Probability Distributions & Confidence Interval Estimation

Nagiza F. Samatova, samatova@csc.ncsu.edu
Professor, Department of Computer Science
North Carolina State University

NC STATE UNIVERSITY
Department of Computer Science

Learning Objectives: Confidence Intervals (CI)

- Distinguish between the proper use of point estimates and interval estimates
- Calculate confidence intervals (via resampling or formula)
- Calculate standard error and explain its difference from standard deviation
- Calculate CI for:
  - a mean
  - a proportion
  - a difference in means
  - a difference in proportion
- Explain the relationship between the Central Limit Theorem and the applicability of Normal approximation for confidence intervals

Descriptive vs. Inferential Statistics

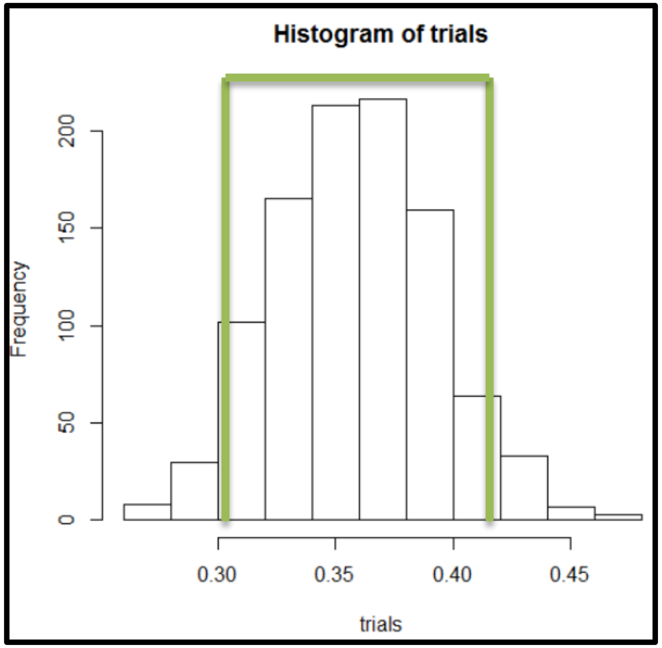
- Descriptive or Summary Statistics
  - **Goal: to describe the features of a collection of data in a quantitative way**
  - **Measures of Central Tendency:**
    - mean, median, and mode
  - **Measures of Variability, Dispersion, or Spread:**
    - range, variance, standard deviation, quartiles
- Inferential or Inductive Statistics
  - **Goal: to summarize a sample of the data to infer or draw conclusions about the population from which the sample is drawn**
    - Hypothesis testing
      - A/B testing
      - p-value
      - t-tests, χ^2-tests, F-tests
    - **Confidence intervals**

Confidence Interval (CI): Bootstrap Procedure

- **Use the observed sample as a good proxy for the population**
- **Make the resample size equal the size of the original sample**
- **Conduct Bootstrap Sampling**
  - **to compute confidence interval for the point estimate of the population statistic;**
  - **with point estimate calculated from the original observed sample**
- **R package:**
  - install.packages (“boot”)
  - library (boot)
  - boot.obj = boot.boot(…)
  - ci.obj = boot::boot.ci(boot.obj)
  - Example: Bootstrap_sampling.R

Ex: CI Estimation via Bootstrap Simulation

```r
# Example: Obama's handling economy example

resample.statistic <- function(data, size=200,
                               replace = TRUE, fun.name) {
  draw <- sample(data, size, replace)
  pos.rate <- fun.name(draw)/size
  return(pos.rate)
}

# positive and negative rates
sample.size <- 200
pos.response <- 72
neg.response <- 128
positive.rate <- pos.response / sample.size

hat <- c(rep(1,pos.response), rep(0,neg.response))

with.replacement <- TRUE
n.repeats <- 1000

trials <- replicate(n.repeats,
                    resample.statistic(hat, sample.size,
                                       with.replacement, sum))
hist(trials)
ci <- quantile(trials, c(0.05, 0.95))
cat("Point Estimate: ", positive.rate, "\n")
cat("90% Confidence Interval: ",ci,"\n")
```



90% CI: [0.3; 0.42]

Statistical Distributions & Functions in R

Distribution	Random Number Generator	Density	Distribution	Quantile
Normal	rnorm	dnorm	pnorm	qnorm
t	rt	dt	pt	qt
F	rf	df	pf	qf
χ^2	rchisq	dchisq	pchisq	qchisq

{dpqr}*distribution_abbreviation*()

- **d** = density
- **p** = distribution function
- **q** = quantile function
- **r** = random generation

- **pnorm(a)** $\equiv \boldsymbol{P}(\boldsymbol{X} \leq \boldsymbol{a})$: probability that a or smaller number occurs
- **pnorm(b) – pnorm(a)** $\equiv \boldsymbol{P}(\boldsymbol{a} \leq \boldsymbol{X} \leq \boldsymbol{b})$: probability that the variable falls between two points
- qnorm(): given the cumulative probability distribution, it returns the quantile

Statistical Distributions: Mean and Variance

Distribution	Degrees of freedom	Mean	Variance
Normal		μ	σ^2
t	n	0	$n/(n-2)$
F	n_1 and n_2	$n_2/(n_2-2)$	a/b
χ^2	r	r	$2r$

$$a = 2n_2^2(n_1 + n_2 - 2)$$

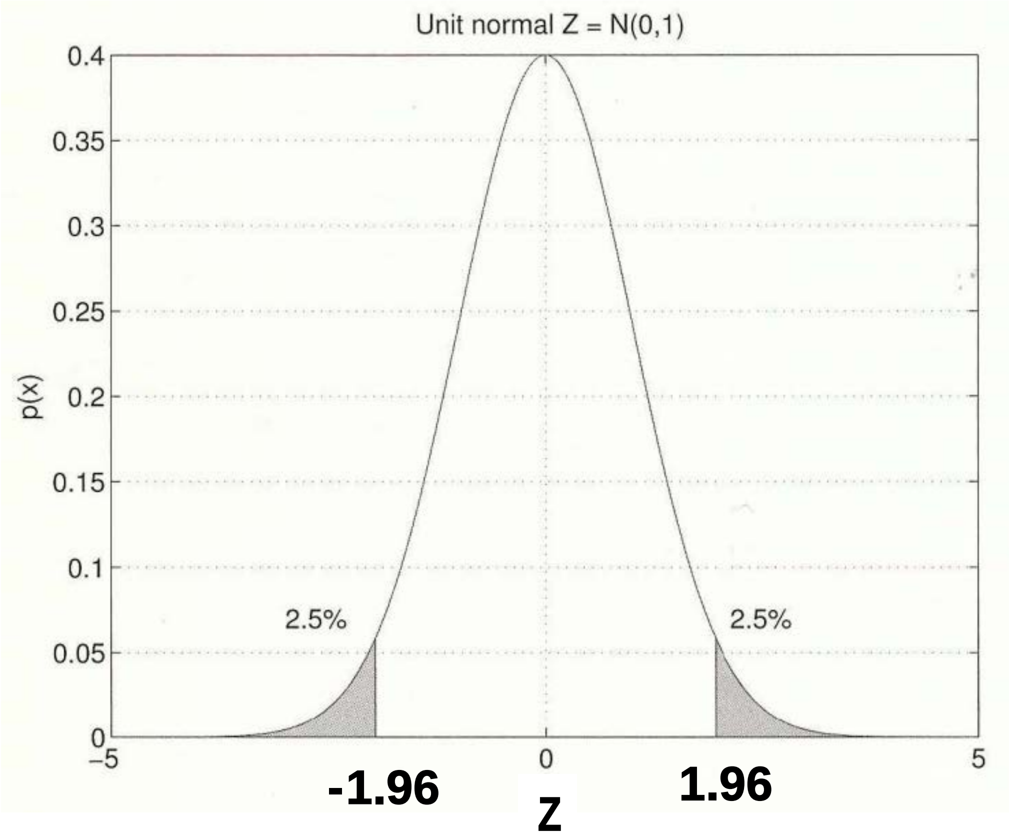
$$b = n_1(n_2 - 2)^2\,(n_2 - 4)$$

Unit Normal Distribution: $N(\mu = 0, \sigma^2 = 1)$

```r
# Generate 1000 points with the unit normal distribution
randUnitNormal <- rnorm(1000, mean=0, sd = 1)

# Calculate their distribution
densityRandUnitNormal <- dnorm(randUnitNormal)

# Plot the ditribution
require(ggplot2)
df <- data.frame(x=randUnitNormal, y=densityRandUnitNormal)
ggplot(df) + aes(x=x, y=y) + geom_point() +
         labs(x="Random Unit Normal Variable", y="Density")

# Compute the probability that x is less than 1.64
pnorm(1.64)

# Compute the probability that x lies between -1.96 and 1.96
pnorm(1.96) - pnorm(-1.96)

# Compute cumulative probability distribution
probabilityRandUnitNormal <- pnorm(randUnitNormal)
df2 <- data.frame(x=randUnitNormal, y=probabilityRandUnitNormal)
ggplot(df2) + aes(x=x, y=y) + geom_line() +
     labs(x="Random Unit Normal Variable", y="Probability")

1 - pnorm(1.96)
```

Sampling_normal_distribution.R

Two-sided Confidence Interval



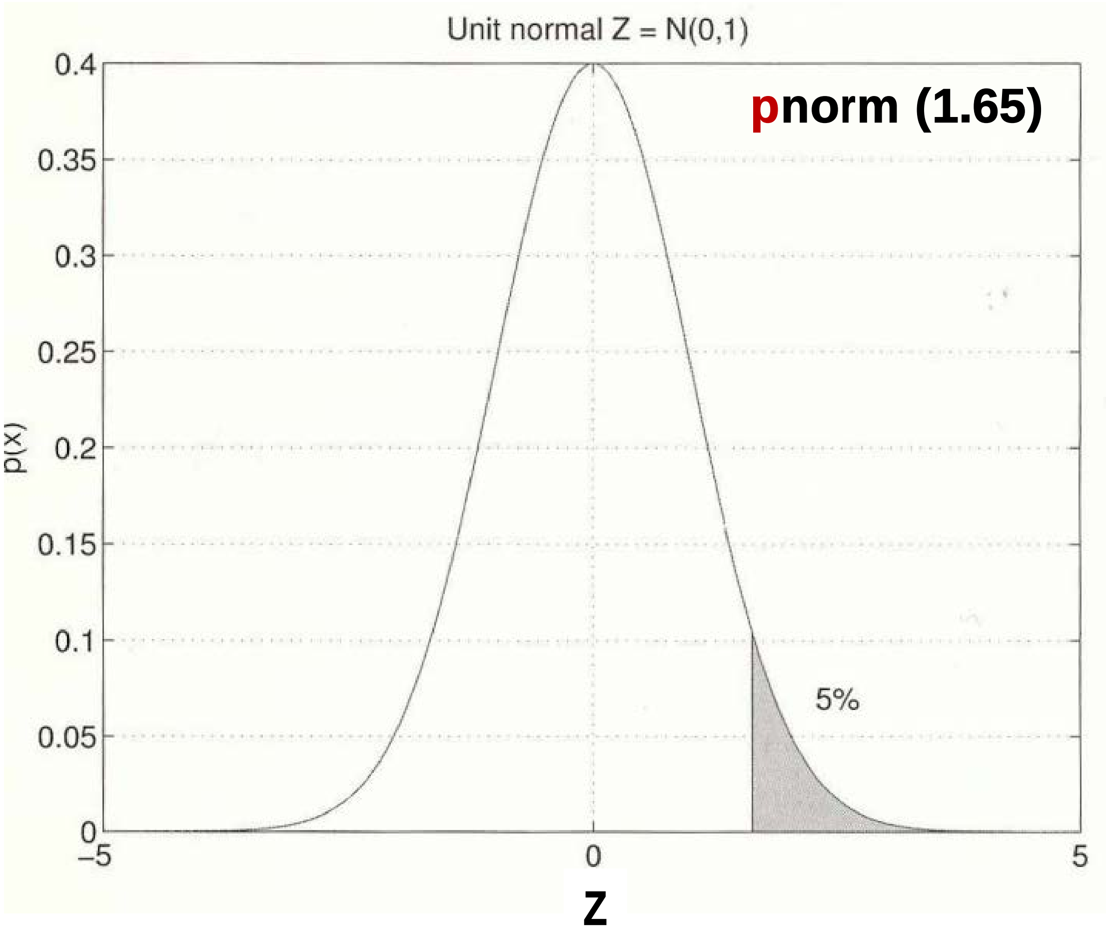
95% of the unit normal distribution lies between - 1.96 and 1.96

$$P\{\ |Z\ -0| < 1.96\ \} = 0.95$$

pnorm (1.96) – pnorm (-1.96)

What is (1 - pnorm(1.96))?

One-sided Confidence Interval



95% of the unit normal distribution lies below 1.64

$$P\{\, Z < 1.64 \,\} = 0.95$$

Normal Distribution: $N(\mu, \sigma^2)$

- Generate a sample of 100 random normal deviates with a mean $\mu = 50$ and a standard deviation $\sigma = 10$:
  - **rand.normal <- rnorm (100, mean = 50, sd = 10)**

- Compute the estimate of the mean ($\boldsymbol{m}$) and the estimate of the standard deviation ($\boldsymbol{S}$) from the rand.normal:
  - **m <- mean (rand.normal)**
  - **S <- sd (rand.normal)**

- Are the estimates equal to the theoretical model parameters?
  - $\boldsymbol{m} == \boldsymbol{\mu}$?
  - $\boldsymbol{S} == \boldsymbol{\sigma}$?

- Repeat the first two steps 1000 times, plot the histogram of the mean estimates, and compute the mean and the variance of the estimates:
  - **m.estimates <- sapply (1:1000,**
    FUN=function(iter) { mean(rnorm(100, mean=50, sd=10)) })
  - **hist (m.estimates)**
  - **mean (m.estimates)**
  - **var (m.estimates)**

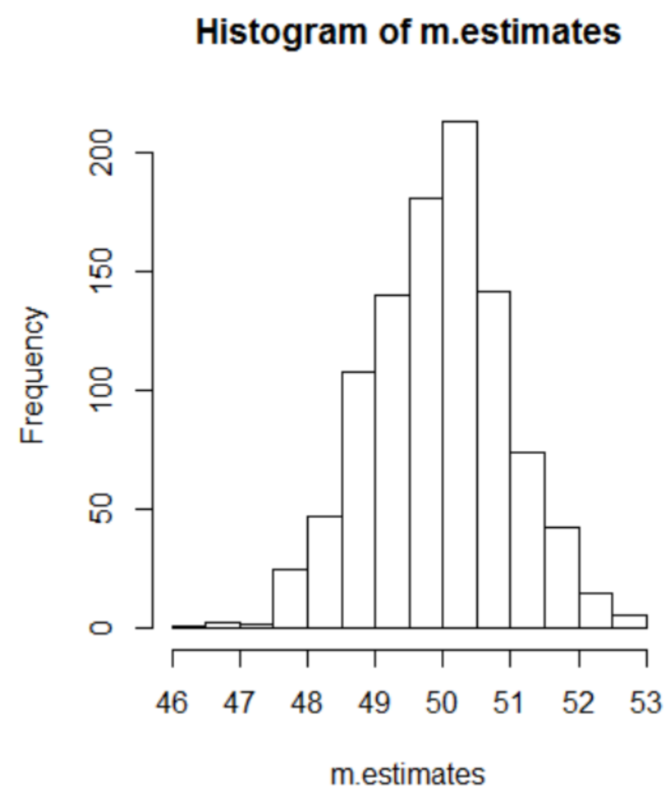
Normal Distribution of Point Estimator: $m \sim N(\mu, \sigma^2/n)$

Suppose we are trying to **estimate the mean μ** of a **population** with a normal density from its **sample** $X = \{x^t\}, t = 1, \dots, n$, i.e., $x^t \sim N(\mu, \sigma^2)$:

- $m = \sum_t x^t / n$: the **sample average**, i.e. the **point estimator to the mean**
- Because m is the sum of normals, it is also **normal**, $m \sim N(\mu, \frac{\sigma^2}{n})$

Illustration of this concept:

- **n <- 100**
- **mu <- 50**
- **sigma <- 10**
- **m.estimates <-** sapply (1:1000,
  FUN=function(iter)
  { mean(rnorm(n, mean=mu, sd=sigma)) })
- **hist (m.estimates)**
- **mean (m.estimates)**
- **var (m.estimates)**
- **sigma * sigma / n**

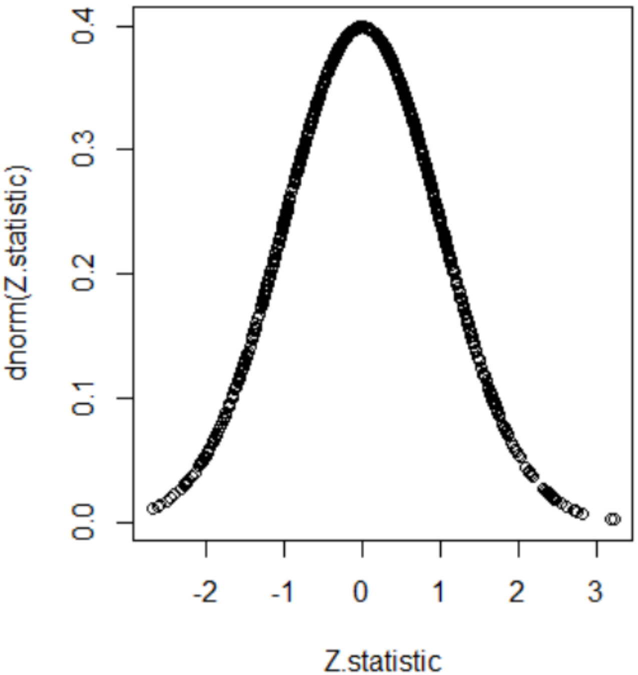


Z-normalization: $Z \sim \frac{m-\mu}{\sigma/\sqrt{n}} \sim N(0,1)$

Suppose we are trying to estimate the mean μ of a normal density from a sample $X = \{x^t\}, t = 1, \dots, n$, i.e., $x^t \sim N(\mu, \sigma^2)$:

- $\boldsymbol{m} = \sum_t \boldsymbol{x}^t / \boldsymbol{n}$: the sample average, i.e. the point estimator to the mean
- Assumption: **the variance σ^2 is known**
- Define: Z-statistic, $\boldsymbol{Z \sim \frac{m-\mu}{\sigma/\sqrt{n}}}$
- Claim: Z-statistic is from a unit normal distribution, $\boldsymbol{Z \sim N(0,1)}$

```
n <- 100
mu <- 50
sigma <- 10
nrep <- 1000
m.estimates <- sapply (1:nrep,
    FUN=function(iter)
    { mean(rnorm(n, mean=mu, sd=sigma)) })
mu.vec <- rep(mu, nrep)
Z.statistic <- (m.estimates-mu.vec)/(sigma/sqrt(n))
plot(Z.statistic, dnorm(Z.statistic),
     x.lab = "Z.statistic", y.lab = "Density")
```



Two-sided Confidence Interval for Z-statistic

Remind that we are trying **to estimate the mean μ** of a population with normal density from its sample $X = \{x^t\}, t = 1, \dots, n,$ i.e., $x^t \sim N(\mu, \sigma^2)$:

- $\boldsymbol{m = \sum_t x^t / n}$: the sample average, i.e. the point estimator to the mean
- Assumption: **the variance σ^2 is known**
- Define: Z-statistic, $\boldsymbol{Z \sim \frac{m-\mu}{\sigma / \sqrt{n}}}$
- Claim: Z-statistic is from a unit normal distribution, $\boldsymbol{Z \sim N(0, 1)}$
  - Remember that 95% of Z lies in (-1.96; 1.96)
    - $\boldsymbol{P\{-1.96 < Z < 1.96\} = 0.95}$
  - Based on the definition of the Z-statistic:
    - $\boldsymbol{P\left\{-1.96 < \frac{m-\mu}{\sigma / \sqrt{n}} < 1.96\right\} = 0.95}$

$$\boxed{\boldsymbol{P\left\{m - 1.96\frac{\sigma}{\sqrt{n}} < \mu < m + 1.96\frac{\sigma}{\sqrt{n}}\right\} = 0.95}}$$

With 95% two-sided confidence,
μ lies within $1.96\ \sigma/\sqrt{n}$ units of the sample average

Exercise: 99% two-sided CI for μ

- **Note:** **the variance σ^2 is known**

$$\boldsymbol{\mu} \in \left(\boldsymbol{m} - \frac{\mathbf{2.58}\boldsymbol{\sigma}}{\sqrt{\boldsymbol{n}}} ; \boldsymbol{m} + \frac{\mathbf{2.58}\boldsymbol{\sigma}}{\sqrt{\boldsymbol{n}}}\right)$$

- $\boldsymbol{m} = \sum_{t} \boldsymbol{x}^{t} / \boldsymbol{n}$: the sample average, i.e. the point estimator to the mean

- The **higher the confidence value** is, the **larger the confidence interval** is
- The interval gets smaller as n, the sample size, increases

Generalization: Two-sided Confidence Interval (CI)

- Assumption: **the variance σ^2 is known**
- Define: Z-statistic, $Z \sim \dfrac{m-\mu}{\sigma/\sqrt{n}}$
- Let's denote z_α such that: $P\{Z > z_\alpha\} = \alpha, 0 < \alpha < 1$
- Because Z is symmetric around the mean:
  - $z_{1-\alpha/2} = -z_{\alpha/2}$
  - $P\{X < -z_{\alpha/2}\} = P\{X > z_{\alpha/2}\} = \alpha/2$
  - $P\{-z_{\alpha/2} < Z < z_{\alpha/2}\} = 1-\alpha$

- Based on the definition of the Z-statistic:
  - $P\left\{-z_{\alpha/2} < \dfrac{m-\mu}{\sigma/\sqrt{n}} < z_{\alpha/2}\right\} = 1-\alpha$

$100(1-\alpha)\%$ **two-sided confidence interval for μ for any α:**

$$\boxed{P\left\{m - z_{\alpha/2}\frac{\sigma}{\sqrt{n}} < \mu < m + z_{\alpha/2}\frac{\sigma}{\sqrt{n}}\right\} = 1-\alpha}$$

Generalization: One-sided Confidence Interval

$\mathbf{100(1-\alpha)\%}$ **one-sided upper confidence interval for μ defines a lower bound:**

$$\boxed{\boldsymbol{P\left\{m - z_\alpha \frac{\sigma}{\sqrt{n}} < \mu\right\} = 1 - \alpha}}$$

- Example: $\left(m - \frac{1.64\sigma}{\sqrt{n}};\ \infty\right)$ is 95% a one-sided upper confidence interval for μ

Summary #1: Confidence Intervals (CI)

- A $(1-\alpha)*100\%$ **confidence interval** is a ***random interval*** $[\boldsymbol{L};\boldsymbol{U}]$ such that
  - If we were to repeat the data gathering and formation of sample average $m = \bar{X}$ and $[\boldsymbol{L};\boldsymbol{U}]$ many times then $[\boldsymbol{L};\boldsymbol{U}]$ would contain μ at least $(1-\alpha)*100\%$ of the time.

- Confidence interval measures how much your estimator will vary from one sample to the next:
  - It is wide when your estimator is based on limited information (small samples) or is approximating a parameter in a noisy setting.

- For i.i.d. normal sample from $N(\mu, \sigma^2)$ with **known variance**:
  - $m \pm 1.96\sigma/\sqrt{n}$ is a 95% CI for μ
  - More generally, $\boldsymbol{m \pm z_{1-\alpha/2}\sigma/\sqrt{n}}$ is a $(1-\alpha)$100% CI for μ, where $z_{1-\alpha}$ satisfies $P(Z > z_{1-\alpha}) = 1-\alpha$ for the Z-statistic

- Smaller values of will result in wider intervals (need to be more sure)
- More data will result in smaller (shorter) intervals

Statistical Distributions & Functions in R

Distribution	Random Number Generator	Density	Distribution	Quantile
Normal	rnorm	dnorm	pnorm	qnorm
t	rt	dt	pt	qt
F	rf	df	pf	qf
χ^2	rchisq	dchisq	pchisq	qchisq

{dpqr}*distribution_abbreviation*()

- **d** = density
- **p** = distribution function
- **q** = quantile function
- **r** = random generation

Distribution	Degrees of freedom	Mean	Variance
Normal		μ	σ^2
t	n	0	$n/(n-2)$
F	n_1 and n_2	$n_2/(n_2-2)$	a/b
χ^2	r	r	$2r$

χ^2-distribution

- X^2**-distribution** plays an important role in several applications:
  - Test for **independence** of two **categorical** variables
  - **Confidence interval** estimation
  - **Hypothesis testing**
  - **Goodness of fit** of the observed data to the expected data under the fitted model

- X^2**-distribution** is the component of a number of distributions:
  - t**-distribution**
  - F**-distribution**

χ^2-distribution: Formal Definition

- If Z_i are independent unit normal random variables ($Z_i \sim N(0,1)$), then
  - $\boldsymbol{Q} = \boldsymbol{Z_1^2} + \boldsymbol{Z_2^2} + \cdots + \boldsymbol{Z_r^2}$ is chi-square with r degrees of freedom, i.e.
  - $\boldsymbol{Q} \sim \boldsymbol{\chi_r^2}$ or $\boldsymbol{Q} \sim \boldsymbol{\chi^2(r)}$, $\boldsymbol{r}$ is positive integer

- The expected value of Q: $E[Q] = r$
- The variance of Q: $Var(Q) = 2r$

- **Example of chi-square distribution**:
  - Consider the sample $X = \{x^t\}, t = 1, \dots, n,$ i.e., $\boldsymbol{x^t \sim N(\mu, \sigma^2)}$:
  - $\boldsymbol{m} = \sum_t \boldsymbol{x^t} / \boldsymbol{n}$: the sample average

  - Consider **the estimated sample variance**:
    - $\boldsymbol{S^2} = \dfrac{\sum_t \left(\boldsymbol{x^t} - \boldsymbol{m}\right)^2}{\boldsymbol{n} - 1}$

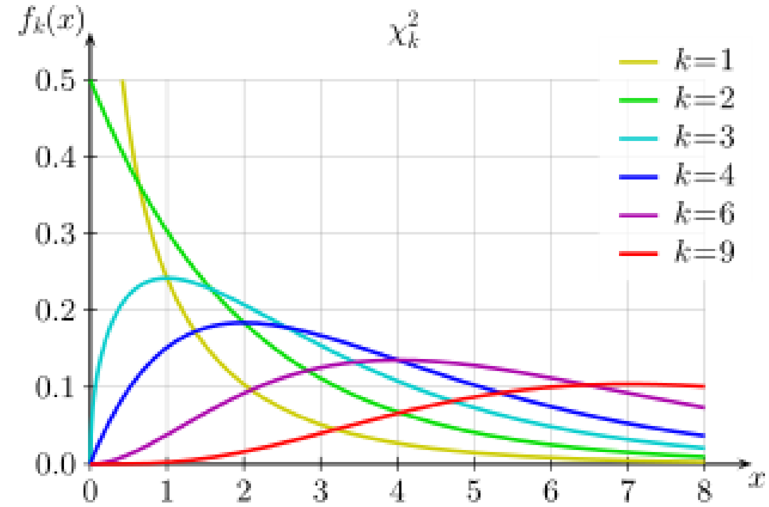
$$\boxed{(\boldsymbol{n} - 1)\frac{\boldsymbol{S^2}}{\boldsymbol{\sigma^2}} \sim \boldsymbol{\chi^2_{n-1}}}$$

  - It is also known that $\boldsymbol{m}$ and $\boldsymbol{S^2}$ are **independent**

χ^2-distribution with different degrees of freedom

```r
# Generate n random numbers from the
# Chi-squared-distribution
# with n degrees of freedom
n = 1000
df <- 7
chiSquaredRandom <- rchisq(n, df=df)

# Confirm that the empirical mean is n
# and the variance is 2n
mean(chiSquaredRandom)
var(chiSquaredRandom)

# Compute and Plot the chi-squared-distribution
require(ggplot2)
densityChiSquared <- dchisq(chiSquaredRandom, df=df)
df.data <- data.frame(x=chiSquaredRandom, y=densityChiSquared)
ggplot(df.data) + aes(x=x, y=y) + geom_line() +
  labs(x="Random Chi-Squared-Distribution Variable", y="Density")
```

File: Explore_distributions.R



Src: Wiki page

- **How does χ^2_n look like?**

Exercise

- Find the 95th percentile of the Chi-Squared distribution with 7 degrees of freedom.

```
> qchisq(.95, df=7)
[1] 14.06714
```

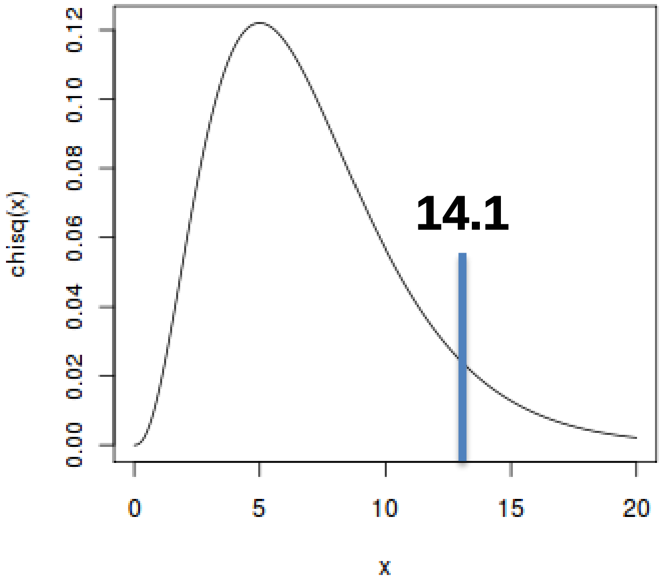


Table of χ^2 values vs. p-values

Degrees of freedom (df)	χ^2 value[18]										
1	0.004	0.02	0.06	0.15	0.46	1.07	1.64	2.71	3.84	6.64	10.83
2	0.10	0.21	0.45	0.71	1.39	2.41	3.22	4.60	5.99	9.21	13.82
3	0.35	0.58	1.01	1.42	2.37	3.66	4.64	6.25	7.82	11.34	16.27
4	0.71	1.06	1.65	2.20	3.36	4.88	5.99	7.78	9.49	13.28	18.47
5	1.14	1.61	2.34	3.00	4.35	6.06	7.29	9.24	11.07	15.09	20.52
6	1.63	2.20	3.07	3.83	5.35	7.23	8.56	10.64	12.59	16.81	22.46
7	2.17	2.83	3.82	4.67	6.35	8.38	9.80	12.02	14.07	18.48	24.32
8	2.73	3.49	4.59	5.53	7.34	9.52	11.03	13.36	15.51	20.09	26.12
9	3.32	4.17	5.38	6.39	8.34	10.66	12.24	14.68	16.92	21.67	27.88
10	3.94	4.87	6.18	7.27	9.34	11.78	13.44	15.99	18.31	23.21	29.59
P value (Probability)	0.95	0.90	0.80	0.70	0.50	0.30	0.20	0.10	0.05	0.01	0.001

- The p-value is the probability of observing a test statistic at least as extreme as in a chi-squared distribution.

Statistical Distributions & Functions in R

Distribution	Random Number Generator	Density	Distribution	Quantile
Normal	rnorm	dnorm	pnorm	qnorm
t	rt	dt	pt	qt
F	rf	df	pf	qf
χ^2	rchisq	dchisq	pchisq	qchisq

{dpqr}*distribution_abbreviation*()

- **d** = density
- **p** = distribution function
- **q** = quantile function
- **r** = random generation

Distribution	Degrees of freedom	Mean	Variance
Normal		μ	σ^2
t	n	0	$n/(n-2)$
F	n_1 and n_2	$n_2/(n_2-2)$	a/b
χ^2	r	r	$2r$

t –Distribution

- If $Z \sim N(0,1)$ and $Q \sim \chi_n^2$ are independent, then
- $\boldsymbol{T_n = \frac{Z}{\sqrt{Q/n}}}$ is ***t*-distributed with $\boldsymbol{n}$ degrees of freedom**
- The expected value of T_n: $\boldsymbol{E[T_n] = 0}$
- The variance of T_n : $\boldsymbol{Var(T_n) = \frac{n}{n-2}, n > 2}$

  - Like the unit normal density, t –distribution is symmetric around 0
  - As n becomes larger, t –density becomes more like the unit normal
    - t –distribution has thicker tails indicating greater variability than does normal

t –distribution is like the unit normal

```r
# Generate n random numbers from the t-distribution
# with n degrees of freedom
n = 1000
degf <- n
tRandom <- rt(n, df=degf)

# Confirm that the empirical mean is zero
# and the variance of n / (n-2)
mean(tRandom)
var(tRandom)

# Compute and Plot the t-distribution
require(ggplot2)
densityTRandom <- dt(tRandom, df=degf)
df <- data.frame(x=tRandom, y=densityTRandom)
ggplot(df) + aes(x=x, y=y) + geom_line() +
  labs(x="Random t-Distribution Variable", y="Density")

# Use a Shapiro-Wilk test to check
# whether tRandom is from the normal distribution
# when the p-value is lower than 0.05,
# conclude that the sample deviates from normality
plot(density(tRandom))
shapiro.test(tRandom)

# Plot using the qqnorm()
qqnorm(tRandom)
qqline(tRandom, col=2)
```

File: Explore_distributions.R

t –Distribution: Applications

- The most used applications are **power calculations for t-tests**

- <u>**Example of t –distribution**</u>:
  - Consider the sample $X = \{x^t\}, t = 1, \ldots, n,$ i.e., $\boldsymbol{x^t \sim N(\mu, \sigma^2)}$:
  - $\boldsymbol{m = \sum_t x^t / n}$: **the estimated sample mean**
  - $\boldsymbol{S^2 = \frac{\sum_t (x^t - m)^2}{n-1}}$: **the estimated sample variance**

- <u>Define</u>: T-statistic, $\boldsymbol{T_{n-1} \sim \frac{m-\mu}{S / \sqrt{n}}}$
- <u>Claim</u>: T_{n-1}-statistic is from a t –distribution with $n-1$ degrees of freedom

t –Distribution: Confidence Interval Estimation

- Consider the i.i.d sample $X = \{x^t\}, t = 1, \dots, n,$ i.e., $\boldsymbol{x^t \sim N(\mu, \sigma^2)}$:
- $\boldsymbol{m = \sum_t x^t / n}$: the estimated sample mean
- $\boldsymbol{S^2 = \frac{\sum_t (x^t - m)^2}{n-1}}$: **the estimated sample variance**
- Define: T-statistic, $\boldsymbol{T_{n-1} \sim \frac{m-\mu}{S/\sqrt{n}}}$
- Claim: T_{n-1}-statistic is from a t –distribution with $n-1$ degrees of freedom
- Let's denote $t_{\alpha,n-1}$ such that: $\boldsymbol{P\left\{T_{n-1} > t_{\alpha,n-1}\right\} = \alpha, 0 < \alpha < 1}$
- Because T_{n-1} is symmetric around the mean:
  - $t_{1-\alpha/2,n-1} = -t_{\alpha/2,n-1}$
  - $P\{t_{1-\alpha/2,n-1} < T_{n-1} < t_{\alpha/2,n-1}\} = 1 - \alpha$

$$P\left\{t_{1-\alpha/2,n-1} < \frac{m-\mu}{S/\sqrt{n}} < t_{\alpha/2,n-1}\right\} = 1 - \alpha$$

$$\boxed{\boldsymbol{P\left\{m - t_{\alpha/2,n-1}\frac{S}{\sqrt{n}} < \mu < m + t_{\alpha/2,n-1}\frac{S}{\sqrt{n}}\right\} = 1 - \alpha}}$$

Central Limit Theorem

- So far we have assumed that the data were generated from a normal distribution
  - This is a reasonable approximation in some cases
- Central Limit Theorem: even if the data are not normal, the sample mean m is approximately normal provided that n is sufficiently large:
  - For our purposes 'sufficiently large' means 30
  - In real life 'sufficiently large' will depend on the underlying distribution

Summary #2: Confidence Intervals (CI)

- A $(1-\alpha) * 100\%$ **confidence interval** is a ***random interval*** $[\boldsymbol{L}; \boldsymbol{U}]$ such that
  - If we were to repeat the data gathering and formation of sample average $m = \bar{X}$ and $[\boldsymbol{L}; \boldsymbol{U}]$ many times then $[\boldsymbol{L}; \boldsymbol{U}]$ would contain μ at least $(1-\alpha) * 100\%$ of the time.

- For i.i.d. normal sample from $N(\mu, \sigma^2)$ with the **Un-known** variance:
  - $\boldsymbol{m \pm t_{n-1,1-\alpha/2} S/\sqrt{n}}$ is a $(1-\alpha)$100% CI for μ,
    - where $t_{1-\alpha,n-1}$ satisfies $P\left(T_{n-1} > z_{1-\alpha,n-1}\right) = 1-\alpha$ for the $\boldsymbol{t}$ **–statistic** $\boldsymbol{T_{n-1}}$ with $n-1$ degrees of freedom and
    - $\boldsymbol{S^2 = \frac{\sum_t (x^t - m)^2}{n-1}}$: the estimated sample variance

- For i.i.d. normal sample from $N(\mu, \sigma^2)$ with the **known** variance:
  - $\boldsymbol{m \pm z_{1-\alpha/2} \sigma/\sqrt{n}}$ is a $(1-\alpha)$100% CI for μ, where $z_{1-\alpha}$ satisfies $P(Z > z_{1-\alpha}) = 1-\alpha$ for the $\boldsymbol{Z}$**-statistic** $\boldsymbol{Z}$