this is different from round(rnorm(36,4.5,2)) where we had asked for rounded/int
```

- We defined a vector, x, with 36 values, a mean of 4.5 and standard deviation of 2.
- So decimal ratings would be ok
- Then we made two histograms
  - First with:

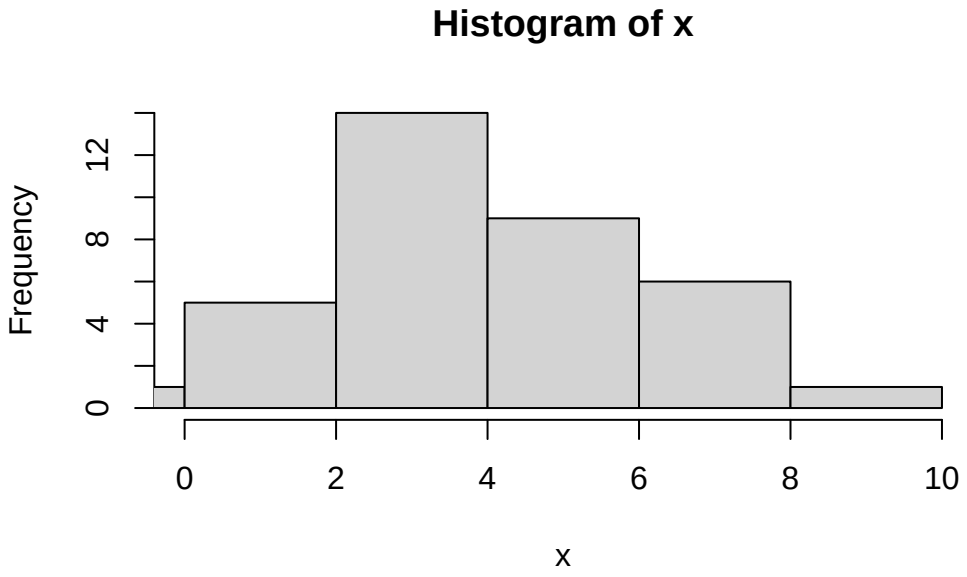
```r
hist(x,breaks=30000, xlim = c(0,10))
```




```r
#here we want to plot a histogram where the width of the cells/bins is very small
```

- Second with:

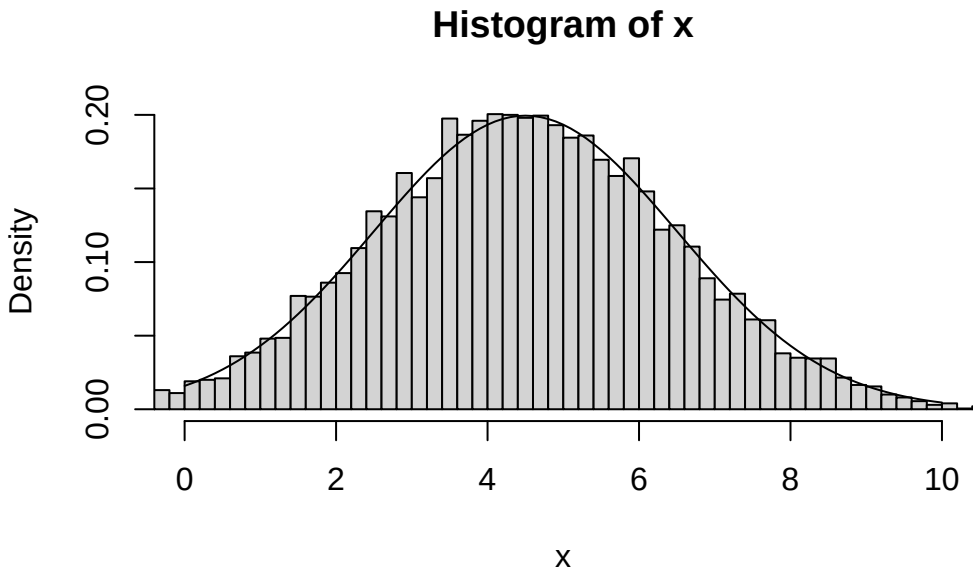
```r
hist(x, xlim = c(0,10))#here we want to plot a histogram where the width of the cells/bins is
```



Theoretical fequency distributions

- Suppose we could draw from an infinite data set
- The larger our data set - more detailed a representation of the frequency distribution
- If we keep collecting sentence grammaticality data for the same sentence, so that instead of ratings from 36 people we have ratings from 10,000 people
- With a histogram that has 1000 bars in it, we see that ratings near 4.5 are more common than those at the edges of the rating scale
- Adding observations up to infinity and reducing the size of the bars in the histogram of the frequency distribution
- Intervals between bars is vanishingly small - i.e. we end up with a continuous curve, almost
- Plotting the normal distribution curve on the frequency distribution

```
x = rnorm(10000, 4.5, 2)
hist(x,breaks=100,freq=FALSE,xlim = c(0,10))
plot(function(x)dnorm(x, mean=4.5, sd=2), 0,10, add=TRUE)
```



Adding the normal curve

- Why the excellent fit between the "observed" and the theoretical distributions?
- The data is generated by random selection
  - rnorm() - observations from the theoretical normal distribution dnorm()
- The "normal distribution" is an very useful theoretical function because…

1. Let's assume that there is an underlying property that we are trying to measure like - grammaticality, or
   - typical duration, or
   - amount of processing time
2. Assume that there is some source of random error that makes it difficult for us to get to this underlying property

- If so, then we can think that - the "true" value of the underlying property we want to measure –
  - Must be at the center of the frequency distribution that we observe in our measurements
  - And, the distribution (we observe) is caused by error - with (the probability of) bigger errors being less likely than smaller errors

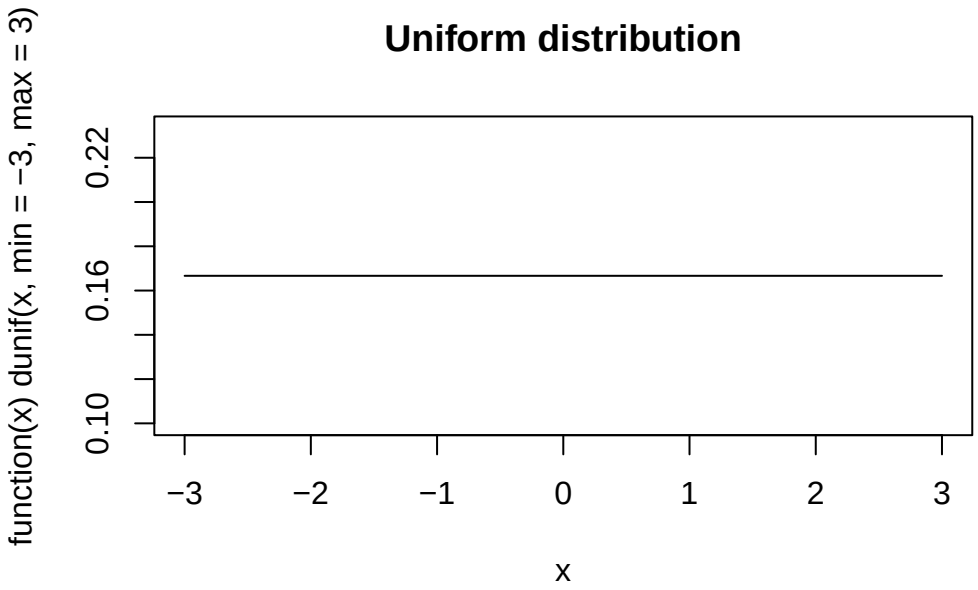
The Normal Distribution

- The normal distribution is described by the normal curve, or the bell-shaped curve
- It is an exponential function of the mean value (μ "mew") and the variance (σ "sigma")
- The sum of the area under the curve, fx is 1
- Derived from just two numbers, the mean value and a measure of how variable the data are
- The area under the curving equalling to 1, is also useful to go from frequency distributions to probability densities
- This is related to hypothesis testing
  - $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$
- e is Euler's constant

Type of distributions

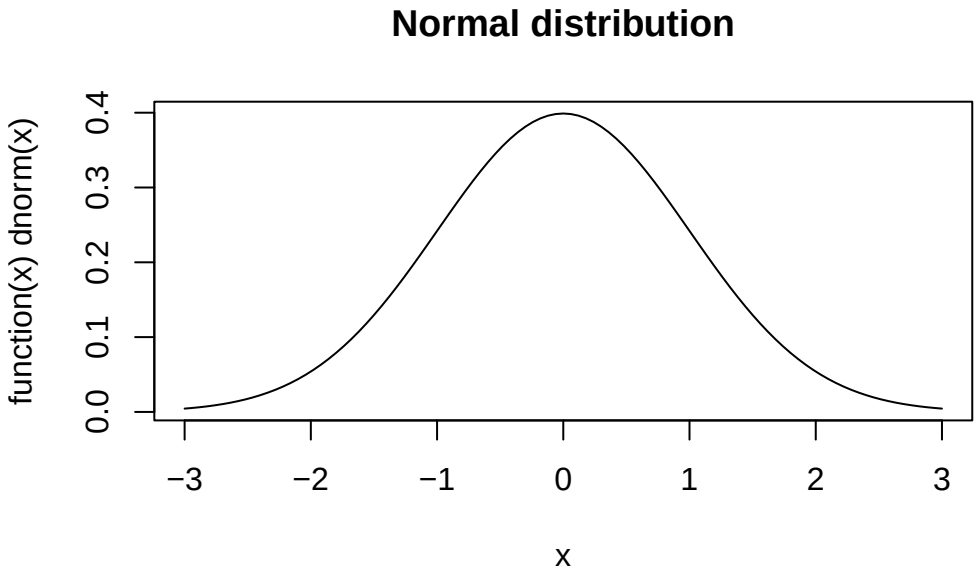
- Uniform distribution: Every outcome is equally likely
  - Six sides of a dice - equal likelihood that either side will be rolled

```
uni=plot(function(x)dunif(x,min=-3,max=3), -3,3, main="Uniform distribution")
```



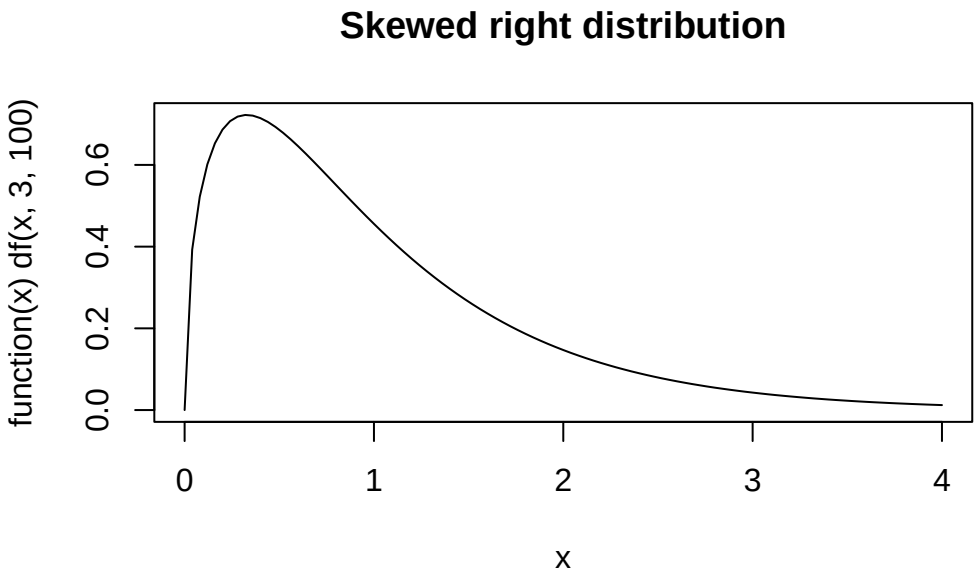
- Normal, bell-shaped distribution, measurements congregate around a typical value and values become less and less likely as they deviate from the central value

```
norm=plot(function(x)dnorm(x), -3,3, main="Normal distribution")
```



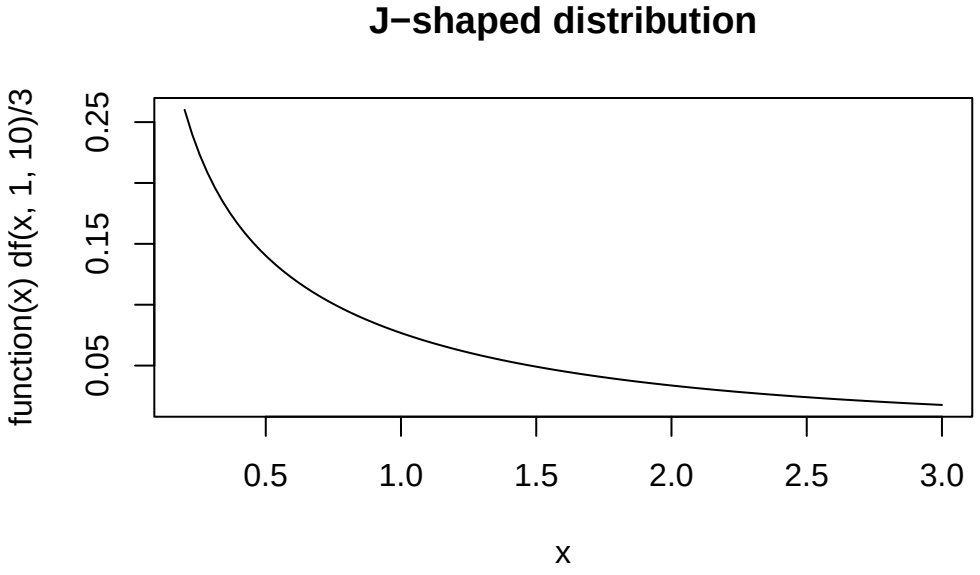
- Skewed right: Skewed frequency distributions
  - percentage data and reaction time data
  - Mean is no longer 'central' to the distribution, or extreme values (from one end of the scale and less from the other) dominate the distribution

```
skewed=plot(function(x)df(x, 3, 100),0,4, main="Skewed right distribution")
```



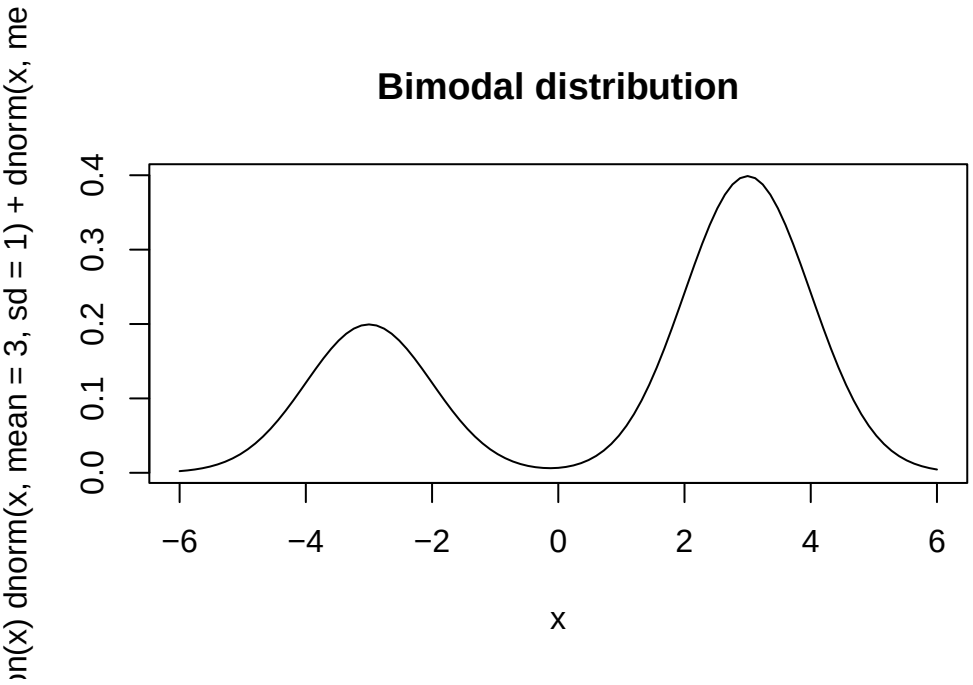
- The J-shaped distribution is a special kind of skewed distribution
  - Most observations come from the end of the measurement scale
  - Most speech errors counts per utterance will have a speech error count of 0

```
j=plot(function(x)df(x, 1, 10)/3,0.2,3, main="J-shaped distribution")
```



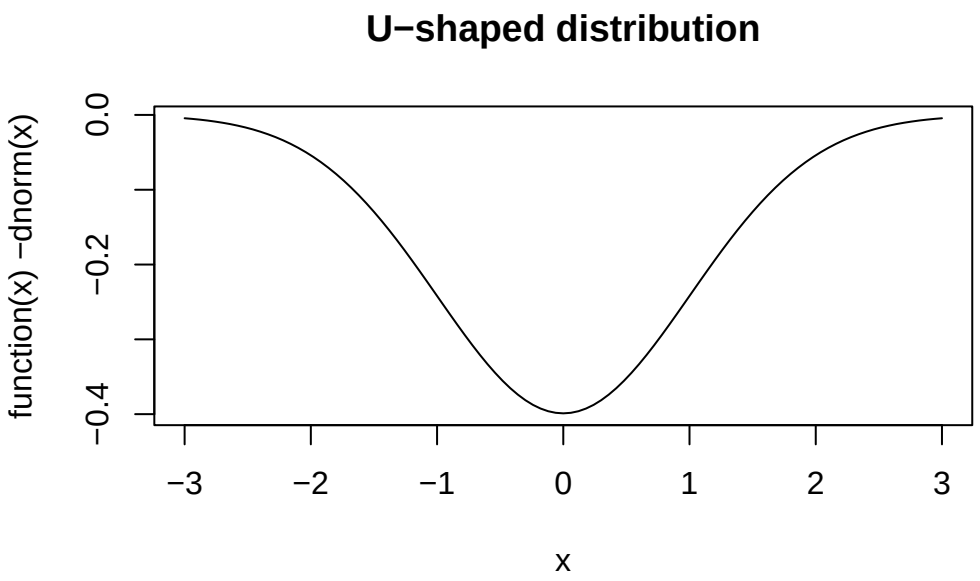
- Bimodal distribution is a frequency distribution where clearly two modalities are involved. For instance
  - f_0 (or pitch) for men and women

```
bimodal=plot(function(x)dnorm(x, mean=3, sd=1)+dnorm(x, mean=-3, sd=1)/2,-6,6,
     main="Bimodal distribution")
```



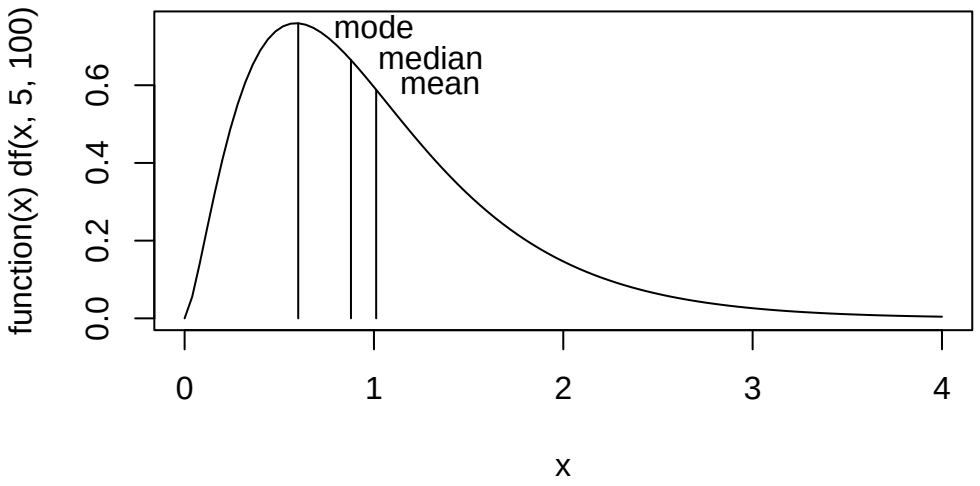
- U shaped distributions result out of polarization where subjects may take drastically one view or the other

```
u=plot(function(x)-dnorm(x),-3,3,
     main="U-shaped distribution")
```



Measures of central tendency

```
plot(function(x)df(x,5,100),0,4, main="Measures of central tendency")
lines(x=c(0.6,0.6),y=c(0,df(0.6,5,100)))
skew.data <- rf(10000,5,100)
lines(
  x=c(mean(skew.data), mean(skew.data)),
  y=c(0,df(mean(skew.data),5,100)))
lines(
  x=c(median(skew.data),median(skew.data)),
  y=c(0,df(median(skew.data),5,100)))
text(1,0.75,labels="mode")
text(1.3, 0.67,labels="median")
text(1.35,0.6,labels="mean")
```



- Normal distribution - the central 'values' (from our samples) have the highest probability of being part of the population
- What are these?
- The most frequently occurring value - *mode* - the tip of the frequency distribution. In the skewed distribution, the mode is 0.6
- The central value, that is in an ordered dataset of the values, the one in the middle is the *median*; aka, the center of gravity
- Arithmetic *mean*, or the sum of values divided by the total number of values, n
- *Least squares estimate of central tendency*

    1. take the difference between the mean and each value in our data set
    2. square these differences and
    3. add them up

- We will get a value that will be smaller than what we would get if we took the median or any other estimate of the "mid-point" of the data set

Weighted means

- Means represent the least squared estimate of the central tendency; say of ratings
- What if we also asked each subject to rate their ratings of grammaticality with a weight, wi
- This way those ratings with a higher weight will give a better estimate of the central tendency; confidence values
- The weights represent the confidence each rater has on her particular rating
- Sample mean = $\bar{x} = \frac{\sum_{i=0}^{n} x_i}{n}$
- Weighted mean = $\bar{x} = \frac{\sum_{i=0}^{n} w_i x_i}{\sum_{i=0}^{n} w_i}$

- Population variance = $\sigma^2 = \sum \frac{(x_i - \mu)^2}{N}$
- Sample variance = $s^2 = \sum \frac{(x_i - \bar{x})^2}{n-1}$

Measures of dispersion

- The mean absolute deviation measures the absolute difference between the mean and each observation
- Absolute deviation could be one measure of difference, where absolute values of the difference for each x_i and sample mean, $\bar{x}$ could be added
- We don't because the mean is the least squares estimator of central tendency
  - so a measure of deviation that uses squared deviations is more comparable to the mean
  - Sum of the squared deviations, $d^2 = \sum_{i=0}^{n} (x_i - \bar{x})^2$
- Variance
  - We square the deviations before averaging them
  - We have definitions for variance of a population and for a sample drawn from a larger population
  - Notice that sample variance, s^2 is calculated by dividing the sum of the squared deviations by n-1 and not n

Why n-1

- Generalize about the process but we only have access to the samples
- Relationship between scores, std. deviation and error
- Accurately talk about the population
  - when we only have access to samples we divide by n-1
  - Taking (n-1) as the denominator in the definition of s^2, sample variance, because $\bar{x}$ is not μ
  - Sample mean $\bar{x}$ is only an estimate of μ, derived from the x_i, so in trying to measure variance we have to keep in mind that our estimate of the central tendency $\bar{x}$ is probably wrong to a certain extent
- The mean of the underlying process (population) we don't know
- The mean of the n points we do, this however contains an error due to statistical noise
- Effect of the error is reduction in the calculated value of s^2
- To make up for this, n is replaced by n-1
- *If n is large, the difference doesn't matter*
- *If n is small, this replacement provides a more accurate estimate of the standard deviation of the underlying process*

Standard deviation

- Variance is the average squared deviation - the differences are squared
- To get to the original unit of deviation we take the square root of the variance; sample and population
- Aka, the RMS (root mean square) sample standard deviation

    1. first square the difference
    2. then take the mean and then
    3. square root of that

- Sample standard deviation
    - $s = \sqrt{\sum \frac{(x_i - \bar{x})^2}{n-1}}$
- Area under the normal distribution is equal to 1
- Measures of the central tendency in terms of $\bar{x}$ (sample mean) and also the sample standard deviation, s
- Normal distribution can be defined for any mean value μ, and any standard deviation σ
- This distribution is also used to calculate probabilities, where the total area under the curve is equal to 1
- That means that the area under any portion of the curve is equal to some proportion of 1
- This happens, when the mean of the bell-shaped distribution is 0 and the standard deviation is 1
    - $f_x = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$

Distributions

- Throwing a six sided dice 20 times
- Let's note down all the 20 outcomes
- Drawing from a uniform distribution
- Sample distribution
- For every outcome count the number of times it appears

Z-score and normalization

- Two things to remember:

    1. Since the area under the normal distribution curve is 1, we can state the probability (area under the curve) of finding a value larger than any value of x, smaller than any value of x, or between any two values of x; relating individual scores to the normal distribution

2. Since, we can approximate our data with a normal distribution - we can state these probabilities for our data given the mean and standard deviation; under the assumption that our data are normally distributed

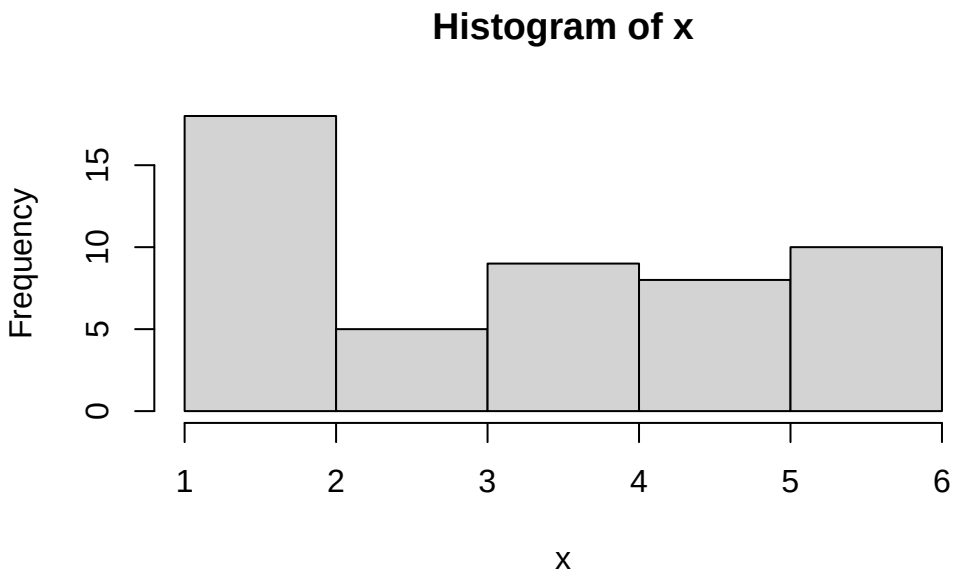
- Relate the frequency distribution of our data to the normal distribution because we know the mean and standard deviation of both
- Key is to be able to express any value in a data set in terms of its distance in standard deviations from the mean
- z-score normalization, $z_i = \frac{x_i - \bar{x}}{s}$

```
#------- shade.tails --------------------
# draw probability density functions of t with critical regions shaded.
#   by default the function draws the 95% confidence interval on the normal
#   distribution.
#
# Input parameters
#   crit - the critical value of t (always a positive number)
#   df - degrees of freedom of the t distribution
#   tail - "upper", "lower" or "both"
#   xlim - the x axis range is -xlim to +xlim

shade.tails <- function(crit=1.96, df = 10000, tail = "both",xlim=3.5)
{

curve(dt(x,df),-xlim,xlim,ylab="Density",xlab="t")

ylow = dt(xlim,df)
pcrit = pt(crit,df)
caption = paste(signif(1-pcrit,3))

if (tail == "both" | tail == "lower") {
    xx <- seq(-xlim,-crit,0.05)
    yy <- dt(xx,df)
    polygon(c(xx,-crit,-xlim),c(yy,ylow,ylow),density=20,angle = -45)
    text(-crit-0.7,dt(crit,df)+0.02,caption)
}
if (tail =="both" | tail == "upper") {
    xx2 <- seq(crit,xlim,0.05)
    yy2 <- dt(xx2,df)
    polygon(c(xx2,xlim,crit),c(yy2,ylow,ylow),density=20,angle = 45)
    text(crit+0.7,dt(crit,df)+0.02,caption)
}
}
```

Sampling from a uniform distribution

- Storing outputs of functions in vectors
- Here, x, is a vector that stores the outout of the function *sample*
-

```
x <- sample(1:6,50,TRUE)
hist(x)
```



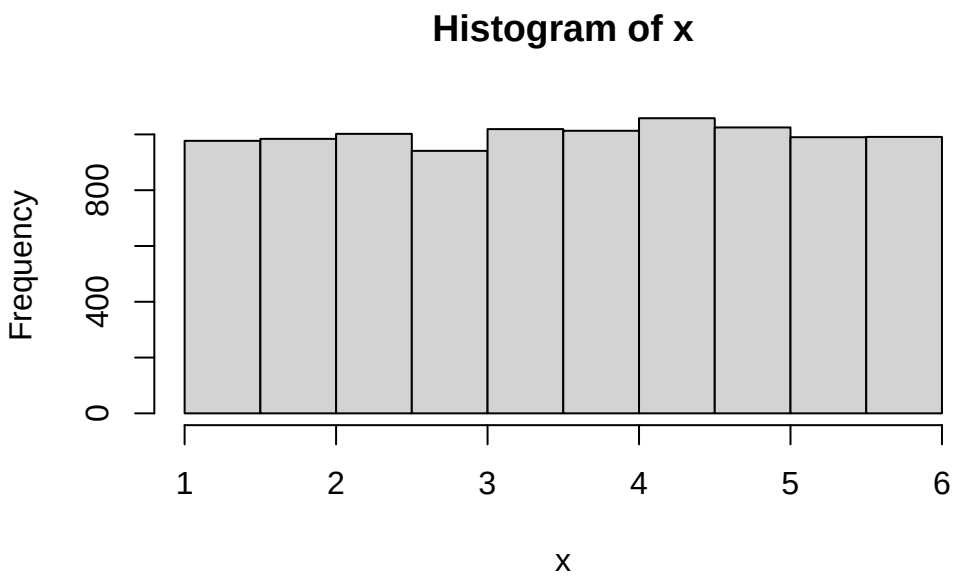

```
x
```

```
 [1] 5 2 3 4 6 2 4 4 1 6 5 6 5 6 4 2 1 1 6 1 6 1 6 2 5 2 3 2 1 3 1 5 4 4 6 3 5 1
[39] 5 4 6 5 1 1 2 4 6 4 3 1
```

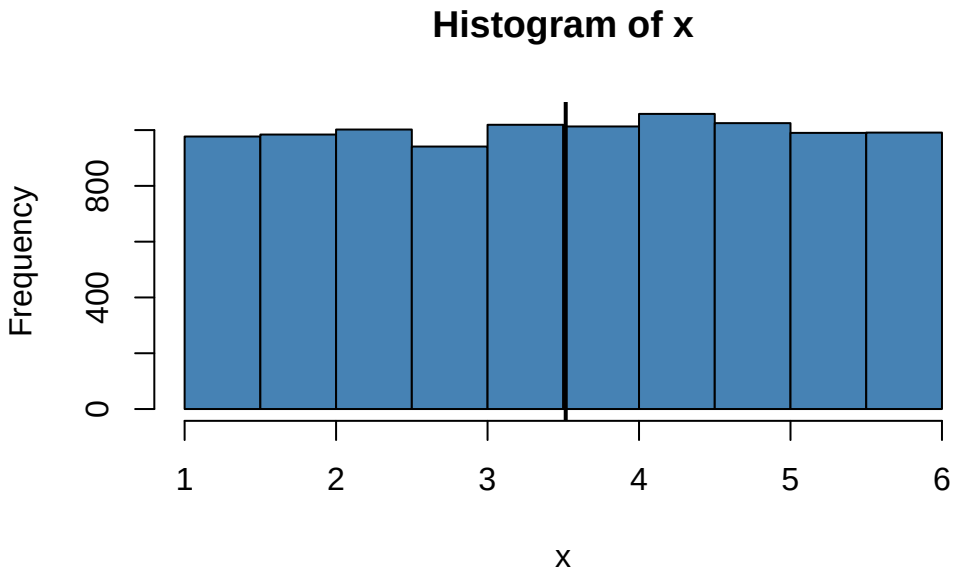
- Every time we run this code chunk the out of the sampling will change

---

```
x <- runif(10000, min = 1, max = 6)
hist(x)
```

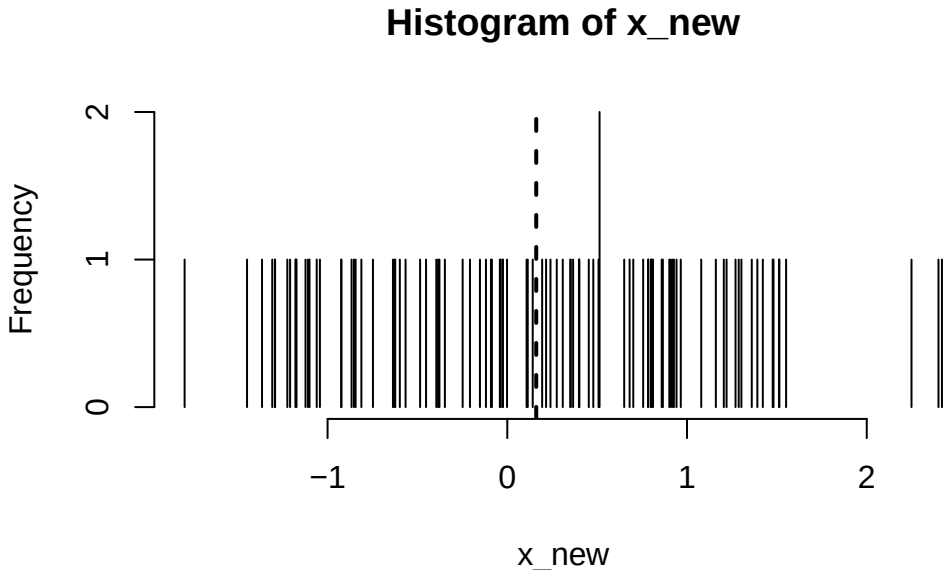


```
hist(x, col = 'steelblue')
abline(v = mean(x), lty = 1, lwd = 2)
```



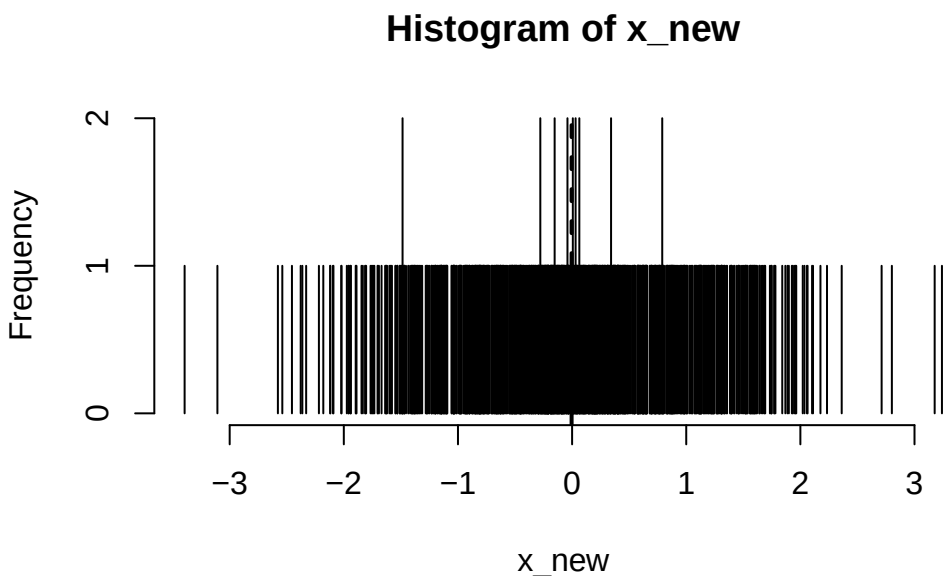
Uniform Distribution

```
x_new <- rnorm(100)
hist(x_new, breaks=100000,col = 'steelblue')
abline(v = mean(x_new), lty = 2, lwd = 2)
```



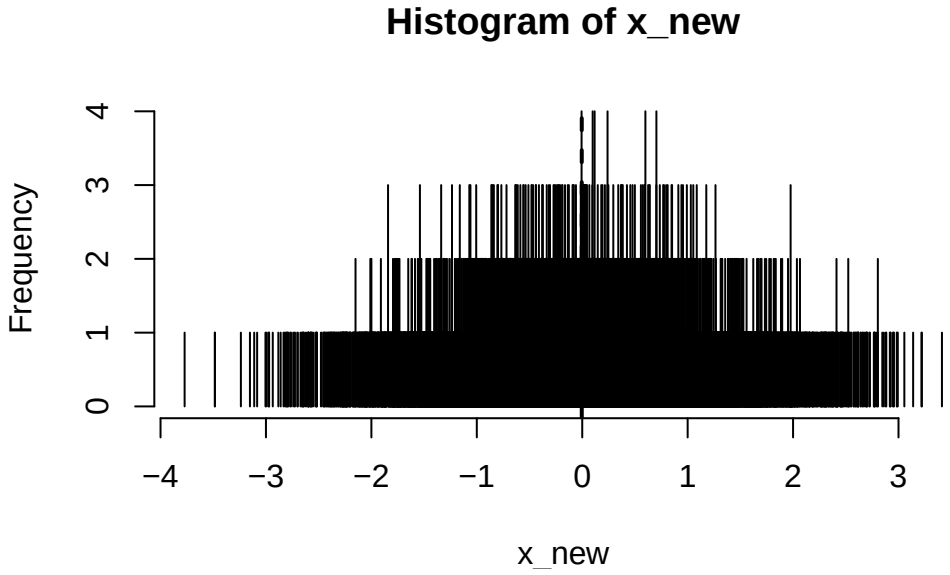
Still Uniform Distribution

```
x_new <- rnorm(1000)
hist(x_new, breaks=100000,col = 'steelblue')
abline(v = mean(x_new), lty = 2, lwd = 2)
```



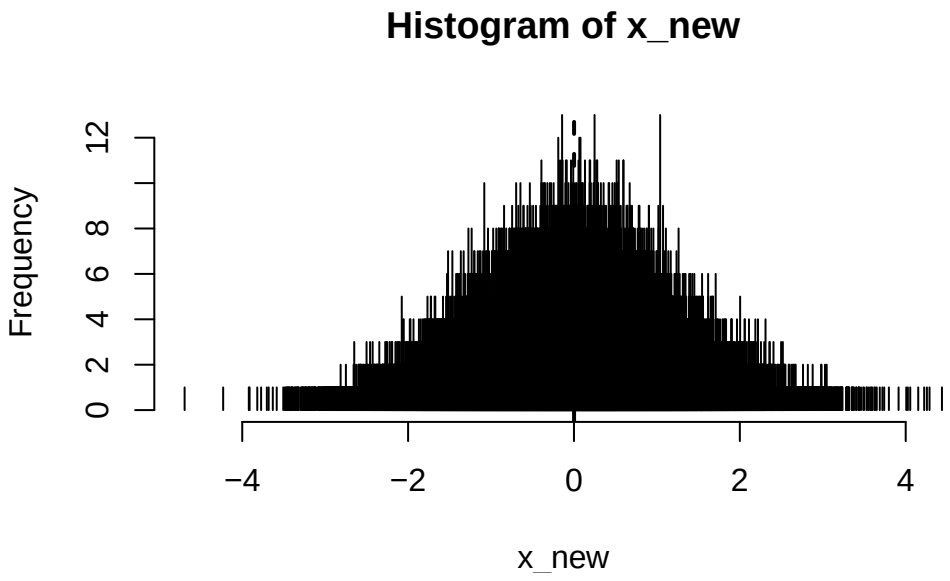
Normal or Gaussian Distribution

```
x_new <- rnorm(10000)
hist(x_new, breaks=100000,col = 'steelblue')
abline(v = mean(x_new), lty = 2, lwd = 2)
```



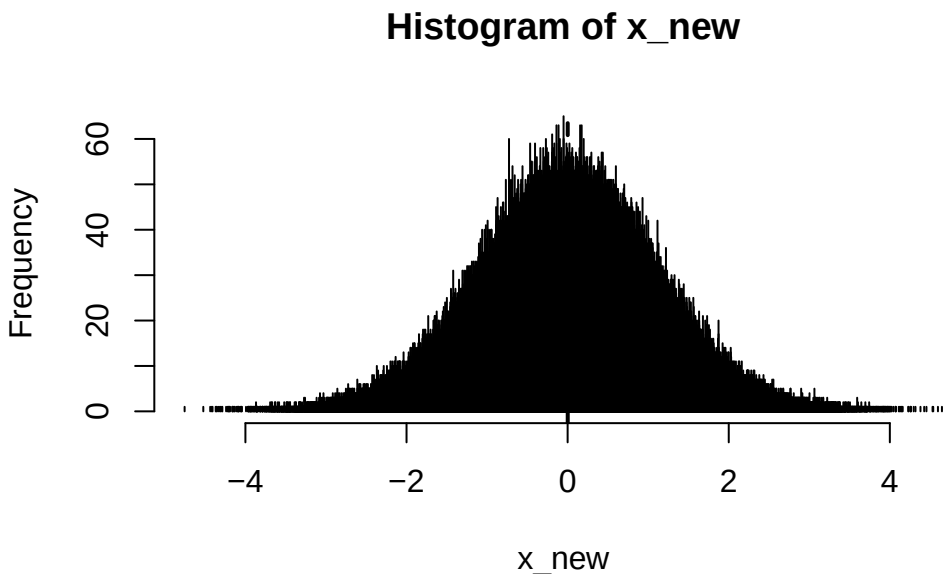
Increasing sampling in a normal or Gaussian Distribution

```
x_new <- rnorm(100000)
hist(x_new, breaks=100000,col = 'steelblue')
abline(v = mean(x_new), lty = 2, lwd = 2)
```



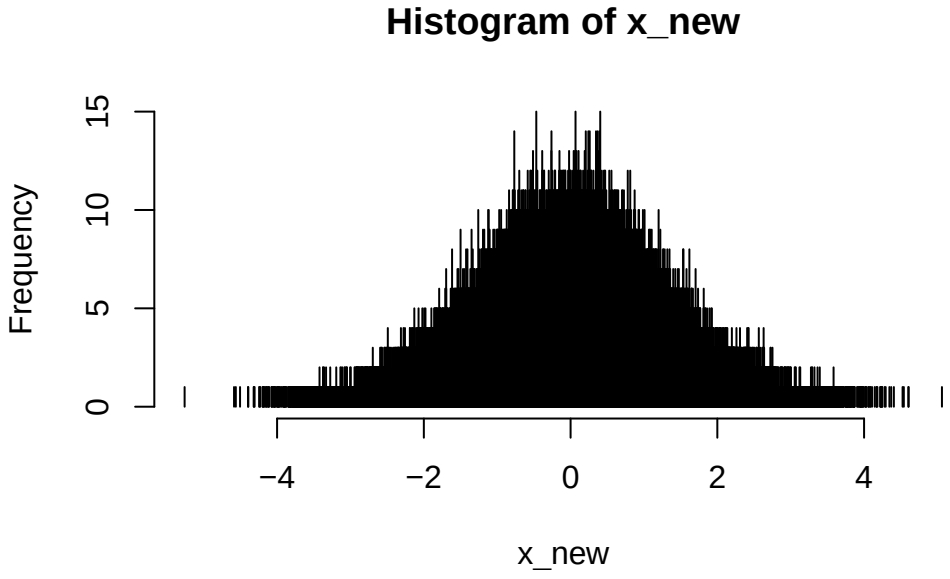
Further increasing sampling in a normal or Gaussian Distribution

```
x_new <- rnorm(1000000)
hist(x_new, breaks=100000,col = 'steelblue')
abline(v = mean(x_new), lty = 2, lwd = 2)
```



Increasing breaks now in a normal or Gaussian Distribution

```
x_new <- rnorm(1000000)
hist(x_new, breaks=1000000,col = 'steelblue')
```



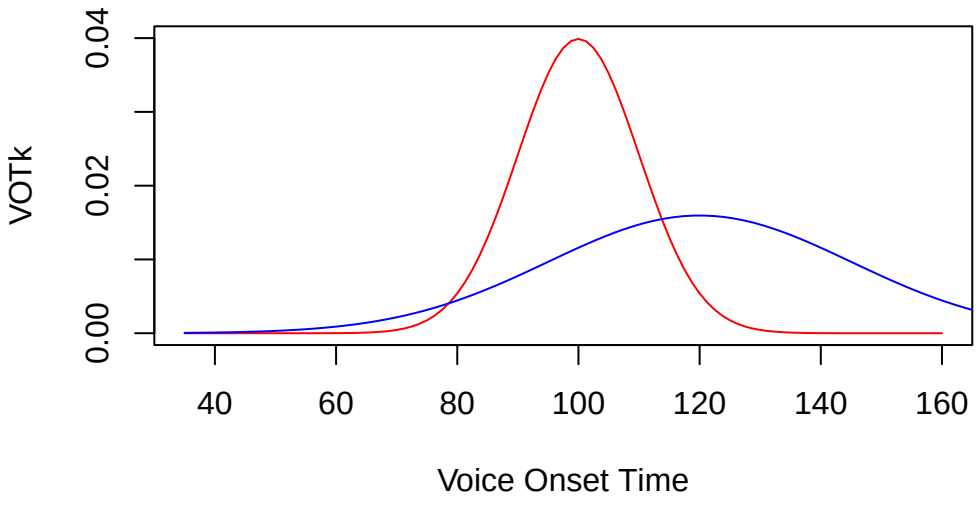
Getting some invariant parts of the sample: mean and standard deviation

- Sum of x $\sum x_i$
  - $\sum x_i^2$
  - $\sum x_i y_i$
- Mean of x $\frac{1}{n}\sum_{i=i}^{n} x_i$
- $StandardDeviation$

  $s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}$

  σ is the population parameter
- Variance = σ^2

```
VOTk <- function(x) dnorm(x, mean = 100, sd = 10)
VOTp <- function(x) dnorm(x, mean = 120, sd = 25)
myYLim <- c(0, 0.04)
myXlim <- c(0,140)
plot(VOTk, from = 35, to = 160, ylim = myYLim, col="red",
     xlab="Voice Onset Time", myXlim)
plot(VOTp, from = 35, to = 200, add = TRUE, col="blue", ylim = myYLim, xlim=myXlim)
```



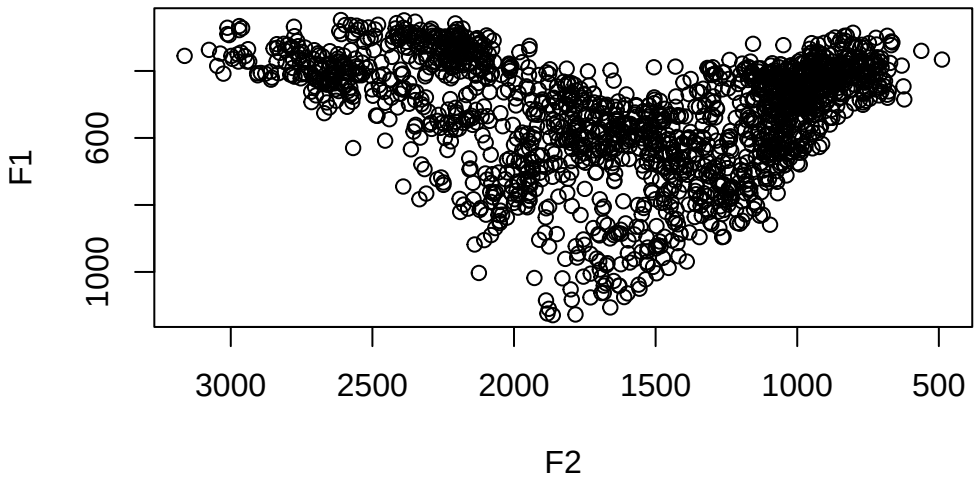
Plotting Vowels using PhonR

```
library(phonR)
#(indo)
#head(indo)
summary(indo)
```

```
     subj           gender  vowel         f1              f2
 Length:1725        f:867   a:349   Min.   : 248.0   Min.   : 489
 Class :character   m:858   e:335   1st Qu.: 402.0   1st Qu.:1055
 Mode  :character           i:348   Median : 493.0   Median :1509
                            o:346   Mean   : 531.1   Mean   :1594
                            u:347   3rd Qu.: 632.0   3rd Qu.:2097
                                    Max.   :1129.0   Max.   :3163
```

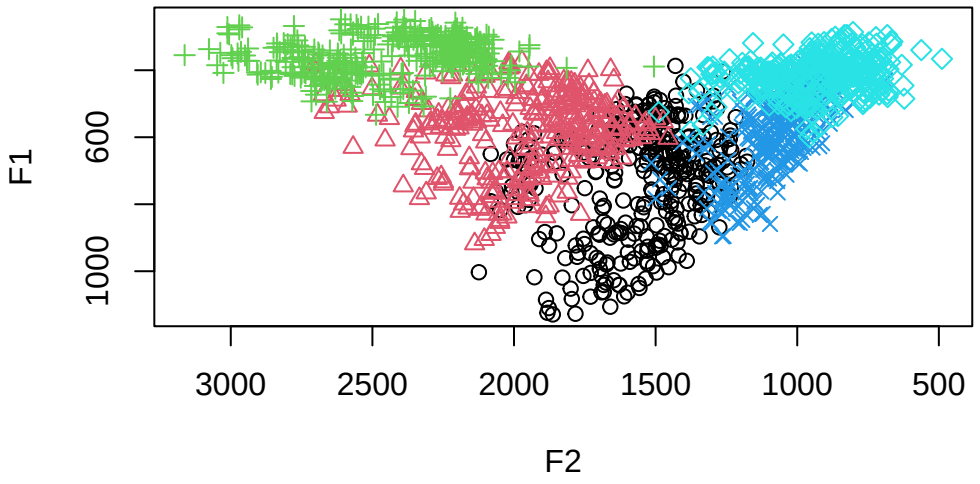
Plotting Vowels using PhonR

```
with(indo, plotVowels(f1, f2))
```



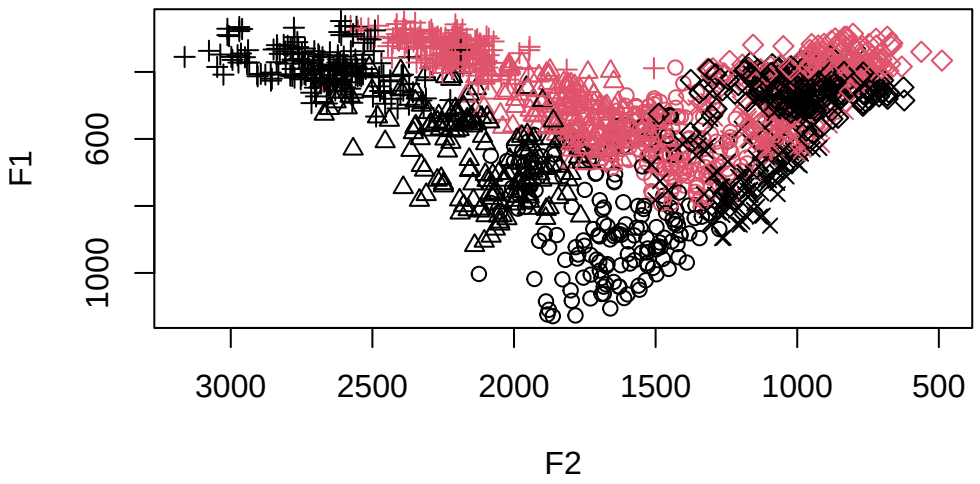
Plotting Vowels using PhonR

```
with(indo, plotVowels(f1, f2, var.sty.by = vowel, var.col.by = vowel))
```



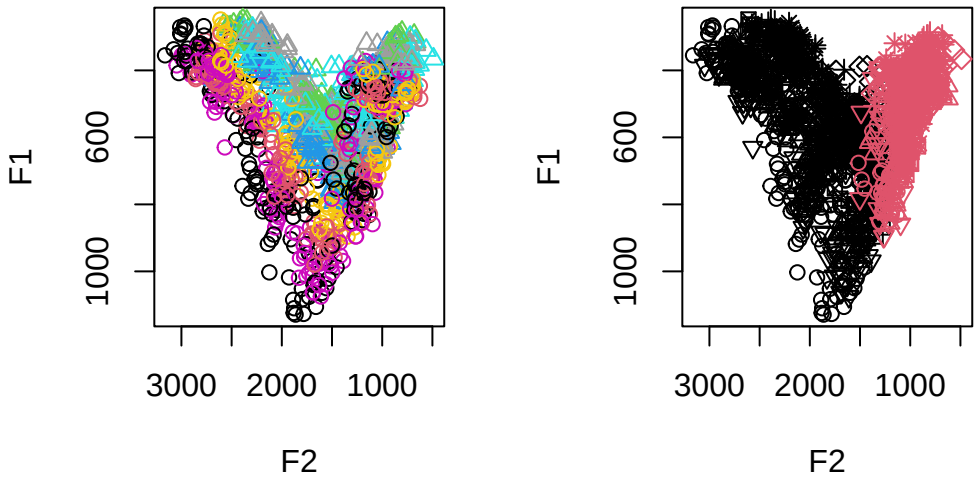
Plotting Vowels using PhonR

```
with(indo, plotVowels(f1, f2, var.sty.by = vowel, var.col.by = gender))
```



Plotting Vowels using PhonR

```
par(mfrow = c(1, 2))
rounded <- ifelse(indo$vowel %in% c("o", "u"), "round", "unround")
with(indo, plotVowels(f1, f2, var.sty.by = gender, var.col.by = subj))
with(indo, plotVowels(f1, f2, var.sty.by = subj, var.col.by = rounded))
```



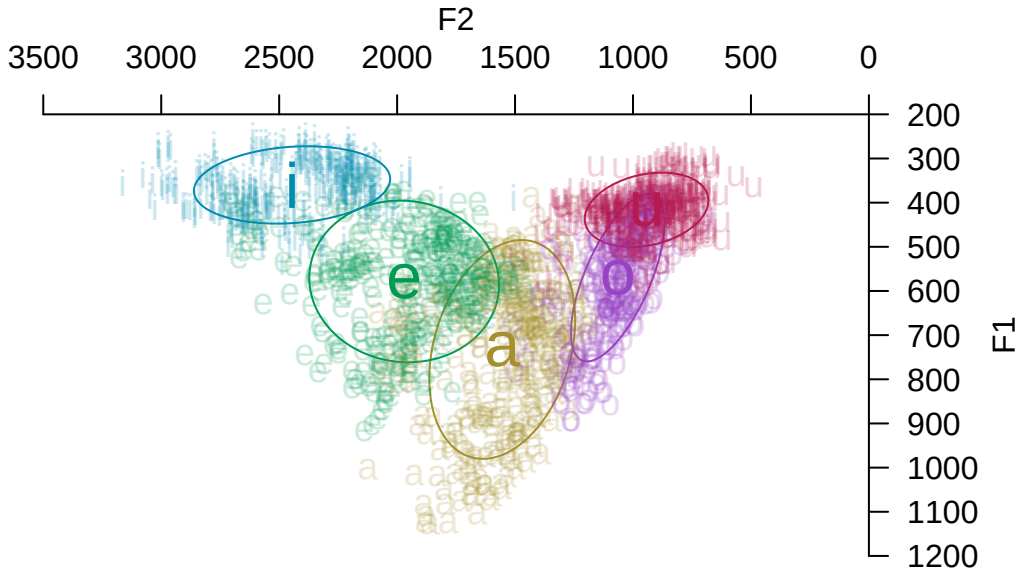
Calculating vowel space areas

```
poly.area <- with(indo, vowelMeansPolygonArea(f1, f2, vowel, poly.order = c("i",
    "e", "a", "o", "u"), group = subj))
hull.area <- with(indo, convexHullArea(f1, f2, group = subj))
rbind(poly.area, hull.area)
```

```
               F02      F04     F08      F09      M01      M02      M03
poly.area  485051.4 337364.0  434816 302064.9 197746.1 229501.7 215713.3
hull.area 1254575.0 866109.5 1020835 751327.0 517212.5 666246.0 477518.5
               M04
poly.area 177131.1
hull.area 568364.0
```

Ellipses, polygons, and hulls

```
#par(mfrow = c(2, 2))
with(indo, plotVowels(f1, f2, vowel, plot.tokens = TRUE, pch.tokens = vowel, cex.tokens = 1.2
    alpha.tokens = 0.2, plot.means = TRUE, pch.means = vowel, cex.means = 2, var.col.by = vow
    ellipse.line = TRUE, pretty = TRUE))
```



Normalizing data

- Speaker vocal tracts are variable - different lengths and cross-sections
- Implies variable resonances
- $F_n = \frac{(2n-1)c}{4L}$, for a tube that is open at one end and closed in the other

Minimizing variation

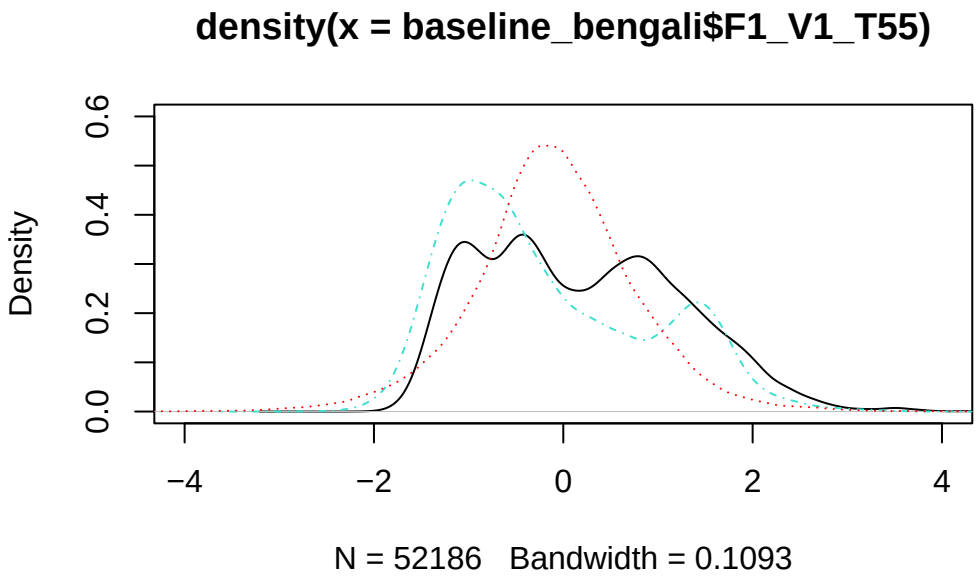
- In order to minimize the variation brought about by the variable vocal tract parameters, often we do a type of normalization that we call z-score normalization

Z-Score Normalization

- This serves two purposes
    1. Allows us to reduce individual differences (between subjects)
    2. Makes data comparable
- Z-Score normalization
- $z = \frac{x_i - \overline{x}}{\sigma}$
- Where $\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2}$

Error in estimating population parameters

- Two major sources of errors
    1. The underlying distribution
    2. The number of samples
- $SE = \frac{\sigma}{\sqrt{n}}$



Day 2

Normal distribution and standardization

- With standardized values we can make probability statements

- In this figure, the area under the normal curve between -1.96 and 1.96 is 0.95.
- *95% of the values we draw from a normal distribution will be between 1.96 standard deviations below the mean and 1.96 standard deviations above the mean*

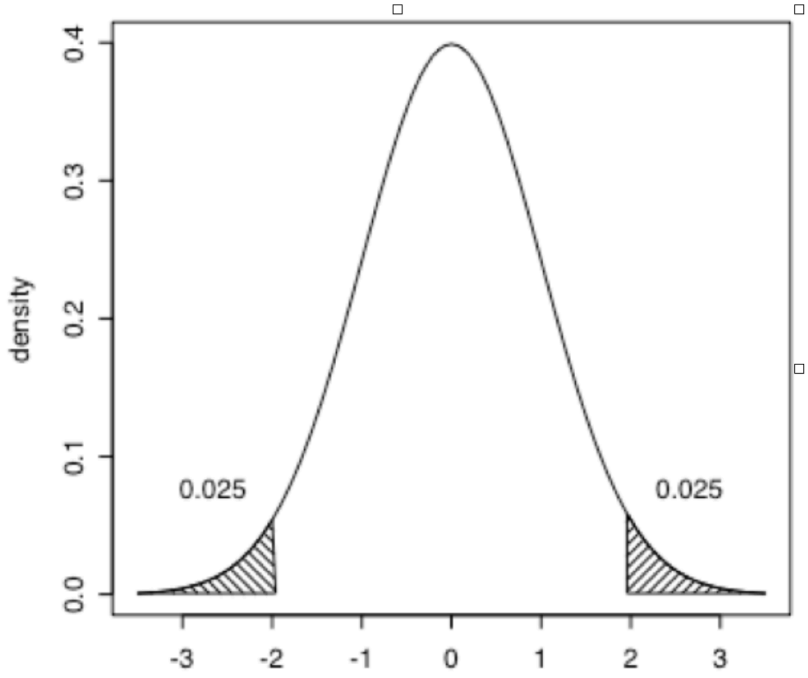


Figure 1: Z-score

How normal

- The normal distribution is a helpful way to describe data; Why? Because from this distribution, given a value, we can state the probability of its occurrence
- The normal distribution also provides a basis for making inferences about the accuracy of our statistical estimates.
- In data reduction, we use just the mean and standard deviation to describe the whole frequency distribution.
- It is important to find out whether or not the frequency distribution of our data is shaped like the normal distribution.
- First we will find out if our data are normally distributed, and then we'll look at a couple of transformations that we can use to MAKE data more normal

Cherokee dataset

What is VOT

- Voice Onset Time
- Duration of time it takes for regular voicing to get initiated following a stop into the vowel

Cheroke VOT

- VOT data Longitudinal data collected first in 1971 and then again 2001 Let's define two vectors that represent the two sets of data

```
vot01 = c(84, 82, 72,193, 129, 77, 72, 81, 45, 74, 102,
          77, 187, 79, 86, 59, 74, 63, 75, 70, 106, 54, 49, 56, 58, 97)
# And then
vot71 =c(67, 127, 79, 150, 53, 65, 75, 109, 109,
         126, 129, 119, 104, 153, 124, 107, 181, 166)

#The simplest way to means and standard deviations in R are to simply ask for them
mean(vot01)#mean of the data set vot01, 84.65385
```

```
[1] 84.65385
```

```
mean(vot71)#mean of the data set vot71, 113.5
```

```
[1] 113.5
```

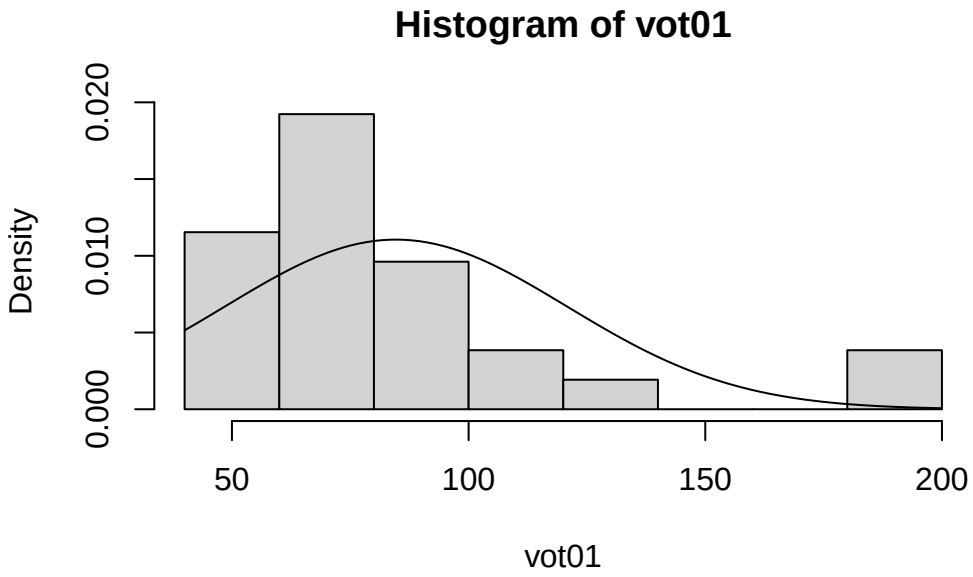
```
sd(vot01)
```

```
[1] 36.08761
```

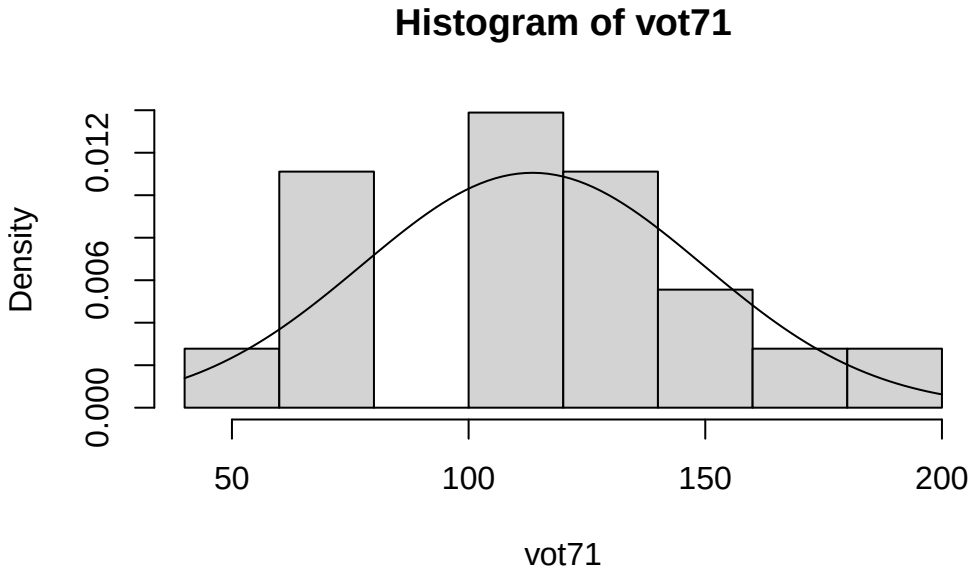
```
sd(vot71)# We will get 36.08761 and 35.92844, respectively
```

```
[1] 35.92844
```

```
hist(vot01,freq=FALSE)#make a histogram by using probability densities and not the actual cou
#Getting help from R: type
?hist
plot(function(x)dnorm(x, mean=84.654, sd=36.088), 40, 200, add=TRUE)#For 2001
```




```
hist(vot71,freq=FALSE)
plot(function(x)dnorm(x, mean=113.5, sd=36.087), 40, 200, add=TRUE)#For 1971
```



- We have two distributions; plot the frequency distribution as a histogram and then compare the observed distribution with the best-fitting normal curve
- Both the 2001 and the 1971 data sets are fairly similar to the normal curve
- The 2001 set has a pretty normal looking shape, but there are a couple of measurements at nearly 200 ms. that don't 'fit'.
- The 1971 set also looks like a normally distributed data set, though there are no observations between 80 and 100 ms in this data set. If these data came from a normal curve we would expect several observations in this range.
- Making things normal
- Let's see what happens if we remove the outliers from the 2001 dataset; does the fit get better? Let's assume that these outliers are caused due to speech errors for the moment
- Let's assume that the two VOT measurements in vot01 that are greater than 180 ms are outliers
- Calculate the mean and standard deviation for only those numbers in the vector that are less than 180

```
mean(vot01[vot01<180])
```

```
[1] 75.875
```

```
sd(vot01[vot01<180])
```

```
[1] 19.218
```

- **You must have good reasons for trimming data**

q-q plots

- Frequency distribution graphs give an indication of whether our data is distributed on a normal curve
- It would be nice to be able to measure just how "normally distributed" these data are
- Quantile means the fraction (or percent) of points/scores below the given value
- So the 0.3 (or 30%) quantile is the point at which - 30% percent of the data fall below - and 70% fall above that value.
- Quantiles are values that divide the distribution so that a given proportion of observations falls below the quantile. The median is a good example of a quantile.
- q-q plots: Measure the degree of fit between the data and the normal curve
- quantile/quantile plot are a correlation between

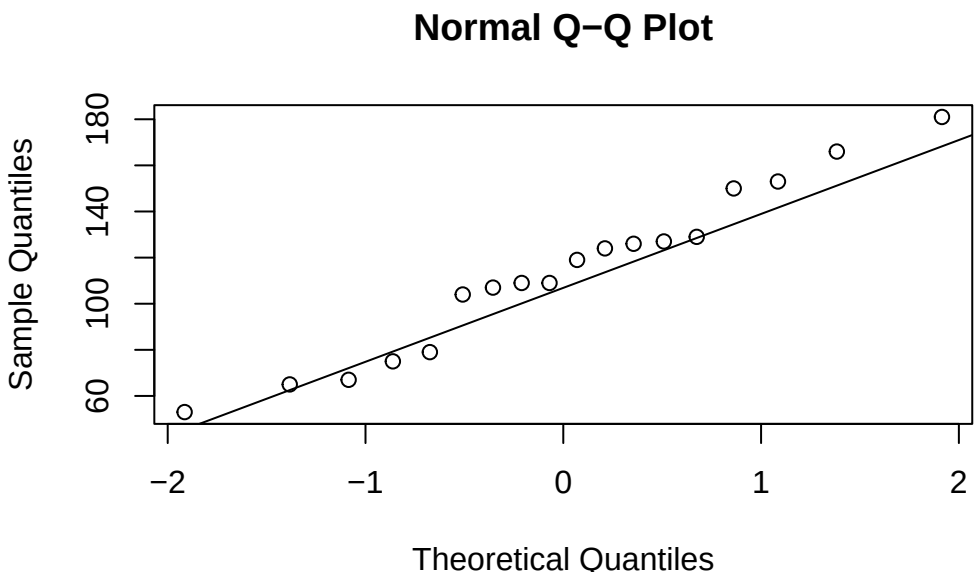
    1. the actual quantile scores and
    2. the quantile scores that are predicted by the normal curve

- The quantile-quantile (q-q) plot is a graphical technique for determining if two data sets come from populations with a shared distribution.
- Plot of the quantiles of the first data set against the quantiles of the second data set.
- The median is the central value of the distribution, such that half the points are less than or equal to it and half are greater than or equal to it.
- Quantiles are easily understood if you think about quartiles (3+ 1 Median);
- We use two tertiles to split data into three groups, four quintiles to split them into five groups, and so on.
- A 45-degree reference line is also plotted.
- If the two sets come from a population with the same distribution, the points should fall approximately along this reference line.
- Greater the departure from this reference line, the greater the evidence for the conclusion that the two data sets have come from populations with different distributions.
- Advantages of the q-q plot are:

    1. The sample sizes do not need to be equal.
    2. Many distributional aspects can be simultaneously tested

- Shifts in location, shifts in scale, changes in symmetry, and the presence of outliers can all be detected from this plot.
- If the two data sets come from populations whose distributions differ only by a shift in location, the points should lie along a straight line that is displaced either up or down from the 45-degree reference line.
- We expect our 71 data to be fairly normally distributed

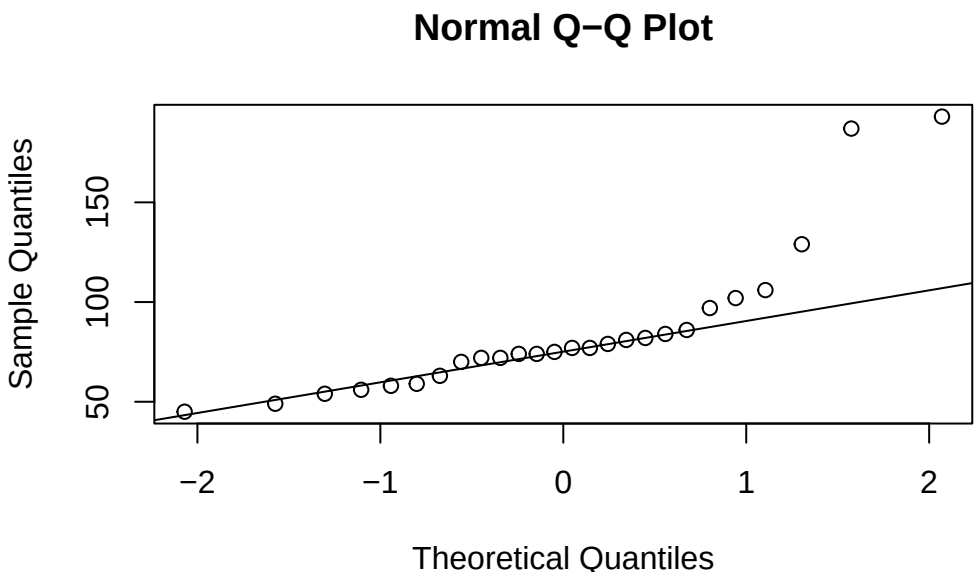
```
vot71.qq = qqnorm(vot71)$x # make the quantile/quantile plot
qqline(vot71) # put the line on the plot
```



```
cor(vot71,vot71.qq) # compute the correlation
```

```
[1] 0.9868212
```

```
vot01.qq = qqnorm(vot01)$x # make the quantile/quantile plot
qqline(vot01) # put the line on the plot
```




```
cor(vot01,vot01.qq) # compute the correlation
```

```
[1] 0.8700187
```

- Departures from the straight line indicate departures from the specified distribution
- The 1971 data shows that there is a good fit between expected and actual quantiles; reflected in a correlation coefficient of 0.987 - almost a perfect 1
- Contrast this with the 2001 data
- Most of the data points in the 2001 data set are just where we would expect them to be in a normal distribution
- However, there are two or three large VOT values that are much larger than expected.
- Because of this the correlation between expected and observed quantiles for this data set (r = 0.87) is lower than what we found for the 1971 data.

Summary so far

- We took two data sets and calculated their means and standard deviations
- We also learned how to manipulate data to see how we can remove outliers
- We compared the sample distributions to a theoretical distribution to see how well our data are correlated with the theoretical distribution
- We answered the question, "How normal are our data?"

Hypothesis testing

- How to test hypotheses regarding means …
    1. We can make probability statements about variables in normal distributions
    2. We can estimate the parameters of empirical distributions as the least squares estimates of $\bar{x}$ and s
    3. Means of samples drawn from a population, fall in a normal distribution
    4. We can estimate the standard error (SE) of the normal distribution of $\bar{x}$ values from a single sample.
- What this means is that we can make probability statements about means, and hence relate them to our hypotheses…let's start with Hypothesis 0, or the null hypothesis

H0: $\mu = 100$

- We want to make probability statements about observations using the normal distribution
- Remember, we converted our observation scores into z scores (the number of standard deviations different from the mean) using the z score formula.
- To test a hypothesis about the population mean (μ) on the basis of our sample mean and the standard error of the mean we use a similar approach
- Big problem is !!! We don't know the population standard deviation.
- Instead, we estimate it with the sample standard deviation, and the uncertainty introduced by using s instead of σ means that we are off a bit and can't use the normal distribution to compare $\bar{x}$ to μ.
- To be a little more conservative, we use a distribution (or family of distributions), called the t-distribution
- Taking into account how certain we can be about our estimate of σ.
- Since, a larger sample size gives us a more stable estimate of the population mean, similarly we get a better estimate of the population standard deviation with larger sample sizes. So the larger the sample size, the closer the t distribution is to normal

One-sample t-test

- We use a slightly different distribution to talk about mean values, but the procedure is similar to using the normal distribution
- To make a probability statement about a z-score we refer to the normal distribution, and to make a probability statement about a t value we refer to the t distribution.
- It may seem odd to talk about comparing the sample mean to the population mean because we can easily calculate the sample mean but the population mean is not a value that we can know
- But, if we think of this as a way to test a hypothesis, then we have something.
- For example, with the Cherokee VOT data, we observed that $\bar{x}$ = 84.7 and s = 36.1 for the stops produced in 2001
  - We can now ask whether the population mean μ is different from 100. Plug the numbers into the formula.
- Here, $s\bar{x}$ is the standard error
  - $SE = s\bar{x} = \frac{\sigma}{\sqrt{N}}$; population
  - $SE = s\bar{x} = \frac{s_x}{\sqrt{n}}$
  - $t = \frac{\bar{x}-\mu}{s\bar{x}} = \frac{84.7-100}{36.1/\sqrt{26}} = \frac{-15.3}{7.08} = -2.168$

Interpreting t values

- t value in this test is -2.168.
- But what does that mean?
- We were testing the hypothesis that the average VOT value of 84.7 ms is not different from 100 ms.
- This can be written as H0: μ = 100.
- Meaning that the null hypothesis (the "no difference" hypothesis H0) is that the population mean is 100.
- Now we know that, observations that are more than 1.96 standard deviations away from the mean in a normal distribution are pretty unlikely - only 5% of the area under the normal curve.
- So this t value of -2.168 (-15.3 is a little more than 2 standard errors than the hypothesized mean) will be a pretty unlikely one to find if the population mean is actually 100 ms.
- The more likely conclusion that we could draw is that the population mean is less than 100.

Type of errors

- We want to test the hypothesis (null) that the true Cherokee VOT in 2001 (μ) is 100ms by taking a sample from a larger population of possible measurements.
- If the sample mean $\bar{x}$ is different enough from 100ms then we reject this hypothesis otherwise we accept it.

- How different is different enough?
- We can quantify the difference between the sample mean and the hypothesized population mean in terms of a probability.
- If the population mean is 100 ms, then only 2 times in 100 could we get a sample mean of 84.7 or less.
- Suppose we decide then that this is a big enough difference
- The probability of a sample of 84.7 mean coming from a population that has a mean of 100 ms is preeeeeetty low - so we reject the hypothesis that $\mu = 100$ (let's call it H0)
- Instead, we accept the alternative hypothesis that $\mu < 100$ (call this H1; this is only one of several possible alternatives)
- H0: $\mu = 100$ (Reject); H1: $\mu < 100$ (Accept)

Type of errors - Type I

- But wait… 2 times out of 100 we will be **wrong** to reject the null hypothesis
- This error probability (0.02) is called the probability of making a type I error.
- A type I error is that we incorrectly reject the null hypothesis
- We claim that the population mean is less than 100, when actually we just got unlucky and happened to draw one of the 2 out of 100 samples for which the sample mean was equal to or less than 84.7.
- No matter what the sample mean is, we can't reject the null hypothesis with certainty because the normal distribution extends from negative infinity to positive infinity
- In practice, going with our best guess means choosing a type I error probability that we are willing to tolerate.
- Most often we are willing to accept a 1 in 20 chance (5 in 100, if you will) that we just got an unlucky sample that led us to make a type I error.
- This means that if the probability of the t value that we calculate to test the hypothesis is less than 0.05, we are willing to reject H0 ($\mu = 100$)
- And conclude that the sample mean comes from a population that has a mean that is less than 100 ($\mu < 100$).
- This criterion probability value ($p<0.05$) is called the "alpha" α level of the test.
- The α level is the acceptable type I error rate for our hypothesis test

Type of errors - Type II

- Where there is a type I error, there is a type II error as well
- A type II error occurs when we incorrectly accept the null hypothesis.
- Suppose we test the hypothesis that the average VOT for Cherokee (or at least this speaker) is 100 ms, but the actual true mean VOT is 95 ms.
- If our sample mean is 95 ms and the standard deviation is again about 35 ms we are surely going to conclude that the null hypothesis (H0: $\mu = 100$) is probably true.

- At least our data is not inconsistent with the hypothesis because 24% of the time (p=0.24) we can get a t value that is equal to or less than -0.706.
- By accepting the null hypothesis we made a type II error. Just as we can choose a criterion α level for the acceptable type I error rate, we can also require that our statistics avoid type II errors.
- The probability of making a type II error is called β, and the value we are usually interested in is 1-β, called the power of our statistical test
  - $t = \frac{\bar{x}-\mu}{s\bar{x}} = \frac{95-100}{36.1/\sqrt{26}} = \frac{-5}{7.08} = -0.706$
- Preregistration of studies in linguistics and setting the power

t-tests in R

```
t.test(vot01,mu=100, alternative="less")
```

```
    One Sample t-test

data:  vot01
t = -2.1683, df = 25, p-value = 0.01993
alternative hypothesis: true mean is less than 100
95 percent confidence interval:
     -Inf 96.74298
sample estimates:
mean of x
 84.65385
```

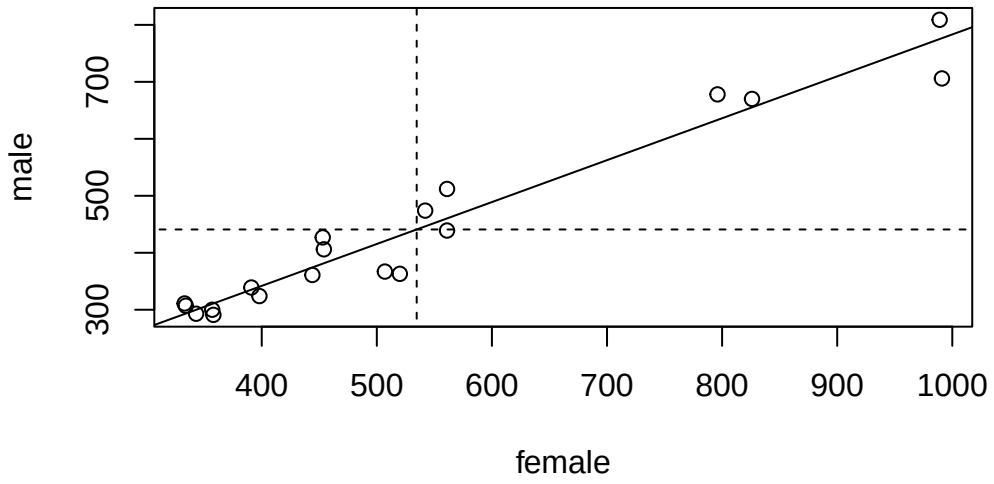
- In this t.test(), we entered the name of the vector that contains our data, the hypothesized population mean for these data, and that we want to know how likely it is to have a lower t value

Correlations

- So far we have been looking at the statistical background assumptions that make it possible to test hypotheses about the population mean.
- The aim is to explain some of the key concepts that underlie studies of relationships among variables.
- One way to explore the relationship between two variables is by looking at counts in a contingency table.
- We have a data set of two measurements of the first formant (F1).

- We have F1 values for men and women for the vowels /i/, /e/, /a/, /o/, and /u/ in four different languages
- Women tend to have shorter vocal tracts than men and thus have higher resonance frequencies. - The average F1 of the women is 534.6 Hz and the average F1 for men is 440.9.
- We can construct a contingency table by counting how many of the observations in this data set fall above or below the mean on each of the two variables being compared.
- For example, we have the five vowels in Sele measured on two variables - male F1and female F1 - and we are interested in studying the relationship or correlation between male and female F1 frequency.

```
F1_data <- read.csv("F1_data.csv", header = TRUE, sep=",")
attach(F1_data)
plot(female,male)
lines(x=c(mean(female),mean(female)),y=c(200,900),lty=2)
lines(x=c(200,1100),y=c(mean(male),mean(male)),lty=2)
abline(lm(male~female))
```



- The grid lines mark the average female (vertical line) and male (horizontal line) F1 values. The diagonal line is the best fitting straight line (the linear regression) that relates female F1 to male F1
- If the male F1 falls below the average male F1, then the female F1 for that vowel will probably also fall below the average F1 for female speakers. In only one case does this relationship not hold.
- Contingency tables are a useful way to see the relationship, or lack of one, between two variables
- From this plot/table all we know is that if the male F1 is above average so is the female F1
- But we don't know whether they tend to be the **same amount above average** or if sometimes the amount above average for males is much more than it is for females. It would be much better to explore the relationship of these two variables without throwing out this information

		female F1	
		below	above
male F1	above	0	6
	below	12	1

Figure 2: Contingency table

- Here, we can see the four cells of the contingency table
- There are 6 data points in the upper right quadrant of the graph
- 12 data points in the lower left
- And 1 that just barely ended up in the lower right quadrant.
- These quadrants were marked in the graph by drawing a dashed line at the mean values for the male (441 Hz) and female (535 Hz) talkers.
- We can see, that the relationship between male and female F1 values goes beyond simply being in one quadrant of the graph or not.
- In fact, if we can divide the lower left and the upper right quadrants into quadrants again
- We would still have the relationship, higher male F1 is associated with higher female F1.
- **We need a measure of association that will give us a consistent indication of how closely related two variables are**
- Developing a measure of association between two variables is to measure deviation from the mean $(x_i - \bar{x})$
- The association of male F1 and female F1 can be captured by seeing that when female F1 (let's call this variable x) was higher than the female mean, male F1 (y) was also higher than the male mean.
- That means that if $(x_i - \bar{x})^-$ is positive
  - then $(y_i - \bar{y})$ is also positive
- The association is strongest when the **magnitudes of these deviations are matched**
- when x_i is quite a bit larger than the $\bar{x}$ and y_i is also quite a bit larger than the $\bar{y}$

Correlations and covariance

- The strength of the association can be gauged by multiplying the deviations
- If indeed is x_i quite a bit larger than $\bar{x}$ and y_i is also much larger than $\bar{y}$
  - then the product will be greater than if y_i is only a little larger than the $\bar{y}$
- Also if x_i is quite a bit less than $\bar{x}$ and y_i is also quite a bit less than $\bar{y}$ the product will again be a large positive value
- Product of the deviations will be larger as we have a larger and larger data set,
- So we normalize this value to the size of the data set by taking the average of the paired deviations.
- This average product of the deviations is called the covariance of X and Y
- Sum of the product of the deviations, $\sum_{i=0}^{n}(x_i - \bar{x})(y_i - \bar{y})$
- Covariance of x and y, $\frac{\sum_{i=0}^{n}(x_i-\bar{x})(y_i-\bar{y})}{n}$
- The size of a deviation from the mean can be standardized so that we can compare deviations from different data sets on the same measurement scale.
- Deviation can be expressed in units of standard deviation with the z-score normalization.
- This is also done when we measure association as well.
- The correlation coefficient rxy is simply a scaled version of the sum of the product of the deviations using the idea that this value will be highest when x and y deviate from their means in comparable magnitude.
- **Correlation is identical to covariance, except that correlation is scaled by the standard deviations**
- While covariance can have any value, correlation ranges from 1 to -1 (perfect positive correlation is 1 and perfect negative correlation is -1)
- Correlation of x and y, $\frac{\sum_{i=0}^{n}\frac{(x_i-\bar{x})}{s_x}\frac{(y_i-\bar{y})}{s_y}}{n} = \frac{\sum_{i=0}^{n}(z_x)(z_y)}{n} = r_{xy}$

Day 3

The LM function

```
summary(lm(male~female))
```

```
Call:
lm(formula = male ~ female)

Residuals:
    Min      1Q  Median      3Q     Max
-70.619 -18.170   3.767  26.053  51.707

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 47.59615   23.85501   1.995   0.0623 .
female       0.73564    0.04162  17.676 2.23e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 37.49 on 17 degrees of freedom
Multiple R-squared:  0.9484,    Adjusted R-squared:  0.9454
F-statistic: 312.4 on 1 and 17 DF,  p-value: 2.23e-12
```

```
cor(male,female)
```

```
[1] 0.9738566
```

Finding the best fitting line

- Assuming we have a perfect correlation between x and y we can say that:
- $\frac{y_i-\bar{y}}{s_y} = \frac{x_i-\bar{x}}{s_x}$ Assuming deviations are equivalent, i.e., $r_{xy} = 1$
- Now we can predict the $\widehat{y}_i$, estimating y_i from x_i when $r_{xy} = 1$
- $\widehat{y}_i = \frac{s_y}{s_x}(x_i - \bar{x}) + \bar{y}$
- Even if the correlation is not perfect (not equal to 1) we would like to get the predicted $\widehat{y}_i$
- The best prediction of z_x is r_{xy} times z_x, making our prediction of $\widehat{y}_i$
- $\widehat{y}_i = r_{xy}\frac{s_y}{s_x}(x_i - \bar{x}) + \bar{y}$

- Now we can fit this into a line equation of the form:
  - "slope-intercept", y = Bx +A
  - Here, "B" is the slope and "A" gives the y-intercept
  - Or where the line crosses the y-axis

T-test for two-samples

Comparing Cherokee VOTs from 1971 and 2001

- Previously, we saw that we can test the hypothesis that a sample mean value x is the same as or different from a particular hypothesized population mean μ
- **The key question of interest** could be about comparison of two sample means; such as in the Cherokee 1971/2001 data
- **Is the mean VOT in 1971 different from the mean VOT in 2001, as the boxplot suggests**
- We want to test whether the average VOT in 1971 was equal to the average VOT in 2001
- We think that for this speaker there may have been a slow drift in the aspiration of voiceless stops as a result of language contact
- This question provides us with the null hypothesis that there was no reliable difference in the true, population, means for these two years - that is: H0: μ1971 = μ2001

```
vot <- read.delim("cherokeeVOT.txt")
vot
```

```
   VOT year Consonant
1   67 1971         k
2  127 1971         k
3   79 1971         k
4  150 1971         k
5   53 1971         k
6   65 1971         k
7   75 1971         k
8  109 1971         k
9  109 1971         t
10 126 1971         t
11 129 1971         t
12 119 1971         t
13 104 1971         t
14 153 1971         t
15 124 1971         t
16 107 1971         t
```

```
17 181 1971         t
18 166 1971         t
19  84 2001         k
20  82 2001         k
21  72 2001         k
22 193 2001         k
23 129 2001         k
24  77 2001         k
25  72 2001         k
26  81 2001         k
27  45 2001         k
28  74 2001         k
29 102 2001         k
30  77 2001         k
31 187 2001         k
32  79 2001         t
33  86 2001         t
34  59 2001         t
35  74 2001         t
36  63 2001         t
37  75 2001         t
38  70 2001         t
39 106 2001         t
40  54 2001         t
41  49 2001         t
42  56 2001         t
43  58 2001         t
44  97 2001         t
```

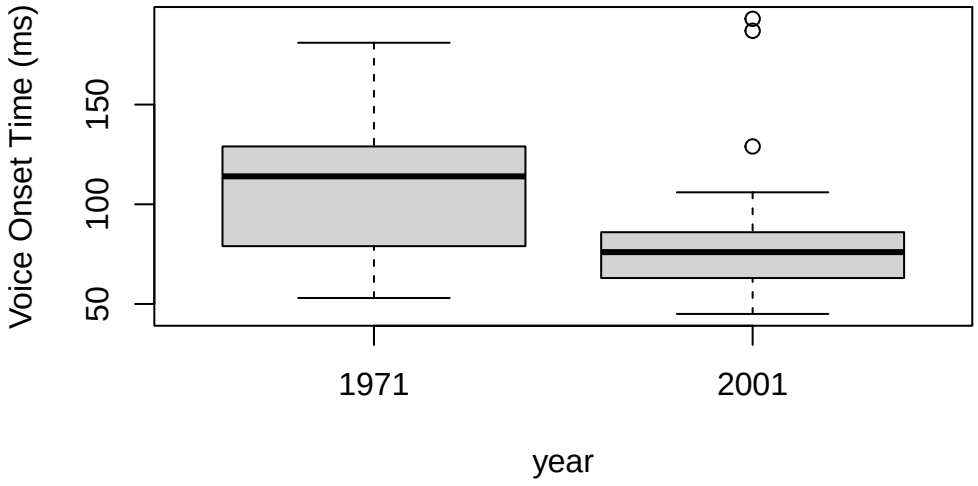
```
attach(vot)
summary(vot)
```

```
      VOT              year       Consonant
 Min.   : 45.00   Min.   :1971   Length:44
 1st Qu.: 71.50   1st Qu.:1971   Class :character
 Median : 81.50   Median :2001   Mode  :character
 Mean   : 96.45   Mean   :1989
 3rd Qu.:120.25   3rd Qu.:2001
 Max.   :193.00   Max.   :2001
```

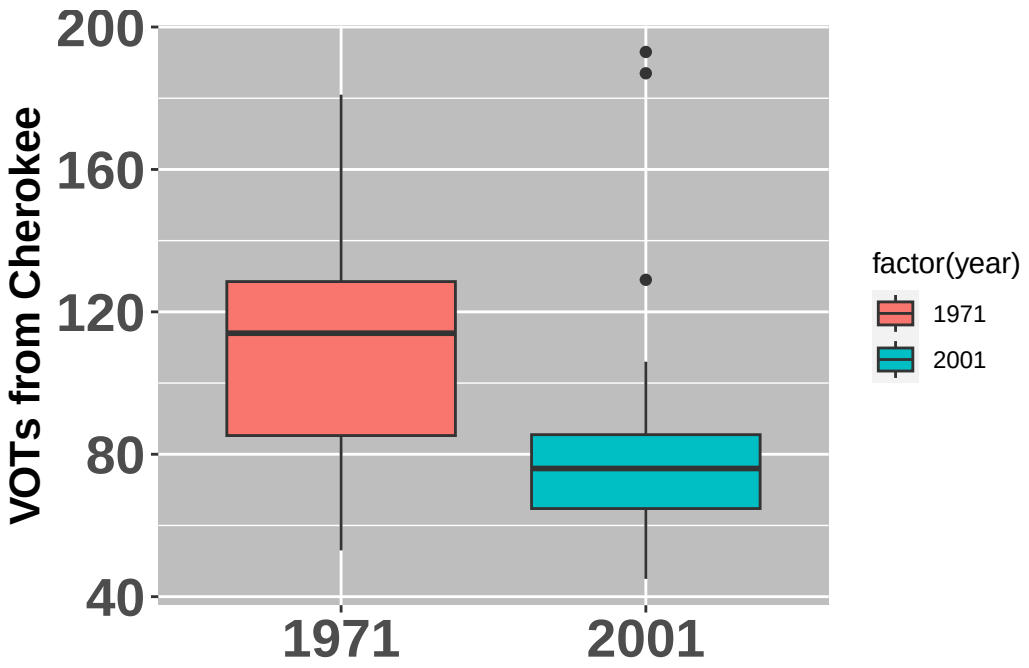
```r
# We need to treat year as a nominal variable and not a continuous variable
vot$year <- factor(vot$year)
summary(vot)
```

```
      VOT           year     Consonant
 Min.   : 45.00   1971:18   Length:44
 1st Qu.: 71.50   2001:26   Class :character
 Median : 81.50             Mode  :character
 Mean   : 96.45
 3rd Qu.:120.25
 Max.   :193.00
```

```r
boxplot(VOT~year, data = vot, col="lightgrey", ylab = "Voice Onset Time (ms)")
```




```r
library(ggplot2)
vot_bp<-ggplot(vot, aes(x=year, y=VOT)) + geom_boxplot(aes(fill = factor(year))) +
  ylab("VOTs from Cherokee") +
  theme(axis.text.x = element_text(size=20, face="bold"),axis.text.y = element_text(face="bo
        axis.title.x=element_blank(), axis.title.y = element_text(size=16, face="bold"))
vot_bp= vot_bp + theme(panel.background = element_rect(fill = "gray",colour = NA), legend.pos
vot_bp
```



Testing our two-samples

- We can test this hypothesis with a t- test similar to the "one sample" t-test that we discussed earlier
- There, we tested the null hypothesis: H0: μ1971 = μhyp
- We supplied the hypothesized population mean.
- The idea with the t-test is that we expect the difference between means to be zero
- The null hypothesis is that there is no difference
  - and we measure the magnitude of the observed difference relative to the magnitude of random or chance variation we expect in mean values (the standard error of the mean)
- If the difference between means is large, more than about 2 standard errors (a t value of 2 or -2)
- We are likely to conclude that the sample mean comes from a population that has a different mean than the hypothesized population mean
- In testing whether the mean VOT in 1971 is different from the mean VOT in 2001 for this talker we are combining two null hypotheses:
  - H0: μ1971 = μ
  - H0: μ2001 = μ
  - H0: μ1971 = μ2001
- In other words, the expected mean value of the 1971 sample is the same as the expected value of the 2001 sample
- Same as with a one-sample t-test the expected value of the difference is 0.

- Therefore we can compute a t statistic from the difference between the means of our two samples.
- **But, when we compare the two means in this computation of t. We have two samples of data; one from 1971 and one from 2001**
- We have *two estimates* of the standard error of the mean (SE). In calculating the t statistic we need to take information from both the 1971 data set and the 2001 data set when we compute the SE for this test.
- $t = \frac{\bar{x}_{1971} - \bar{x}_{2001}}{SE}$; the two-sample *t* value
- We need to compute the SE for this test, and we have two SEs, one for 1971 and one for 2001
- **What is our estimate of the standard error of the mean?**
- With only one sample we used the standard deviation or the variance of the sample to estimate the standard error
- With two samples, there are two estimates of variance, s^2_{1971} and s^2_{2001}

- If we can assume that these two represent essentially the same value then we can pool them by taking the weighted average as our best estimate of SE
- Before pooling the variances from our 1971 and 2001 samples we need to test the hypothesis that they do not differ
- This hypothesis can be tested using the F distribution
  - a theoretical probability distribution that gives probabilities for ratios of variances
- If we want to know whether the two estimates of variance are equal to each other
- We can simply take their ratio and test the probability of this ratio, given the degrees of freedom that went into each variance estimate
- We are testing the hypothesis that H0=$s^2_{1971} = s^2_{201}$
- $SE = \frac{s}{\sqrt{n}} = \sqrt{\frac{s^2}{n}}$

Establishing equal variance(s)

- We do this with the F distribution because this distribution lets us specify degrees of freedom for the numerator and the denominator of the ratio
- The variances are not very different from each other (36.1 ms versus 35.9 ms)
- Also, the variances are very similar in magnitude
- Thus the F ratio is close to one
- $F = \frac{s^2_{2001}}{s^2_{1971}} = \frac{36.0876^2}{35.9284^2} = \frac{1302.32}{1290.85} = 1.0089$, F-test of equality of variance
- We look up the probability of getting an F of 1.0089 or higher using the R pf() function
- In this function, we specify the F value, the degrees of freedom the numerator (n2001 - 1 = 25) and of the denominator (n1971 - 1 = 17)
- We also specify that we are looking at the upper tail of the F distribution because, we put the larger of the two variances as the numerator.
- The probability of getting an F value of 1.0089 or higher when the variances are in fact equal is quite high p=0.5 so we have no reason to believe that the variance of the 1971 data is any different

from the variance of the 2001 data

```
pf(1.0089,25,17,lower.tail=F)
```

```
[1] 0.5034847
```

Pooling variance

- We can estimate SE for our test of whether VOT was different in 2001 than it was in 1971 by pooling the two sample variances
- This is done using the weighted average of the variances where each variance is weighted by its degrees of freedom.
- Let's calculate the weighted average of the pooled variances?

```
attach(vot)
```

```
The following objects are masked from vot (pos = 4):

    Consonant, VOT, year
```

```
pooled_variance=(var(VOT[year=="1971"])*17 + var(VOT[year=="2001"])*25) / (17+25)
pooled_variance
```

```
[1] 1297.676
```

- The pooled variance for our Cherokee VOT data is 1297.7 and hence the pooled standard deviation is s = 36.02
- The t statistic that we use to compare two means uses the pooled variance from the two samples to estimate SE - the standard error of the mean(s), and t is a ratio of

    1. The difference between the two means $\bar{x}_a - \bar{x}_b$
    2. SE calculated from the pooled variance

- $t = \frac{\bar{x}_a - \bar{x}_b}{\sqrt{\frac{s_p^2}{(n_a + n_b)}}}$

```
t.test(VOT[year=="1971"],VOT[year=="2001"], var.equal=TRUE, alternative="greater")
```

```
    Two Sample t-test

data:  VOT[year == "1971"] and VOT[year == "2001"]
t = 2.6116, df = 42, p-value = 0.006223
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 10.2681     Inf
sample estimates:
mean of x mean of y
113.50000  84.65385
```

Paired t-test

- Naturally occurring pairs of observations
- F1 data for men and women for each language and vowel in the data set
- The male F1 of /a/ in Sele ~ the female F1 of /a/ in Sele
- The male F1 of /i/ in Sele ~ the female F1 of /i/ in Sele
- Men and women tend to have different vowel F1 frequency
- However, the difference between vowels can be bigger than the overall male/female difference
- To test the male/female difference we have to have a control for the vowel differences
- Pairing the male/female differences by vowel gives us this control
- Define a derived variable
- The difference between the paired observations, $d_i = x_{ai} - x_{bi}$
- Calculate the mean and variance of this difference such as for any other variable
- Test the null hypothesis - there is no difference between the paired observations, $H_0 : d = 0$

```
attach(F1_data)
```

```
The following objects are masked from F1_data (pos = 6):

    female, language, male, vowel
```

```
t.test(female, male, alternative="greater", var.equal = TRUE) # Two Sample t-test
```

```
    Two Sample t-test

data:  female and male
t = 1.5356, df = 36, p-value = 0.06669
```

```
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 -9.323753       Inf
sample estimates:
mean of x mean of y
 534.6316  440.8947
```

```r
t.test(female, male, paired="TRUE", alternative="greater") # Paired t-test
```

```
    Paired t-test

data:  female and male
t = 6.1061, df = 18, p-value = 4.538e-06
alternative hypothesis: true mean difference is greater than 0
95 percent confidence interval:
 67.11652      Inf
sample estimates:
mean difference
       93.73684
```

Paired test in comparison to two-samples

- Paired t-tests help us remove systematic differences due to vowel or language influence
- This is because the F1 difference is immune to any vowel or language influences
- F1 variation due to language or vowel category is automatically controlled by taking paired F1 measurements from the same vowels spoken by speakers of the same language
- The paired t-test, due to the underlying controls, tends to be more sensitive compared to the two sample t test
- Comparing between the independent samples t-test and the paired t-test, the paired t-test gives us more reliable results
- True difference in means between males and females; controlled by vowel quality and language

Multiple regressions

- X-ray Microbeam uses a narrow high-energy x-ray beam to track gold pellets attached to articulators while synchronously acquiring the physiological data. Not very portable: 15 tons. Gold pellets are secured by wires

- EMA (Electromagnetic Articulograph) consists of a helmet with three transmitters, microscopic sensor coils (which are attached to a subject's articulators such as the tongue, the jaw, and the lips), and a control computer
- These articulatory tracking systems tell us about speech production directly
- Tongue modeling; but without the root of the tongue
- Interpolation: Let us assume that we can use the data from the location of points on the top surface of the tongue to make predictions about the location of a point on the root of the tongue
- We have seen that we can define a regression line $y = a + bx$
- This simple equation allows us to express a linear relationship that might exist between two variables
- We also have the ability to measure the strength of this linear association with the Pearson's correlation coefficient r
- Assume that each pellet that is placed on the tongue can take two variable positions, x and y; x representing the front-back axis and y representing the high-low axis
- So there are 30 variables in all. Why?
- Some highly correlated pellets can be seen
- A regression formula predicts 99.7% of the variance of pellet 14's y location if we know the y location of pellet 15
- Aside from predicting y-y relationships, we could also predict the xy location of one of the back pellets from one of the front pellets.
- The correlation between the y locations of pellet 1 and pellet 14 is r = 0.817
- Highest correlation between the x location of pellet 14 and any of the front pellets is with the y location of pellet 6 (r = 0.61)
- We could keep looking into individual correlations (435 in all) and find out more about the predictions, but…that would become tedious

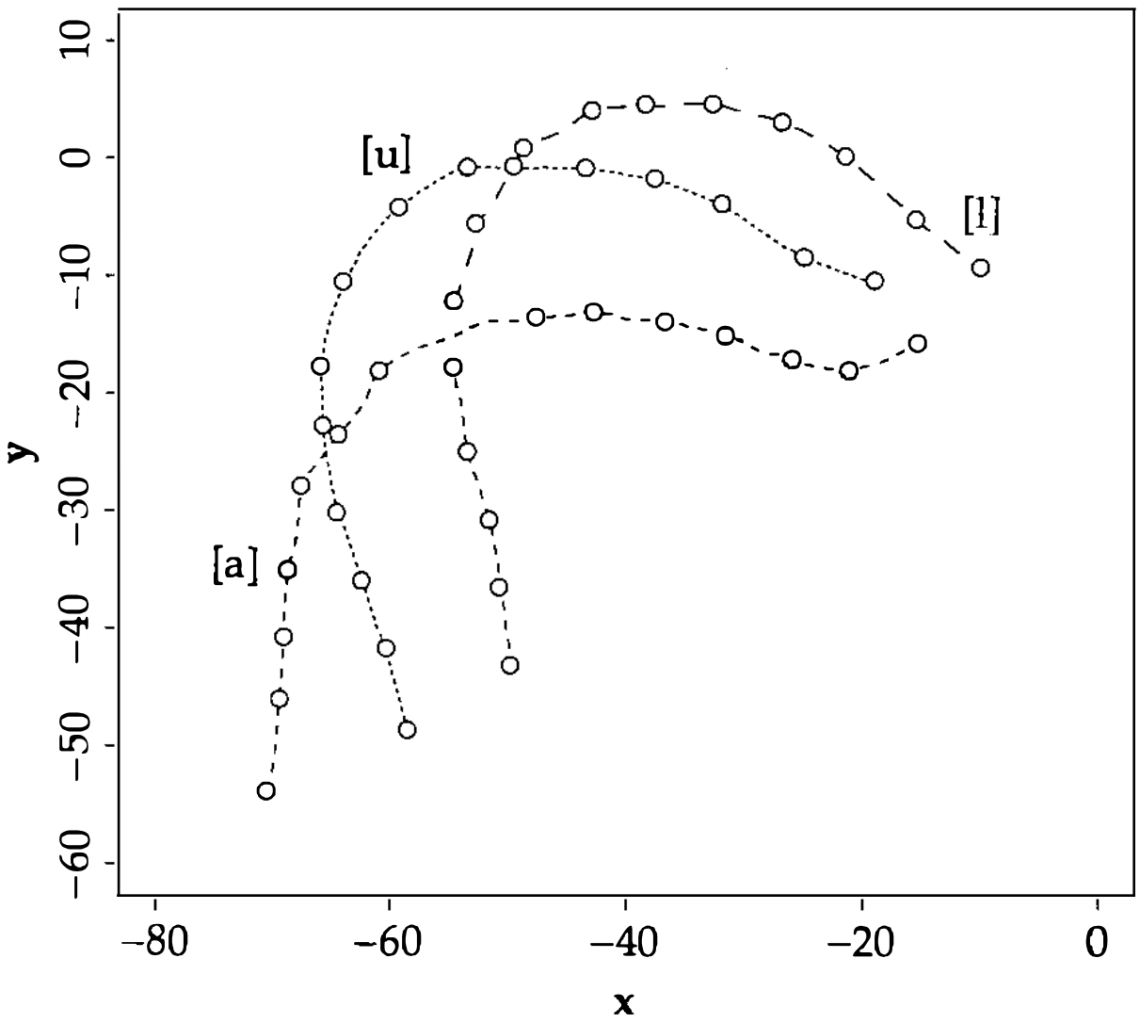


Figure 3: Tongue shape recorded in the x and y locations of the pellets for the corner vowels [i], [a], and [u]

- $y = a + b_1x_1 + b_2x_2 + ... + b_nx_n$
- We are working with the assumption that there are 15 points on the tongue that model the dynamic nature of the tongue
- Are these 15 adjacent points independent of each other; both statistically and physically?
- The question we are trying to answer: how many independent parameters of tongue movement are there?
- If we are able to identify the actually independent parameters/factors that influence the movement of the tongue, then we have successfully modeled the dynamic nature of the tongue
- There are many patterns that suggest some inter-relationships between the different x and y points and some of these could be causal

- Now we need to employ techniques that will help us unearth the nature of these relationships
- Let's take a close look at $y = a + b_1x_1 + b_2x_2 + ... + b_nx_n$
- The above linear equation can be thought of expressing a complex relationship between y and parameters $x_1...x_n$
- $x_{15} = -51.69 - O.97y_5 + 1.05x_2 - 4.04x_6 + 4.66x_4 + O.61y_2 - 3.69x_3 + 2.66x_5 + 1.48y_4$
- So a linear combination of some of the tongue pellet xy variables, produces an estimate of x_{15} that accounts for 98% of the variance of the x location of pellet 15
- Estimating $\hat{y} = A + Bx$, where A is the intercept and B is the slope
- Multiple regression is an extension of linear regression where the y position is estimated based on both x_i and y_i values and their corresponding coefficients

```
chain<-read.delim("chaindata.txt")
PL1<-subset(chain, talker=="PL1",x1:y15)
cor(PL1)
```

```
            x1         x2         x3         x4         x5         x6
x1   1.0000000  0.9972439  0.9586665  0.9727643  0.9485555  0.9313837
x2   0.9972439  1.0000000  0.9737532  0.9839730  0.9647815  0.9452476
x3   0.9586665  0.9737532  1.0000000  0.9911912  0.9933240  0.9844554
x4   0.9727643  0.9839730  0.9911912  1.0000000  0.9939075  0.9706192
x5   0.9485555  0.9647815  0.9933240  0.9939075  1.0000000  0.9791682
x6   0.9313837  0.9452476  0.9844554  0.9706192  0.9791682  1.0000000
x7   0.8826467  0.9045604  0.9676897  0.9444815  0.9590505  0.9857084
x8   0.8848730  0.9027061  0.9631517  0.9355931  0.9485909  0.9857265
x9   0.8670081  0.8768510  0.9138043  0.8855591  0.8877869  0.9476435
x10  0.8180217  0.8152159  0.8197234  0.7963674  0.7849567  0.8546101
x11  0.7099233  0.6971155  0.6680835  0.6508381  0.6230325  0.7074034
x12  0.6150218  0.5978171  0.5523024  0.5423943  0.5083533  0.5912593
x13  0.5427050  0.5190964  0.4503409  0.4550445  0.4125353  0.4861107
x14  0.5594826  0.5376614  0.4534726  0.4700785  0.4215692  0.4701265
x15  0.4264830  0.4005368  0.2945995  0.3247048  0.2748709  0.3018348
y1  -0.6430144 -0.6515223 -0.6004651 -0.6607319 -0.6310148 -0.4970681
y2  -0.4468121 -0.4607197 -0.4403215 -0.4937839 -0.4830498 -0.3399075
y3  -0.2591833 -0.2814760 -0.3021502 -0.3399317 -0.3519970 -0.2159412
y4  -0.1896134 -0.2149814 -0.2485273 -0.2868717 -0.3030113 -0.1660916
y5  -0.1680348 -0.1951401 -0.2350771 -0.2700908 -0.2879589 -0.1536507
y6  -0.1812421 -0.2092955 -0.2552343 -0.2880069 -0.3045802 -0.1778198
y7  -0.1591502 -0.1854801 -0.2325480 -0.2632983 -0.2796905 -0.1613945
y8  -0.2306270 -0.2541891 -0.2864455 -0.3227920 -0.3288238 -0.2151308
y9  -0.3291291 -0.3440023 -0.3410436 -0.3888807 -0.3750269 -0.2659207
y10 -0.4209402 -0.4301827 -0.4063643 -0.4585186 -0.4326318 -0.3191451
y11 -0.4373993 -0.4447500 -0.4124883 -0.4642736 -0.4339063 -0.3286466
```

```
y12 -0.4517399 -0.4559899 -0.4146855 -0.4709292 -0.4366739 -0.3293244
y13 -0.4538007 -0.4595669 -0.4191213 -0.4753218 -0.4421529 -0.3303649
y14 -0.4631173 -0.4692994 -0.4240054 -0.4842038 -0.4507894 -0.3339151
y15 -0.4793341 -0.4842853 -0.4374637 -0.4991052 -0.4651097 -0.3408050
           x7          x8          x9          x10        x11       x12
x1   0.8826467  0.88487298  0.86700811  0.818021743 0.70992333 0.6150218
x2   0.9045604  0.90270612  0.87685096  0.815215866 0.69711553 0.5978171
x3   0.9676897  0.96315167  0.91380429  0.819723404 0.66808354 0.5523024
x4   0.9444815  0.93559314  0.88555909  0.796367409 0.65083815 0.5423943
x5   0.9590505  0.94859089  0.88778693  0.784956716 0.62303248 0.5083533
x6   0.9857084  0.98572653  0.94764355  0.854610068 0.70740345 0.5912593
x7   1.0000000  0.99081763  0.95285546  0.859488426 0.71648508 0.6019182
x8   0.9908176  1.00000000  0.97320880  0.885494088 0.74509298 0.6294228
x9   0.9528555  0.97320880  1.00000000  0.964336028 0.87235904 0.7844257
x10  0.8594884  0.88549409  0.96433603  1.000000000 0.96678427 0.9146643
x11  0.7164851  0.74509298  0.87235904  0.966784275 1.00000000 0.9828830
x12  0.6019182  0.62942279  0.78442573  0.914664271 0.98288296 1.0000000
x13  0.4901337  0.52116807  0.69360066  0.850881714 0.94789022 0.9852569
x14  0.4701821  0.49331761  0.66258850  0.819814045 0.91795544 0.9626874
x15  0.2892777  0.31663918  0.49956409  0.678650925 0.79812560 0.8787212
y1  -0.4493588 -0.39762296 -0.25567549 -0.128058327 0.00701521 0.1030256
y2  -0.2874986 -0.23956208 -0.05595084  0.113137557 0.27798263 0.3818132
y3  -0.1729971 -0.12558195  0.08611065  0.288820384 0.47602392 0.5837413
y4  -0.1320502 -0.08491869  0.13110202  0.339396744 0.52740956 0.6301562
y5  -0.1215132 -0.07950614  0.13694187  0.350869624 0.54158884 0.6450023
y6  -0.1481857 -0.10968771  0.10508806  0.323834232 0.51731325 0.6226740
y7  -0.1264282 -0.09670945  0.11505580  0.335960567 0.52879959 0.6350133
y8  -0.1734058 -0.14670145  0.04807574  0.258860153 0.44284226 0.5435993
y9  -0.2138385 -0.18813869 -0.02048560  0.161114087 0.31756533 0.4108731
y10 -0.2628619 -0.23662992 -0.08265659  0.076472242 0.22077876 0.3090629
y11 -0.2706012 -0.24680259 -0.10638710  0.042671931 0.17508219 0.2615567
y12 -0.2671119 -0.24499111 -0.11066590  0.028645171 0.15467899 0.2373482
y13 -0.2703928 -0.24499080 -0.10717815  0.034424910 0.16248729 0.2466154
y14 -0.2723349 -0.24611021 -0.11711575  0.016633350 0.14085239 0.2167362
y15 -0.2799517 -0.25335091 -0.12059365  0.008672668 0.13438856 0.2130937
          x13        x14       x15          y1          y2          y3
x1  0.5427050 0.55948263 0.4264830 -0.64301440 -0.44681211 -0.25918335
x2  0.5190964 0.53766139 0.4005368 -0.65152225 -0.46071973 -0.28147602
x3  0.4503409 0.45347265 0.2945995 -0.60046514 -0.44032148 -0.30215021
x4  0.4550445 0.47007852 0.3247048 -0.66073190 -0.49378393 -0.33993165
x5  0.4125353 0.42156921 0.2748709 -0.63101478 -0.48304978 -0.35199697
x6  0.4861107 0.47012650 0.3018348 -0.49706814 -0.33990750 -0.21594119
x7  0.4901337 0.47018206 0.2892777 -0.44935875 -0.28749855 -0.17299711
```

```
x8  0.5211681 0.49331761 0.3166392 -0.39762296 -0.23956208 -0.12558195
x9  0.6936007 0.66258850 0.4995641 -0.25567549 -0.05595084  0.08611065
x10 0.8508817 0.81981405 0.6786509 -0.12805833  0.11313756  0.28882038
x11 0.9478902 0.91795544 0.7981256  0.00701521  0.27798263  0.47602392
x12 0.9852569 0.96268739 0.8787212  0.10302563  0.38181315  0.58374133
x13 1.0000000 0.98764307 0.9335377  0.14083720  0.42054153  0.63215955
x14 0.9876431 1.00000000 0.9561889  0.05900981  0.34379768  0.57172069
x15 0.9335377 0.95618892 1.0000000  0.14094886  0.40557629  0.61428783
y1  0.1408372 0.05900981 0.1409489  1.00000000  0.95124981  0.83663103
y2  0.4205415 0.34379768 0.4055763  0.95124981  1.00000000  0.96163146
y3  0.6321596 0.57172069 0.6142878  0.83663103  0.96163146  1.00000000
y4  0.6740785 0.61544424 0.6469020  0.79590430  0.93787888  0.99406470
y5  0.6896241 0.63247447 0.6620363  0.77225258  0.92332726  0.98793904
y6  0.6723244 0.61637935 0.6612791  0.76690338  0.91666672  0.97896353
y7  0.6833332 0.63081356 0.6799293  0.73377745  0.89563802  0.96441262
y8  0.5902782 0.52604757 0.5962323  0.76191321  0.89837013  0.93565489
y9  0.4481634 0.37199594 0.4657648  0.81415716  0.90197027  0.88510293
y10 0.3463403 0.26631727 0.3643715  0.86650669  0.91433732  0.86069505
y11 0.2960953 0.21342315 0.3244851  0.85178958  0.88665380  0.81906475
y12 0.2659346 0.18139017 0.2883868  0.85763003  0.88372778  0.80562102
y13 0.2757862 0.18911409 0.2945650  0.87247430  0.89811754  0.82049735
y14 0.2393083 0.14369153 0.2409664  0.86625112  0.88421973  0.79830903
y15 0.2333982 0.13841843 0.2333638  0.89128607  0.90463227  0.81328223
             y4          y5         y6          y7          y8         y9
x1  -0.18961340 -0.16803484 -0.1812421 -0.15915018 -0.23062704 -0.3291291
x2  -0.21498142 -0.19514007 -0.2092955 -0.18548007 -0.25418914 -0.3440023
x3  -0.24852731 -0.23507711 -0.2552343 -0.23254798 -0.28644546 -0.3410436
x4  -0.28687173 -0.27009083 -0.2880069 -0.26329831 -0.32279199 -0.3888807
x5  -0.30301133 -0.28795894 -0.3045802 -0.27969055 -0.32882377 -0.3750269
x6  -0.16609155 -0.15365074 -0.1778198 -0.16139454 -0.21513077 -0.2659207
x7  -0.13205022 -0.12151323 -0.1481857 -0.12642824 -0.17340577 -0.2138385
x8  -0.08491869 -0.07950614 -0.1096877 -0.09670945 -0.14670145 -0.1881387
x9   0.13110202  0.13694187  0.1050881  0.11505580  0.04807574 -0.0204856
x10  0.33939674  0.35086962  0.3238342  0.33596057  0.25886015  0.1611141
x11  0.52740956  0.54158884  0.5173133  0.52879959  0.44284226  0.3175653
x12  0.63015619  0.64500231  0.6226740  0.63501328  0.54359930  0.4108731
x13  0.67407847  0.68962408  0.6723244  0.68333324  0.59027821  0.4481634
x14  0.61544424  0.63247447  0.6163794  0.63081356  0.52604757  0.3719959
x15  0.64690203  0.66203632  0.6612791  0.67992930  0.59623228  0.4657648
y1   0.79590430  0.77225258  0.7669034  0.73377745  0.76191321  0.8141572
y2   0.93787888  0.92332726  0.9166667  0.89563802  0.89837013  0.9019703
y3   0.99406470  0.98793904  0.9789635  0.96441262  0.93565489  0.8851029
y4   1.00000000  0.99796289  0.9905223  0.97736258  0.94129214  0.8762571
```

```
y5   0.99796289  1.00000000  0.9955177  0.98596275  0.94973687  0.8791254
y6   0.99052225  0.99551771  1.0000000  0.99577617  0.97170807  0.9083825
y7   0.97736258  0.98596275  0.9957762  1.00000000  0.98254780  0.9216446
y8   0.94129214  0.94973687  0.9717081  0.98254780  1.00000000  0.9742688
y9   0.87625708  0.87912541  0.9083825  0.92164455  0.97426878  1.0000000
y10  0.84139587  0.83899599  0.8677333  0.87500402  0.93783735  0.9868552
y11  0.79468406  0.79186422  0.8244852  0.83547263  0.91262295  0.9793339
y12  0.77951007  0.77520441  0.8068095  0.81736482  0.89745218  0.9713757
y13  0.79416606  0.78906622  0.8181364  0.82578330  0.90193599  0.9726823
y14  0.77153634  0.76600388  0.7949487  0.80206132  0.88636719  0.9631291
y15  0.78591615  0.77850325  0.8030402  0.80594877  0.88231860  0.9562594
            y10         y11         y12         y13         y14          y15
x1  -0.42094022 -0.43739926 -0.45173988 -0.45380071 -0.46311733 -0.479334083
x2  -0.43018266 -0.44475002 -0.45598990 -0.45956686 -0.46929945 -0.484285271
x3  -0.40636425 -0.41248831 -0.41468546 -0.41912132 -0.42400543 -0.437463686
x4  -0.45851857 -0.46427360 -0.47092916 -0.47532177 -0.48420376 -0.499105231
x5  -0.43263181 -0.43390631 -0.43667390 -0.44215292 -0.45078936 -0.465109692
x6  -0.31914506 -0.32864664 -0.32932443 -0.33036488 -0.33391514 -0.340804991
x7  -0.26286193 -0.27060122 -0.26711187 -0.27039277 -0.27233492 -0.279951654
x8  -0.23662992 -0.24680259 -0.24499111 -0.24499080 -0.24611021 -0.253350909
x9  -0.08265659 -0.10638710 -0.11066590 -0.10717815 -0.11711575 -0.120593647
x10  0.07647224  0.04267193  0.02864517  0.03442491  0.01663335  0.008672668
x11  0.22077876  0.17508219  0.15467899  0.16248729  0.14085239  0.134388558
x12  0.30906295  0.26155672  0.23734818  0.24661543  0.21673617  0.213093726
x13  0.34634035  0.29609531  0.26593457  0.27578619  0.23930832  0.233398232
x14  0.26631727  0.21342315  0.18139017  0.18911409  0.14369153  0.138418428
x15  0.36437145  0.32448509  0.28838678  0.29456500  0.24096644  0.233363825
y1   0.86650669  0.85178958  0.85763003  0.87247430  0.86625112  0.891286071
y2   0.91433732  0.88665380  0.88372778  0.89811754  0.88421973  0.904632267
y3   0.86069505  0.81906475  0.80562102  0.82049735  0.79830903  0.813282233
y4   0.84139587  0.79468406  0.77951007  0.79416606  0.77153634  0.785916146
y5   0.83899599  0.79186422  0.77520441  0.78906622  0.76600388  0.778503246
y6   0.86773334  0.82448524  0.80680947  0.81813640  0.79494867  0.803040241
y7   0.87500402  0.83547263  0.81736482  0.82578330  0.80206132  0.805948771
y8   0.93783735  0.91262295  0.89745218  0.90193599  0.88636719  0.882318605
y9   0.98685523  0.97933388  0.97137567  0.97268234  0.96312908  0.956259357
y10  1.00000000  0.99510071  0.99255503  0.99373309  0.98464383  0.983356475
y11  0.99510071  1.00000000  0.99861383  0.99832613  0.99246803  0.987339321
y12  0.99255503  0.99861383  1.00000000  0.99922809  0.99499543  0.991746225
y13  0.99373309  0.99832613  0.99922809  1.00000000  0.99585279  0.994107054
y14  0.98464383  0.99246803  0.99499543  0.99585279  1.00000000  0.996442112
y15  0.98335648  0.98733932  0.99174623  0.99410705  0.99644211  1.000000000
```

```
cov(PL1)
```

```
            x1         x2         x3         x4         x5         x6         x7
x1   18.688494  20.257932  19.559742  20.71310  18.317303  16.912481  16.101781
x2   20.257932  22.080714  21.595530  22.77404  20.251032  18.657079  17.936760
x3   19.559742  21.595530  22.274955  23.04179  20.941653  19.516232  19.272782
x4   20.713102  22.774042  23.041793  24.26056  21.867947  20.081253  19.631059
x5   18.317303  20.251032  20.941653  21.86795  19.953692  18.372171  18.078109
x6   16.912481  18.657079  19.516232  20.08125  18.372171  17.643432  17.471889
x7   16.101781  17.936760  19.272782  19.63106  18.078109  17.471889  17.807368
x8   15.551510  17.244770  18.480238  18.73449  17.226421  16.832648  16.998009
x9   16.117945  17.718694  18.546457  18.75717  17.053744  17.117325  17.291245
x10  16.514806  17.889602  18.067466  18.31831  16.374907  16.764130  16.937971
x11  16.328849  17.428832  16.776299  17.05613  14.807417  15.809411  16.086590
x12  16.052530  16.960580  15.738059  16.12989  13.710178  14.994615  15.335684
x13  15.201030  15.804351  13.771202  14.52200  11.939739  13.229662  13.400977
x14  16.474682  17.209121  14.578163  15.77118  12.826962  13.450844  13.514787
x15  12.715161  12.980217   9.589003  11.02991   8.467855   8.743671   8.418755
y1  -19.571558 -21.555265 -19.953256 -22.91359 -19.845805 -14.700260 -13.350903
y2  -15.262113 -17.105902 -16.420292 -19.21719 -17.049283 -11.281189  -9.586016
y3   -9.272095 -10.945398 -11.800895 -13.85561 -13.011704  -7.506036  -6.041188
y4   -6.921835  -8.530463  -9.904844 -11.93173 -11.429720  -5.891206  -4.705482
y5   -6.344121  -8.008257  -9.689553 -11.61837 -11.233813  -5.636522  -4.478251
y6   -6.450443  -8.096731  -9.917236 -11.67876 -11.200994  -6.149148  -5.148129
y7   -5.524963  -6.999046  -8.813655 -10.41438 -10.032854  -5.443966  -4.284292
y8   -7.362861  -8.820897  -9.983885 -11.74147 -10.847360  -6.673340  -5.403966
y9   -9.271154 -10.532909 -10.488145 -12.48094 -10.915772  -7.278202  -5.879850
y10 -11.032171 -12.254987 -11.627258 -13.69181 -11.716111  -8.127052  -6.724825
y11 -11.514051 -12.725811 -11.854494 -13.92475 -11.802406  -8.405889  -6.953325
y12 -12.027375 -13.196448 -12.053758 -14.28569 -12.013350  -8.519434  -6.942059
y13 -11.958642 -13.163907 -12.058067 -14.27143 -12.039645  -8.458920  -6.955438
y14 -13.034671 -14.357489 -13.028719 -15.52746 -13.110136  -9.131656  -7.482127
y15 -14.487783 -15.910517 -14.435336 -17.18775 -14.525915 -10.008614  -8.259607
           x8        x9        x10        x11       x12       x13       x14
x1   15.551510 16.117945 16.5148062 16.3288491 16.052530 15.201030 16.474682
x2   17.244770 17.718694 17.8896015 17.4288323 16.960580 15.804351 17.209121
x3   18.480238 18.546457 18.0674657 16.7762994 15.738059 13.771202 14.578163
x4   18.734491 18.757169 18.3183079 17.0561251 16.129886 14.521997 15.771176
x5   17.226421 17.053744 16.3749066 14.8074169 13.710178 11.939739 12.826962
x6   16.832648 17.117325 16.7641299 15.8094105 14.994615 13.229662 13.450844
x7   16.998009 17.291245 16.9379712 16.0865904 15.335684 13.400977 13.514787
x8   16.527567 17.014134 16.8116996 16.1165440 15.449440 13.727907 13.660745
```

```
x9   17.014134 18.492631 19.3664149 19.9595892 20.366528 19.325508 19.408257
x10  16.811700 19.366415 21.8093624 24.0219379 25.789862 25.746176 26.078351
x11  16.116544 19.959589 24.0219379 28.3083108 31.573636 32.676617 33.267630
x12  15.449440 20.366528 25.7898616 31.5736358 36.452861 38.542276 39.590809
x13  13.727907 19.325508 25.7461763 32.6766168 38.542276 41.980164 43.587842
x14  13.660745 19.408257 26.0783509 33.2676298 39.590809 43.587842 46.396644
x15   8.877734 14.815748 21.8575230 29.2860793 36.588949 41.714490 44.917958
y1  -11.381342 -7.741159 -4.2106279  0.2627939  4.379542  6.424761  2.829991
y2   -7.695293 -1.901116  4.1747539 11.6863054 18.214586 21.529480 18.503284
y3   -4.224889  3.064360 11.1617667 20.9589428 29.165530 33.894746 32.226327
y4   -2.915230  4.760733 13.3842567 23.6957293 32.127662 36.880569 35.399497
y5   -2.822865  5.143048 14.3104157 25.1658543 34.010426 39.022894 37.624547
y6   -3.671180  3.720454 12.4505200 22.6596829 30.950650 35.862800 34.564845
y7   -3.157246  3.973223 12.5992546 22.5934810 30.788156 35.554099 34.504788
y8   -4.404413  1.526773  8.9276243 17.4002407 24.237840 28.244119 26.461687
y9   -4.983828 -0.574022  4.9027013 11.0095734 16.164192 18.920773 16.510565
y10  -5.832135 -2.154916  2.1651038  7.1214375 11.312697 13.604374 10.997552
y11  -6.109661 -2.785809  1.2134635  5.6723351  9.616004 11.681982  8.852129
y12  -6.134089 -2.930950  0.8238880  5.0685481  8.825656 10.611875  7.609433
y13  -6.071329 -2.809539  0.9799949  5.2699428  9.076441 10.892413  7.852297
y14  -6.514123 -3.278963  0.5057351  4.8791379  8.519598 10.094894  6.372302
y15  -7.201176 -3.625770  0.2831722  4.9991450  8.995244 10.572948  6.591946
          x15          y1         y2         y3         y4         y5
x1  12.715161 -19.5715582 -15.262113  -9.272095  -6.921835  -6.344121
x2  12.980217 -21.5552651 -17.105902 -10.945398  -8.530463  -8.008257
x3   9.589003 -19.9532560 -16.420292 -11.800895  -9.904844  -9.689553
x4  11.029913 -22.9135946 -19.217192 -13.855609 -11.931725 -11.618371
x5   8.467855 -19.8458050 -17.049283 -13.011704 -11.429720 -11.233813
x6   8.743671 -14.7002597 -11.281189  -7.506036  -5.891206  -5.636522
x7   8.418755 -13.3509027  -9.586016  -6.041188  -4.705482  -4.478251
x8   8.877734 -11.3813421  -7.695293  -4.224889  -2.915230  -2.822865
x9  14.815748  -7.7411592  -1.901116   3.064360   4.760733   5.143048
x10 21.857523  -4.2106279   4.174754  11.161767  13.384257  14.310416
x11 29.286079   0.2627939  11.686305  20.958943  23.695729  25.165854
x12 36.588949   4.3795416  18.214586  29.165530  32.127662  34.010426
x13 41.714490   6.4247614  21.529480  33.894746  36.880569  39.022894
x14 44.917958   2.8299908  18.503284  32.226327  35.399497  37.624547
x15 47.562648   6.8440336  22.100809  35.058113  37.673557  39.874920
y1   6.844034  49.5718479  52.919379  48.745568  47.319866  47.485598
y2  22.100809  52.9193787  62.431705  62.877443  62.576959  63.715229
y3  35.058113  48.7455684  62.877443  68.480548  69.464573  71.400089
y4  37.673557  47.3198660  62.576959  69.464573  71.306677  73.597742
y5  39.874920  47.4855982  63.715229  71.400089  73.597742  76.272852
```

```
y6  37.545775  44.4530388  59.628971  66.695015  68.860883  71.577617
y7  37.655796  41.4874727  56.829022  64.088714  66.275936  69.148150
y8  30.366708  39.6161503  52.421219  57.180575  58.700087  61.254430
y9  20.930523  37.3513447  46.438187  47.726307  48.214435  50.028360
y10 15.234590  36.9864695  43.798824  43.180366  43.074365  44.422018
y11 13.626700  36.5184907  42.659879  41.272881  40.862275  42.111301
y12 12.249084  37.1888535  43.004743  41.059122  40.539846  41.696201
y13 12.383508  37.4455081  43.257888  41.389514  40.879536  42.007610
y14 10.819606  39.7084788  45.486726  43.010703  42.417334  43.554983
y15 11.252344  43.8743948  49.974808  47.054522  46.399978  47.535918
           y6         y7         y8         y9        y10        y11
x1   -6.450443  -5.524963  -7.362861  -9.271154 -11.032171 -11.514051
x2   -8.096731  -6.999046  -8.820897 -10.532909 -12.254987 -12.725811
x3   -9.917236  -8.813655  -9.983885 -10.488145 -11.627258 -11.854494
x4  -11.678756 -10.414383 -11.741467 -12.480939 -13.691806 -13.924747
x5  -11.200994 -10.032854 -10.847360 -10.915772 -11.716111 -11.802406
x6   -6.149148  -5.443966  -6.673340  -7.278202  -8.127052  -8.405889
x7   -5.148129  -4.284292  -5.403966  -5.879850  -6.724825  -6.953325
x8   -3.671180  -3.157246  -4.404413  -4.983828  -5.832135  -6.109661
x9    3.720454   3.973223   1.526773  -0.574022  -2.154916  -2.785809
x10  12.450520  12.599255   8.927624   4.902701   2.165104   1.213464
x11  22.659683  22.593481  17.400241  11.009573   7.121438   5.672335
x12  30.950650  30.788156  24.237840  16.164192  11.312697   9.616004
x13  35.862800  35.554099  28.244119  18.920773  13.604374  11.681982
x14  34.564845  34.504788  26.461687  16.510565  10.997552   8.852129
x15  37.545775  37.655796  30.366708  20.930523  15.234590  13.626700
y1   44.453039  41.487473  39.616150  37.351345  36.986470  36.518491
y2   59.628971  56.829022  52.421219  46.438187  43.798824  42.659879
y3   66.695015  64.088714  57.180575  47.726307  43.180366  41.272881
y4   68.860883  66.275936  58.700087  48.214435  43.074365  40.862275
y5   71.577617  69.148150  61.254430  50.028360  44.422018  42.111301
y6   67.777648  65.832453  59.078338  48.729556  43.309477  41.332252
y7   65.832453  64.486695  58.269054  48.225747  42.598912  40.853590
y8   59.078338  58.269054  54.537902  46.882249  41.988479  41.039640
y9   48.729556  48.225747  46.882249  42.458131  38.984080  38.857438
y10  43.309477  42.598912  41.988479  38.984080  36.754186  36.735282
y11  41.332252  40.853590  41.039640  38.857438  36.735282  37.078818
y12  40.908117  40.424649  40.818382  38.981892  37.059816  37.450341
y13  41.058067  40.423200  40.602658  38.635007  36.724229  37.056544
y14  42.609282  41.933829  42.617186  40.858914  38.864622  39.346069
y15  46.222882  45.250040  45.556580  43.564487  41.681261  42.034495
          y12         y13         y14         y15
x1  -12.027375 -11.9586417 -13.0346709 -14.4877828
```

```
x2  -13.196448 -13.1639068 -14.3574887 -15.9105167
x3  -12.053758 -12.0580673 -13.0287191 -14.4353358
x4  -14.285690 -14.2714346 -15.5274605 -17.1877480
x5  -12.013350 -12.0396446 -13.1101356 -14.5259148
x6   -8.519434  -8.4589204  -9.1316556 -10.0086140
x7   -6.942059  -6.9554380  -7.4821273  -8.2596068
x8   -6.134089  -6.0713291  -6.5141234  -7.2011756
x9   -2.930950  -2.8095395  -3.2789631  -3.6257700
x10   0.823888   0.9799949   0.5057351   0.2831722
x11   5.068548   5.2699428   4.8791379   4.9991450
x12   8.825656   9.0764410   8.5195978   8.9952438
x13  10.611875  10.8924133  10.0948936  10.5729479
x14   7.609433   7.8522967   6.3723019   6.5919463
x15  12.249084  12.3835081  10.8196062  11.2523443
y1   37.188854  37.4455081  39.7084788  43.8743948
y2   43.004743  43.2578876  45.4867265  49.9748083
y3   41.059122  41.3895140  43.0107028  47.0545218
y4   40.539846  40.8795357  42.4173341  46.3999782
y5   41.696201  42.0076104  43.5549831  47.5359181
y6   40.908117  41.0580667  42.6092823  46.2228822
y7   40.424649  40.4231998  41.9338291  45.2500396
y8   40.818382  40.6026578  42.6171860  45.5565803
y9   38.981892  38.6350074  40.8589137  43.5644865
y10  37.059816  36.7242291  38.8646223  41.6812612
y11  37.450341  37.0565438  39.3460694  42.0344954
y12  37.930670  37.5136590  39.8968150  42.7043660
y13  37.513659  37.1585761  39.5226960  42.3681159
y14  39.896815  39.5226960  42.3880823  45.3576371
y15  42.704366  42.3681159  45.3576371  48.8824460
```

```
summary(lm(y15 ~ y2 + y6 + y5 + x6 + x5, data = PL1))
```

```
Call:
lm(formula = y15 ~ y2 + y6 + y5 + x6 + x5, data = PL1)

Residuals:
     23      24      25      26      27      28      29      30      31      32
-0.7107 -1.0186  0.6163  1.4168  0.3217  0.5271 -0.5677 -1.2789  1.2641 -0.7737
     33
 0.2036
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -16.8100     6.8275  -2.462  0.05708 .
y2            1.2785     0.1937   6.599  0.00120 **
y6            3.8458     0.6110   6.294  0.00149 **
y5           -4.1140     0.6104  -6.740  0.00109 **
x6            1.4703     0.8134   1.808  0.13048
x5           -1.1467     0.8165  -1.404  0.21919
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.304 on 5 degrees of freedom
Multiple R-squared:  0.9826,    Adjusted R-squared:  0.9652
F-statistic: 56.46 on 5 and 5 DF,  p-value: 0.000213
```

Multiple regressions: Step by step

Model selection

- Regression coefficients can be found by using the lm() function
- That is the b_i values in the regression equation $y = a + b_1x_1 + b_2x_2 + ... + b_nx_n$
- But how should we go about selecting a model that predicts the y
- Principle followed so far: predicting the root of the tongue from the top of the tongue
- This involves models that use some combination of the x and y locations of pellets $2, 3, 4, 5$ and 6
- Using five pellets to predict the root of the tongue gives us 10 variables That is a whopping 2^{10} regression models
- Why?
- The balance hangs between 1. high within-dataset accuracy on the one hand, and 2. high predictive accuracy for new data on the other
- More the number of parameters, more our ability to account for all the variation that is present in a set of data; that being said
- "Over-fitted" models or ones with too many predictors/parameters suffer from the problem of breaking down when new data are presented
- We want to find a model with few parameters and get as good a fit as possible with a minimum of predictive variables.

- For this, the step() function in R is used, which in turn uses the Akaike Information Criterion.
- This is a log-likelihood measure of model fitness that adds a penalty for each new parameter.
- Adding parameters incurs penalties, and so the predictions have to improve not by tiny increments but by large values for a new model with added parameters to be acceptable.
- Log-likelihood is similar to least-squares in terms of selecting a model that is the best fit.
- Recall, that the arithmetic average, i.e., the least-squares estimate of central tendency minimizes the squared deviations.
- A log likelihood estimate maximizes the likelihood of the model; is similar to least squares in that regard

Likelihood of a Model

- L(M) the likelihood of model M, has two terms
    1. Model fitness and the
    2. Model size, where nm is the number of coefficients in the regression equation
- AIC = -2logL(M) + 2nm
- step() is used to select a model to predict the y location of pellet 15 in the PL1chain data
- Let's say, step 1, the initial model has only one parameter - the intercept value
- This is specified with y15~1.
- The second argument, is the largest model we would like to consider; i.e., has the xy locations for pellets 2,3,4,5,and 6.

```
summary(y.step <- step(lm(y15 ~ 1,data=PL1),y15~ x2+y2 + x3+y3+ x4+y4 + x5+y5+ x6+y6))
```

```
Start:  AIC=43.74
y15 ~ 1

       Df Sum of Sq    RSS    AIC
+ y2    1    400.03  88.79 26.972
+ y3    1    323.32 165.50 33.822
+ y6    1    315.23 173.59 34.347
+ y4    1    301.93 186.90 35.159
+ y5    1    296.26 192.56 35.488
+ x4    1    121.77 367.06 42.584
+ x2    1    114.65 374.18 42.795
```

```
+ x5    1    105.75 383.08 43.054
+ x3    1     93.55 395.28 43.399
<none>              488.82 43.735
+ x6    1     56.78 432.05 44.377

Step:  AIC=26.97
y15 ~ y2

       Df Sum of Sq    RSS    AIC
+ y3    1     20.84  67.95 26.030
+ y4    1     15.87  72.92 26.806
<none>               88.79 26.972
+ y5    1     10.68  78.11 27.562
+ x2    1      2.83  85.96 28.616
+ y6    1      2.10  86.69 28.709
+ x4    1      1.78  87.01 28.750
+ x3    1      0.93  87.86 28.857
+ x6    1      0.61  88.18 28.896
+ x5    1      0.50  88.29 28.910
- y2    1    400.03 488.82 43.735

Step:  AIC=26.03
y15 ~ y2 + y3

       Df Sum of Sq     RSS    AIC
+ y6    1    32.429  35.525 20.896
+ y5    1    11.557  56.397 25.980
<none>               67.954 26.030
- y3    1    20.836  88.790 26.972
+ x2    1     3.243  64.711 27.492
+ y4    1     3.049  64.905 27.525
+ x5    1     2.612  65.342 27.599
+ x4    1     2.292  65.662 27.653
+ x3    1     2.175  65.779 27.672
+ x6    1     1.706  66.248 27.751
- y2    1    97.548 165.502 33.822

Step:  AIC=20.9
y15 ~ y2 + y3 + y6

       Df Sum of Sq     RSS    AIC
+ y5    1    13.104  22.421 17.833
+ y4    1    11.615  23.910 18.541
```

```
<none>               35.525 20.896
+ x4    1     3.829  31.696 21.641
+ x5    1     3.688  31.837 21.690
+ x3    1     3.323  32.202 21.815
+ x6    1     2.795  32.730 21.995
+ x2    1     2.636  32.890 22.048
- y6    1    32.429  67.954 26.030
- y3    1    51.163  86.689 28.709
- y2    1   129.423 164.949 35.785

Step:  AIC=17.83
y15 ~ y2 + y3 + y6 + y5

       Df Sum of Sq    RSS    AIC
+ x6    1    11.437 10.984 11.984
+ x3    1    11.097 11.324 12.320
+ x4    1    10.624 11.797 12.770
+ x5    1     9.989 12.432 13.346
+ x2    1     9.918 12.503 13.409
- y3    1     2.215 24.636 16.869
<none>              22.421 17.833
+ y4    1     0.435 21.987 19.618
- y5    1    13.104 35.525 20.896
- y6    1    33.976 56.397 25.980
- y2    1    50.218 72.639 28.764

Step:  AIC=11.98
y15 ~ y2 + y3 + y6 + y5 + x6

       Df Sum of Sq    RSS    AIC
- y3    1     0.879 11.863 10.831
+ x5    1     2.514  8.470 11.126
<none>              10.984 11.984
+ y4    1     1.222  9.762 12.687
+ x4    1     1.114  9.870 12.807
+ x2    1     0.598 10.386 13.368
+ x3    1     0.004 10.980 13.980
- x6    1    11.437 22.421 17.833
- y5    1    21.746 32.730 21.994
- y6    1    44.623 55.607 27.825
- y2    1    58.529 69.513 30.280

Step:  AIC=10.83
```

```
y15 ~ y2 + y6 + y5 + x6

       Df Sum of Sq     RSS    AIC
+ x5    1     3.355   8.507  9.173
+ y4    1     2.059   9.804 10.733
<none>                11.863 10.830
+ x4    1     1.893   9.970 10.918
+ x2    1     1.192  10.670 11.665
+ y3    1     0.879  10.984 11.984
+ x3    1     0.152  11.711 12.689
- x6    1    12.773  24.636 16.869
- y6    1    65.692  77.555 29.484
- y5    1    74.744  86.607 30.698
- y2    1   126.711 138.573 35.869

Step:  AIC=9.17
y15 ~ y2 + y6 + y5 + x6 + x5

       Df Sum of Sq    RSS    AIC
<none>                8.507  9.173
+ x3    1     1.323  7.184  9.314
+ y4    1     0.847  7.661 10.020
+ x4    1     0.332  8.175 10.735
- x5    1     3.355 11.863 10.830
+ x2    1     0.098  8.409 11.046
+ y3    1     0.037  8.470 11.126
- x6    1     5.559 14.066 12.705
- y6    1    67.409 75.916 31.249
- y2    1    74.084 82.591 32.176
- y5    1    77.288 85.795 32.595



Call:
lm(formula = y15 ~ y2 + y6 + y5 + x6 + x5, data = PL1)

Residuals:
     23      24      25      26      27      28      29      30      31      32
-0.7107 -1.0186  0.6163  1.4168  0.3217  0.5271 -0.5677 -1.2789  1.2641 -0.7737
     33
 0.2036

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -16.8100     6.8275  -2.462  0.05708 .
y2            1.2785     0.1937   6.599  0.00120 **
y6            3.8458     0.6110   6.294  0.00149 **
y5           -4.1140     0.6104  -6.740  0.00109 **
x6            1.4703     0.8134   1.808  0.13048
x5           -1.1467     0.8165  -1.404  0.21919
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.304 on 5 degrees of freedom
Multiple R-squared:  0.9826,    Adjusted R-squared:  0.9652
F-statistic: 56.46 on 5 and 5 DF,  p-value: 0.000213
```

References

Johnson, K. 2008. *Quantitative Methods in Linguistics*. Wiley. https://books.google.co.in/books?id=kJJpAAAAMAAJ.

Winter, B. 2020. *Statistics for Linguists: An Introduction Using r*. Routledge. https://books.google.co.in/books?id=IXhpxQEACAAJ.