Modelling and Simulation with the t and F distributions

COSC/DATA 405/505

Modelling Continuous Data (Cont'd)

Distributions based on the Normal: χ^2, t and F

Random Variables Constructed from Normals

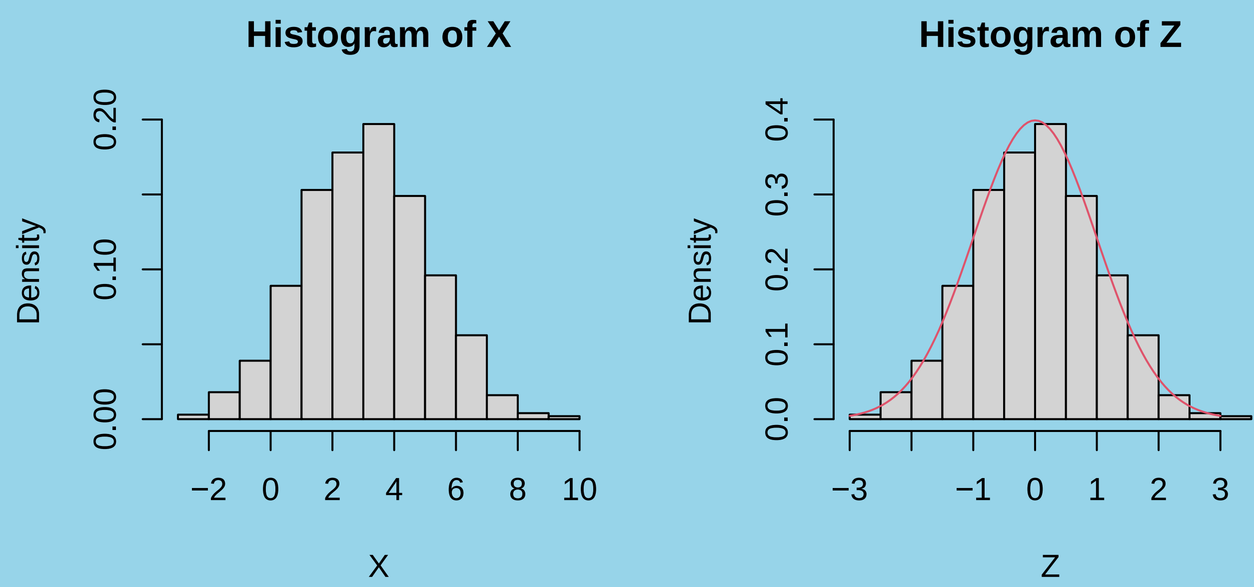
Construction starts with the standard normal random variable

- **Let Y be a normal random variable with mean μ and standard deviation σ**
- $$Z = \frac{Y - \mu}{\sigma} \quad (1)$$
  is a standard normal random variable.

Transforming Normal to Standard Normal

Check standardization by simulation:

```
X <- rnorm(1000, mean =3, sd = 2); Z <- (X-3)/2
par(mfrow=c(1, 2))
hist(X, freq=FALSE); hist(Z, freq=FALSE)
curve(dnorm(x), -3, 3, col=2, add=TRUE)
```



The distribution of `Z` is identical to that of `X`, therefore normal. N(0,1) pdf curve matches.

The χ^2 Random Variables

- **Squaring Z leads to a χ^2 random variable on 1 degree of freedom.**
- **Note that**

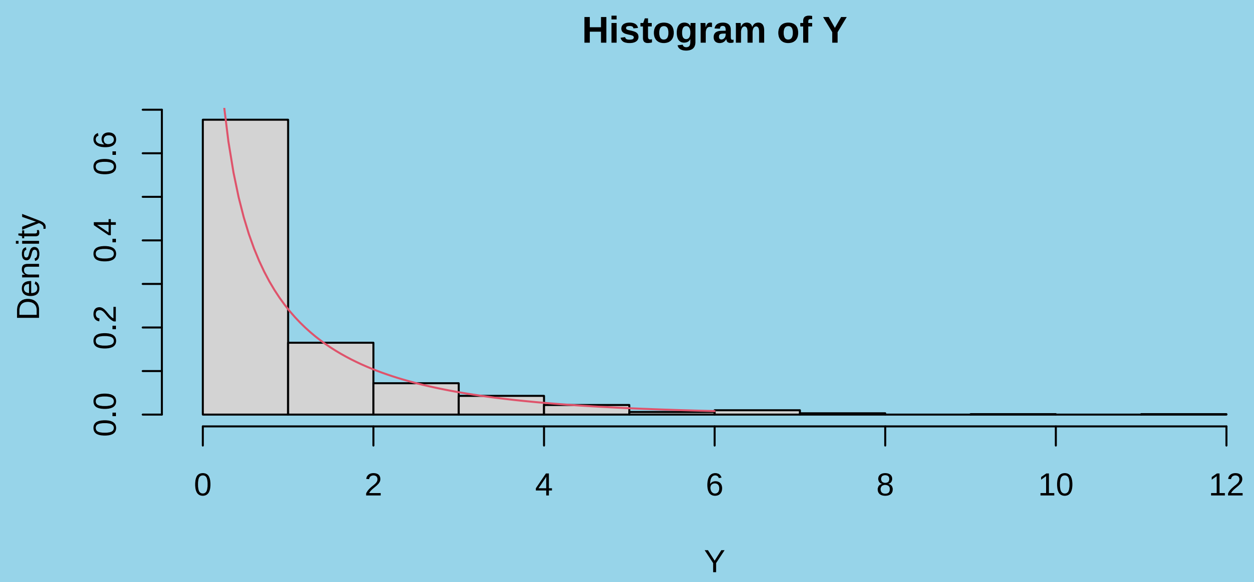
$$E[Z^2] = 1 \tag{2}$$

$\rightsquigarrow$ **a χ^2 random variable on 1 degree of freedom has expected value 1.**

On the next slide, we check that Z^2 is χ^2 by simulation, using `dchisq()`**.**

The χ^2 Random Variables

```r
Y <- Z^2
hist(Y, freq=FALSE)
curve(dchisq(x, df = 1), 0, 6, add=TRUE, col=2)
```



χ^2 random variables can be generated using `rchisq()`:

```r
rchisq(5, df = 1)

## [1] 0.5790044 3.7902630 2.0388091 0.3716699 0.2288695
```

The χ^2 Random Variables

- If $Z_1, \ldots, Z_n$ is a sequence of n independent standard normal random variables, then

$$X = \sum_{j=1}^{n} Z_j^2 \tag{3}$$

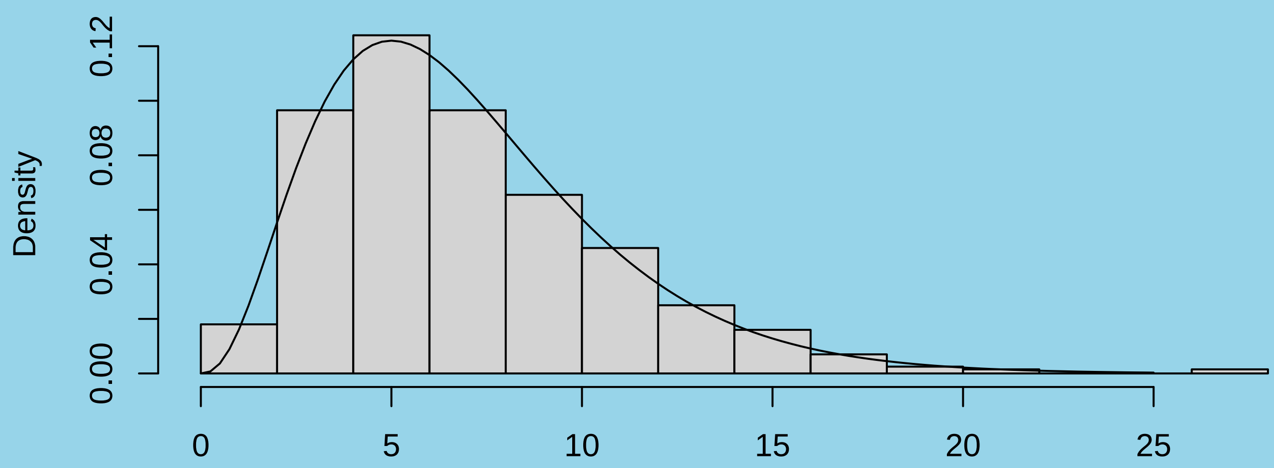
is a $\chi^2_{(n)}$ random variable on n degrees of freedom.

- $$E[\sum_{j=1}^{n} Z_j^2] = \sum_{j=1}^{n} E[Z_j^2] = n \tag{4}$$

The χ^2 Random variables

1000 simulated values of X for the case where $n = 7$

```
X <-  rchisq(1000, df = 7)
hist(X, freq = FALSE, main = " ")
curve(dchisq(x, df = 7), from = 0, to = 25, add = TRUE)
```



The F random variable

If X_1 and X_2 are independent χ^2 random variables on m and n degrees of freedom, respectively, then

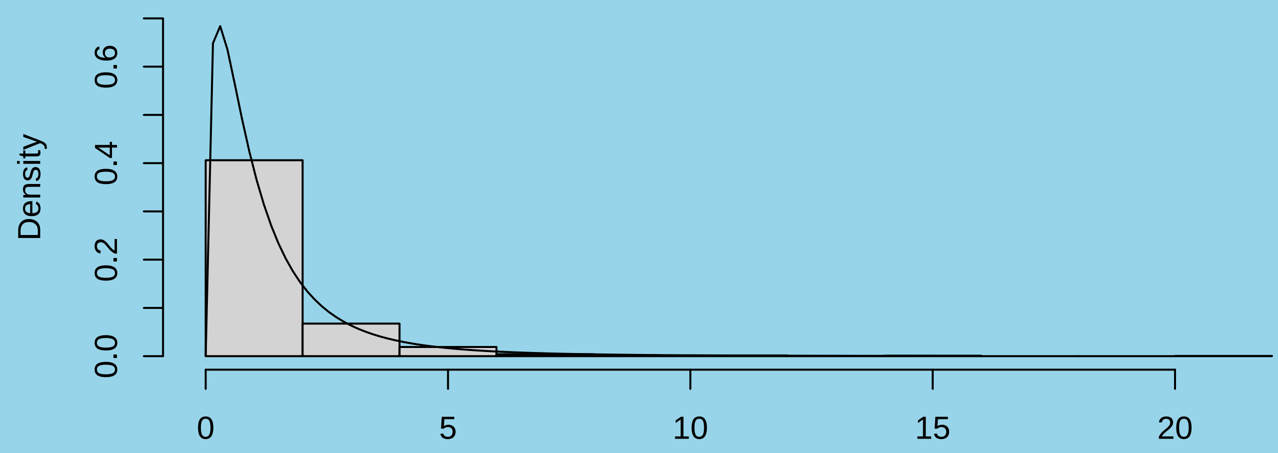
$$F = \frac{X_1/m}{X_2/n} \tag{5}$$

is an F random variable on m and n degrees of freedom. m is sometimes referred to as the numerator degrees of freedom, and n is the denominator degrees of freedom.

The F random variable

1000 simulated values of F for the case where $m = 3$ and $n = 7$, $F_{(3,7)}$.

```
F <-  rf(1000, df1 = 3, df2 = 7)
hist(F, freq = FALSE, ylim = c(0, 0.7), main = " ")
curve(df(x, df1 = 3, df2 = 7), from = 0, to = 15, add = TRUE)
```



The t random variable

Suppose Z is a standard normal random variable and suppose X is a χ^2 random variable on n degrees of freedom, then

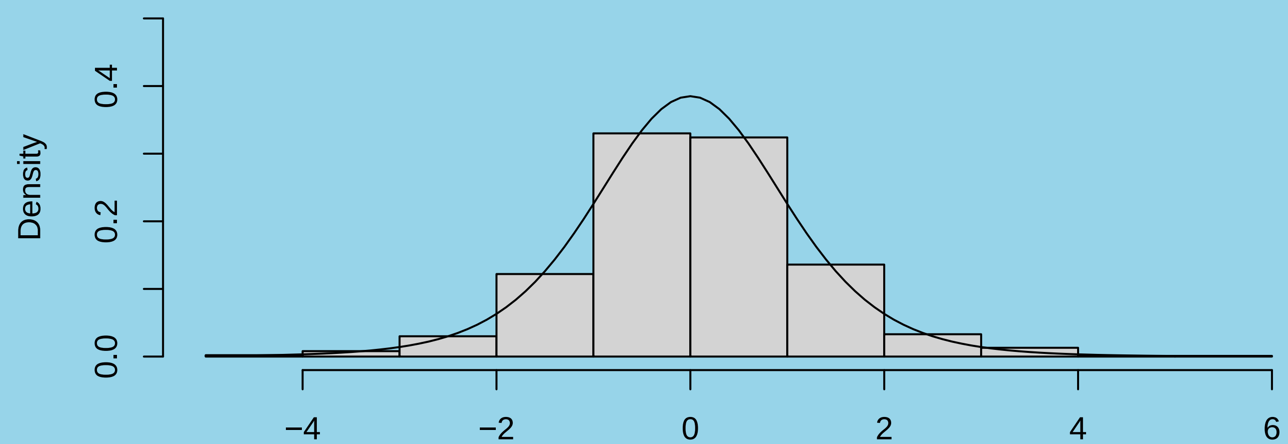
$$T = \frac{Z}{\sqrt{X/n}} \tag{6}$$

is a t random variable on n degrees of freedom, provided that Z and X are independent of each other.

The t random variable

1000 simulated values of t for the case where $n = 7$

```
T <-  rt(1000, df = 7)
hist(T, freq = FALSE, ylim = c(0, .5), main = " ")
curve(dt(x, df = 7), from = -5, to = 5, add = TRUE)
```



Studentizing yields a t random variable

- $\bar{Y}$ is normally distributed with mean μ and variance σ^2/n, if the underlying sample consists of n uncorrelated normal random variables with common mean μ and common variance σ^2.

- We will demonstrate empirically that $\bar{Y}$ and S_Y^2 are independent

- $(n-1)S_Y^2/\sigma^2$ is a $\chi^2_{(n-1)}$ random variable

- We will now show by simulation that

$$\frac{\bar{Y}-\mu}{S_Y/\sqrt{n}} \tag{7}$$

is a t random variable on $n-1$ degrees of freedom.

Simulation of Distribution of t Statistic

Let us consider a random samples of $n = 20$ normal random variables, each with mean 3 and standard deviation 2, and let us draw 1000 such samples.

We will show that $(\bar{X} - \mu)\sqrt{n}/S$ has a t distribution on 19 degrees of freedom:

```
m <- 1000; n <- 20; sigma <- 2
# m samples of size n:
Z <- matrix(rnorm(m*n, mean = 3, sd = sigma), nrow=n)
Sz <- apply(Z, 2, sd); xbar <- apply(Z, 2, mean)
T <- sqrt(n)*(xbar - 3)/Sz  # t statistics
T[1:5] # first 5 t statistic values

## [1] -0.2478585 -0.2443742  1.5762034  0.3657780  1.4227428
```

These are scattered about 0.0.

Independence of the Sample Mean and Standard Deviation

Development of t and F statistics only worked because of independence of the sample mean and standard deviation.

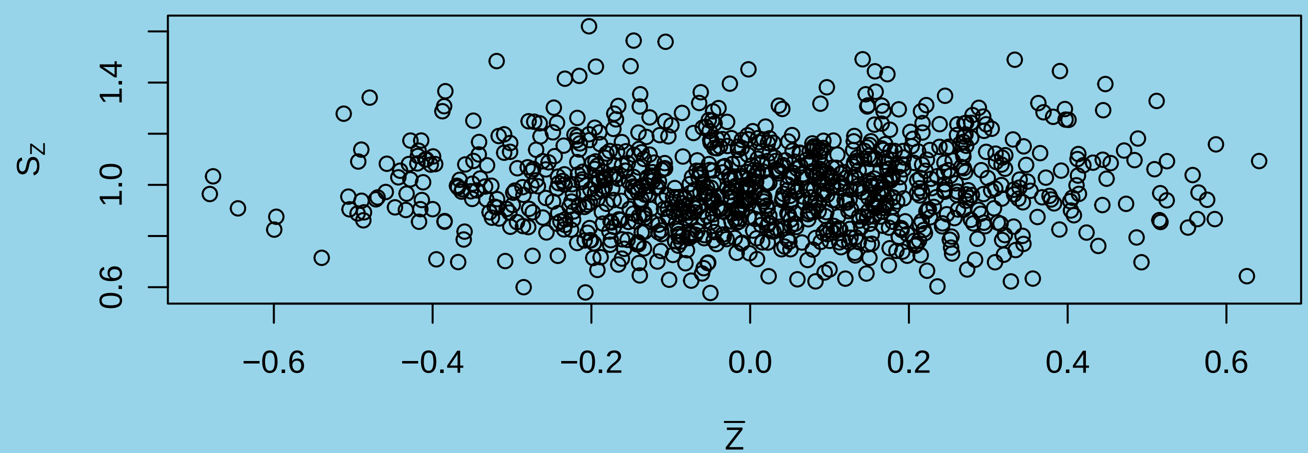
For normally distributed samples, the sample mean and standard deviation are independent.

We can see evidence for this from simulated data. Let us consider a samples of $n = 20$ uncorrelated standard normal random variables, and let us draw 1000 such samples. Here is a way to do this:

```
m <- 1000; n <- 20
Z <- matrix(rnorm(m*n), nrow=n)
zbar <- apply(Z, 2, mean); Sz <- apply(Z, 2, sd)
```

Independence of the Sample Mean and Standard Deviation

```
plot(Sz ~ zbar)
```

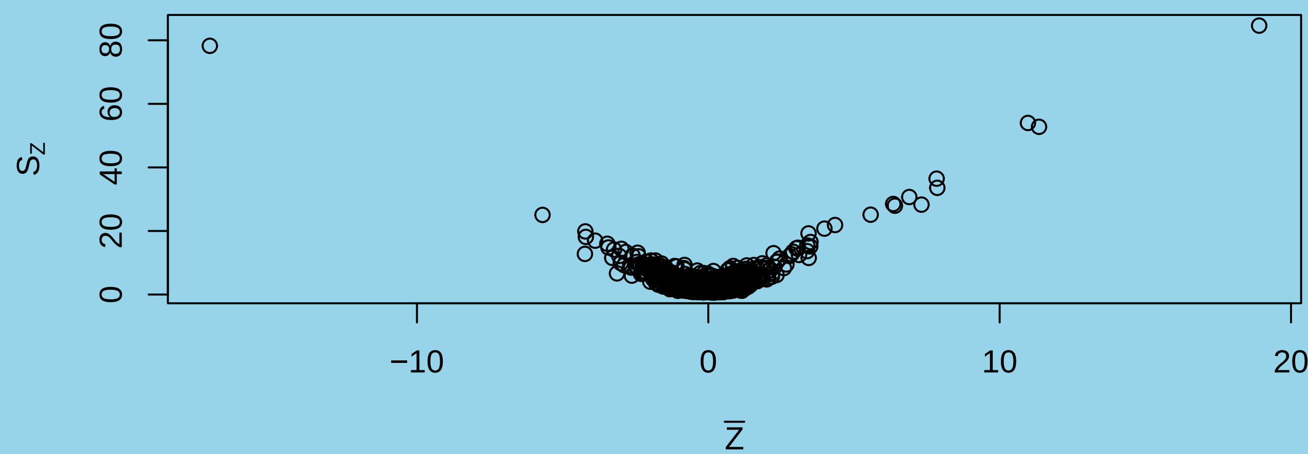


No pattern. It appears to be impossible to predict the standard deviation from the sample mean for normal data.

Dependence of the Sample Mean and Standard Deviation

For non-normal data, the picture is different. The sample mean and standard deviation are no longer independent. t and F statistics will no longer be accurate.

```
m <- 5000
Z <- matrix(rt(m*n, df=2), nrow=n) # t data on 2 df
zbar <- apply(Z, 2, mean); Sz <- apply(Z, 2, sd)
plot(Sz ~ zbar)
```



Clear pattern. The standard deviation is quite predictable from the sample mean for averages of t random variables.

Confidence Intervals for the Mean

Given data of the form $X_1, X_2, \ldots, X_n$ which are a random sample of independent normal random variables from a normal population with mean μ and variance σ^2, we want to estimate μ with a confidence interval.

We will use the usual statistical notation for the sample mean:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

and for the sample variance:

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

Frosted Flakes Example

Two Methods of Measuring Sugar Content:

1. **Lab Analysis - slow, but accurate**
2. **High Performance Liquid Chromatography (HPLC) - fast,**

   ... but is HPLC accurate?

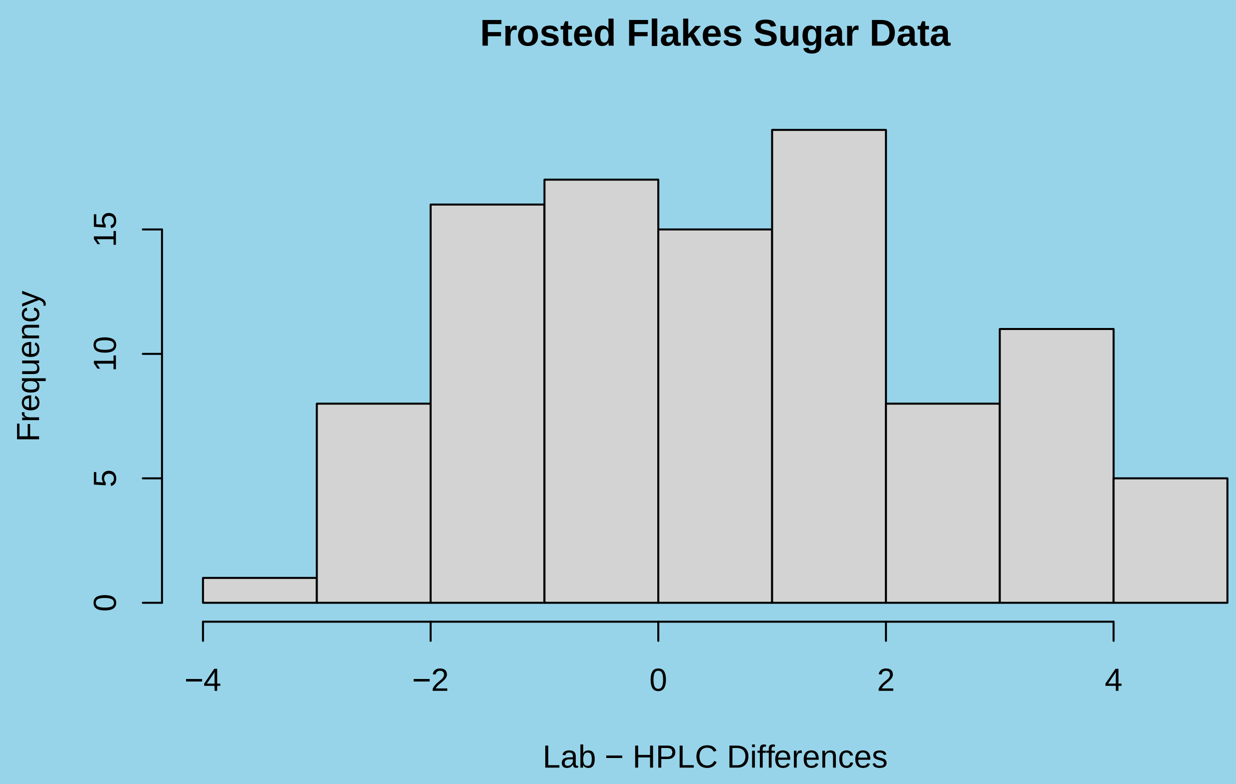
Measurements of each type were taken on 100 frosted flakes samples ...

Frosted Flakes Measurements

```r
FFdiff <- scan("FFdiff.txt")
length(FFdiff) # how many sample elements?

## [1] 100

FFdiff[1:10] # first 10 observations

##  [1] -1.2  2.7  1.1 -1.8 -2.8  1.1  2.7  1.9  3.3  3.1
```

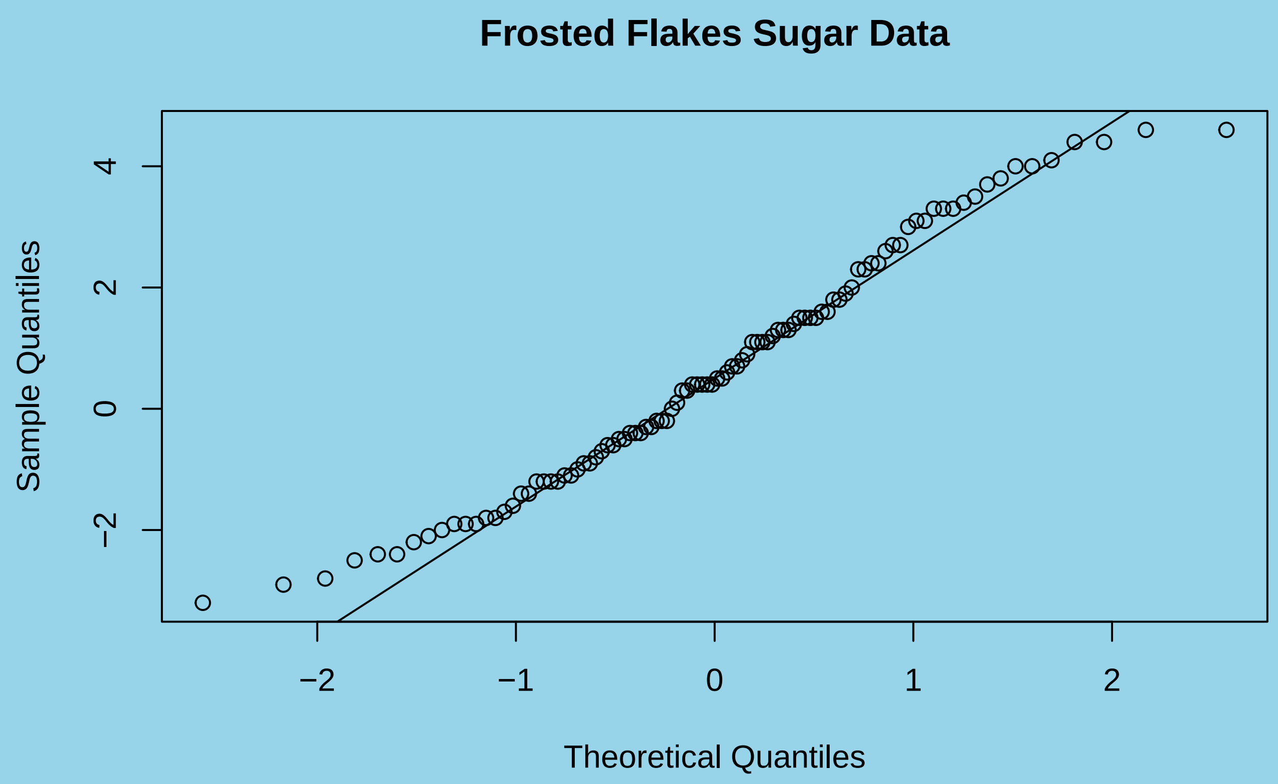
Frosted Flakes Measurements

```r
hist(FFdiff, main = "Frosted Flakes Sugar Data", xlab =
  "Lab - HPLC Differences")
```



Frosted Flakes Measurements

```
qqnorm(FFdiff, main = "Frosted Flakes Sugar Data")
qqline(FFdiff)
```



... reasonably normal-looking

What is the expected value of the difference ($\mu = E[X]$)?

Answer: We don't know.

Estimate: $\bar{x} = .622$**.**

How much error is there in this estimate?

Standard Error of Estimator: $\sqrt{\text{Var}(\textbf{Estimator})}$

Standard Error of $\bar{X}$: $\sqrt{\text{Var}(\bar{X})} = \frac{\sigma}{\sqrt{n}}$.

Estimated Standard Error (S.E.): $s/\sqrt{n}$ **= 1.98/10 = .198.**

Example (cont'd)

The approximate probability that $\bar{X}$ differs from μ by less than 2 standard errors is

$$P(-2S.E. < \bar{X} - \mu < 2S.E.) =$$

$$P(-2 < Z < 2) = .9772 - .0228$$

$$= .9544$$

($\bar{X}$ is approximately normal with mean μ and variance σ/n, if n is large enough.)

since

```
pnorm(2) - pnorm(-2)

## [1] 0.9544997
```

Conclusion

We can be 95.44% confident that the true expected value of the difference in sugar content measurements lies within 2 S.E. of .622:

$$.622 \pm .396.$$

This is an example of a 95.44% confidence interval.
We conclude that HPLC is not accurate. Calibration is required, if HPLC is to be used.

Confidence Interval Formula (Large n)

n independent measurements taken from a population with expected value μ and variance σ^2.

If n is large, then an approximate $100\%(1-\alpha)$ confidence interval for μ is given by

$$\bar{x} \pm z_{\alpha/2}\frac{s}{\sqrt{n}}$$

where $z_{\alpha/2}$ is defined so that

$$P(Z > z_{\alpha/2}) = \alpha/2$$

e.g. $z_{.2/2} = 1.28$ since

```
1 - pnorm(1.28)  # Obtain 1.28 using  " > 1 - qnorm(.1) "

## [1] 0.1002726
```

Exercise.

Find a 95% confidence interval for the expected difference in sugar content measurement.

$\alpha = .05$ $z_{.025} = 1.96$ **from**

```
qnorm(1 - .025)

## [1] 1.959964
```

The 95% c.i. for μ is given by

$$\bar{x} \pm z_{.025}\textbf{S.E.} = .622 \pm 1.96(.198) = .622 \pm .388$$

Exercise.

Find a 90% confidence interval for the expected difference in sugar content measurement.

$\alpha = .1$

$z_{.05} = 1.645$ **from**

```
qnorm(1 - .05)

## [1] 1.644854
```

The 90% c.i. for μ is given by

$$.622 \pm 1.645(.198) =$$

$$.622 \pm .326$$

A Small Sample Confidence Interval for μ

Define the upper percentile of the t distribution as $t_{\alpha,\nu}$ in

$$P(T > t_{\alpha,\nu}) = \alpha.$$

Here T has a t distribution on ν degrees of freedom. Use `qt(1-alpha, nu)`.

Then we can say that

$$P\left(-t_{\alpha/2,n-1} < \frac{\bar{X} - \mu}{S/\sqrt{n}} < t_{\alpha/2,n-1}\right) = 1 - \alpha.$$

A Small Sample Confidence Interval for μ

Therefore,

$$P\left(\bar{X} - t_{\alpha/2,n-1}S/\sqrt{n} < \mu < \bar{X} + t_{\alpha/2,n-1}S/\sqrt{n}\right) = 1 - \alpha.$$

and

$$\left(\bar{X} - t_{\alpha/2,n-1}S/\sqrt{n}, \bar{X} + t_{\alpha/2,n-1}S/\sqrt{n}\right) \tag{8}$$

defines a $1 - \alpha$ confidence interval for μ.

Example: Small Sample Confidence interval for μ

Find a 95% confidence interval for the expected value of concentration measurements taken from a chemical process. Sample measurements are

204	190	202	207
204	202	201	195

Example: Small Sample Confidence interval for μ

If X denotes a concentration measurement, then $\bar{x} = 201.$, $s = 5.50$, and $n = 8$ so a 95% confidence interval for $\mu = E[X]$ is

$$\bar{x} \pm t_{.025,7} \frac{s}{\sqrt{8}}$$

$$= 201 \pm 2.365(5.5)/\sqrt{8}$$

$$= 201 \pm 4.60$$

since

```
qt(1 - .025, 7)

## [1] 2.364624
```

Application to Monte Carlo Integration

Suppose $g(x)$ is any function that is integrable on the interval $[a, b]$.

The integral

$$\int_a^b g(x)dx$$

gives the area of the region with $a < x < b$ and y between 0 and $g(x)$ (where negative values count towards negative areas).

Monte Carlo integration uses simulation to obtain approximations to these integrals. It relies on the law of large numbers.

Monte Carlo Integration

This law says that a sample mean from a large random sample will tend to be close to the expected value of the distribution being sampled.

If we can express an integral as an expected value, we can approximate it by a sample mean.

We can assess the error in the simulation using the standard error and a confidence interval.

Monte Carlo Integration

For example, let $U_1, U_2, \ldots, U_n$ be independent uniform random variables on the interval $[a, b]$. These have density $f(u) = 1/(b-a)$ on that interval. Then

$$E[g(U_i)] = \int_a^b g(u)\frac{1}{b-a}du$$

so the original integral $\int_a^b g(x)dx$ can be approximated by $b-a$ times a sample mean of $g(U_i)$.

Example

To approximate the integral $\int_0^1 x^4 dx$, use the following lines:

```
u <- runif(100000)
mean(u^4)   #  Compare with the exact answer, $0.2$

## [1] 0.2018032
```

Calculate the standard error.

```
SE <- sd(u^4)/sqrt(100000); SE

## [1] 0.000847033
```

A 95% confidence interval for the integral is

```
mean(u^4) + c(-1.96, 1.96)*SE

## [1] 0.2001430 0.2034634
```

Example

To approximate the integral $\int_2^5 \sin(x)dx$, use the following lines:

```
u <- runif(100000, min = 2, max = 5)
mean(sin(u))*(5-2)   #  true value can be shown to be -0.700.

## [1] -0.7073143
```

Calculate the standard error.

```
SE <- sd(sin(u)*(5-2))/sqrt(100000); SE

## [1] 0.006199215
```

A 95% confidence interval for the integral is

```
mean(sin(u))*(5-2) + c(-1.96, 1.96)*SE

## [1] -0.7194648 -0.6951639
```

Multiple Integration

Now let $V_1, V_2, \ldots, V_n$ be an additional set of independent uniform random variables on the interval $[0, 1]$, and suppose $g(x, y)$ is now an integrable function of the two variables x and y. The law of large numbers says that

$$\lim_{n \to \infty} \sum_{i=1}^{n} g(U_i, V_i)/n = \int_0^1 \int_0^1 g(x, y) dx dy$$

with probability 1.

So we can approximate the integral $\int_0^1 \int_0^1 g(x, y) dx dy$ by generating two sets of independent uniform pseudorandom variates, computing $g(U_i, V_i)$ for each one, and taking the average.

Example

Approximate the integral $\int_3^{10}\int_1^7 \sin(x-y)dxdy$ using the following:

```
U <- runif(100000, min = 1, max = 7)
V <- runif(100000, min = 3, max = 10)
mean(sin(U - V))*42

## [1] 0.1160502
```

Calculate the standard error.

```
SE <- sd(sin(U-V)*42)/sqrt(100000); SE

## [1] 0.09382326
```

A 95% confidence interval for the integral is

```
mean(sin(U-V))*42 + c(-1.96, 1.96)*SE

## [1] -0.0678434  0.2999438
```

The factor of $42 = (7-1)(10-3)$ compensates for the joint density of U and V being $f(u,v) = 1/42$.