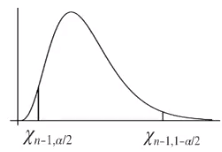
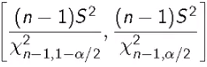
**Confidence Intervals and CI for Normal Variance**

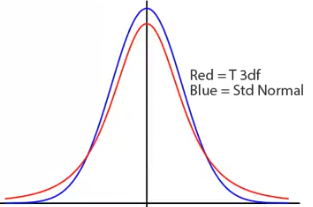
* This is based on the assumption our distribution = Gaussian
* Can use CLT to create CI’s, + to make better ones for smaller samples using **Gosset’s/Student’s t-distribution** (treating the data as if continuous)
* **General Procedure for CI** (creating a probability statement + manipulating it to generate an interval)
* Create a **pivot/**statistic whose distribution does NOT depend on the parameter of interest
* Ex: Use CLT+ take a sample mean, subtract off population mean you're interested in, + divide by standard error, that statistic *clearly* depends on the parameter of interest.
* BUT the *distribution* of that statistic, at least in the limit, *doesn't* depend on the parameter you're interested in the sample mean.
* Solve for probability that the pivot lies between bounds of the parameter of interest
* Must also know **Chi-squared distribution**
* Suppose **S2 = sample variance** from collection of **IID N(µ,σ2) data**, then \*\*\*
* i.e. n - 1, times sample variance, divided by variance, gives a **Chi-squared random variable, χ2n-1, w/ n-1 degrees of freedom**
* i.e the **normalized sample variance** follows a **Chi-squared distribution w/ n-1 dF**
* Chi-squared distribution = skewed w/ support on 0 to Inf, its mean = its dF, + its variance = 2\*dF:
* **E[χ2dF] = dF Var(χ2dF) = 2dF**
* Recall: sample variance = unbiased estimator (why we divide by n, not n-1)
* 
* Note that if  is the α quantile of the Chi-squared distribution, then**:**
* 

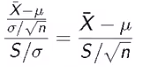
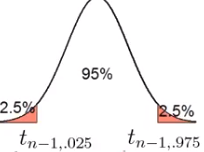
Chi-square density: 

* i.e. Probability our chi-squared random variable  is between the 2 quantiles defined by α must be == 1 – α
* Here,  = our pivot
* Then solve for parameter of interest: 
* i.e. we have a 1 – α probability that the random interval given contains our variance
* So that  == a **100(1-α)% CI for σ2**
* α could be .05, so we wind up w/ a 95% CI for our parameter of interest, σ2
* **In a CI, the \*\*\*interval is random, and the parameter is fixed**\*\*\*
* When we collect data + form the CI, it either contains the parameter or not (no probability in this statement, its either 1 or 0)
* “Intro Stat” interpretation of a CI = a procedure that if you were to repeatedly do the experiment + form CI’s, 95% of the CI’s would contain the parameter of interest
* Much weaker interpretation = you get 2 numbers that = an interval estimate of the parameter to estimate, but said interval estimate incorporates uncertainty.
* Notes about the interval from above (**100(1-α)% CI for σ2**)
* \*\*\*This CI relies *heavily* onassumed normality + is not too robust to departures from normality\*\*\*
* We *could* get the CI in other ways, if not normal, such as **bootstrapping**
* Square-rooting the endpoints = CI for **σ** (SD) = 
* Turns out that **(n-1)S2 ~ Gamma{(n-1)/2, 2σ2}** == “follows a gamma distribution w/ **shape** = (n-1)/2 and **scale** = 2σ2”
* Therefore, this can be used to plot the likelihood function for σ2
* Bit difficult since underlying data is Gaussian w/ 2 parameters µ, σ2, so our likelihood is a **bivariate function** (µ on 1 axis, σ2 on the other, likelihood as the height)
* Could create a **marginal likelihood** for σ2
* \*\*\*If we *don’t* divide by σ2, then (n-1)S2 still contains its units (dividing by σ2 makes it unitless, which is a requirement of the Chi-squared distribution, which is unitless\*\*\*)
* If we *don’t* divide by σ2, we end up w/ a Gamma distribution which is indexed by 2 parameters (**shape, scale**) as seen above
* So, we have data + a single number, (n-1)S2, + if we’re willing to assume the DP’s that make up this number are Gaussian, we can take the Gamma density, plugin the data, view it as a function of the parameters, + plot a likelihood function
* Ex: Study of 513 organo-lead manufacturing workers reported an average TBV of 1150.315 cm3 w/ **σ =** 105.997. Assuming normality of the underlying measurements, calculate a CI for population variation in TBV
* **## Create CI for variance**
* **x\_bar <- 1150.315**
* **s <- 105.997**
* **sample\_var <- s\*\*2**
* **n <- 513**
* **alpha <- .05**
* **## grab quantiles**
* **qtiles <- qchisq(p = c(alpha/2,1-alpha/2), df = n-1)**
* **## rev() = reverse elements due to order of returned qtiles**
* **(pop\_var\_CI <- rev((n-1)\*sample\_var/qtiles))**
* **[1] 9976.751 12749.451**
* **# interval for sd**
* **(pop\_sd\_ci <- sqrt(pop\_var\_CI))**
* **[1] 99.88369 112.91347**
* This interval (100,113) is created in a way that if assumptions of the interval are correct (data are IID Normal w/ fixed mean **µ** and fixed variance **σ2**), then if we repeat this procedure over and over and over, 95% of intervals obtained would be intervals that contain the true parameter we’re trying to estimate (variance or SD)
* Can plot likelihood:

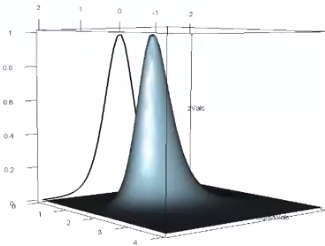
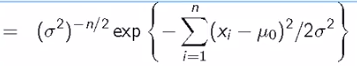
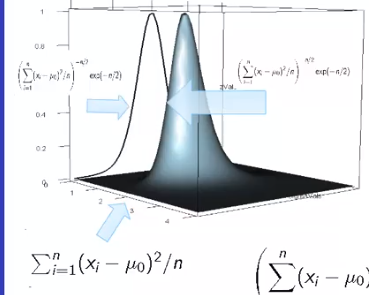
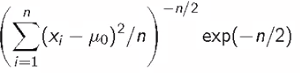
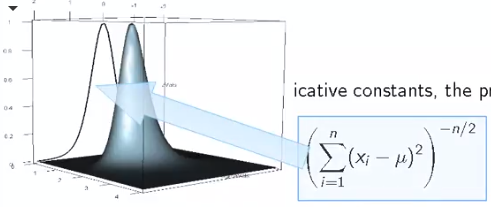
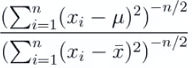
**Student's t Distribution and CI for Normal Means**

* Invented by Williams Gosset (published under pseudonym “Student”
* Has thicker tails than Gaussian, is indexed by **degrees of freedom** + becomes more like the Standard Noemal as dF grows

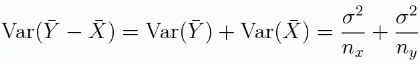
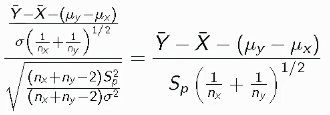
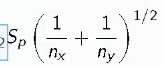
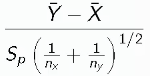
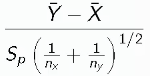
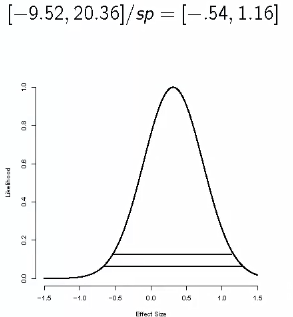


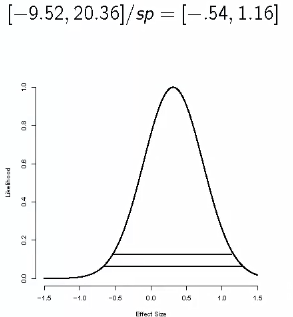
* Obtained via  where Z + χ2 = independent standard Normals + independent Chi-squared distributions, respectively
* Suppose (X1, …, Xn) are IID N(**µ**,**σ2**) then:
* 1) t-statistic =  is standard normal
* b/c linear combinations of normal random variables are themselve normal (i.e. Xbar = normal)
* b/c they’re IID, we know exactly what **SD of Xbar** is (**sigma / sqrt(n)**) + we know its mean = **µ**
* So, when we shift + scale our *nonstandard* normal by **µ** + divide by its SD, we get a standard normal
* 2)  = square root of a Chi-squared divided by its dF
* Therefore, take the 2 values above 🡺  (take a Chi-squared + divide by its dF) will follow Gosset’s t-distribution w/ n-1 dF
* Haven’t *exactly* shown that Xbar and S are independent (they’re from the same data) 🡺 assume for now
* Previously, in constructing CI’s, **Xbar -** **µ** divided by σ/n is a nice pivotal statistic that’s useful for generating CI’s (+ also for doing hypothesis tests.
* All we've done here is replaced σ w/ S 🡺 \*\*\*can take the *unknown population variance* + replace it w/ the *known sample variance* + weget a statistic whose distribution we know*\*\*\**
* This statistic, , also limits to a Standard Normal as X goes to infinity
* Ex: Create CI for the mean
* **t statistic** is a **pivot**, so (*under the assumption of normality of the data*), does NOT depend on parameter of interest (**µ**) + therefore can be used to create a CI for parameter of interest (**µ**)
* \*\*\*Let  =  quantile of the t-distribution w/ **dF** degrees of freedom\*\*\*
* Then, 1 – α =  = probability our t-statistic/random variable lies w/in our CI is true
* Rearrange terms to get CI for **µ** = 
* So, our \*\*\*interval =  == **estimate +/- quantile \* std. error**\*\*\*
* Notes about t-interval
* t-interval *technically* assumes data are IID Normal, though it is robust to this assumption
* Works well whenever distribution of the data is *roughly*  symmetric + mound-shaped
* Paired observations (ex: patients before + after treatment) are often analyzed using a t-interval by taking differences (which tend to be much more Gaussian)
* For large dF, t-quantiles becomes the *same as Standard Normal quantiles*
* Therefore, this interval converges to the same interval as from the CLT
* For skewed distributions, t-interval assumptions are violated
* Also, for skewed distributions, doesn’t make sense to center interval @ the mean
* In this case, consider taking **logs** or using a different summary stat (i.e. median)
* For *highly discrete data* (like binary), other intervals are available
* Can use t-distribution to create a likelihood for a single parameter (which itself is function of 2 parameters, **µ and σ2**)
* If X is N(µ,σ2) + χ2 is a Chi-squared random variable w/ dF degrees of freedom, then  = **non-central t random variable w/ non-centrality parameter = µ/σ**
* µ/σ = the mean in SD units 🡺 unit-free quantity called the **effect size**
* Useful for creating **likelihood effect size**
*  == did not subtract off µ, so X/σ still has a mean (= µ/σ), so we have NOT taken a Standard Normal + divided by the square root of an indepentn Chi-squared divided by its dF, but instead took a NON-Standard Normal + divided by the square root of an indepentn Chi-squared divided by its dF
* So, it *can’t* work out to be a t-random variable b/c we haven’t satisfited the definision of a t-random vairble, so that’s why it’s a **non-central t random variable**
* But when µ = 0, we DO have a normal t random variable
* Non-central t random variable has a dF, but also has a 2nd parameter, the non-centrality parameter µ/σ
* For context, Xbar is Normal(µ, σ2/n) and (n-1)S2/σ2 = chi-squared w/ n-1 dF
* Then,  = a non-central t w/ non-centrality parameter = 
* This can be used to create a 1D likelihood for **effect size =** µ/σ w/out any further tricks

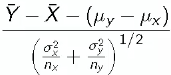
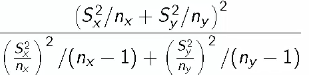
**Profile Likelihoods**

* **Profiling** = preferred method for univariate likelihoods from multivariate likelihoods
* We’re going to look at the bivariant Normal distribution which has 2 parameters µ, σ2, + we’re going to obtain a likelihood for µ alone (can also obtain a likelihood for σ2 alone)
* **Profile Likelihood (PL)** gets name b/c results = shadow we’d get if we shined a light on the 2D likelihood for µ and σ2
* PL is a bivariant surface w/ µ and σ2 on the axis + height = likelihood
* Ex: for shining light along σ axis
* Shadow = likelihood placed on the plane defined by the µ direction
* In other words, we want to get the function obtained onto the wall where the shadow occurs
* PL for parameter value µ0 = obtained by maximizing **joint likelihood** for σ w/ µ fixed @ µ0
* *Finding value of the curve in the shadow @ µ0*
* Light will go through all values above the likelihood + get stopped anywhere on the likelihood + up until the maximum value.
* This process is then repeated for many values of µ0
* Ex: **joint likelihood w/** µ fixed @ µ0  == the Gaussian density , and since we have independent data we take a product out from == 
* Collect all terms == 
* Having µ0 fixed, MLE for σ2 🡺 **log** the likelihood + take derivatives + solve for σ2 🡺 (generalization of the variance)
* Remember to fix σ2 as the parameter we’re deriving w/ respect to, not σ
* So, if we fix µ @ a particular value, our MLE for σ2 = sample variance but instead of subtracting deviations from the sample mean, we’re subtracting deviations from our fixed value for σ2
* == peak of our likelihood == the point where the light switches from *not* going through the likelihood to the point right above it where the light passes over the likelihood, which is the point that gets shadowed onto the wall @ µ0
* 
* Plugging this peak back into likelihood again, we get 
* This e(-n/2) is irrelevant since it doesn’t involve µ0
* All the above is done for 1 value of µ0, so if we do it for every µ0 then we’d get a function that = our profile likelihood 🡺 
* **profile likelihood** = a summation of x minus µ squared raised to -n/2 over all n
* **this function is clearly maximized at** 
* In general, a nice property of the profile likelihood is the **maximizer of the profile likelihood/maximum profile likelihood estimand is also the MLE for the parameters**
* In this case  is the same as the MLE for parameter µ for the complete likelihood
* If we wanted to divide this by its peak value, divide it by the same thing w/ x\_bar plugged in for µ to normalize the function to tap out @ a value of 1 🡺 
* We treat profile likelihood as if it was a Standard Univariate Likelihood forµ (higher values = better supported, peak = where MLE occurs, horizontal lines to get likelihood-based intervals for µ)
* **NOTE:** The specific technique of the t-CI is a very robust interval, as long as the data look roughly mount-shaped
* the t-CI + the Standard Normal CI look the same, except w/ the t-quantile replaced by a standard normal quantile
* t-CI limits to the standard normal CI, so just always do a t-CI (don’t worry about which sample size to switch between) 🡺 Just never do a standard normal CI + then you don't even have to worry about it
* ***If your sample size is big enough, the t-quantile looks like a normal quantile anyway***

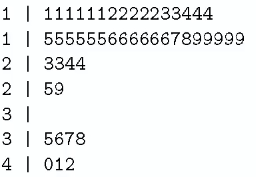
**t-Confidence Intervals**

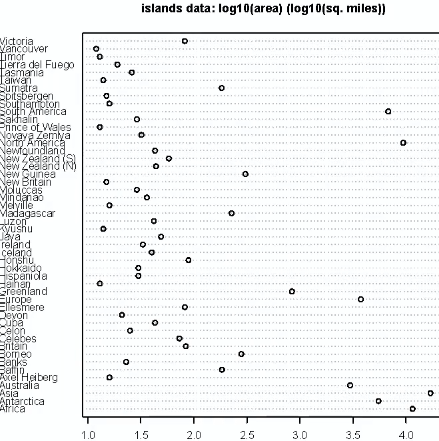
* **Independent groups t-CI’s**
* Want to compare 2 mean BP between 2 groups in a randomized trial (received Tx vs. placebo)
* Cannot use **paired** t-test b/c groups = independent + may have different sample sizes
* Let both **X1,…,Xnx+ Y1,…Yny** be **IID N(μx, δ2) + N(μy, δ2)** respectively, **Xbar, Ybar, Sx, Sy** be the means **+ SD’s**
* Using the fact that **\*\*\*linear combinations of normals are again normal\*\*\*,** we know **Ybar-Xbar** (*what we want to estimate w/ some uncertainty)*isalso normal w/ **mean = μx + μy** and **variance = δ2(1/nx+ 1/ny)**
* 
* **The \*\*\*Pooled variance estimator\*\*\* =** S2p = is used as a good estimator (*an MLE*) of **δ2**
* Note:  == i.e. complements
* So, S2p = a weighted average of the 2 group variances (for group X and group Y)
* If we have the same sample sizes (nx= ny), then ends up being ½, in which case the pooled variance estimate ends up being the arithmetic averages of the 2 variances
* 
* But if X contains much more data, nx – 1 >>> ny – 1, so the numerator in π above gets much larger, so the weight on S2xover S2y is much larger
* Then, the weighted averages takes whichever group has more measurements + weights the variance from that group more heavily (which is what we want)
* i.e. more data = good estimator places more weight = variance estimated better
* **Pooled variance estimator** = mixture of group variances w/ greater weights on variances from groups w/ larger sample sizes
* If sample sizes are =, Pooled variance estimator = normal average of group variance
* **\*\*\*Pooled variance estimator is unbiased\*\*\***
* Take expected value of the PVE + use the fact that the individual group variance estimators are unbiased 🡺  
* Pooled variance estimator is also independent of Ybar-Xbar since Sx is independent of Xbar and Sy is independent of Ybar
* Recall: **\*\*\*sum of 2 independent Chi-squared random variables is Chi-squared w/ dF = sum of the dF in the summands**
* Therefore, 
* 1st term is chi-squared w/ nx-1 dF, and 2nd = chi-squared w/ ny-1=1 dF 🡺 
* From the fact in bold above + b/c our random variables are independent 
* Would like to create a t-CI, which needs a standard normal divided by the square root of an independent chi-squared which itself is divided by its dF
* Need some function of the PVE to be Chi-squared (?)
* Now construct t 🡺 
*  == standardizing ( = (value – mean) / std. err) == results in a Standard Normal
* *Original data are Gaussian == sample means are Gaussian == different in sample means is Gaussian 🡺 take a Gaussian, subtract mean + divide by SD, we get a Standard Normal*
* Divide Standard Normal it by a chi-squared divided by its dF 🡺  🡺 must result in a t random variable w/ (nx + ny – 2) dF
* Collecting terms + working through the math gives the RHS above == observed difference in means minus population difference in means divided by Std. Err (w/ δ replaced w/ out data estimate of δ, Sp)
* Therefore, this t statistic follows Gosset’s t distribution w/ nx + ny – 2 dF
* Like before, we took our statistic, replaced unknown SD w/ its estimate, + what would be a Gaussian random variable turned into a t random variable in form == (estimator–true value) / SE
* Therefore, a (1-α)\*100 CI for **μx - μy is** 
* **This == \*\*\*estimate (Ybar-Xbar) +/- appropriate quantile from appropriate distribution (t-quantile form the t-distribution =** **) \* by SE** \*\*\*
* This interval assumes constant variance across groups (there are tests to check for this = **F-test**, but these are typically bad)
* If doubt in this, assume different variance per group
* Textbooks typically suggest testing equality of the variances + if equal, do one CI + if unequal, do another CI 🡺 bad strategy
* Instead, look at the data (graphs) + make assessments as to whether or not the variances are equal/unequal + use that to decide.
* If you *must* estimate ratio of the variances in the groups, try bootstrapping (unless sample sizes are very small)
* Safe + conservative thing = just always assume the variances are unequa
* How to get a likelihood for **μy – μx**
* Getting a likelihood for **μy – μx** divided by δ (which is still a single parameter) is very easy.
* The reason is that  (Ybar - Xbar divided by its standard error) follows a **non-centrality distribution**, + the **non-centrality parameter** depends on **μy – μx** over δ + something about the n’s that we know 🡺 **non-centrality parameter** == 
* We can use the statistic  ro create a likelihood for **μy – μx / δ (= a standardized measure of the change (difference) in group means, relative to the inter-group SD/in SD units)**
* “in SD units/relative to inter-group SD” = useful for calibrating difference in means across studies
* Ex: Constructing t-CI with a PVE
* Comparing Sys BP for 8 oral contraceptive users vs. 21 controls
* Xbaroc = 132.86 mmHg, w/ soc = 15.34 mmHg
* Xbarc = 124.44 mmHg, w/ sc = 18.23 mmHg
* **PVE s2p** **=** 
* Remember to pool variances, NOT standard deviations
* W/out squaring, we’d get a pooled SD instead of pooled variance + if we treated the result as if it was a variance, it’s units would be on the wrong order of magnitude
* 1 way to check 🡺 it's an average of 15.34^2 and 18.23^2, so it must be between those 2 #’s
* The 15.34^22 and 18.23^22. So, so then when you square root it, that number has to be between 15.34 and 18.23. So if that hasn't happened, you've really screwed up. Okay, so we've got our pooled variance, and if we square root that, we get our pooled standard deviation. And then we need the appropriate T quantiles. So we need the T quantile for 97.5 if we want a 95% confidence interval.
* For a 95% CI w/ 27 dF, want t-quantile for 1–(1-.95) / 2 = .975 **🡺 t27,.975 = (qt.975,df=27) = 2.052**
* 27 dF 🡺 comes form 🡪 **8 + 21 - 2**
* **Interval =** 
* Now, check **if the interval contains 0**, b/c if so, then apparently a reasonable estimate for the difference in BP between the 2 groups is 0 (i.e. they’re identical)
* This means there’s evidence that there is no difference/that this oral contraceptive use does not appear to be presenting evidence of an associated increase in BP
* Checking **if the interval contains 0 == equivalent to a 2-sided hypothesis test**
* For now, interpret as “ -9.5 to 20.4 is reasonable estimate, accounting in the measurements for comparing the average systolic BP between oral contraceptive users vs. controls”.
* Another thing to keep track of whenever creating these intervals == **what order you've subtracted things in** (we did contraceptive users – controls)
* Pick a rule + stick with it, such as Treated – Control
* Would just get negative of the interval if done the opposite way
* If the entire interval was > 0, we’d say oral contraceptive users had a higher estimated systolic BP than controls
* Reasonable values for **effect size** from above CI 🡺 **divide it by pooled SD** 
* [-.54,1.16] is NOT a valid interval for effect size b/c uncertainty in estiamteing sp is not accounted for yet



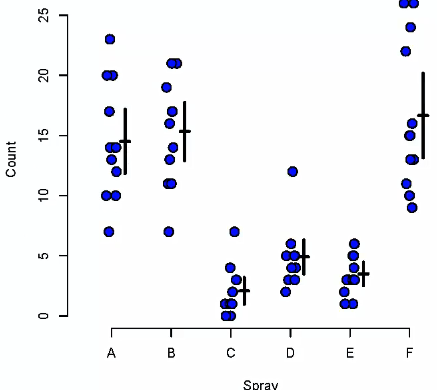
* **Unequal Group Variances**
* Under unequal variances 🡪 
* Ybar – Xbar is still normal, it’s mean is still , but its variance has changed  + we can’t factor out δ2
* Resulting statistic used to generate the CI *approximately* (NOT exactly) follows a Gosset’s t-distribution w/ dF =
* This dF value is the best possible value to make it *look like* a t-distribution
* We have the value depend on the data + the variances + have it be a **fractional dF**
* We evaluate the statistic w/ the empirical variances plugged into denominator, + then act like it’s a t-distribution w/ the dF formula above
* Comparing Sys BP for 8 oral contraceptive users vs. 21 controls
* Xbaroc = 132.86 mmHg, w/ soc = 15.34 mmHg
* Xbarc = 124.44 mmHg, w/ sc = 18.23 mmHg
* dF = 15.04, t15.04, .975= 2.13
* Interval = 

**Plotting**

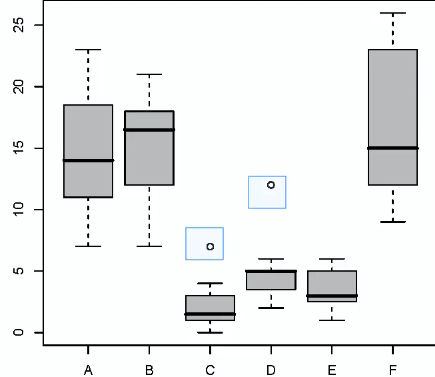
* **Histograms** = “connect” bar grapsh thos show sample estimates of the
* Discrete **= density** via frequencies of times a variable takes on specific values within a sample
* Continuous **=** **mass function** via proportions of times a variable takes on a range of values (**bins**) within a sample
* Pros: easy, useful, make sense, apply to: continuous, discrete, and *unordered* data
* Cons: use a lot of space + display little info, difficult to display several @ same time for comparison
* NOTE: Probability preferable to consider log base 10 (orders of magnitude) for some histograms where most values are small (`islands` dataset in R)
* **Stem-and-leaf plots** = useful for getting distribution info on-the-fly compared to histogram
* **Pick a digit + “break” the data on it, stacking digits to the right of it 🡪** 
*  🡺 See 6 islands w/ a log10 land area = 1.1
* Pros: display complete dataset + waste little space, 2 data sets’ S&L plots can be shown back-to-back for comparison
* **Dot chart** = display a data set w/ 1 DP per dot
* Ordering of dots + labeling of axes = displays additional helpful info
* Show complete data sets == therefore have **high density**
* May be impossible to construct/difficult to interpret for data sets w/ many DP’s

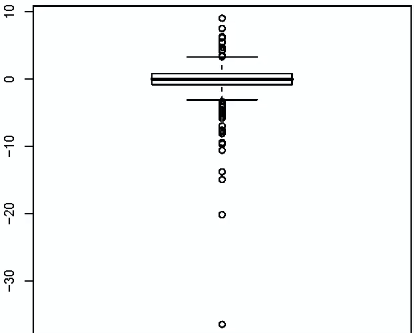
 🡪 maybe try grouping alphabetically rather than ordering alphabetically

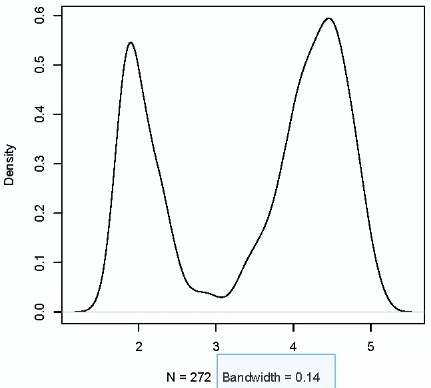
* If we have a small # of DP’s, plotting whole data set is preferable
* Can compare groups easily/display density info by group
* Can plot lines to show mean + median (horizontal) or spread, CI’s + IQR’s (vertical)
* **Jitter** DP’s to overvoid over plotting



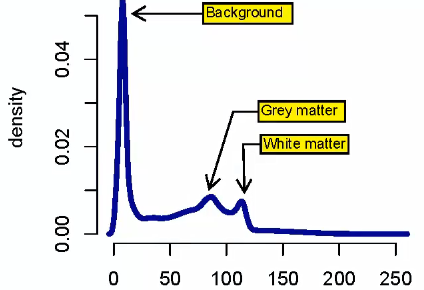
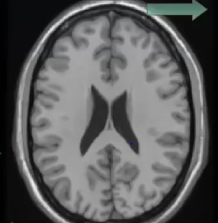
* See all DP’s, can see sprays C, D, E are different from A, B, F, we/ some means + CI’s (black crosses) for each group
* **Boxplots** = useful for same displays as dot plots but in where displaying all DP’s is not possible
* Center line = median, box edges = quartiles, whiskers extend to a constant\*IQR or max value, sometime potential outliers denoted by DP’s outside whiskers
* Skewness indicated by centerline being closer to 1 box edge

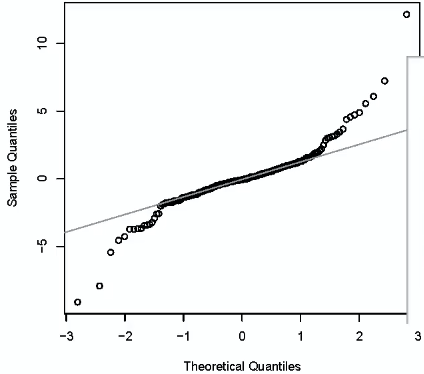
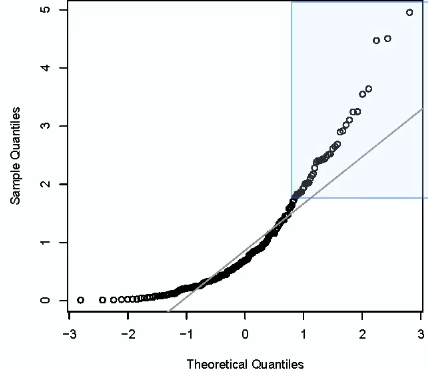
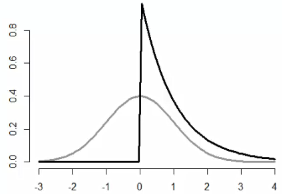
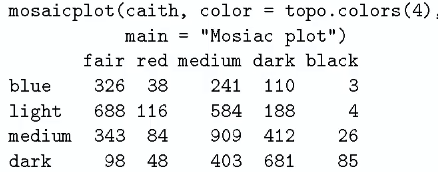
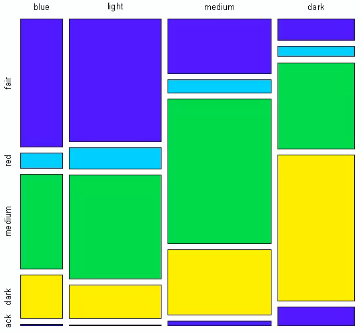


* Don’t use for small #’s of DP’s, just plot all data (dot plots)
* Try **log**ging if some boxes are too squished relative to others (can convert axis to unlogged units, though they won’t be equally space anymore)
* For many DP’s, omit outliers when plotting if we get so many we can’t see the “good” points
* Ex: 
* **Kernel Density Estimates** = more modern versions of histograms, providing density estimates for continuous data
* Observations = weighted according to a **kernel** (most cases, a Gaussian one)
* **Bandwidth** of kernel = parameter that effectively plays role of bin size for a histogram = determines how smooth/jagged density estimate will be
* Too low = too variable/jagged measure of densityδδ
* Too high = over-smooth
* Ex: Density estimate for waiting + eruption times between eruptions @ Old Faithful

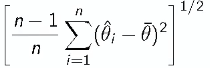
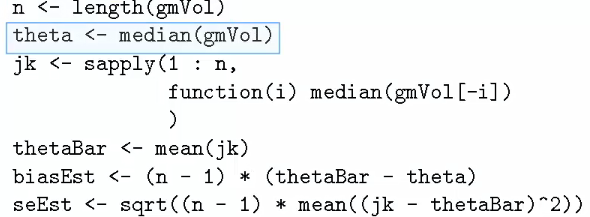
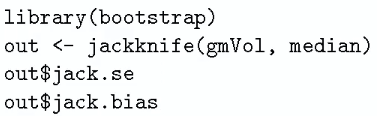


* Eruptions seem to occur in 2 time periods + we get a sense of variation around those eruption times as well
* Ex: Image slice (single axial slice of a 3D image) from high-resolution MRI of the brain
* Discarding location info, plot KDE of greyscale color intensities

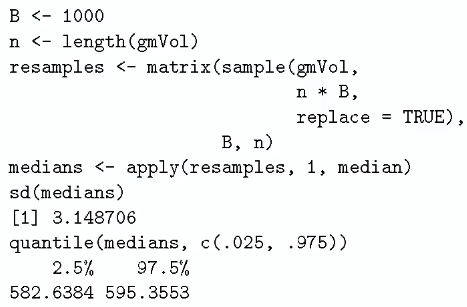
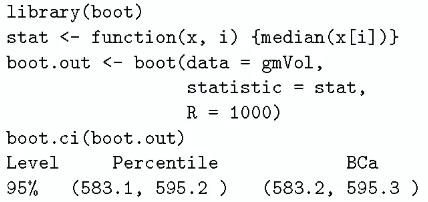


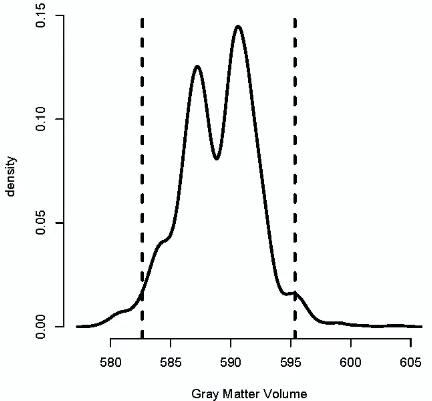
* **Quantile-Quantile/QQ Plots** = useful for comparing a data’s distribution to a theoretical distribution
* Plot **empirical** distributions **against** **theoretical** quantiles
* **\*\*\*Most useful for diagnosing normality\*\*\***
* Unlike histograms, they focus in *exactly*  on the two distributions, quantile by quantile, + tend to highlight difference much more effectively than overlaying 2 histograms
* Want to check to make sure the result is a line
* Let xp = pth quantile from a N(μ,δ2), then probability of non-standard normal X **P(X<=x) = p**, so clearly, for the standardized normal Z, **,** and therefore **xp = μ + zpδ**
* The result of this is that quantiles from a N(μ,δ2) population should be linearly related to Standard Normal quantiles
* A Normal QQ-plot plots empirical quantiles against theoretical Standard Normal quantiles
*  = At the high end, our sample quantiles are “too large”, and they are “too small” at the low end == our data is more heavily-tailed than a Standard Normal
*  = At the high end, our sample quantiles are “too large”, and they are all squished up at 0 at the low end == our data right-skewed (like a Gamma distribution)
* 
* The empirical sample quantiles are measured + contain some noise, + the Normal QQ-plot does not account for these, so maybe some uncertainty around the plot should be plotted
* **Mosaic plots** = useful for displaying **contingency table** data
* distribution of hair + eye color
*   🡺 size = count
* Gives good way to immediately look at the bivariate distribution (low count for red hair among all eye colors, fair hair seems to vary among eye color, while medium hair stays consistent)

**The Jackknife**

* **Bootstrapping** =incredibly useful, handy result in statistics that can be uses in a variety of settings, made, sort of coincided with the personal CPU revolution.
* gives us a way to avoid an awful lot of math in biostatistics
* **Jackknife = tool for estimating Std. Errors + bias of estimators**
* Jackknife : handy tool :: bootstrap : entire workshop of tools
* Key idea in both jackknife + bootstrap = **resampling** data = repeatedly creating new datasets form the original data to get quantities that’re otherwise difficult to get (variance, biases, etc.)
* have something we don’t know + we use data to get a sense of the thing
* **Jackknife** 🡺 deletes each observation + calculates an *estimate* (formulates a statistic)based on remaining n-1 observations to see how well the statistic does at estimating the held-out observation
* **i.e. like leave-out Cross-validation in ML**, but goal of jackknife = use this collection of estimates to estimate bias + Std. Error, etc.
* These 2 estimates are NOT needed for things like sample mean (already know they’re **unbiased estimates** of population means + what their standard errors are)
* Consider jackknife for univariate data:
* Let X1, …, Xn = collection of data (univariate DP’s) used to estimate parameter ϴ
* Let **ϴ^** = estimate based on full dataset, **ϴi^** = estimate after deleting observation i
* Let  🡺 average deleted observation estimates after deleting each observation
* Then, **jackknife estimate of the bias = (n-1)( - ϴ^)** == how far average delete-one-estimate is from the full dataset estimate
* n-1 = good value/factor that’s a good estimate of the right multiplier to get the bias estimate to be near the true bias
* Then, **jackknife estimate of Std. Error =** == n-1/n times sum of the squared deviations of delete-one-estimates from the average delete-one-estimate == **deviance of delete-one-estimates from the average delete-one-estimate**
* Sort of like square root of n-1 times the variance of the delete-one-estimates
* Ex: 630 measurements of gray matter volume (GMV) for workers from a lead manufacturing plant
* Median = ~589 cm2 🡺 want to estimate bias + SE of the median
*  
* Both code snippets above give **estimated bias = 0, SE = 9.94**
* \*\*\*Jackknife estimate of bias for median = *ALWAYS* 0 *when # of observations is even*
* Has been shown that jackknife = a **linear approximation** to the bootstrap
* Generally don’t use jackknife for sample quantiles like median, as it’s been shown to have some poor qualities
* Jackknife works well for smooth functions + empirical quantiles like median often don’t satisfy that
* Think of jackknife as using **pseudo observations**
* Let **PseudoObs = nϴ^ - (n-1)ϴ^­I 🡺** think of these as “whatever observation I contributes to estimate of ϴ”
* When ϴ^ = the sample mean, the PseudoObs = the data themselves
* Then, the Sample Std. Error of PseudoObs = previous jackknife Estimated Std. Error **+** the mean of PseudoObs = *bias-corrected* estimate of ϴ

**Bootstrap**

* Very useful for creating CI’s + calculating Std. Errors for difficult statistics (i.e. how to derive a CI for a median)
* useful for creating basically anything involving a statistic where you don't know the distribution
* Ex: Have a statistic that estimates some population parameter ϴ, + we don’t know the sampling distribution
* **Bootstrap Principle suggests using the distribution defined by the data to approximate its sampling distribution**
* distribution defined by the data, if not putting any constraints on it, puts **probability = 1/n** on each DP (discrete distribution)
* kind of weird distribution b/c DP’s generally have fractional amounts
* this empirical distribution = a reasonable distribution to work with
* **we look @ the sampling distribution of the median based on the *empirical* distribution from our data, where we place probability = 1/n over on each DP**
* Bootstrap principle is saying “IDK what population distribution is, so w/out that + w/out some tool like the CLT, I can't figure out what the distribution of my test statistic is”
* “So, I take my empirical distribution + figure out what the distribution of sample medians are like *from that distribution,* b/c we know it’s a distribution that's p = 1/n on each DP + I can work with that”
* Bootstrap principle can be executed in a parametric way or non-parametric way
* **Non-parametric bootstrap = empirical distribution just places p = 1/n on each DP**
* This distribution is actually not very convenient to work w/ to figure out things about it.
* 100 observations 🡪 distribution places p = 1/100 on every DP + I want to know what the distribution of medians of a 100 observations from that distribution is 🡺 hard to work with
* Bootstrap principle is instead carried out via **re-sampling Simulation**
* General procedure 🡺 taking the data set, simulating complete datasets from observed data w/ replacement
* i.e. approximately drawing IID samples from the sampling distribution (p = 1/n on each DP) of that statistic, at least as far as the data is able to approximate the true population distribution
* Then, for each re-sampled dataset, calculate the statistic (such as median), + then use all simulated statistics to define a CI or take the SD of them to get a Std. Error
* Ex: 100-sided die w/ p =1/100 for each side + the die shaped funny (not *exactly* p = 1/100 for each side)
* So we have a distribution on the #’s between 1-100 + want to know the sampling distribution of the median of ten die rolls from this 100-sided die
* Roll the die 10 times, get a sample median, record it, repeat thousands of times
* If done enough, we’d get exactly the distribution of the sample median of ten die rolls
* Now we know that if we can actually sample from the population distribution over + over, we can get the sampling distribution of a statistic.
* **\*\*\*But when confronted with *real* data, we don't know what the population distribution is\*\*\***
* What we CAN do is roll from a some die, where on every side of the die we've put the # associated w/ an observed DP, then **now we're not drawing from the population distribution but instead from the empirical distribution**
* If we had 10 DP’s + want to know the distribution of the sample median of 10 observations:
* can't draw from the population distribution, but can **draw samples** of size = 10 **from the distribution defined by the data observed** + look at what the distribution of the sample median is for *those*.
* **That is exactly what the bootstrap does**
* Bootstrap, in practice via re-sampling, basically says “we know exactly what we’d do if we actually knew the population distribution, so why don't we *still* do that + use the sample distribution + see how that works”
* Ex: 630 measurements of gray matter volume (GMV) for workers from a lead manufacturing plant w/ median = ~589 cm2 🡺 want a CI for the median of these measurements
* Sample *n* observations **with replacement** from our observed data (630) resulting in 1 simulated, complete dataset
* Take median of simulated dataset
* Repeat above 2 steps *B* times, getting *B* simulated medians
* These *B* medians = approximately draws from the sampling distribution of the median of *n* observations
* In reality = *actual* draws from empirical distribution
* Therefore we can:
* Draw their histogram
* Calculate their SD to estimate Std. Error of the median
* Take the 2.5­th + 97.5th percentiles as the CI for the median
*  
* each row = complete dataset
* Set B large enough we don't have to worry about the error in the **Monte Carlo** re-sampling (don't want # of times you rolled the die to be a factor in what you’re doing)
* Histogram of resampled medians w/ the CI:

 🡺 95% of resampled medians are between the 2 lines

* Notes on Bootstrap
* Here, it was **non-parametric** 🡺 knew nothing/makes very little assumptions about distribution
* **\*\*\*Theoretical arguments validating the bootstrap rely on large samples\*\*\***
* The “Better” percentile bootstrap CI’s correct for bias
* There’re lots of variations on bootstrap + jackknife (Intro to the Bootstrap, Efron & Tibshirani)