Data Analysis (534)

Textbook: Modern Applied Statistics with S, 4th ed.
by Venables and Ripley
https://www.researchgate.net/publication/224817420_Modern_Applied_Statistics_With_S/link/5f5481f092851c250b96c0a9/download
If you cannot find it, let me know.
Office: WH 132
Office hours: M, Tu 7:00pm-8:00pm through zoom
https://binghamton.zoom.us/j/8265526594?pwd=d3l6OGx1cmZ4M3cxZEJwVGd1RGcrUT09
Meeting ID: 826 552 6594
Passcode: 031320
Classroom: WH 329 MWF 1:10-2:10 pm
Homework due: Wednesday in class.
Or Email me at qyu@math.binghamton.edu.
Sample of the Latex homework is in **report and hw-solution** items in my website.
Grading policy: 40% homework and quizzes + 60% Exams.
B = 75 $\pm$
You can bring one page with R commands and formulas in exams, bring a simple calculator.
Midterm: Mar. 18 (M)
Final: May 9 Th. 5:40pm-7:40pm LN 2402 (WH 329)
Quiz: Once a week at a random day,
quiz problems: formulas for Math 447-448 (see my website)
Homework assigned during a week is due next Wednesday in class.
It is on my website: http://www.math.binghamton.edu/qyu/qyu_personal
Remind me if you do not see it by Saturday morning !
There is a homework due this Friday. I will send you the problems.

The lecture note is also on my website
http://www.math.binghamton.edu/qyu/qyu_personal
note and note2 are updated one,

Chapter 0. Introduction.

This Data analysis course is to teach how to analyze data (using R program).
Usual steps in data analysis:

1. For a given random sample, *e.g.*, regression data,
   (X_i, Y_i), $i = 1, ..., n$, input them to a computer software, say R.
2. Assume a proper probability model, say a parametric model
   $Y_i = \beta' X_i + \epsilon_i$, where $\epsilon \sim N(\alpha, \sigma^2)$ (NID);
   or a semiparametric model
   $Y_i = \beta' X_i + \epsilon_i$, where $\epsilon \sim F$, an unknown cumulative distribution function (cdf),
   or a non-parametric model
   $(X_i, Y_i) \sim F(x, y)$, where F is unknown.
3. Compute an estimate of (α, β, σ) if it is parametric,
   or an estimate of (β, F) if it is semi-parametric,
   or an estimate of F, if it is non-parametric.
4. Check whether the model assumption is valid (*e.g.*, iid, NID, LR model *etc.*.
5. If No, go to Step 2, otherwise, carry out the other statistics inferences, e.g.,
   testing statistical hypotheses,
   or constructing confidence intervals,
   or drawing inferences on some other parameters, *e.g.* $P(Y \in A | X = x) = ?$

Example 1. **An example how to hand in homework.**

X_i: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30

Y_i: 1.40 1.40 3.36 4.69 6.05 7.35 7.27 6.80 8.94
8.68 11.24 11.62 12.85 14.38 14.28 15.43 16.86 18.08 18.45
19.73 20.63 20.73 23.30 23.06 26.15 27.45 27.67 27.64 28.28 32.30

Suppose it is saved in a file called "data" in a directory /home/qyu/try in a PC.
cd /home/qyu/try
Two ways to work on R:

1. Write a program file, say ch0,
   R - -vanilla < ch0 # figure is in the file Rplots.pdf
   R - -vanilla < ch0 > output # all commands and output in the file called "output".
2. Open R in that directory directly by typing:
   R or click the icon of R on a laptop.

You can find R download site through Google.
Or login to department computer.
ssh qyu@ssh1.math.binghamton.edu
sftp qyu@ssh1.math.binghamton.edu

```
> library(MASS)
> sink("ch0.out") # put output in ch0.out file
> x=matrix(scan("data"), ncol=1, byrow=T)
> y=x[31:60]
> x=x[1:30]
> z=lm(y~x)
> summary(z)
> plot(x,y)   # scatter plot
```

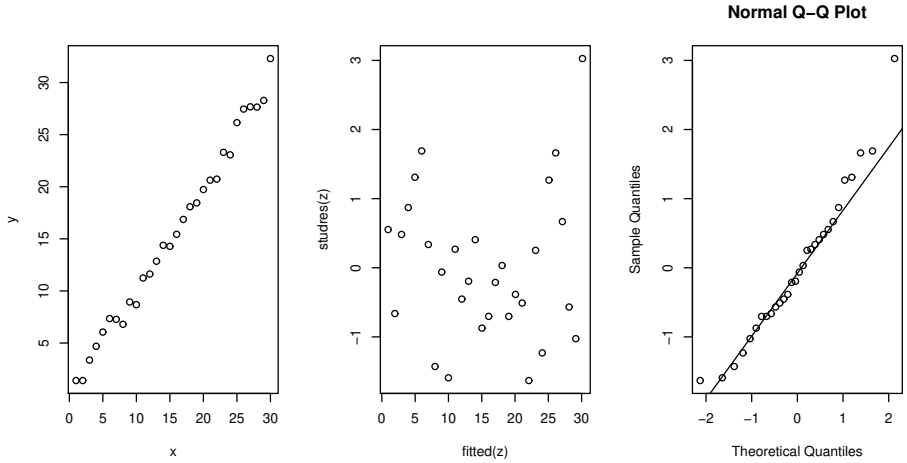
```
> plot(fitted(z),studres(z))
> qqnorm(studres(z))
> qqline(studres(z))
> makepsfile = function() {
    ps.options(horizontal = F)
    ps.options(height=4.0, width=7.5)
    postscript("ch1.ps")
    par(mfrow =c(1,3))
    plot(x,y)
    plot(fitted(z),studres(z))
    qqnorm(studres(z))
    qqline(studres(z))
    dev.off()
    }
> makepsfile()
> sink() # close sink function
> rm(x,y)
> q()
```

The output is as follows.

```
    Call:
    lm(formula = y ~ x)
    Residuals:
    Min     1Q     Median     3Q     Max
    -1.3470 -0.5934 -0.1120 0.4434 2.1720
    Coefficients:
```

	Estimate	*Std. Error*	*t value*	$Pr(>\|t\|)$	
$(Intercept)$	-0.06299	0.32684	-0.193	0.849	—
x	1.00636	0.01841	54.663	$< 2e-16$	$***$

```
    Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
    Residual standard error: 0.8728 on 28 degrees of freedom
    Multiple R-squared: 0.9907, Adjusted R-squared: 0.9904
    F-statistic: 2988 on 1 and 28 DF, p-value: < 2.2e-16
```



Write a report using Tex (or LaTex).

Edit a file called report.tex (see example),

need a postscript file: ch1.ps (which is created by makepsfile in page 3),

attach relevant R outputs into your report (see *e.g.* report (report.tex) and hw-solution (report.pdf) in my website).

Some commands in the linex system:

tex report.tex (create report.dvi file) (or latex ..., or pdflatex ...)

xdvi report (view the file)

dvipdf report (create a pdf file)

dvips report -o report.ps (create a postscript file)

dvips −p 2 −l 3 report −*o* page2.ps

ps2pdf page2.ps (create a two-page pdf file)

pdf2ps report.pdf

ps2eps ch1.ps # change ps file to eps file, needed in Latex.

For each homework, send me **2 files** by email **(do not compress them)**:

1. junk.pdf — the formal report file (pdf file)

2. junk.tex – the Latex file or the Tex file preparing junk.pdf

You need to organize them so that they are readable.

A brief manual for LaTex is on my website: short-math-guide

A brief introduction of R is in given in Math 531

One can google the pdf file "An introduction to R".

In my website, report.pdf is a sample of homework using LaTex (report.tex). Mimic it in your homework.

Chapter 5. Univariate Statistics

5.1. Probability Distributions.

Let X be a random variable (rv).

Its cdf $F(t) = P\{X \le t\}$, **domain ?**

density function (df) $f(t) = \begin{cases} F'(t) & \text{if } X \text{ is continuous} \\ F(t) - F(t-) & \text{if } X \text{ is discrete,} \end{cases}$ **domain ?**

quartile $Q(u) = F^{-1}(u) = \min\{t : \ F(t) \ge u\}$ **domain ?**

survival function $S(t) = 1 - F(t)$.

Example 1. $X \sim$ Weibull distribution with cdf $F(x|\gamma,\tau) = 1 - \exp(-(x/\tau)^\gamma), \ x > 0$, $S(x|\gamma,\tau) = \exp(-(x/\tau)^\gamma \mathcal{I}(x > 0))$ and $E(X) = \int x f(x) dx = \tau\Gamma(1 + 1/\gamma))$

γ– shape, τ – scale,

pweibull(x,shape,scale) — $F(x)$,

qweibull(x,shape,scale) — $Q(x)$,

dweibull(x ,shape,scale) — $f(x)$,

rweibull(10 ,1 ,3) — 10 observations from Exp(3) with $E(X) = 3$.

Remark. The list of all distributions in R is given in Table 5.1.

Distributions	*R name*	*parameters*	$f(x;\theta)$
$beta$	$beta$	$shape1, shape2$	$\frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)},\ x \in (0,1)$
$uniform$	$unif$	min, max	$\frac{1}{b-a},\ x \in (a,b)$
$gamma$	$gamma$	$shape, scale$	$\frac{x^{\alpha-1}e^{-x/\beta}}{\Gamma(\alpha)\beta^{\alpha}},\ x,\ \alpha,\ \beta > 0$
$exponential$	exp	$rate$	$\rho e^{-\rho x},\ x > 0$
$chi-square$	$chisq$	df	
$Cauchy$	$cauchy$	$location, scale$	$\frac{1}{\pi(1+x^2)} \to \frac{1}{\beta} f(\frac{x-\alpha}{\beta})$
$binomial$	$binom$	$size, prob$	$\binom{n}{x} p^x (1-p)^{n-x},\ x \in \{0,1,...,n\}$
$negative\ binomial$	$nbinom$	$size, prob$	
$geometric$	$geom$	$prob$	$p(1-p)^x,\ x = 0,1,...$
$hypergeometric$	$hyper$	m, n, k	
$normal$	$norm$	$mean, sd$	
$log-normal$	$lnorm$	$meanlog, sdlog$	
F	f	$df1, df2$	
T	t	df	
$logistic$	$logis$	$location, scale$	
$Poisson$	$pois$	$lambda$	
$Weibull$	$weibull$	$shape, scale$	
$Wilcox$	$wilcox$	m, n	

Example 1 (continued).
R
> x=rweibull(100,1,5)
> round(x,2)
> mean(x)
Q: What do you expect to see ?
QQplot: quantile-quantile plot.

1. Given data X_i, $i = 1, ..., n$.
2. Order them as $X_{(1)} \le \cdots \le X_{(n)}$.
3. Plot $(X_{(i)}, F^{-1}(\tilde{F}(X_{(i)})))$, where $\tilde{F}$ is a step function, and
   $\tilde{F}(X_{(i)}) = \frac{i}{n}$ (ecdf), or $\frac{i-\frac{1}{2}}{n}$ (ppoints(x)), or $\frac{i}{n+1}$.
   Since $\tilde{F}(t) \to F(t)$ w.p.1, we expect the qqplot is roughly a straight line.

Remark. If the assumption $X_i \sim F$ is correct
(and thus $\tilde{F} \approx F$ in the ideal situation),
then qqplot is plotting (X_i, X_i), $i = 1, ..., n$, as $F^{-1}(F(X_i)) = X_i$.
Thus the qqplot is expected to be a straight line roughly.

Example 2. Given $X_1, ..., X_{100}$, 100 observations in the file data_ex2,
Estimate F and $P\{X \in (1,2]\}$ and $E(X)$.
It is desirable to do parametric analysis, say assume that they are from a Weibull distribution. $F(x|\gamma,\tau) = 1 - \exp(-(x/\tau)^{\gamma}),\ x > 0$

Solution: We first find the MLE of (γ, τ), that is,
a value of (γ, τ) that maximizes the joint density function

$\mathcal{L}(\gamma, \tau) = \prod_{i=1}^{n} f(X_i|\gamma, \tau)$, where $f(t) = F'(t)$, $t > 0$.

Carry out data analysis using **R codes:**

```
x=matrix(scan("data_ex2"), ncol=1, byrow=T)
summary(x)
y=fitdistr(x,"weibull") # compute MLE
```

Remark. Distributions "beta", "cauchy", "chi-squared", "exponential", "gamma", "geometric", "log-normal", "lognormal", "logistic", "negative binomial", "normal", "Poisson", "t" and "weibull" are recognized.

```
y
summary(y)
pweibull(2,y$e[1],y$e[2])-pweibull(1,y$e[1],y$e[2])        # P(X ∈ (1,2])
(y$e[2])*gamma(1+1/y$e[1])        # E(X) = ∫ xf(x)dx = τΓ(1 + 1/γ))
```

Output:

```
> summary(x)
      V1
 Min. :1.030
 1st Qu.:1.840
 Median :3.000
 Mean :2.992
 3rd Qu.:4.070
 Max. :4.970
> y
    shape             scale
 2.7761986    3.3746473
 (0.2257762) (0.1280903)
> summary(y)                    different from summary(lm() and summary(x))
            Length   Class      Mode
estimate      2      -none-     numeric
sd            2      -none-     numeric
vcov          4      -none-     numeric
loglik        1      -none-     numeric   #loglikelihood
n             1      -none-     numeric
```

Question: What is the use of summary(y) here ?

```
> y$estimate
    shape scale
 2.776199 3.374647
> y$e
    shape scale
 2.776199 3.374647
> y$v # y$vcov
              shape       scale
shape  0.050974887  0.009118663
scale  0.009118663  0.016407135
> pweibull(2, 2.776,3.3746)-pweibull(1, 2.776,3.3746))        # P(X ∈ (1,2])
    [1] 0.1750694
> pweibull(2,y$e[1],y$e[2])-pweibull(1,y$e[1],y$e[2])
```

[1] 0.1750694

> ((y$e[2])*gamma(1+1/y$e[1])) # E(X)

3.003995

Ans to Ex. 2: The MLEs under the Weibull model are $\hat{\tau} = 3.4$ with $\hat{\sigma}_{\hat{\tau}} = 0.13$ and $\hat{\gamma} = 2.8$ with $\hat{\sigma}_{\hat{\gamma}} = 0.23$.

$\hat{F}(t) = 1 - \exp(-(t/3.4)^{2.8})$, $t > 0$ and $\hat{P}(X \in (1, 2]) \approx 0.175$.

$\hat{E}(X) = \hat{\tau}\Gamma(1 + 1/\hat{\gamma}) \approx 3.004$ versus $\overline{X} = 2.992$.

Question:

1. Can the model be simplified ?

   e.g., $X \sim Exp(1)$? ($\tau = 1$ and $\gamma = 1$ as $F(x) = 1 - e^{-(\frac{x}{\tau})^{\gamma}}$, $x > 0$).

   If the model is valid, then it can be shown that the MLEs $\hat{\gamma}$ and $\hat{\tau}$ have approximately normal distributions, $N(\gamma, \hat{\sigma}^2_{\hat{\gamma}})$ and $N(\tau, \hat{\sigma}^2_{\hat{\tau}})$.

   H_0: $\gamma = 1$ v.s. H_1: $\gamma \neq 1$. Check $|\hat{\gamma} - 1| < 2\hat{\sigma}_{\hat{\gamma}}$?

   H_0: $\tau = 1$ v.s. H_1: $\tau \neq 1$. Check $|\hat{\tau} - 1| < 2\hat{\sigma}_{\hat{\tau}}$?

   Ans: It seems that the model cannot be simplified as Exp(1) **Why ?**

2. $\hat{E}(X) = \hat{\tau}\Gamma(1 + 1/\hat{\gamma}) \approx 3.004$ is the MLE of $E(X)$ and $\overline{X} = 2.992$ is the non-parametric estimator of $E(X)$ ($\overline{X} = \sum_i x\hat{f}(x)$, where $\hat{f}(x) = \sum_{i=1}^n \mathbf{1}(X_i = x)/n$ is the density of the edf $\hat{F}(x) = \sum_{i=1}^n \mathbf{1}(X_i \leq x)/n$, a non-parametric MLE (NPMLE) of $F_X(t)$. **Which is better ?**

3. Is the model assumption valid ?

   We can use the qqplot, confidence band (CB) of the edf and ks.test to check.

Remark. The ks.test assumes the parameter are true one, not estimates.

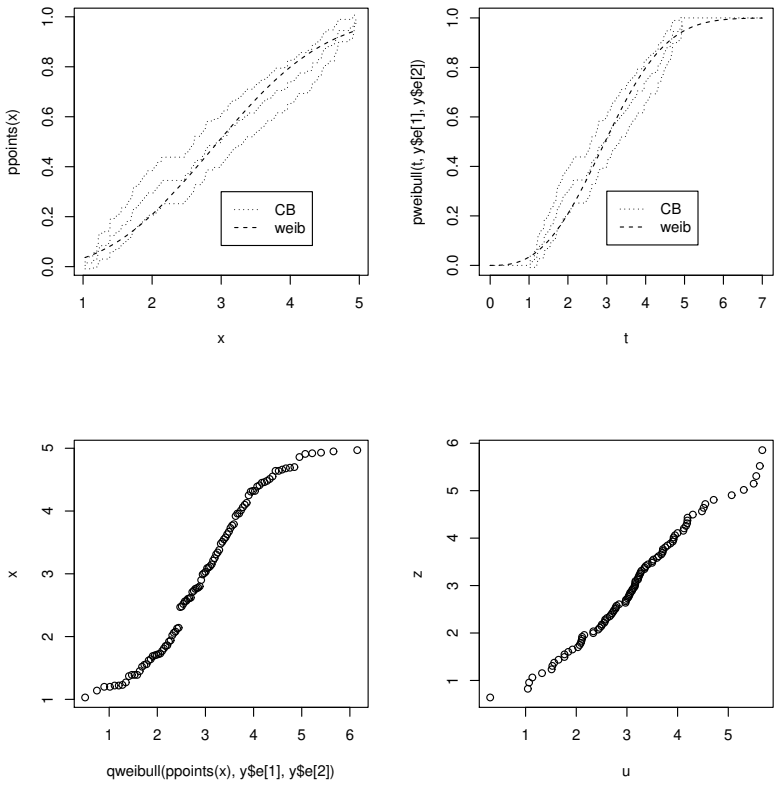
CB of the edf is the pointwise confidence interval based on the edf.

Example of qqplot and CB codes:

```
makepsfile = function() {
ps.options(horizontal = F)
ps.options(height=8.0, width=7.5)
postscript("ch1.2.ps")
par(mfrow =c(2,2))
x=sort(x)
plot(x,pweibull(x,y$e[1],y$e[2]),type="l",lty=2)
lines(x,ppoints(x),type="S",lty=3)
s=1.96*sqrt(ppoints(x)*(1-ppoints(x))/100)
lines(x,ppoints(x)+s,type="S",lty=3)
lines(x,ppoints(x)-s,type="S",lty=3)
leg.names=c("CB", "weib")
legend(3, 0.3, leg.names, lty=c(3,2),cex=1.0)
t=(0:70)/10
plot(t,pweibull(t,y$e[1],y$e[2]),type="l",lty=2)
lines(x,ppoints(x),type="S",lty=3)
s=1.96*sqrt(ppoints(x)*(1-ppoints(x))/100)
lines(x,ppoints(x)+s,type="S",lty=3)
lines(x,ppoints(x)-s,type="S",lty=3)
lines(c(0,1),c(0,0),type="l",lty=3)
lines(c(5,7),c(1,1),type="l",lty=3)
```

```
leg.names=c("CB", "weib")
legend(3, 0.3, leg.names, lty=c(3,2),cex=1.0)
u=rweibull(100,y$e[1],y$e[2])
plot(qweibull(ppoints(x),y$e[1],y$e[2]),x) # or qqplot(u,x) compare weibull to data
z=qweibull((1:100)/101,y$e[1],y$e[2])
qqplot(u,z) # compare to qqplot weibull v.s. weibull
dev.off()
}
makepsfile()
```

The output is in Figure 5.1 below.



$Fig(1,1)$: *similar to* $Fig(1,2)$: *cdf of weibull v.s. edf with its CB*
$Fig(2,1)$: *qqplot data vs weibull* $Fig(2,2)$: *qqplot Weibull vs Weibull*

Figure 5.1.

```
> pweibull(1,y$e[1],y$e[2]) +1-pweibull(5,y$e[1],y$e[2])
    [1] 0.08444813 # Difference in the two CB's
```

It seems from Figure 5.1 that the Weibull assumption is not valid, as

(1) No observation in $U_o = (0,1] \cup [5,\infty)$, but $P(X \in U_o) \approx 0.084$ under Weibull.

(2) qqplot does not look linear.

qqplot is quite subjective. One may consider a test, *e.g.*, ks.test in R.

Kolmogorov-Smirnov Goodness-of-Fit Test

Performs a one or two sample Kolmogorov -Smirnov test, which tests the relationship

between two distributions.
One-sample. Suppose that X_1, ..., X_n are a random sample from F.
ks.test(x, "pweibull", shape, scale)
ks.test(x, "pweibull", shape, scale)$p
ks.test(x, distribution = "punif", min(x),max(x), alternative = "gr")$p
ks.test(x, distribution = "punif", min(x),max(x), alternative = "less")$p
Remark. The ks-test statistic is one sided test $\mathbf{1}(D > c)$, where
$D = \sup\{|\tilde{F}(t) - F_o(t)| : t \in R\}$. **(Can we use a two-sided test $\mathbf{1}(D \notin [c_1, c_2])$?**

> ks.test(x, "pweibull", y$e[1], y$e[2], alternative="less")
One-sample Kolmogorov-Smirnov test
data: x
D^ = 0.08086, p-value = 0.2704
alternative hypothesis: the CDF of x lies below the null hypothesis

> ks.test(x, "pweibull", y$e[1], y$e[2], alternative="gr")
One-sample Kolmogorov-Smirnov test
data: x
$D\hat{+} = 0.096506$, p-value = 0.1553
alternative hypothesis: the CDF of x lies above the null hypothesis

> ks.test(x, "pweibull", y$e[1], y$e[2])
One-sample Kolmogorov-Smirnov test
data: x
D = 0.096506, p-value = 0.3094 $\approx 2 \times 0.1553$
alternative hypothesis: two-sided
H_0: $F = F_o$ a Weibull distribution(shape,scale), versus
H_1: $F \neq F_o$, where F_o is given (together with the parameter).
Remark. P-value= $P\{D > D_o\}$ (for alternative="gr" given in R),
where D_o is the observed value of D for the given X_1, ..., X_n.
We reject H_0: $F = F_o(\cdot|\theta)$ assuming θ is known if P-value is small (< 0.05). θ **= ?** For Section 5.1, alternative="gr" is not included in the codes, as it is not noticed that the program modified for "gr" and "less".
> ks.test(x, "pweibull", y$e[1],y$e[2])
One-sample Kolmogorov-Smirnov test
data: x
D = 0.0965, p-value = 0.3094
alternative hypothesis: two-sided
Question: What is our conclusion about the test ?
Does it agree with qqplot ?
Remark. In ks.test, the P-value is determined under these two assumptions:
(1) θ is the true value and
(2) the sample size n is very large.
However, θ is estimated by its MLE here and n is not **very** large, the P-value is not true.
One can find the critical value in D by empirical quantiles of 0.05 for a given sample size n.

(See the simulation studies in Examples 3, 4 and 5.)

Example 3. Generate data from U(1,5) with $n = 100$ or 1000.
Test against Weibull, Uniform and Uniform(1,5) with ks(). **Why uniform of U(1,5) ?**
Question: What is the difference between the last two tests ?
Summarize the findings.

We expect to reject Weibull and accept Uniform and Uniform(1,5).
How to find the MLE of the parameter θ for:

Weibull: θ **? true value of θ ?**

Uniform: θ **? true value of θ ?**

```
> (y=fitdistr(x,"unif"))
        Error in fitdistr(x, "unif") : unsupported distribution
```

MLE of U(a,b) ?

Uniform(1,5): $\theta = ?$

```
> fun3 = function(n) {
    x=runif(n,1,5)
    y=fitdistr(x,"weibull")
    a=ks.test(x, "pweibull", y$e[1], y$e[2])
    b=ks.test(x, "punif", min(x),max(x))
    c=ks.test(x, "punif", 1, 5)
    return(c(u=a$p.value, v=b$p, w=c$p)) }
> fun3(100) # What is the output ? What do you expect ?
```

u	v	w	
0.4267747	0.7190210	0.6058055	Are they expected ?
?	?	?	**Is it possible ?**

Repeat 1000 times with $n = 100$:

```
m=1000
u=rep(0,m)
v=rep(0,m)
w=rep(0,m)
for(i in 1:m) {
    z=fun3(100)
    u[i]=as.numeric(z[1]<0.05)
    v[i]=as.numeric(z[2]<0.05)
    w[i]=as.numeric(z[3]<0.05)
}
mean(u)
[1] 0.013  # (Power or size of the test φ ? Or an estimate ?)
```

$$\underbrace{\text{(Power or size of the test } \phi \text{ ? Or an estimate ?)}}_{E(\phi(\mathbf{U}))\ or\ \ \mathrm{P}(H_1|H_o)}$$

```
mean(v)
[1] 0.043  # (Power or size of the test ? Or an estimate ?)
mean(w)
[1] 0.044  # (Power or size of the test ? Or an estimate ?)
n=1000 # Repeat but with larger sample size n size
u=rep(0,m)
v=rep(0,m)
```

```
w=rep(0,m)
for(i in 1:m) {
z=fun3(n)
u[i]=as.numeric(z[1]<0.05)
v[i]=as.numeric(z[2]<0.05)
w[i]=as.numeric(z[3]<0.05)
}
>c(mean(u), mean(v), mean(w))
[1] 1    0.05    0.053   # Are they expected ?
```

Summary: Uniform(1,5) data test for

	$Weibull$	why	$Uniform$		$Uniform(1,5)$
n	$P(H_0\|H_1)$		$P(H_1\|H_0)$	$P(H_0\|H_1)$	$P(H_1\|H_0)$
100	≈ 0.987	**?**	≈ 0.043?	0.957?	0.044? *or* 0.05?
1000	0?	?	≈ 0.05?	0.95?	0.053? *or* 0.05??
∞	0		≈ 0.05	1	0.05

Findings:

1. If n is very large, then it seems that ks.test works for U(a,b).
2. OW, $P(H_0|H_1)$ can be 99%, instead of $< 50\%$, this explains the discrepancy in Ex. 2.
3. If n is moderate, the level of the ks.test seems fine for U(a,b). **why ?**

Remark. The P-value given in ks.test is an approximation when n is very large and the assumption is correct. Otherwise, it can be arbitrary, either > 0.5 or < 0.5.

Review on 502.

Case 1. For testing H_o: $\mu = \mu_o$ v.s. H_1: $\mu > \mu_o$ under $N(\mu, 1)$ with size $\alpha = 0.05$.
A test is $\phi = \mathbf{1}(\overline{X} - \mu_o > 1.64/\sqrt{n})$. **What does it mean ?** H_o v.s. H_a or H_0 v.s. H_1.

$\beta(\mu) = E(\phi|\mu)$ – the power function of the test.

$\beta(\mu)$ is the power of the test ϕ if $\mu > \mu_o$.

$\alpha = \sup_{\mu \le \mu_o} \beta(\mu) = 0.05$ is the size of the test.

ϕ is a level 0.1 test, but not a level 0.01 test.

If a random sample is from $N(\mu_o + 1, 1)$, the size of ϕ is 0.05 and the power of the test is $\beta(\mu_o + 1)$. $P(H_1|H_o) = 0.05$ and $P(H_o|H_1) = 1 - \beta(\mu_o + 1)$.

If a random sample is from $N(\mu_o - 1, 1)$, the size of ϕ is 0.05.

If a random sample is from $N(\mu_o, 2)$, the size of ϕ is not 0.05 and $P(H_o|H_1)$ is not relevant, as the assumption for ϕ is not valid.

Homework 5.1.1. In Case 1 above, what is the probability of type I or II error for $\mu = \mu_o \pm 2$, and the size of the test ϕ ?

Case 2. For testing H_o: $p = 0.5$ v.s. H_1: $p \neq 0.5$, assuming data satisfy that $X_1, ..., X_n$ are i.i.d. from $bin(1, p)$ with size $\alpha = 0.05$. First assume $n = 2$.

A test is $\psi = 0.1 \times \mathbf{1}(\overline{X} \neq 0.5)$. **What does it mean ?**

$\beta(p) = E(\psi(\overline{X})) = 0.1(q^2 + p^2)$ is the power function of the test.

$\beta(0.5) = 0.1P(\overline{X} \neq 0.5) = 0.1 \times 0.5 = 0.05$ is the size of the test.

If X_i's are from $bin(1, 0.4)$, $\beta(0.4)$ is the power of the test and the size of the test is 0.05.
If X_i's are from $N(1, 4)$, the size of the test ψ is no longer 0.05, but can be computed. **What is it ??**

Homework 5.1.2. Derive the most powerful test ψ_2 of size 0.05 for Case 2 if $X_1, ..., X_{10}$ are i.i.d. from bin(1,p) **(instead of bin(n,p))**. What is the probability of type I or II error for $p \in \{0.3, 0.9\}$, and the size of the test ψ_2 ?

Hint: Check 447 [21].

Example 4. Generate data from Weibull(1,0.2) with $n = 100$ or 1000.

Test against Weibull and Weibull(1,0.2). Summarize the findings.

R codes:

```
>fun3 = function(n) {
    x=rexp(n,5)                    # Why not rexp(n,0.2) ?
    y=fitdistr(x,"weibull")
    a=ks.test(x, "pweibull", y$e[1], y$e[2])      true value of (γ, τ) ?
    c=ks.test(x, "pweibull", 1, 0.2)                 (y$e[1], y$e[2]) = (1, 0.2) ?
    return(c(u=a$p.value, w=c$p))
}
>n=100
>fun3(n)
```

output:

u	w	**Are they what you expect ?**
0.4647952	0.5927737	

Repeat 1000 times again.

```
> m=1000
> u=rep(0,m)
> w=rep(0,m)
> for(i in 1:m) {
    z=fun3(n)
    u[i]=as.numeric(z[1]<0.05)
    w[i]=as.numeric(z[2]<0.05)
    }
> c(mean(u) , mean(w))
    [1] 0 0.045
```

What happens if n is larger ? (It was $n = 100$ above).

```
> n=1000
> u=rep(0,m)
> w=rep(0,m)
> for(i in 1:m) {
    z=fun3(n)
    u[i]=as.numeric(z[1]<0.05)
    w[i]=as.numeric(z[2]<0.05)
    }
> c(mean(u) , mean(w))
    [1] 0 0.046
```

Remark. The above code can be revised by R code apply(), which is faster.

Summary: Weibull(1,0.2) data test for

	$Weibull$	$Weibull(1, 0.2)$	
n	$P(H_1\|H_0)$	$P(H_1\|H_0)$	
100	0?	0.045?	*or* 0.05?
1000	≈ 0	≈ 0.046?	
∞	$\approx$ 0	0.05	

Finding: $\hat{P}(H_1|H_0) = 0$ if Weibull data test Weibull with (τ, γ) replaced by the MLE.

It is too small or the critical value for size 0.05 is too large.

Thus the ks.test does not work under Weibull if the parameters are not true values.

Notice that for a test ϕ, if $P(H_1|H_0) = 0$ then it is often $P(H_0|H_1) =$ **??**

(If one always accepts H_o, it means H_1 is always rejected).

Examples 3 and 4 suggest that ks.test is not reliable for $n = 100$.

Example 2 (continued). Examples 3 and 4 suggest that one needs to modify the ks.test for the case θ being replaced by the MLE.

The test statistic is $D = \sup_t |\tilde{F}(t) - F_o(t)|$ if $H_1 : \tilde{F}(t) \neq F_o(t)$.

How to find the empirical critical value of size 0.05 for ks.test ?

Ans: A modified ks.test:

```
> x=matrix(scan("data_ex2"), ncol=1, byrow=T)
> y=fitdistr(x,"weibull"))
> b=ks.test(x, "pweibull", y$e[1], y$e[2])$s                    # What is b ?
```

Ans: (see the next command).

```
> summary(b)
```

	Length	Class	Mode
statistic	1	−none−	numeric
p.value	1	−none−	numeric
alternative	1	−none−	character
method	1	−none−	character
data.name	1	−none−	character

```
>u=rep(0,1000)
> for (i in 1:1000){
    x=rweibull(100, y$e[1], y$e[2])
    z=fitdistr(x,"weibull")
    a=ks.test(x, "pweibull", z$e[1], z$e[2])   # v.s. a=ks.test(data, "pweibull",z$e[1],z$e[2])
    u[i]=a$s                    # or u[i]=a$p
    }
> sort(u)[950]                  # or sort(u)[c(25,975)] anything wrong ?
    [1] 0.08622978                  # what is this ?
> b                   # b= ks.test(x, "pweibull", y$e[1], y$e[2])$s
    [1] 0.09650574 # D_o= 0.09650574
```

Q: Can we have conclusion now ?

```
> mean(u>b) # what is this ?
```

[1] 0.024 # the p-value for the ks.test H_o: data in Example 2 are from Weibull distribution **verse 0.3094 using ks.test directly**

Thus we are quite sure to reject H_o.

What is the reasoning of this approach ?

1. First derive the test statistic value b from the data.
2. Pretend the true θ =MLE to generate pseudo random numbers.
3 Repeat the ks.test m times with the same n and unknown θ.
4. It results i.i.d. ks.test statistic value D_i, $i = 1, ..., m$
5. SLLN ($\overline{\mathbf{1}(D > b)} \to P(D > b)$ **anything wrong with the statement ??**).

What is conclusion for testing H_0: the data in Ex. 2 are from Weibull distribution ?

Question: Ideally, if we reject H_0 when ks.test\$$p < 0.05$, the size of the test is **??**
How to find a "ks.test\$<**??** for a size 0.05 for the data in Ex. 2 ?
Ans:

```
x=matrix(scan("data_ex2"), ncol=1, byrow=T)
y=fitdistr(x,"weibull"))
for (i in 1:10000){
    x=rweibull(100, y$e[1], y$e[2])
    z=fitdistr(x,"weibull")
    a=ks.test(x, "pweibull", z$e[1], z$e[2])
    u[i]=as.numeric(a$p<0.43) # try to increase from 0.05 to achieve mean(u)≈ 0.05
}
mean(u)
[1] 0.0494 (≈ 0.05)
```

If replacing the last command

```
u[i]=as.numeric(a$p<0.43)
```

by

```
u[i]=as.numeric(a$p<0.05)
```

The output is

```
mean(u)=0.00
```

Comment: The above approach is not as direct and simple as the modified ks.test specified around page 13 with the R codes for the critical value for ks-statistic.

Recall what we have discussed so far:

Usual steps for data analysis:

1. Input data to a computer software, say R.
2. Assume a proper probability model, say, a parametric model,
   or a semiparametric model,
   or a non-parametric model.
3. Compute an estimate of parameter if it is parametric,
   or an estimate of parameter and F if it is semi-parametric,
   or an estimate of F, if it is non-parametric.
4. Check whether the model assumption is valid.
5. If No, go to Step 2, otherwise, carry out the other statistics inferences, e.g., testing, or confidence intervals, *etc.*

Example 2. Given X_1, ..., X_{100}, 100 observations in the file data_ex2,
Estimate F and $P\{X \in (1, 2]\}$ and $E(X)$.

Sol. It is desirable to do parameteric analysis, say, Weibull distribution.

Solution to the MLE of (γ, τ):

The MLEs under the Weibull model are $\hat{\tau} = 3.4$ with $\hat{\sigma}_{\hat{\tau}} = 0.13$
and $\hat{\gamma} = 2.8$ with $\hat{\sigma}_{\hat{\gamma}} = 0.23$.
$\hat{F}(t) = 1 - \exp(-(t/3.4)^{2.8})$, $t > 0$ and $\hat{P}(X \in (1, 2]) \approx 0.175$.
$\hat{E}(X) = \hat{\tau}\Gamma(1 + 1/\hat{\gamma}) \approx 3.004$ versus $\overline{X} = 2.992$.

Question: Step 4. Is the model assumption valid ?
We can use the qqplot, confidence band (CB) of the edf and ks.test to check.

The qqplots are in Figure 5.1.

It seems from Figure 5.1 that the Weibull assumption is not valid, as

(1) No observation in $U_o = (0, 1] \cup [5, \infty)$, but $P(X \in U_o) \approx 0.084$ under Weibull.

(2) qqplot does not look linear.

qqplot is quite subjective. One may consider a test, *e.g.*, ks.test in R for

H_o: the data are from Weibull or $\hat{F}(t) \approx F_o(t)$, the cdf of Weibull.

```
> ks.test(x, "pweibull", y$e[1],y$e[2])
```

D = 0.0965, p-value = 0.3094

Question: What is our conclusion about the test ?

The ks.test does not agree with qqplot !

Remark. In ks.test, the P-value is determined under these two assumptions:

(1) the parameter θ is the true value and

(2) the sample size n is very large.

However, θ is estimated by its MLE here and n is not large, the P-value is not true.

One can find the critical value in D by empirical quantiles of 0.05 for a given sample size n. (as explained in the simulation exercises in Examples 3 and 4.)

Example 2 (continued). Examples 3 and 4 suggest that one needs to modify the ks.test for the case θ is replaced by the MLE.

The test statistic is $D = \sup_t |\tilde{F}(t) - F_o(t)|$ if $H_1 : \tilde{F}(t) \neq F_o(t)$.

How to find the empirical critical value of size 0.05 for ks.test:

Recall in Example 4, the codes are

```
> u=rep(0,1000)
> for (i in 1:1000) {
    x=rexp(100,5)
    y=fitdistr(x,"weibull"))
    u[i]=ks.test(x, "pweibull", y$e[1], y$e[2])$p
    }
> mean(u)        # This is a wrong P̂(H1|H0) (≈ 0 here), not ≈ 0.05 in general.
    [1] 0
```

How to find the empirical critical value of size 0.05 for ks.test:

A modified ks.test:

```
x=matrix(scan("data_ex2"), ncol=1, byrow=T)
y=fitdistr(x,"weibull"))
b=ks.test(x, "pweibull", y$e[1], y$e[2])$s
for (i in 1:1000){
        x=rweibull(100, y$e[1], y$e[2])
        z=fitdistr(x,"weibull")
        a=ks.test(x, "pweibull", z$e[1], z$e[2])
        u[i]=a$s                    # or u[i]=a$p
    }
> sort(u)[950]
    [1] 0.08622978
> b
    [1] 0.09650574 # D_o= 0.09650574
```

Q: Can we have conclusion now ?

> mean(u>b)

what is this ?
[1] 0.024 # the p-value for the ks.test H_o: data in Example 2 are from Weibull distribution,
verse 0.3094 using ks.test directly

What is conclusion for testing H_0: the data are from Weibull distribution ?

Weibull is not an appropriate assumption.

It coincides with qqplot in Ex. 2.

We end here last lecture.

Since the ks.test and qqplots suggest that the data are not from a Weibull distribution. Then there are two choices:

1. empirical distribution function (edf),

2. other parametric distributions.

1. Use the edf to estimate F, $\hat{F}(t) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}(X_i \le t)$.
R codes:

mean(x)

sum((x>1& x<=2))/length(x) mean((x>1& x<=2))

Outcomes: $\hat{\mu} = \overline{X} = 2.99$ and $\hat{P}(X \in (1, 2]) = 0.29$
2. Try other parametric cdf's,

Notice that in the program fitdistr(), distributions "beta", "cauchy", "chisq", "exp", "f", "gamma", "geom", "lnormal", "logis", "nbinom", binom", "norm", "pois", "t" and "weibull" are recognized.
Which of them are inappropriate ? Notice that $(X_{(1)}, X_{(n)}) \subset (1, 5)$.

beta ? cauchy ?

geom ? nbinom ? binom ? pois ?

Only try gamma, uniform, normal as follows.

par(mfrow =c(1,3))

y=fitdistr(x,"gamma")

n=length(x)

s=(1:n)/(n+1) # or s=(1:n)/n, or s=ppoint(sort(x))

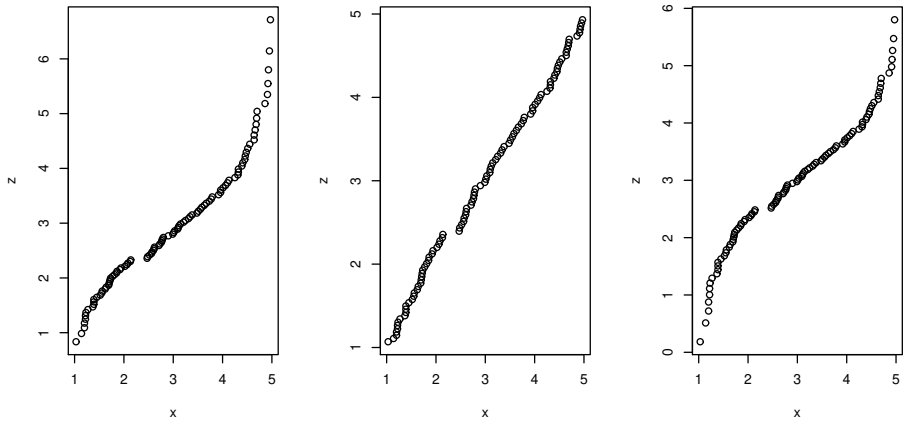
z=qgamma(s,y$e[1],y$e[2]) # or z=rgamma(n,y$e[1],y$e[2])

qqplot(x,z)

z=qunif(s,min(x),max(x))

qqplot(x,z)

qqnorm(x)



In view of the qqplots, we may test whether the data are from a uniform distribution,
> ks.test(x, distribution = "punif", min(x),max(x))

data: x

D = 0.99, p-value = 0.2864

alternative hypothesis: two-sided

Do we need to modify ks.test ?

Example 4 suggests that if $X \sim$ Weibull, the ks.test needs to be modified for all n.

Homework 5.1.3. Do the modified ks.test anyway and find out the p-value for the data data_ex2 and compare to p-value 0.2864 above.

data_ex2:

1.39 3.63 2.06 4.25 2.76 4.64 1.85 1.72 1.37 1.64 4.01 3.25 1.14 4.70 4.69
1.73 2.57 3.96 3.12 3.55 1.77 1.62 2.02 1.39 4.93 2.14 1.52 2.80 3.67 3.01
4.95 1.45 4.41 4.06 3.09 2.08 3.51 4.92 4.48 4.97 4.51 4.45 3.21 4.68 1.71
1.39 4.32 1.86 4.64 3.15 2.13 4.39 1.56 2.61 2.71 4.66 3.48 3.38 1.20 2.90
1.94 2.99 3.10 2.52 2.60 2.77 2.56 1.03 4.91 1.23 1.22 3.96 1.81 1.92 1.69
2.62 2.48 2.73 3.31 3.79 4.86 4.46 1.22 3.92 3.77 1.20 2.47 3.03 1.27 3.58
2.78 4.13 4.31 4.55 3.73 3.34 4.10 1.70 4.32 1.55

Actually Example 3 suggests that if $X \sim U(a,b)$, no need to modify for $n \geq 100$.

Can we assume $X \sim U(a,b)$?

$$F(t) = \begin{cases} \frac{t-a}{b-a} & \text{if } t \in (a,b), \\ 1 & \text{if } t \geq b. \end{cases}$$

then the MLE is $(\hat{a}, \hat{b}) = (\min_i X_i, \max_i X_i) = (1.03, 4.97)$,

as it maximizes the likelihood function $\mathcal{L}(a,b) = \prod_{i=1}^n \frac{1}{b-a}\mathbf{1}(X_i \in (a,b))$.

R codes:

(max(x)+min(x))/2

punif(2,min(x),max(x))-punif(1,min(x),max(x))

Or assume $X \sim U(1,5)$ based on ks.test(x,"unif", 1,5).

Final solution:

$\hat{F}$ is $U(1,5)$.

$\tilde{\mu} = 3$ and

$\tilde{P}(X \in (1,2]) = 0.25$

Comment 5.1.: Various estimates of $P(X \in (1,2]) = 0.25$ and their SE's are as follows.

1. edf=> $\hat{P}(X \in (1,2]) = 0.29$, with SE $\sqrt{\hat{P}(1-\hat{P})/n} \approx 0.045$.
   and with CI $[0.20, 0.38]$ **difference between SE and SD ?**)
2. U(a,b) (assuming $(a,b) \supset (1,2)$) => $Z = \tilde{P}(X \in (1,2]) = (2-1)/(\hat{b}-\hat{a}) \approx 0.254$ $SE_Z \approx$? **HW**)

   Hint: $(\hat{a}, \hat{b}) = (X_{(1)} \wedge 1, X_{(n)} \vee 2)$, $\sigma_Z^2 =$? $f_{X_{(1)},X_{(n)}}(x,y) =$??
3. U(a,b) => $Z = \tilde{P}(X \in (1,2]) = \frac{(2 \wedge \hat{b} - 1 \vee \hat{a}) \mathbf{1}_{(X_{(1)},X_{(n)}) \cap (1,2) \neq \emptyset)}}{\hat{b}-\hat{a}}$, ($SE_Z \approx$? **HW**)

   Hint: $(\hat{a}, \hat{b}) = (X_{(1)}, X_{(n)})$, $\sigma_Z^2 =$? $f_{X_{(1)},X_{(n)}}(x,y) =$??
4. U(1,5) => $P(X \in (1,2]) = 0.25$, SD = ?
5. Weibull MLE=> $\check{P}(X \in (1,2]) = 0.18$, differ≈ half due to wrong assumption.

Question: # of parameters using the EDF ? (non-parametric model)

# of parameters using the uniform distribution ? (parametric model)

Both models are correct, but there are more parameters in the edf.

Homework 5.1.4. Answer all the question marks (?) in Comment 5.1.

Example 5. (simulation study). Generate 100 data from Exp(1/2) with mean 1/2. Now pretend that we do not know the underlying distribution of the data. Assume Weibull distribution. Estimate F and $P\{X \in (1,2]\}$ and $E(X)$.

Sol. **Simulation data:**

> x=rexp(100,2)

> mean(x)

[1] 0.5153382 # **rate =2 or scale=2 ?**

Now pretend we assume but do not really know the true distribution is

$F(t) = 1 - \exp(-(t/\tau)^\gamma)$, $t > 0$.

The MLE is computed:

>fitdistr(x,"weibull")

shape	*scale*
1.16690389	0.54517822
(0.08866768)	(0.04934344)

We may test

H_o: $\gamma = 1$ v.s. H_1: $\gamma \neq 1$,

or

H_o: $\tau = 1$ v.s. H_1: $\tau \neq 1$.

That is, we check whether the data is from Exp(μ) or further Exp(1).

If $X \sim Weibull(\gamma, \tau)$, $\hat{\mu} = \hat{\tau}\Gamma(1 + 1/\hat{\gamma})$ with 2 parameters and SE by Delta method;

Note: $\Gamma(\alpha) = \int_0^\infty t^{\alpha-1}e^{-t}dt$, $\Gamma'(\alpha) = \int_0^\infty (\ln t) t^{\alpha-1}e^{-t}dt$ can be computed numerically,

If $X \sim Exp(\mu)$, $\hat{\mu} = \overline{X}$ with 1 parameter and SE= $\hat{\sigma}_X/\sqrt{n}$=?

If $X \sim Exp(1)$, $\hat{\mu} = 1$ with no parameter and SE=?

Conclusion ?

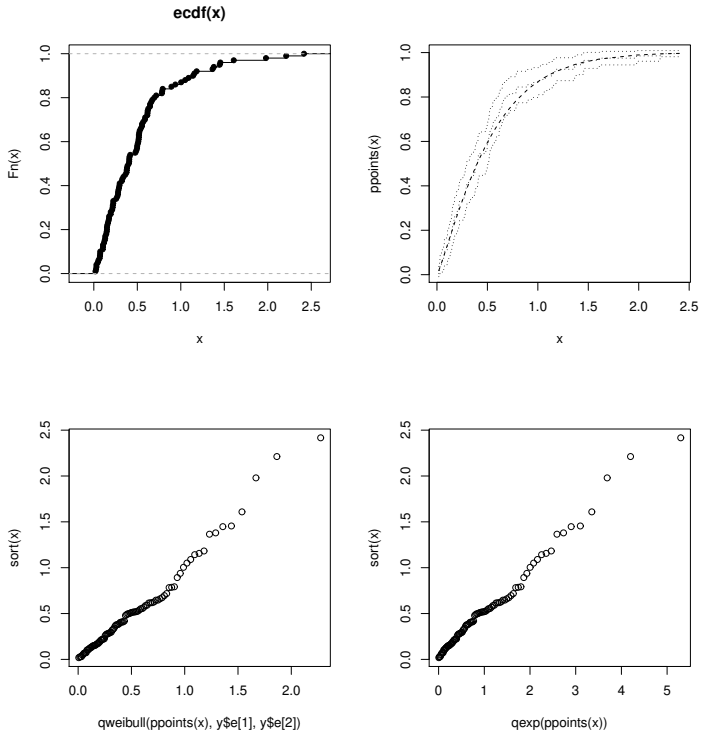
$\hat{\mu}_X = 0.55$ or $\check{\mu}_X = 0.52$? **pretend we do not know the truth.**

$\hat{\sigma}_{\hat{\mu}_X} =$? $\hat{\sigma}_{\check{\mu}_X} = 0.52/10$ **why ?** **which is smaller ?** **why ?**

$\hat{F}(t) = 1 - e^{-t/0.52}$, $t > 0$.

$\hat{P}(X \in (1,2)) = e^{-1/0.52} - e^{-2/0.52}$.

Done ?
The qqplots (see Figure 2) appear linear.



Fig(1,1): empirical cdf, Fig(1,2): cdf of weibull v.s. CB
Fig(2,1): qqplot weibull, Fig(2,2): qqplot Exp(2).

Figure 2. CB plot and QQplot in Example 5

It supports that the data are from the Weibull model or Exponential model.

> ks.test(x, "pexp", 1/mean(x)) # **Do we need to test weibull or others ?**

One-sample Kolmogorov-Smirnov test
data: x
D = 0.079181, p-value = 0.5575

Done ?

```
> m=1000
> u=rep(0,m)
> n=100
> b=ks.test(x, "pexp", 1/mean(x))$s
> for (i in 1:m){
    z=rexp(n, 1/mean(x))
    u[i]=ks.test(z, "pexp", 1/mean(z))$s }
> sort(u)[950]                    # estimated critical value for 5%
    [1] 0.1068376                    # v.s. ??
> sum((u>b))/length(u)
    [1] 0.322                   # estimated P-value
```

Q: Is it possible that the simulation study suggests that the data do not fit the Weibull model ?

So far, we consider examples of data analysis on non-regression data. Next, consider an example of regression data.

Example 6. Use Prostate data to estimate $E(Y|X=4)$, where (Y,X) = (lpsa,lweight).
Sol. This is a regression data. There are 3 approaches:

(1) parametric, (2) semi-parametric, (3) non-parametric.

First try to fit the linear regression model $y=\alpha+\beta x+\epsilon$, where $\epsilon\sim N(0,\sigma^2)$.
Is it approach (1) ? or (2) ? or (3) ?

```
>library(MASS)
>library(faraway)
>prostate[96:98, ]
```

	lcavol	*lweight*	*age*	*lbph*	*svi*	*lcp*	*gleason*	*pgg*45	*lpsa*
96	2.882564	3.7739	68	1.558145	1	1.55814	7	80	5.47751
97	3.471967	3.9750	68	0.438255	1	2.90417	7	20	5.58293

```
>(y=lm(lpsa~lweight,data=prostate))
> summary(y)
```

	Estimate	*Std. Error*	*t value*	$Pr(>\|t\|)$
(*Intercept*)	−0.5281	0.8220	−0.642	0.522120
lweight	0.8231	0.2230	3.691	0.000373

The model is $y=-0.5281+0.8231x+\epsilon$, where $\epsilon\sim N(0,\sigma^2)$.

```
>lm(lpsa~lweight-1,data=prostate)$co
```

why do this ?

0.6811

The updated model is $y=0.68x+\epsilon$, where $\epsilon\sim N(0,\sigma^2)$. **Done ?**
We need to check whether

(1) whether the model $Y=\alpha+\beta x+\epsilon$ is valid;

(2) whether $\epsilon\sim N(0,\sigma^2)$.

```
> x=y$resid
> sd(x)
```

[1] 1.079527

```
> ks.test(x, "pnorm", 0,1)                    # or ks.test(x, "pnorm", 0,1.08)
```

D = 0.05809, p-value = 0.8798

modified ks.test:

```
length(prostate[,2]) # [1] 97
u=rep(0,1000)
t=ks.test(x,"pnorm",0,s)$p                    $p instead of $s here, equivalently
for (i in 1:1000) {
z=rnorm(97,0,s)
x=lm(z~1)$resid
v=sd(x)
u[i]=ks.test(z,"pnorm",0,v)$p }
mean(u<t)
```

[1] 0.694 # v.s. t= 0.8798 above

What is the suggestion from the ks.test ?

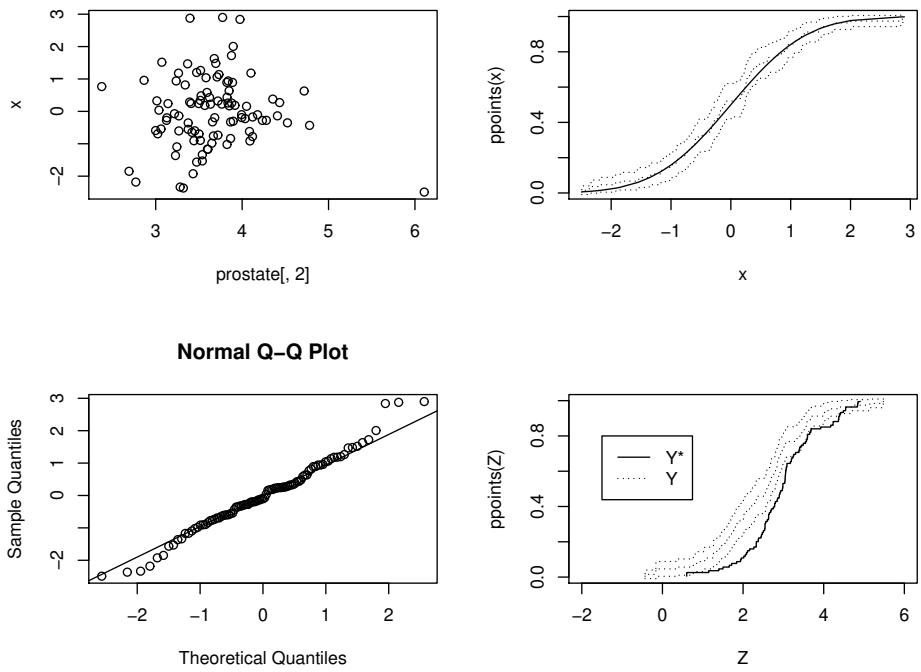


Figure 3. *scater plot of residuals* *CB of residual* *QQ − plot of residuals* *MD plot (of $F_Y(z)$)*

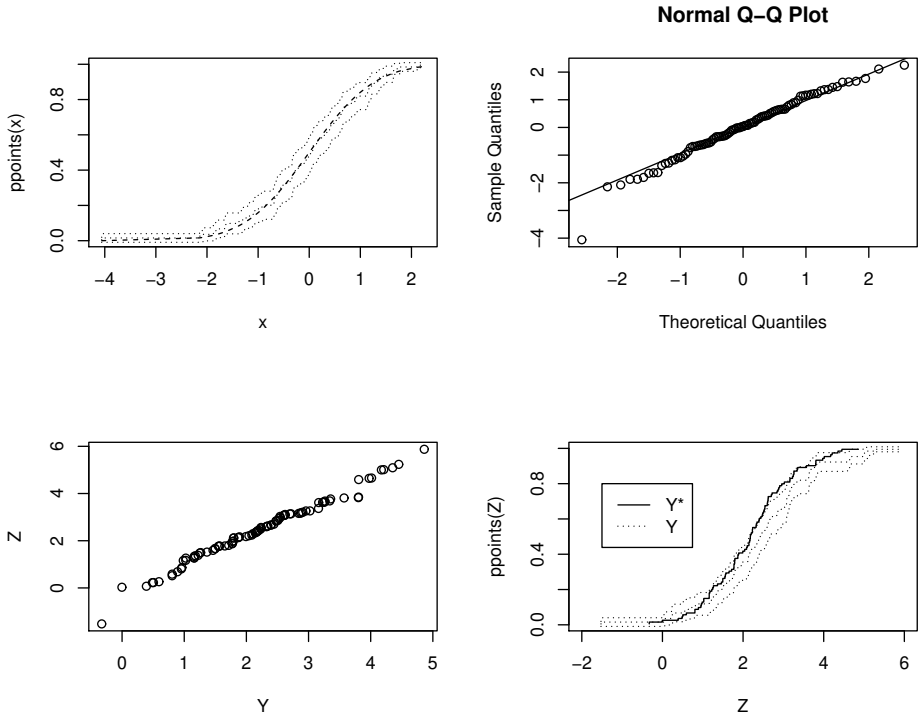


Figure 4. *CB of residuals* *QQ − plot of residuals* *qqplot(Y, Z)* *MD plot*

For regression data, instead of plotting the MLE of F_Y and the CB of the EDF of F_Y, or

the CB of $\hat{\epsilon}_i$'s vs. the MLE of the cdf of the normal distribution (see Figure 3), the MD plot is more appropriate.
The MD plot is the plots of the edf $\hat{F}_Y(t)$ and the edf of $F_Z(t)$ where $Z_i = \hat{\beta}X_i + \hat{\alpha} + \epsilon_i$, under the given assumptions *i.e.*, $\epsilon \sim N(\mu, \sigma^2)$ and $X \perp \epsilon$. Notice that the residuals $\hat{\epsilon}_i$'s depend on X_i's.

What does Figure 3 tell ?

For comparison, using the MLE of the parameter derive above, we generate simulated data and give a similar plots in Figure 4 (compare the MD plots).

CB is plot $(\hat{F}(t) - 2SE, \hat{F}(t), \hat{F}(t) + 2SE)$, $t \in (0, \infty)$.

$P(\hat{F}(t) - 2SE < F(t) < \hat{F}(t) + 2SE) << 0.95$, $t \in (0, \infty)$.

Notice that

if $P(A) = P(B) = 0.9$ and $A \perp B$ then $P(A)P(B) = ?$

The codes for Figures 3 and 4 are given as follows.

```
library(faraway)
x=prostate[,2]
Z=prostate[,9]                    Z=Y
y=lm(lpsa~lweight,data=prostate)                    Z = Y = βx + α + ϵ
(N=length(Z))
# Figure 3:
sort(x)
[1] 2.3749 2.6912 2.7695 2.8651 2.9982 3.0131 3.0229 3.0374 3.0611 3.0704
[11] 3.1246 3.1290 3.1420 3.2169 3.2288 3.2367 3.2445 3.2638 3.2677 3.2677
[21] 3.2828 3.3196 3.3411 3.3690 3.3759 3.3759 3.3928 3.3962 3.4095 3.4194
[31] 3.4324 3.4420 3.4516 3.4735 3.4735 3.4980 3.5010 3.5160 3.5160 3.5190
[41] 3.5249 3.5395 3.5395 3.5681 3.5821 3.5932 3.5987 3.6041 3.6230 3.6336
[51] 3.6494 3.6571 3.6674 3.6776 3.6826 3.6951 3.7099 3.7197 3.7197 3.7317
[61] 3.7647 3.7739 3.8232 3.8254 3.8254 3.8362 3.8384 3.8491 3.8512 3.8660 mode
[71] 3.8660 3.8764 3.8785 3.8888 3.8969 3.8969 3.9170 3.9750 3.9936 3.9954 near 3.85
[81] 4.0351 4.0509 4.0851 4.0902 4.1018 4.1198 4.1215 4.1782 4.2348 4.2801
[91] 4.3548 4.4085 4.4338 4.5245 4.7181 4.7804 6.1076
z=Z[x>3.8 & x<3.9]
w= sample(z, N, replace = TRUE)
X=x−3.85
```

Remark 5.1.1. codes for Figure 4 are the same hereafter.

```
X=sample(X, N, replace=T)
Y=X*y$co[2]+w #Y*                    Y* = β̂X + W, E(W) ≠ 0.
x=y$resid
ps.options(horizontal = F)
ps.options(height=6.0, width=7.5)
postscript("try.ps")
par(mfrow =c(2,2))
plot(prostate[,2],x)                 # scater plot of residuals vs. covariate
x=sort(x)
plot(x,ppoints(x),type="S",lty=3)          Next 5 lines is
```

```
s=1.96*sqrt(ppoints(x)*(1-ppoints(x))/N)      CB plot for residuals against normal dist.
lines(x,ppoints(x)+s,type="S",lty=3)
lines(x,ppoints(x)-s,type="S",lty=3)
lines(x,pnorm(x,0,1),lty=1)
qqnorm(x)
qqline(x)
Z=sort(Z)
plot(Z,ppoints(Z),type="S",xlim=c(-2,6),lty=3)      EDF of Z (which is response "Y")
s=1.96*sqrt(ppoints(Z)*(1-ppoints(Z))/N)
lines(Z,ppoints(Z)+s,type="S",lty=3)                  CB of response data
lines(Z,ppoints(Z)-s,type="S",lty=3)
Y=sort(Y)
lines(Y,ppoints(Y),type="S",lty=1)                    EDF of Y*.
leg.names=c("Y*", "Y")
legend(-1.5, 0.8, leg.names, lty=c(1,3),cex=1.0)
# Figure 4:
x=c(rnorm(N-20,3.85,1),rep(3.85,20))          # N x's, with 20 at 3.85.
Z=x*y$co[2]+y$co[1]+rnorm(N)
y=lm(Z~ x)
z=Z[x==3.85]
w= sample(z, N, replace = TRUE)
```

Hereafter, the codes are the same as Remark 5.1.1.

```
X=x-3.85 ....
```

Remark. The idea of the MD plot will be discussed later on.

The previous MD plot codes is to implement following steps:

b1. Suppose that the data fit the model $Y_i = \beta X_i + W_i$, where W_i's are i.i.d. $\sim N(\mu, \sigma^2)$. Moreover, X_i's are dense around x_o. Let $m = ||\{X_i : |X_i - x_o| < \delta_n\}||$ for some $\delta_n > 0$. By making a transformation $X_i^* = X_i - x_o$ and $W_i^* = W_i + \beta x_o$, WLOG, we can assume $x_o = 0$.

b2. Obtain $\hat{\beta}$, the LSE of β based on $(\mathbf{X}_i, Y_i)$'s.

b3. Take a random sample of size m from the $\mathbf{X}_i$'s in a neighborhood of x_o (before $X_i - x_o$), *i.e.*, $|X_i - x_o| \le \delta_n$, or a neighborhood of $\mathbf{0}$ (after $X_i - x_o$), where m and δ_n are as in (b1), take another random sample of size $n - m$ from the $\mathbf{X}_i$'s satisfying $|X_i - x_o| > \delta_n$, and take a random sample of size n from Y_i's satisfying $|X_i - x_o| \le \delta_n$.
It yields a sample of (X, W)'s, say (X_1^*, W_i^*), $i = 1, ..., n$.
(Note that $Y = \beta X + W = W$ if $X = 0$.)

b4. Let $Y_i^* = \hat{\beta} X_i^* + W_i^*$.

Since the parametric linear regression (LR) model is not appropriate, one can use the semiparametric LR model or the edf to estimate $E(lpsa|lweight \approx 4]$ **How ?**

Remark. The test statistic is one sided test $\mathbf{1}(D > c)$, where
$D = \sup\{|\tilde{F}(t) - F_o(t)| : t \in R\}$. **(can we use a two-sided test $\mathbf{1}(D \notin [c_1, c_2])$?**
However, it seems that it somewhat is modified to either one-sided or two-sided as seen from the outputs below. Thus we can use default anyway.

```
> ks.test(x, "pweibull", y$e[1], y$e[2], alternative="less")
```

```
        One-sample Kolmogorov-Smirnov test
        data: x
        D^- = 0.08086, p-value = 0.2704
        alternative hypothesis: the CDF of x lies below the null hypothesis

> ks.test(x, "pweibull", y$e[1], y$e[2], alternative="gr")
        One-sample Kolmogorov-Smirnov test
        data: x
        D^+ = 0.096506, p-value = 0.1553
        alternative hypothesis: the CDF of x lies above the null hypothesis

> ks.test(x, "pweibull", y$e[1], y$e[2])
        One-sample Kolmogorov-Smirnov test
        data: x
        D = 0.096506, p-value = 0.3094
        alternative hypothesis: two-sided
        H_0: F = F_o a Weibull distribution(shape,scale), versus
        H_1: F ≠ F_o, where F_o is given (together with the parameter).
> n=10000
> x=rexp(n,2)
> m=10000
> mean(x) [1] 0.4917744
> u=rep(0,m)
> v=rep(0,m)
> w=rep(0,m)
> for (i in 1:m){
        z=rexp(n, 2)
        u[i]=as.numeric(ks.test(z, "pexp", 2, alternative="less")$p<0.05)
        v[i]=as.numeric(ks.test(z, "pexp", 2, alternative="gr")$p<0.05)
        w[i]=as.numeric(ks.test(z, "pexp", 2)$p<0.05)
        }
> mean(u)
        [1] 0.0476≈ 0.05                    # p-value using true parameters.
> mean(v)
        [1] 0.0548≈ 0.05
> mean(w)
        [1] 0.0542≈ 0.05

> for (i in 1:m){
        z=rexp(n, 2)
        u[i]=as.numeric(ks.test(z, "pexp", 1/mean(z), alternative="less")$p<0.05)
        v[i]=as.numeric(ks.test(z, "pexp", 1/mean(z), alternative="gr")$p<0.05)
        w[i]=as.numeric(ks.test(z, "pexp", 1/mean(z))$p<0.05)
}
> mean(u)
```

[1] 0.0086 # rarely reject H_0

> mean(v)

[1] 0.0104 # p-value using MLE for parameters.

> mean(w)

[1] 0.0054

```
> for (i in 1:m){
    z=rexp(n, 2)
    u[i]=as.numeric(ks.test(z, "pexp", 1/mean(z), alternative="less")$s)
    v[i]=as.numeric(ks.test(z, "pexp", 1/mean(z), alternative="gr")$s)
    w[i]=as.numeric(ks.test(z, "pexp", 1/mean(z))$s)
    }
> sort(u)[9500]
```

[1] 0.03069644 estimated critical value of D for testing size of 5%

$D = \sup_t\{|\hat{F}(t) - F_o(t)|\} >$ sort(v)[9500]

[1] 0.03171528

> sort(w)[9500]

[1] 0.03460668

Section 5.2. Tests on means.

t.test, wilcox.test, binom.test.

1. t.test: (based on normal assumption). Performs a one-sample, two-sample, or paired t-test, or a Welch modified two-sample t-test.

t.test(x, y=NULL, alternative=c("two.sided", "less", "greater"), mu=0, paired=F, var.equal=F, conf.level=.95)

2. wilcox.test: (nonparametric)

Computes Wilcoxon rank sum test for two sample data (equivalent to the Mann-Whitney test) or the Wilcoxon signed rank test for paired or one sample data.

wilcox.test(x, y=NULL, alternative="two.sided", mu=0, paired=F, exact=T, correct=T, conf.level=.95)

Remark. Use ?wilcox.test to find out more information.

3. binom.test: (binomial distribution)

Test hypotheses about the parameter p in a binomial(n,p) model given x, the number of successes out of n trials.

binom.test(x, n, p=0.5, alternative="two.sided") (x is transformed).

One sample, H_0: $\mu = \mu_0$, v.s. H_1: $\mu \neq \mu_0$ (or $>$, or $<$).

Two-sample, H_0: $\mu_X - \mu_Y = \mu_0$, v.s. H_1: $\mu_X - \mu_Y \neq \mu_0$ (or $>$, or $<$).

Remark. These tests all assume i.i.d.

binom.test assumes binomial distribution.

wilcox.test is a non-parametric test, assuming symmetric distribution;

The small-sample t.test is a parameter inference, making use of $N(\mu, \sigma^2)$;

The large-sample t-test can ignore the normal assumption, though it needs finite σ_X or σ_Y.

Section 5.2.1. One sample.

t.test.

Assumption:

The random sample size is large $n > 30$, otherwise, $X_1, ..., X_n$ are i.i.d. from $N(\mu, \sigma^2)$.
Test statistic $T = \frac{\overline{X}-\mu_0}{S/\sqrt{n}}$.

wilcox.test:

Assumptions: X_i's are i.i.d. from a symmetric distribution.
Rank $X_i - \mu$'s by their *absolute* values.
Let S_n (S_p) be the sum of negative (positive) ranks.
Let $S = |S_n| \wedge |S_p|$.
The Wilcoxon sign rank test statistic is $Z = \frac{S+\frac{1}{2}-\frac{1}{4}n(n+1)}{\sqrt{n(n+1)(2n+1)/24}}$

Example. Observations: $3, 1, 7$, H_o: $\mu = 4$. $S_n = ?$ $S_p = ?$ $S = ?$

binom.test.

Assumption: X_i's are i.i.d..
Test statistics is $Z = \sum_{i=1}^{n} \mathbf{1}(X_i > \mu)$.

Remark: If n is large, t.test is very close to z.test by CLT on $\overline{X}$ under the assumption:
X_1, ..., X_n are i.i.d., provided $\sigma_X < \infty$.

Steps in one-sample test on mean μ:

1. Input data;

2. qqnorm, CB plot or modified ks.test to check normality;

3. Check i.i.d. assumption;

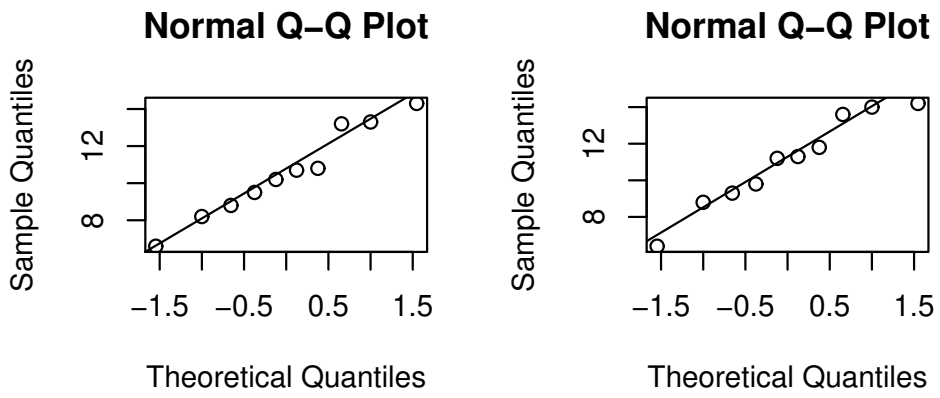
4. If $X \sim N(\mu, \sigma^2)$ then t.test;

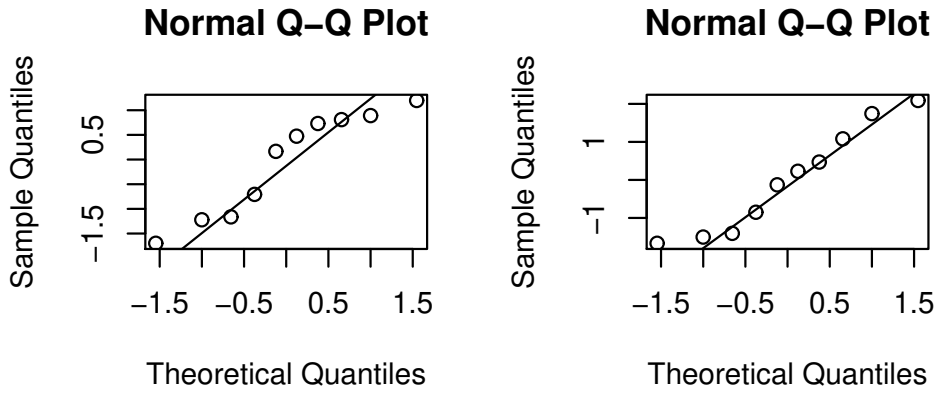
5. O.W. use hist() or stem() to check symmetry;

6. If it looks like symmetric, use wilcox.test.

7. O.W. let $Z = \sum_{i=1}^{n} \mathbf{1}(X_i > \mu)$, binom.test(Z,n,0.5)

Example 1. Data on shoe wear (10 pairs).
shoe=list(A=c(13.2, 8.2, 10.2, 14.3, 10.7, 6.6, 9.5, 10.8, 8.8, 13.3),
B=c(14.0, 8.8, 11.2, 14.2, 11.8, 6.4, 9.8, 11.3, 9.3, 13.6))
Mean for data A = 10 ?





$qqplot(A)$ $qqplot(B)$
$qqplot(rnorm(10))$ $qqplot(rnorm(10))$ **(why 10 ?)**

It seems from qqplot that the normal assumption is OK.

> library(coin) for checking i.i.d. ?

> (x=A[2*(1:5)])

[1] 8.2 14.3 6.6 10.8 13.3 **where are they from ?**

> (y=A[-2*(1:5)])

[1] 13.2 10.2 10.7 9.5 8.8

> independence_test(x∼y)

Z = -1.1982, p-value = 0.2308 **what is the conclusion ?**

The P-value of the ks.test is not accurate as $n = 10$ is small. One needs to estimate it by modifying ks.test, but we skip it here.

> (z=t.test(A,mu=10))

t = 0.722, df = 9, p-value = 0.4886

95 percent confidence interval:

8.805406 12.314594

mean of x

10.56

> z$c # c=conf.int from z

[1] 8.805406 12.314594
attr(,"conf.level")
[1] 0.95

Conclusion:

For testing H_0: $\mu = 10$ v.s. H_1: $\mu \neq 10$ P-value > 0.4. Do not reject H_0.

Mean = 10

> stem(A) # **Do we need to do this ?**

06 | 6
08 | 285
10 | 278
12 | 23
14 | 3

The decimal point is at the "|".

What can we conclude ?

> wilcox.test(A,mu=10)

V = 33, p-value = 0.625

> y=sum(A>10) # **Do we need to do this ? Why not** < 10 **?**

> binom.test(y,10,0.5)

number of successes = 6, number of trials = 10, p-value = 0.7539

Comments: For this data set, 3 tests are valid, and they do not reject H_o.

But it is more appropriate and better to use the t.test. **Why ?**

5.2.2. Two-sample.

Data: X_1, ..., X_n, Y_1, ..., Y_m.

H_0: $\mu_X - \mu_Y = \mu_0$, v.s. H_1: $\mu_X - \mu_Y \neq \mu_0$

If both sample-sizes are very large a Z-test

$$\phi = \begin{cases} \mathbf{1}(\frac{|\overline{X}-\overline{Y}|}{\sqrt{S_X^2/n+S_Y^2/m}} > z_{\alpha/2}) & \text{if two samples are independent,} \\ t.test(x-y) & \text{if two samples are paired.} \end{cases}$$

Steps if n and m are small or moderate :

1. Check i.i.d.
2. Check normal assumptions by qqnorm or ks.test. Use t.test if normal, o.w. use wilcox.test.
3. Determine independence by data feature (*e.g.* $n \neq m$? paired ?) or use cor.test. If dependent, use one-sample test with $Z_i = X_i - Y_i$. Otherwise, go on.
4. If normal, check whether $\sigma_X = \sigma_Y$ by var.test.

Questions:

X and Y are uncorrelated => $X \perp Y$?

X and Y are uncorrelated <= $X \perp Y$?

X and Y are correlated => $X \not\perp Y$?

X and Y are correlated <= $X \not\perp Y$?

t.test.

Test statistic $T = (\overline{X} - \overline{Y} - \mu_0)/\hat{\sigma}$, where

$\hat{\sigma}$ is an estimate of σ, depending on the assumption.

Possible assumptions:

1. $X_i \sim N(\mu_X, \sigma_X^2)$ and $Y_i \sim N(\mu_Y, \sigma_Y^2)$,
2. $\sigma_X = \sigma_Y$?
3. Are two samples dependent ?

cor.test.

cor.test(x,y,method="pearson", "kendall","spearman")

Given (X_i, Y_i), $i = 1$, ..., n, test for correlation ρ (**= ?**).

"pearson" test statistics:

$$T = \sqrt{n-2} * R/\sqrt{1-R^2} \qquad (\ T \sim t_{n-2} \text{ if } (X,Y)) \sim N(\mu, \Sigma))$$

where $R = S_{xy}/\sqrt{S_{xx}S_{yy}}$.

"kendall" test statistics:

$$\tau = \frac{n_c - n_d}{n(n-1)/2}$$

where $n_c = \sum_{i<j} \mathbf{1}((Y_i - Y_j)(X_i - X_j) > 0)$, the number of concordant

(i.e., numbers of $b = \frac{Y_i - Y_j}{X_i - X_j} > 0$ or #'s of 2 points with form: $\begin{matrix} * & (X_i, Y_i) \\ * & (X_j, Y_j) \end{matrix}$),

and $n_d = \sum_{i<j} \mathbf{1}((Y_i - Y_j)(X_i - X_j) < 0)$, the number of discordant

(i.e., numbers of $b = \frac{Y_i - Y_j}{X_i - X_j} < 0$ or **2 points has the form ??** $\begin{matrix} (X_i, Y_i) \\ (X_j, Y_j) \end{matrix}$).

Critical values for testing Kendall's tau is tabulated.

"spearman" test statistics:

$$\hat{\rho} = \frac{S_{rs}}{S_r S_s} = \frac{\sum_i r_i s_i - C}{\sqrt{\sum_i r_i^2 - C}\sqrt{\sum_i s_i^2 - C}}$$

where $C = n(n+1)^2/4$,

$r_i = rank$ of x_i among x_j's and

$s_i = rank$ of y_i among y_j's.

Critical values for testing Spearman's rho is tabulated.

Steps:

1. Input data,
2. qqnorm and qqline on X_is and Y_is separately,
3. If normal assumption is valid use pearson,

otherwise, use kendal or spearman. (**Are they related to t.test or wilcox.test ?**)

var.test

Performs an F test to compare variances of two independent samples from $N(\mu_i, \sigma_i^2)$'s.

var.test(x, y, alternative="two.sided", conf.level=.95)

H_0: $\sigma_X = \sigma_Y$.

Test statistics $F = \sqrt{S_X^2/S_Y^2}$

wilcox.test. Wilcoxon Rank Sum Tests for testing two means.

Data: X_1, ..., X_n, Y_1, ..., Y_m.

Assumptions: The X_i's and Y_j's are independent samples

H_o: $F_Y(t) = F_X(t - \mu)$

The test statistic is $W = \sum_{j=1}^{m} R_{n+j}$,

where R_{n+j} =rank(Y_j) among $X_i - \mu$'s and Y_j's.

Example 1 (continued). Data on shoe wear (10 pairs).
Which of them are appropriate ?

cor.test(A,B,alternative="two.sided",method="pearson")
var.test(A,B)
t.test(A,B,pair=T)
t.test(A,B)
t.test(A,B, alternative="two.sided", paired=F, var.equal=T)
wilcox.test(A,B)
wilcox.test(A-B)

Applying tests to this data set yields output as follows.
> cor.test(A,B,alternative="two.sided",method="pearson")

t = 16.50071, df = 8, p-value = 1.831e-07
95 percent confidence interval:
0.9383049 0.9967172
cor
0.9856358

Are A and B correlated ?
Do we need to check i.i.d. based on the above result ? # If A and B are correlated, check i.i.d. as follows.
> library(coin)
> rbinom(1,0.5) $0 \to$ A; $1 \to$ B.

[1] 1

> (x=B[2*(1:5)])
> (y=B[-2*(1:5)])
> independence_test(x∼y) # or cor.test(x,y)
> var.test(A,B)

F = 0.9485, num df = 9, denom df = 9, p-value = 0.9385
95 percent confidence interval:
0.2355932 3.8186432
ratio of variances
0.948497

Q: $\sigma_A^2 = \sigma_B^2$ **?** Yes, No, DNK.
Do we need to check it ?
> t.test(A,B)

t = -0.4318, df = 17.987, p-value = 0.671
95 percent confidence interval:
-2.815702 1.855702
mean of x mean of y
10.56 11.04

Does it suggest $\mu_A = \mu_B$ **?** Yes, No, DNK.
> t.test(A,B,var.equal=T)

t = -0.4318, df = 18, p-value = 0.671 (compare to no "var.equal=T") df =17.987
95 percent confidence interval:
-2.815585 1.855585
mean of x mean of y

10.56 11.04

Does it suggest $\mu_A = \mu_B$? Yes, No, DNK.

> t.test(A,B,pair=T)

t = -3.5602, df = 9, p-value = 0.006118

95 percent confidence interval:

-0.7849953 -0.1750047

mean of the differences

-0.48

Does it suggest $\mu_A = \mu_B$? Yes, No, DNK.

Which t.test result should be used ?

> wilcox.test(A,B)

W = 42.5, p-value = 0.5966

> wilcox.test(A,B,pair=T)

V = 3, p-value = 0.01437

Conclusion:

It seems from qqplot that the normal assumption is OK.

Do we know that the normal assumption is indeed true ?

cor.test gives $\rho = 0.98$ and P-value 0.00. In fact, we knew that X and Y are paired.

Thus the var.test is not valid, even though

it seems from var.test that the variances are equal (P-value= 0.94).

If we use correct test (paired t.test), P-value is 0.006 and

we reject H_0. That is, there is a difference in mean.

If we use the incorrect test (two sample test), P-value is 0.67

and we do not reject H_0.

The paired Wicoxon test gives P-value 0.014, which is not as significant as the paired t.test.

Remark. In two sample problems, if $n = m$ and i.i.d. then paired tests are valid, just not not as powerful as the unpaired ones.

Example 2 (a simulation study).

Generate two independent samples from N(0,1) and N(0,25).

Test for equal means.

```
x=rnorm(10)
y=rnorm(10,0,5)
qqnorm(x)
qqline(x)
qqnorm(y)
qqline(y)                    # H_o ? Expect to reject H_o ? Yes, No, DNK
cor.test(x,y,method="pearson")          # expect to reject H_o ? Yes, No, DNK
var.test(x,y)                  # expect to reject H_o ? Yes, No, DNK
t.test(x,y,pair=T)                  # expect to reject H_o ? Yes, No, DNK
t.test(x,y)                  # expect to reject H_o ? Yes, No, DNK
t.test(x, y, alternative="two.sided", paired=F, var.equal=T) # What do you expect ?
```

Outputs:

cor.test(x,y,method="pearson") **Pearson's product-moment correlation**

t = 1.8239, df = 8, p-value = 0.1056

95 percent confidence interval:

-0.1331101 0.8735067

cor

0.5419356 # **what is the real correlation ?**

What's your conclusion ? Is it what you expected ?

What can you say based on the CI ?

var.test(x,y) **F test to compare two variances**

F = 0.0227, num df = 9, denom df = 9, p-value = 4.522e-06

95 percent confidence interval:

0.005646828 0.091527345

ratio of variances

0.0227341

What's your conclusion ? Is it what you expected ?

t.test(x,y,pair=T) **Paired t-test**

t = 0.2158, df = 9, p-value = 0.834

t.test(x,y) **Welch Two Sample t-test**

t = 0.1978, df = 9.409, p-value = 0.8474

t.test(x, y, alternative="two.sided", paired=F, var.equal=T) **Two Sample t-test**

t = 0.1978, df = 18, p-value = 0.8454

Which of the 3 t.test can be used **based on outputs ?**

Which of the 3 t.test should be used based **on the true model ?**

Can we tell which test of the last 3 is more powerful from this simulation ?

Which of the 3 t.test and 2 wilcox.test is valid ?

Look at the following simulation results:

```
> m=200
> r=rep(0,5)
> for(i in 1:m){
    x=rnorm(n,0,5)
    y=rnorm(n)
    r[1]=r[1]+as.numeric(t.test(x,y,pair=T)$p.value<0.05)
    r[2]=r[2]+as.numeric(t.test(x,y)$p.value<0.05)
    r[3]=r[3]+as.numeric(t.test(x,y,var.equal=T)$p.value<0.05)
    r[4]=r[4]+as.numeric(wilcox.test(x-y)$p.value<0.05)
    r[5]=r[5]+as.numeric(wilcox.test(x,y)$p.value<0.05) }
> r/m
    [1] 0.045 0.050 0.055 0.045 0.050
```

Which of the 3 t.test and 2 wilcox.test is valid ?

```
> fun1=function(n){
    x=runif(10,0,10)
    y=x+rnorm(10)
    r[1]=r[1]+as.numeric(t.test(x,y,pair=T)$p.value<0.05)
    r[2]=r[2]+as.numeric(t.test(x,y)$p.value<0.05)
    r[3]=r[3]+as.numeric(t.test(x,y,var.equal=T)$p.value<0.05)
    r[4]=r[4]+as.numeric(wilcox.test(x-y)$p.value<0.05)
    r[5]=r[5]+as.numeric(wilcox.test(x,y)$p.value<0.05)
    return(r) }
```

> u=matrix(rep(0,m*n),m)
> s=apply(u,1,fun1)
> apply(s,1,mean)
[1] 0.05 0.00 0.00 0.05 0.00

Which of the 3 t.test and 2 wilcox.test is valid ?

What does Example 2 tell us ?

Example 3 (a simulation study).

Generate two independent samples from N(0,1) and N(2,9)

Test for equal means.

```
x=rnorm(10)
y=rnorm(10,2,3)
qqnorm(x)
qqline(x)
qqnorm(y)
qqline(y)
cor.test(x,y,alternative="two.sided",method="pearson")
var.test(x,y)
t.test(x,y,pair=T)          # expect to reject H_o ? Yes, No, DNK
t.test(x,y)           # expect to reject H_o ? Yes, No, DNK
t.test(x, y, alternative="two.sided", paired=F, var.equal=T)
          # expect to reject H_o ? Yes, No, DNK
wilcox.test(x,y)          # expect to reject H_o ? Yes, No, DNK
wilcox.test(x-y)          # expect to reject H_o ? Yes, No, DNK
```

Q: Which of the 7 tests is valid ? (*i.e.*, the distribution for the test statistic is valid).

Q: Which of the last 5 tests is more appropriate ?

> cor.test(x,y,alternative="two.sided",method="pearson")

t = -1.3413, df = 8, p-value = 0.2167
95 percent confidence interval:
-0.8332983 0.2754580
cor
-0.4284824 **What is the conclusion based on p-value or CI ?**

> var.test(x,y) **F test to compare two variances**

F = 0.2261, num df = 9, denom df = 9, p-value = 0.03726
95 percent confidence interval:
0.05615047 0.91012210
ratio of variances
0.2260615

> t.test(x,y,pair=T) **Paired t-test**

t = -2.0923, df = 9, p-value = 0.06594
95 percent confidence interval:
-3.1563001 0.1231082
mean of the differences
-1.516596

> t.test(x,y) **Welch Two Sample t-test**

t = -2.4151, df = 12.871, p-value = 0.03136

95 percent confidence interval:
-2.874619 -0.158573
mean of x mean of y
0.3581944 1.8747904

> t.test(x, y, alternative="two.sided", paired=F, var.equal=T) **Two Sample t-test**

t = -2.4151, df = 18, p-value = 0.02659
95 percent confidence interval:
-2.8359078 -0.1972842
mean of x mean of y
0.3581944 1.8747904

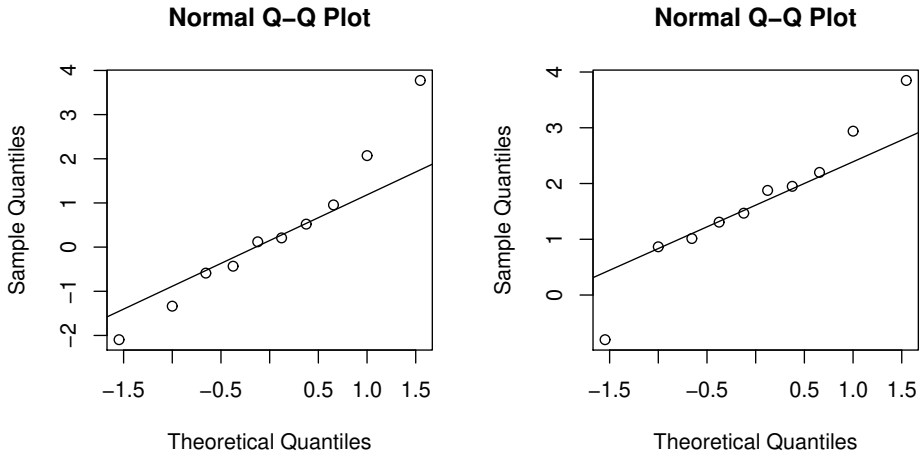
> wilcox.test(x,y) **Wilcoxon rank sum test**

W = 27, p-value = 0.08921

> wilcox.test(x-y) **Wilcoxon signed rank test**

data: x - y
V = 10, p-value = 0.08398

Which of the 3 t.test should be uses based **on outputs ?**
Which of the 3 t.test should be uses based **on the true model ?**
A replication of 1000 similar to Example 2 yield ratio of reject H_o:
[1] 0.417 0.441 0.479 0.404 0.436
Is the 3rd t.test more powerful here ?
Which test of the last 5 is more powerful from this simulation ?
Example 4 (a simulation study). Generate two independent samples. Test for equal means. **Ignore the data distribution for the moment.**
Do the two qqplots of data look like straight lines ?



What will you do if you are not sure ?
Can we use ks.test ?
It seems from qqplot that the normal assumption is not likely.

> cor.test(x,y,alternative="two.sided",method="kendall")

T = 15, p-value = 0.2164

Try: t.test(x,y,pair=T)
t.test(x,y)
t.test(x, y, alternative="two.sided", paired=F, var.equal=T)

wilcox.test(x,y)
wilcox.test(x-y)

Which of the previous tests is likely to be valid ?

Two possible answers:

a. DNK.

b. Based on QQ-plot, the last two.

Which of the previous tests is more appropriate ?

>t.test(x,y,pair=T) **Paired t-test**

t = -1.7041, df = 9, p-value = 0.1226

>t.test(x,y) **Welch Two Sample t-test**

t = -2.0313, df = 16.632, p-value = 0.05852

>t.test(x, y, alternative="two.sided", paired=F, var.equal=T) **Two Sample t-test**

t = -2.0313, df = 18, p-value = 0.05725

>wilcox.test(x,y) **Wilcoxon rank sum test**

W = 23, p-value = 0.04326

>wilcox.test(x-y) **Wilcoxon signed rank test**

data: x - y

V = 13, p-value = 0.1602

Conclusion ?

a. We correctly reject H_0: equal mean if use wilcox(x,y).

b. We incorrectly do not reject H_0: equal mean if use wilcox(x-y) or t.test.

Remark. The two samples are from double exponential+0, +2.

```
y=rexp(10)
z=c(-1,1)
u=sample(z,10,replace=T)
y=u*y+2
x=rexp(10)
z=c(-1,1)
x=sample(z,10,replace=T)*x
```

Example 5 (simulation study).

```
> n=10
> m=100
> r=rep(0,5)
> ptm < − proc.time()
> for(i in 1:m){
    x=runif(n,0,10)
    y=x+rnorm(n)
    r[1]=r[1]+as.numeric(t.test(x,y,pair=T)$p.value<0.05)
    r[2]=r[2]+as.numeric(t.test(x,y)$p.value<0.05)
    r[3]=r[3]+as.numeric(t.test(x,y,var.equal=T)$p.value<0.05)
    r[4]=r[4]+as.numeric(wilcox.test(x-y)$p.value<0.05)
    r[5]=r[5]+as.numeric(wilcox.test(x,y)$p.value<0.05) }
> r/m
[1] 0.06 0.00 0.00 0.05 0.00
> (proc.time()- ptm)[3]
```

r/m :	0.02	0.03	0.07	0.08	0.09	0.1	0.11	0.12
$2SD$:	0.03	0.03	0.05	0.05	0.06	0.06	0.06	0.06

```
    elapsed
    0.373
> ptm <- proc.time()
> fun1=function(n)
    x=runif(10,0,10)
    y=x+rnorm(10)
    r[1]=r[1]+as.numeric(t.test(x,y,pair=T)$p.value<0.05)
    r[2]=r[2]+as.numeric(t.test(x,y)$p.value<0.05)
    r[3]=r[3]+as.numeric(t.test(x,y,var.equal=T)$p.value<0.05)
    r[4]=r[4]+as.numeric(wilcox.test(x-y)$p.value<0.05)
    r[5]=r[5]+as.numeric(wilcox.test(x,y)$p.value<0.05)
    return(r) }
> u=matrix(rep(0,m*n),m)
> s=apply(u,1,fun1)
> apply(s,1,mean)/m
    [1] 0.05 0.00 0.00 0.05 0.00
> (proc.time()- ptm)[3]
    elapsed
    0.274                          # compare to 0.373 above
```

What does this simulation study tell us ?

Summary of Examples 2, 3, 4 and 5.

1. From Example 5, we can see that if model assumption is not satisfied by the data, the sizes of the 5 tests are wrong except for wilcox.test(x-y), and so are their p-value, thus the 4 tests are invalid.
2. In Example 2, H_0 is true, only t.test(x,y,var.equal=T) is invalid. P-values given for that t.test is wrong.
3. In Example 3, H_0 is false and only t.test(x,y, var.equal=T) is invalid. t.test(x,y) is more powerful than the other 3 valid tests.
4. In Example 4, H_0 is false and all 3 t.test are invalid. wilcox(x,y) is more powerful.

Remark. If the test is not valid then
the size α of the test is often not as claimed. Notice

If $\alpha = P(H_1|H_o) \uparrow$ then $P(H_o|H_1) \downarrow$.

If $\alpha = P(H_1|H_o) \downarrow$ then $P(H_o|H_1) \uparrow$.

$\alpha = P(H_1|H_o) = 1 - P(H_o|H_1)$???

Under given assumptions, if H_1 is true, α remains the same and

$\alpha = P(H_1|H_o) \neq 1 - P(H_o|H_1)$.

H_o is associated with the given assumptions. For example, H_o for equal variance t-test assumes $\mu_1 - \mu_2 = \mu_o$ and $\sigma_1 = \sigma_2$, not just $\mu_1 - \mu_2 = \mu_o$.

5.2.3 Tests on mean with multiple samples

Standard approach is the one-way anova: assuming

$$Y_{ij} = \alpha_i + \epsilon_{ij},\ \epsilon_{ij},\ i = 1, ..., t,\ j = 1, ..., n_i, \text{ are i.i.d. } \sim N(0, \sigma^2). \tag{5.2.3.1}$$

H_o: $\alpha_1 = \cdots = \alpha_t$ v.s. H_1: at least one inequality.

What is the standard LR model for one-way anova in terms of $y = \beta' \mathbf{x} + \epsilon$?

```
    # same as
```

anova(lm(y$\sim$ x)) or anova(lm(y$\sim$ x-1)) $x \in \mathcal{R}^t$

Remark. Here the one-way-anova is a linear regression model

$Y_h = \sum_i \alpha_i \mathbf{1}(X_h = i) + \epsilon_h$, $h = 1, ...,$ n $(= \sum_{i=1}^t n_i)$ (corresponding to $Y_{ij} = \alpha_i + \epsilon_{ij}$).

1. **kruskal.test**. (Kruskal-Wallis (K-W) Rank Sum Test).
Performs a Kruskal-Wallis rank sum test on data following a one-way layout.

kruskal.test(y, groups)

Assumption: There are t (independent) samples and the ith sample observations satisfy

$$X_{ij} = \alpha_i + \epsilon_{ij},\ j \in \{1, ..., n_i\},\ i = 1, ..., t, \text{ and } F_{\epsilon_{ij}} = F_o\ \forall\ (i, j). \tag{5.2.3.2}$$

H_o: $\alpha_1 = \cdots = \alpha_t$, v.s. H_1: $\alpha_i \neq \alpha_j$ for at least one pair.

Remark. Here α_i can be either the mean or the median. H_o can be written as

$F_{X_{ij}} = F_o\ \forall\ (i, j)$.

This is a nonparametric alternative to one-way anova whereas the latter needs $N(\alpha_i, \sigma^2)$.

The K-W test statistic is

$$T = \frac{(N-1)(S_t^2 - C)}{S_r^2 - C}, \quad \textbf{where}\ \ N = \sum_i n_i.$$

Rank all N observations from 1 to N.

Let $r_{ij} = \text{rank}(X_{ij})$ and

s_i be the sum of the ranks in the ith sample, $i = 1, ..., t$.

Let $S_r^2 = \sum_{i,j} r_{ij}^2$, $S_t^2 = \sum_{i=1}^t (s_i/n_i)^2$ and $C = N(N+1)^2/4$.

T has approximately $\chi^2(t-1)$ distribution for moderate N.

Critical values for T are tabulated for small N.

Example 1. A real data (holl data). Total of 14 data from 3 groups.

>holl.y = c(2.9,3.0,2.5,2.6,3.2,3.8,2.7,4.0,2.4,2.8,3.4,3.7,2.2,2.0)

>holl.grps = factor(c(1,1,1,1,1,2,2,2,2,3,3,3,3,3), labels=c("Normal Subjects", "Obstr. Airway Disease","Asbestosis"))

$t = 3$, $n_1 = n_3 = 5$, $n_2 = 4$. Test for equal means.

>kruskal.test(holl.y, holl.grps)

Kruskal-Wallis chi-squared = 0.7714, df = 2, p-value = 0.68

>z=lm(holl.y$\sim$ holl.grps)

>anova(z)

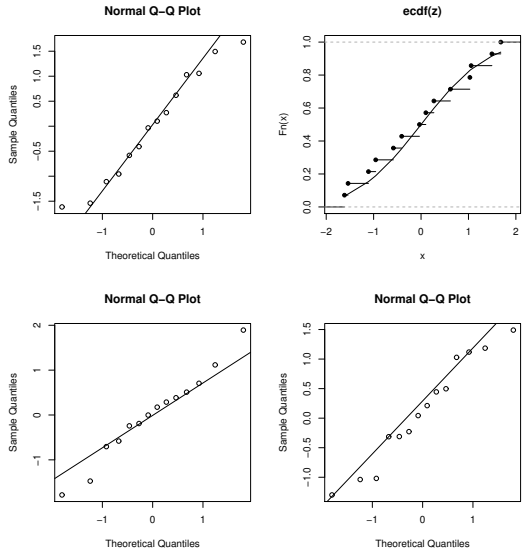
	Df	$Sum\ Sq$	$Mean\ Sq$	$Fvalue$	$Pr(>F)$
holl.grps	2	0.4468	0.22339	0.5601	0.5866
Residuals	11	4.3875	0.39886		

>z=studres(z)

> plot(ecdf(z))

> x =rnorm(14)

> y =rnorm(14)



$qqnorm(z)$ $cedf(z)$
$qqnorm(x)$ $qqnorm(y)$, where x, y =rnorm(14)

Q: Does the normal assumption hold ?
Conclusion of the tests ?

Example 2. Cancer relapse time data: n= 90, three groups.

> x

[1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
[39] 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
[77] 0 0 0 0 0 0 0 0 0 0 0 0 0 0

> y

[1] 789.0 496.5 260.0 434.5 463.0 412.0 658.5 576.5 280.5 823.5
[11] 694.5 198.5 677.5 937.5 549.5 613.5 1168.5 734.0 1100.0 646.0
[21] 717.0 396.0 5592.0 5051.0 3477.0 1483.0 6337.0 5088.0 2450.0 3354.0
[31] 1717.0 3199.0 3875.0 3171.0 5717.0 2890.0 4410.0 3298.0 3998.0 5175.0
[41] 383.0 1259.5 563.5 1829.0 1979.0 1440.0 1388.5 146.5 1897.0 41.0
[51] 253.0 1595.0 669.0 504.0 1669.0 600.0 359.5 451.0 11.5 1560.0
[61] 1625.0 1608.0 891.0 979.0 1541.0 1288.0 1189.0 1132.0 1387.0 1063.0
[71] 1359.0 535.0 1403.0 1272.0 134.0 1045.0 1007.0 1322.0 1309.0 1315.0
[81] 1296.0 930.0 636.0 1277.0 329.0 231.0 820.0 1241.0 988.0 903.0

> x=factor(x)

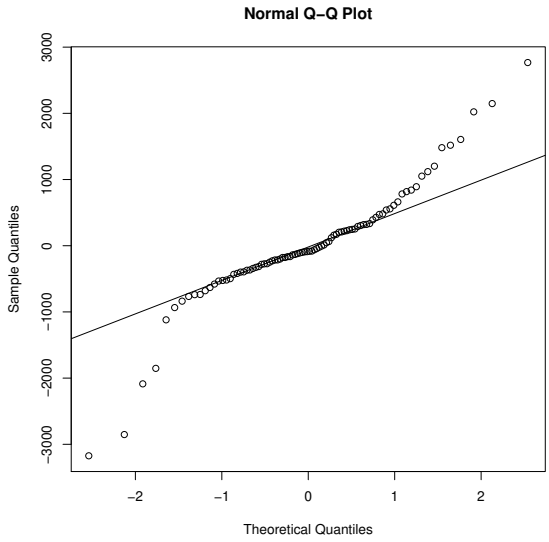
> anova(lm(y∼x))

	Df	$Sum\ Sq$	$Mean\ Sq$	$F\ value$	$Pr(>F)$	
x	2	112123133	56061567	71.826	$< 2.2e-16$	$***$
$Residuals$	87	67904709	780514			

> kruskal.test(y,x)

Kruskal-Wallis chi-squared = 37.343, df = 2, p-value = 7.781e-09

What is the conclusion ?
Are these two tests valid ?



Does it support $\epsilon \sim N(0, \sigma^2)$?
Can we believe the p-value ≈ 0 given by anova(z) here ?

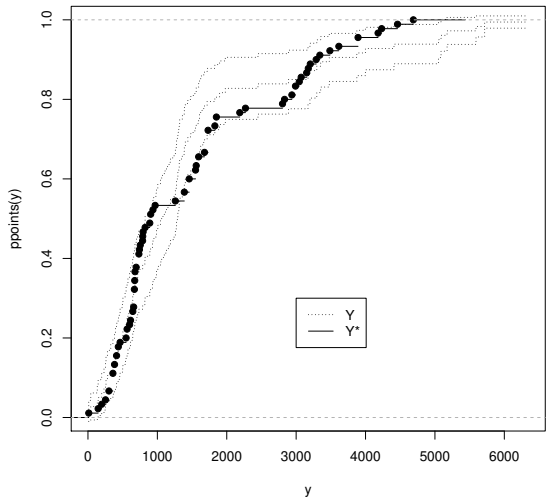


Figure 5.2.3.2. MD plot for cancer data with baseline centered at $x = 2$. why 2 ?

Remark. For regression data, CB is not reliable, MD plot is more reliable.

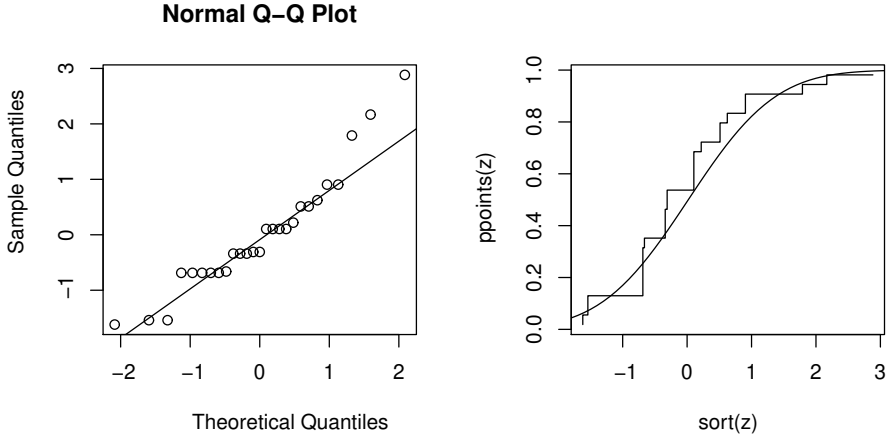
> u=2*(1:45)
> v=y[u]
> u=y[-u]
> cor.test(u,v,method="kendall") # **why do this ?**
pearson, kendall, spearman in cor.test

Example 3 (Simulation study). Generate 4 random samples.
Test H_o: $\mu_1 = \mu_2 = \mu_3 = \mu_4$.
The output:
> kruskal.test(x,y)
Kruskal-Wallis chi-squared = 8.0894, df = 3, p-value = 0.0442

> summary(aov(x∼y))

	Df	$Sum\ Sq$	$Mean\ Sq$	$F\ value$	$Pr(>F)$
y	3	13.78	4.594	2.254	0.109
$Residuals$	23	46.88	2.038		

Conclusion from 2 P-values ?



Remark. In order to use aov or anova, one needs to check the normal assumption.
Notice that there are obvious ties in qqplot of z.
It implies that there are many ties in residuals, thus ϵ is not continuous.

Q: Which test is more appropriate ?

Conclusion of the test ?

Remark. The example illustrates that it is important to check the validity of the test.

Remark. The 4 random samples in Example 3 are from $\mathcal{P}ois(\lambda_i)$,
with 4 different λ_i and 4 different sample sizes n_i.

```
n=c(3,4,5,15)
p=c(0.8,1,0.9,2)
x=rpois(n[1],p[1])
for (i in 2:4) x=c(x,rpois(n[i],p[i]))
y=c(rep(1,n[1]),rep(2,n[2]),rep(3,n[3]),rep(4,n[4]))
y=as.factor(y)
z=lm(x∼ y)
z=studres(z)
qqnorm(z)
qqline(z)
plot(sort(z),ppoints(z),type="S") Or plot(ecdf(z))
x=sort(z)
lines(x,pnorm(x,mean(x),sd(x)))
kruskal.test(x,y) #if normal assumption is not likely
anova(z) #if normal assumption seems likely
```

2. **friedman.test** (Rank sum test).

```
friedman.test(B) # matrix B_{b×t} v.s. column factor (called treatment)
```

Remark. The test is a non-parametric alternative of two-way anova (parametric one).

Review of two-way anova: Suppose we have t treatments each applied to one of the t

treatments in each of b blocks in a randomized block design. We denote by X_{ji} the response (observation) from treatment i in block j.

$$\begin{array}{c} \\ block\ 1 \\ \cdot \\ block\ b \end{array} \begin{array}{c} \begin{array}{ccc} treatment\ 1 & \cdots & treatment\ t \end{array} \\ \begin{pmatrix} X_{11} & \cdots & X_{1t} \\ \cdot & \cdots & \cdot \\ X_{b1} & \cdots & X_{bt} \end{pmatrix} \end{array} \stackrel{def}{=} B \quad (friedman.test(B)) \tag{1}$$

Assumption for two-way anova: $X_{ij} = \mu + \alpha_i + \beta_j + \epsilon_{ij}$,

X_{ij}'s are independent $N(\mu + \alpha_i + \beta_j, \sigma^2)$, $i = 1, ..., b$, $j = 1, ..., t$.

It is to test H_0: $\beta_1 = \cdots = \beta_t$ or

H_0^*: $\alpha_1 = \cdots = \alpha_b$.

R commands:

z=aov(y∼column+row)

summary(z) # or summary(aov(y∼column+row))

anova(lm(y∼column+row) # present the same output

It gives two p-values.

The command lm(y∼column+row) means

$$\begin{aligned} y_{ij} =& \mu + \alpha_2 \mathbf{1}_{(i=2)} + \cdots + \alpha_b \mathbf{1}_{(i=b)} + \beta_2 \mathbf{1}_{(j=2)} + \cdots + \beta_t \mathbf{1}_{(j=t)} + \epsilon_{ij} \\ & (= \mu + \alpha_2 \mathbf{1}_{(i=2)} + \beta_2 \mathbf{1}_{(j=2)} + \epsilon_{ij} \text{ if } t = b = 2). \end{aligned} \tag{2}$$

Recall the LSE is $\hat{\theta} = (X'X)^{-1}X'Y$, where $\theta = ?$ $X = ?$

If $b = t = 2$, then

$$\begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix} \text{ is changed to } \begin{pmatrix} y_{11} \\ y_{12} \\ y_{21} \\ y_{22} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}}_{rank=3} \begin{pmatrix} \mu \\ \alpha_1 \\ \alpha_2 \\ \beta_1 \\ \beta_2 \end{pmatrix} => Y = X\theta \ \textbf{?}$$

$\alpha_1 = \beta_1 = 0$, in Eq. (2) is the default identifiability condition.

Other identifiability conditions:

or $\alpha_1 = \mu = 0$ $(lm(y \sim \alpha + \beta - 1))$

$\sum_i \alpha_i = 0 = \sum_j \beta_j$. control.sum

In order to use aov, one needs to check:

(1) the normal assumption and

(2) independent samples.

In friedman.test, H_o: no treatment effect difference, i.e., $\beta_1 = \cdots = \beta_t = 0$,

where $F_{X_{ij}}(x) = F(x - \alpha_i - \beta_j)\ \forall\ x$ and $\begin{pmatrix} X_{11} \\ \vdots \\ X_{b1} \end{pmatrix}, ..., \begin{pmatrix} X_{1t} \\ \vdots \\ X_{bt} \end{pmatrix}$ are independent.

Data in friedman.test are arranged as the matrix $B = \Big(X_{ij}\Big)_{b \times t}$.

Thus "treatment" is the column factor.

We replace the observations in each block by ranks 1 to t.

This ranking is carried out separately for each block.

The sum of the ranks is then obtained for each treatment, denoted by

$s_j = \sum_{i=1}^{b} r_{ij}$, $j = 1, ..., t$,

where r_{ij} denotes the rank (or mid-rank if there are ties) of X_{ij} within block i,

let $S_r^2 = \sum_{i,j} r_{ij}^2$ $(= bt(t+1)(2t+1)/6$ if there is no tie),

let $S_t^2 = \sum_j s_j^2/b$ and $C = bt(t+1)^2/4$,

$T = b(t-1)(S_t^2 - C)/(S_r^2 - C)$

has approximately $\chi^2(t-1)$ distribution, if b, t are not too small.

Q: What is the difference between the two assumptions ?

(1) $F_{ij}(x) = F(x - \alpha_i - \beta_j)\ \forall\ x$;

(2) $X_{ij} = \mu + \alpha_i + \beta_j + \epsilon_{ij}$, where ϵ_{ij} are i.i.d., with $E(\epsilon_{ij}) = 0$.

If X_{ij} has Cauchy distribution, which model is applicable ?

If X_{ij} has Uniform distribution, which model is applicable ?

Remark. anova(lm()) can test both equal row effects and equal column effects in the same time, but friedman.test can only do column effects.

Example 4. Test for equal treatment effects. 12 data from 3 treatments and 4 subjects.

	$Trt1$	$Trt2$	$Trt3$
$Subject1$	0.73	0.48	0.51
$Subject2$	0.76	0.78	0.03
$Subject3$	0.46	0.87	0.39
$Subject4$	0.85	0.22	0.44

Input data:

```
treatment = factor(rep(c("Trt1", "Trt2", "Trt3"), each=4))              # (1,1,1,1,2,...
sub = factor(rep(c("Subject1", "Subject2", "Subject3", "Subject4"), 3))    # (1,2,3,1,2,...
or sub = factor(rep(c("Subject1", "Subject2", "Subject3", "Subject4"), time=3))
y = c(0.73,0.76,0.46,0.85,0.48,0.78,0.87,0.22,0.51,0.03,0.39,0.44)
(z=lm(y~ treatment+ sub))
summary(aov(y~ treatment+ sub))   # usual approach
qqnorm(studres(z))   # check assumptions
qqline(studres(z))
dim(y)=c(4,3)
v=sample(1:3,2)                    # for next step
cor.test(y[,v[1]],y[,v[2]], method="kendall") # For aov() or friedman.test() ?
    pearson, kendall, spearman in cor.test
friedman.test(y)   # if aov() is not applicable).
kruskal.test(as.vector(y),treatment) # Is y a vector or matrix ?
```

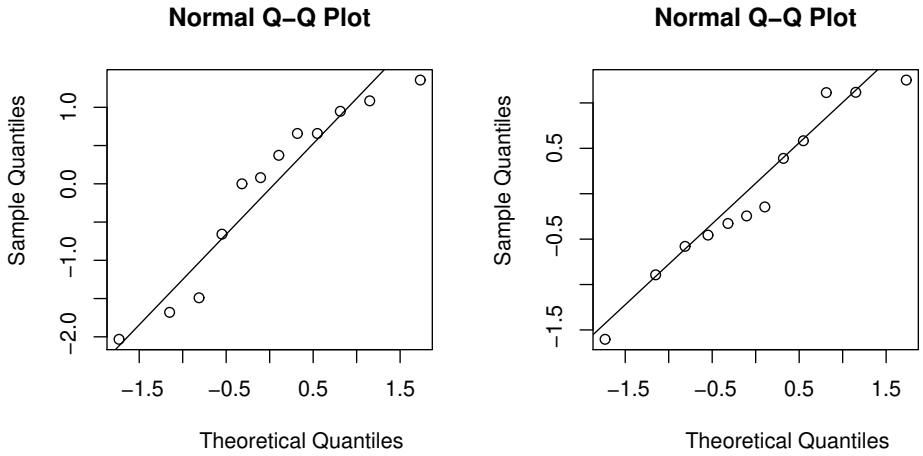
Output:

```
> (z=lm(y~ treatment+ sub))
```

$(Intercept)$	$treatmentTrt2$	$treatmentTrt3$	$subSubject2$	$subSubject3$	$subSubject4$
$7.300e-01$	$-1.125e-01$	$-3.575e-01$	$-5.000e-02$	$1.408e-16$	$-7.000e-02$

```
> summary(aov(y~ treatment+ sub))
```

	Df	$Sum\ Sq$	$MeanSq$	$F\ value$	$Pr(>F)$
$treatment$	2	0.2673	0.13366	1.691	0.262
sub	3	0.0114	0.00380	0.048	0.985
$Residuals$	6	0.4741	0.07902		



$qqnorm(data) \quad qqnorm(rnorm)$

Q: Do we need to continue ?

> cor.test(y[v[1],],y[v[2],], method="kendall") # **Is it a good choice here ?**

T = 9, p-value = 0.7194

> friedman.test(y)

Friedman chi-squared = 2, df = 2, p-value = 0.3679

> kruskal.test(as.vector(y),treatment)

Kruskal-Wallis rank sum test

data: y and treatment

Kruskal-Wallis chi-squared = 3.5769, df = 2, p-value = 0.1672

Conclusion: H_o ? H_1 ? α ? statistic ? ?

Announcement: Final Exam is in this room.

Example 5 (Simulation exercise). Generate 4 random samples from Exp(θ_i), with different θ_i and same sample sizes, say n. Form a $4 \times n$ matrix. Test for equal column effects, and then for equal row effects.

```
n=20
p=matrix(rep(c(1,3,2,5),n),4)
x=matrix(rexp(4*n),ncol=n)
x=x+p
gr= factor(as.vector(row(x)))
bl = factor(as.vector(col(x)))
# skip checking independence
friedman.test(x) # what to expect for P-value
friedman.test(t(x)) # what to expect for P-value
friedman.test(as.vector(x),bl,gr)
friedman.test(as.vector(x),gr,bl)
kruskal.test(as.vector(x),gr)
kruskal.test(as.vector(x),bl)
```

Output:

> friedman.test(x)

Friedman chi-squared = 18.257, df = 19, p-value = 0.5053

> friedman.test(t(x))
Friedman chi-squared = 50.22, df = 3, p-value = 7.172e-11
> friedman.test(as.vector(x),bl,gr)
Friedman chi-squared = 18.257, df = 19, p-value = 0.5053
> friedman.test(as.vector(x),gr,bl)
Friedman chi-squared = 50.22, df = 3, p-value = 7.172e-11
> kruskal.test(as.vector(x),gr)
Kruskal-Wallis chi-squared = 59.436, df = 3, p-value = 7.757e-13
> kruskal.test(as.vector(x),bl)
Kruskal-Wallis chi-squared = 5.0241, df = 19, p-value = 0.9994

Q: Consider testing problem of a mean or median μ, H_o: $\mu = 0$ with $n = 6$ in three cases: (1) $N(\mu, 1)$, (2) $U(a, b)$, (3) Cauchy Distribution.

1. If the sample is from $N(0, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using t.test? 0.05 ?
2. If the sample is from $N(0, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using wilcox.test? 0.05 ?
3. If the sample is from $N(1, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using t.test? 0.05 ?
4. If the sample is from $N(1, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using wilcox.test? 0.05 ?
5. If the sample is from $U(-1, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using t.test? Is it 0.05 ? Y, N, DNK.
6. If the sample is from $U(-1, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using wilcox.test? Is it 0.05 ? Y, N, DNK.
7. If the sample is from $U(0, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using t.test? Is it 0.05 ? Y, N, DNK.
8. If the sample is from $U(0, 1)$ what is the size of the test if we reject with p.value$\leq$ 0.05 using wilcox.test? Is it 0.05 ? Y, N, DNK.
9. How about Cauchy distribution ? Difference between it and U(a,b) ?

Remark. The t.test(x) for H_o: $\mu = 0$ v.s. H_1: $\mu \neq 0$ is $\phi = \mathbf{1}(\frac{|\overline{X}|}{S/\sqrt{n}} > t_{\alpha/2,n-1})$.

$$E(\phi|\mu) = \begin{cases} \textit{power function of } \mu & \text{if } \underbrace{X_i\text{'s are i.i.d. } N(\mu,\sigma^2) \text{ for } \mu \in (-\infty,\infty)}_{\textit{model assumption}}, \\ ? & \text{otherwise} \end{cases}$$

$$= \begin{cases} P(H_1|H_0) = \alpha & \text{if } \mu = 0 \text{ and } X_i\text{'s are i.i.d. } N(\mu,\sigma^2) \\ 1 - P_\mu(H_0|H_1) & \text{if } \mu \neq 0 \text{ and } X_i\text{'s are i.i.d. } N(\mu,\sigma^2) \\ P(H_1|H_0) \neq \alpha \textit{ most likely} & \text{otherwise but } \mu = 0 \\ ? & \text{otherwise, but } \mu \neq 0 \end{cases}$$

On the other hand, under the model assumption:

$$X_i\text{'s are i.i.d. with } F_{X_i}(x) = F_o(x-\mu),\ F_o(x) = 1 - F_o(-x)\ \forall\ x, \mu \in \mathcal{R}, \tag{1}$$

and F_o **may not be** $N(0, \sigma^2)$, the wilcox.test(x) for H_o: $\mu = 0$ v.s. H_1: $\mu \neq 0$ is $\Phi = \mathbf{1}(Z > z_n)$, where Z is defined in §5.2.1 and $P(Z > z_n|\mu = 0) = \alpha$.

$$E(\Phi) = \begin{cases} power\ function\ of\ \mu & \text{if model assumption (1) is true} \\ ? & \text{otherwise} \end{cases}$$
$$= \begin{cases} P(H_1|H_0) = \alpha & \text{if } \mu = 0 \text{ and the model assumption (1) holds} \\ 1 - P_\mu(H_0|H_1) & \text{if } \mu \neq 0 \text{ and the model assumption (1) holds} \\ ? & \text{otherwise} \end{cases}$$

If one is not sure of the distribution, one can use wilcox.test, provided that the model assumption in Eq.(1) holds.

Otherwise, the size of the test is not what you selected for p-value *e.g.* =0.05.

That is why to check the model assumptions of a test.

If one is sure of normal distribution, both tests can be used, as they have the same level.

However, t.test is more powerful.

Remark. We say a test is valid if the model assumption (not including H_o) for the test is satisfied. In the latter case, the p-value given by the test in R is correct. Otherwise, we do not know whether the p-value given by the test in R is correct.

How to find the size of the test $P(H_1|H_0)$?

$P(H_1|H_0) = \int \cdots \int_{RR} \prod_{i=1}^{n} (f(x_i)dx_i)$,

where $RR = \{|T| > c\}$ and $T = \frac{\overline{X}-\mu_o}{S/\sqrt{n}}$.

```
n=6
m=10000
fun=function(n){
x=runif(n,-1,1)
a=t.test(x)$p.value
return(a)
}
u=matrix(rep(0,m*n),m)
s=apply(u,1,fun)
(v=mean((s<0.05)))
[1] 0.0637                    # ≈ 0.05 ??
sqrt(v*(1-v)/m)*2
[1] 0.00488
```

3. **prop.test** (Proportions Tests). Two set-ups:

(1) Compare proportions against hypothesized values p (a vector).

(2) Tests whether underlying proportions are equal.

(1) Suppose that X_i, $i = 1, ..., k$, are independent and $X_i \sim bin(n_i, p_i)$

prop.test(x, n, p, alternative="two.sided", conf.level=.95, correct=T)

x, n, p are $k \times 1$ vectors.

H_o: p_i's are as given.

The test statistic is $T = U'U$, approximately $\chi^2(k)$,

where $U' = (U_1, ..., U_k)$ and $U_i = \frac{X_i - n_i p_{io}}{\sqrt{n_i p_{io}(1-p_{io})}}$.

(2) prop.test(x, n) for H_0: $p_1 = \cdots = p_k$.

Example:

```
> n=c( 18,22,19,23,15)
> x=c( 16,10, 9, 9 ,9)
> p=c(0.9,0.5,0.5,0.5,0.5)
> prop.test(x,n,p)
```

X-squared = 1.9461, df = 5, p-value = 0.8566

alternative hypothesis: two.sided

null values: $prop1$ $prop2$ $prop3$ $prop4$ $prop5$
0.9 0.5 0.5 0.5 0.5 H_o ??

sample estimates:
$prop1$ $prop2$ $prop3$ $prop4$ $prop5$
0.8888889 0.4545455 0.4736842 0.3913043 0.6000000

> prop.test(x,n) # H_o ?

X-squared = 12.079, df = 4, p-value = 0.01677

Conlusion of these tests ?

A simulation study on the level and size of the prop.test:

```
> m=10000
> r=0
> n=c(20,8)
> p=c(0.6,0.5)
> for (i in 1:m){
    x=rbinom(c(1,1), n,p)
    r=r+as.numeric(prop.test(x,n,p)$p.value<0.05)
    }
> (u=r/m)
    [1] 0.0195
> sqrt(u*(1-u)/m)*1.96
    [1] 0.002710174
```

r/m= 0.0195 $\pm$0.0027 for m=10000

r/m= 0.021 for m=1000

Q: For this prop.test:

Is the size of the test 0.05 ? $P(H_1|H_0) = 0.05$?

Is the level of the test 0.05 ? $\alpha = 0.05$?

$P(H_1|H_0) = 0.0195$? $\hat{P}(H_1|H_0) = 0.0195$?

Why $\alpha \neq 0.05$ as desired ?

The midterm will require familiar with Math 447 and 448 formulae.

5.3 Some classical tests in R

5.3.1. Tests for contingency tables:

$r \times c$ contingency table $\left(N_{ij}\right)_{r\times c}$ where N_{ij} are counts.

$r \times c \times l$ contingency table: $\left(N_{ijk}\right)_{r\times c\times l}$

Example 1. Consider a special case of 2×2 contingency table.

Let A – a randomly selected person is a male,

B – a randomly selected person is a democrat.

H_o: $A \perp B$.

<=> $P(AB) = P(A)P(B)$

<=> $P(AB^c) = P(A)P(B^c)$ <=> $P(A^cB^c) = P(A^c)P(B^c)$ <=> $P(A^cB) = P(A^c)P(B)$.

$n = 9$ people are randomly sampled. Data are $\begin{pmatrix} x \\ y \end{pmatrix}$:

```
> x = factor(c(1,1,2,1,2,1,1,2,2), labels=c("male", "female")) # old days
```

> y = factor(c(1,1,1,2,1,2,2,1,1), labels=c("democrat", "none-democrat"))
> table(x,y)

	y	
x	$democrat$	$none-democrat$
$male$	2	3
$female$	4	0

is called a 2×2 contingency table.

One tests H_o base on the data in the form of $r \times c$ or $r \times c \times l$ contingency table.

(1) $r \times c$ **tables**

Original data: Data: X_1, ..., X_n, together with 2-factor classification
$X_i \in \{(a, b) : \ a \in \{a_1, ..., a_r\}, b \in \{b_1, ..., b_c\}\}$.

$r \times c$ contingency table:

	b_1	$\cdots$	b_c
a_1	N_{11}	$\cdots$	N_{1c}
$\vdots$	$\vdots$		$\vdots$
a_r	N_{r1}	$\cdots$	N_{rc}

, leads to probability table

p_{11}	$\cdots$	p_{1c}
$\vdots$	$\cdots$	$\vdots$
p_{r1}	$\cdots$	p_{rc}

,

where $N_{ij} = \sum_{k=1}^{n} \mathbf{1}_{(X_k=(a_i,b_j))}$, $\sum_{ij} p_{ij} = 1$ and $p_{ij} \geq 0$. p_{ij}'s are parameters.
Test H_o: The column and row factors are independent,
that is, $p_{ij} = p_{i\cdot}p_{\cdot j} \ \forall \ (i, j)$, where $p_{i\cdot} = \sum_j p_{ij}$ and $p_{\cdot j} = \sum_i p_{ij}$.
Three tests will be introduced:
(a) fisher.test, (b) chisq.test, (c) mcnemar.test.

(2) $r \times c \times l$ tables

Original data: X_1, ..., X_n, together with 3-factor classification
$X_i \in \{(a, b, w) : \ a \in \{a_1, ..., a_r\}, b \in \{b_1, ..., b_c\}, w \in \{w_1, ..., w_l\}\}$.
Let $N_{ijk} = \sum_{h=1}^{n} \mathbf{1}_{(X_h=(a_i,b_j,w_k))}$,
Array: $(N_{ijk})_{r\times c\times l}$.
mantelhaen.test will be introduced.

1. **fisher.test.** Performs a Fisher's exact test on a two-dimensional contingency table.

e.g., 2×2 table.

	B	B^c	sum
A	N_{11}	N_{12}	r_1
A^c	N_{21}	N_{22}	r_2
sum	c_1	c_2	n

Data: X_1, ..., X_n, together with classification
$X_i \in \{(A, B), (A^c, B), (A, B^c), (A^c, B^c)\}$.
N_{11}, N_{21}, N_{12}, N_{22} are numbers of the 4 types of X_i's.
H_0: the row and column factors are independent ($i.e., P(AB) = P(A)P(B)$).
The test statistic is

$$\phi = \mathbf{1}(N_{11} \geq q_1, \ or \ N_{12} \geq q_2)$$

where q_1 and q_2 are chosen from the hypergeometric tables to make

$$\sum_{s\leq q_1} f(s|c_1, r_1, n) \text{ and } \sum_{s\geq q_2} f(s|c_1, r_1, n), \text{ where } f(s|c_1, r_1, n) = \frac{\binom{c_1}{s}\binom{c_2}{r_1-s}}{\binom{n}{r_1}}$$

each as close to $\alpha/2$ as possible, but not larger (α is the level of the test).
Remark. (1) The P-value is exact, not an approximation.
(2) The size of the test $\leq$ the level of the test, as the distribution is discrete. Thus when we reject H_0 with p-value≤ 0.05, the level (but not the size α) of the test is 0.05.

Example 1. Is it true that gender $\perp$ policital affiliation ?

> x = factor(c(1,1,2,1,2,1,1,2,2), labels=c("male", "female"))

> y = factor(c(1,1,1,2,1,2,2,1,1), labels=c("democrat", "republican"))

> fisher.test(x,y) # x and y are factors

> x=table(x,y) # A second way

> fisher.test(x) # x is a matrix of counts

p-value = 0.1667

alternative hypothesis: true odds ratio is not equal to 1 # odds ratio=$\frac{p_{11}p_{22}}{p_{12}p_{21}}$

95 percent confidence interval: # for odds ratio, $p_{11} = P(AB)$, $p_{22} = P(A^cB^c)$, ...

0.00000 2.64606

Remark. The output sets H_0: $\frac{p_{11}p_{22}}{p_{12}p_{21}} = 1$. In fact, if gender $\perp$ policital affiliation, then

$$\frac{p_{11}p_{22}}{p_{12}p_{21}} = \frac{p_{1\cdot}p_{\cdot 1}p_{2\cdot}p_{\cdot 2}}{p_{1\cdot}p_{\cdot 2}p_{2\cdot}p_{\cdot 1}} = 1. \qquad (p_{11} =?\ \ p_{22} =?)$$

Answer of the test: ??

2. chisq.test. (Pearson's Chi-square Test for Count Data).

Performs a Pearson's chi-square test on a two-dimensional contingency table.

(A large sample test for independence of $r \times c$ contingency table).

H_o, the row and column effects are independent.

Test statistic is

$$T = \sum_{i,j} \frac{(n_{ij} - e_{ij})^2}{e_{ij}}$$

where n_{ij} is the count in cell (i,j),

e_{ij} is expected count of n_{ij}

(in default, $e_{ij} = n_{i+}n_{+,j}/n$) and $n = \sum_{i,j} n_{ij}$.

T is approximately $\chi^2_{(c-1)(r-1)}$ under H_o.

df = df in Θ - df in Θ_o (under H_o) (df of Θ ? $\left(p_{ij}\right)_{r\times c}$)

$((rc-1) - ((r-1)+(c-1)) = (r-1)(c-1))$

There are various functions of chisq.test based on input data: (several chisq.tests)

> x=c(762, 327, 468)

> y= c(484, 239, 477)

Case A. H_o : independent of of column factor and row factor for 3×2 contingency table

	A	B
a	762	484
b	327	239
c	468	477

There are $n = 762 + \cdots + 477$ X_i's, each has two factors.

>(z=matrix(c(x,y),3)) # matrix(c(x,y),nrow=3)

>chisq.test(z)

X-squared = 30.07, df = 2, p-value = 2.954e-07 df=(r-1)(c-1)=(3-1)(2-1)=2

Case B. 762 is treated as a level of the column factor rather than # in cell (1,1) as in case A.

H_o : independent of 3×3 contingency table

>(z=table(x,y))

	$A=$"762"	$B=$"327"	$C=$"468"	
$a=$"484"	1	0	0	Treat as n=3 X_is, with 2 factors
$b=$"239"	0	1	0	and 3 levels each !
$c=$"477"	0	0	1	

>chisq.test(z) # (z is table, not matrix as in case A).

>chisq.test(x,y)

X-squared = 6, df = 4, p-value = 0.1991 df=(r-1)(c-1)=(3-1)(3-1)=4

Case C. test equal probabilities of the 6 elements in $c(x, y)$. H_0: $p_1 = \cdots = p_6$ $(= 1/6)$.

>chisq.test(c(x,y))

X-squared = 345.29, df = 5, p-value < 2.2e-16 df=df in Θ-df in Θ_o=(6-1)-0

Case D. test H_o: $x = cy$ (#x[1]:#x[2]:#x[3]) =(#y[1]:#y[2]:#y[3])

i.e. $p_{xi} = p_{yi}$, $i \in \{1, 2, 3\}$, where $\sum_i p_{xi} = \sum_i p_{yi} = 1$ and p_{yi}'s are given.

>chisq.test(x, p = y) # if y is not a probability vector

>chisq.test(x, p = y/sum(y)) # same as above

X-squared = 66.313, df = 2, p-value = 3.983e-15 df=df in Θ-df in Θ_o=(3-1)-0

The last two cases are applications to $1 \times m$ contingency table application (similar to prop.test).

Suppose that n_i is the count of observations fall in cell i, with expected frequency np_i, $i = 1$, ..., m and $n = \sum_i n_i$. Pearson's χ^2 Goodness-of-fit statistics

$$T = \sum_{i=1}^{m} (n_i - n\hat{p}_i)^2/(n\hat{p}_i) \qquad (T = \textstyle\sum_{i,j} \frac{(n_{ij}-e_{ij})^2}{e_{ij}})$$

H_o: $p_i = p_i(\theta)$ where $\theta \in \Omega_o$,

where $\hat{p}_i$ is the MLE of p_i $(= p_i(\theta))$. $T \sim \chi^2(m - k)$ asymptotically, where k is the degrees of freedom on θ or the dimension of Θ_o.

df = df in Θ - df in Θ_o (under H_o).

If p_i does not depend on some θ, $k = 0$ (see Cases C and D).

Case C: $m = 6 - 1$ and $k = 0$, as $p_1 = \cdots = p_6 = 1/6$ $(\sum_i p_i = 1)$.

Case D: $m = 3 - 1$ and $k = 0$, as (p_1, p_2, p_3) is given.

Example 2. A 3×3 table corresponds to X_1, ..., $X_{12} \in \mathcal{R}^2$.

> y=factor(c(2,2,2,3,3,3,2,1,1,2,1,2),label=c("A","B","C"))

> z=factor(c("a","b","a","b","c","c","c","a","a","a","a","b"))

> table(z,y) #

	A	B	C
a	3	3	0
b	0	2	1
c	0	1	2

> chisq.test(z,y) P-value 0.13,

> fisher.test(z,y) P-value 0.24

H_o**: ?** **Conclusion ?**

An application to an alternative to ks.test().

Data: independent X_1, ..., X_n with distribution F.

H_0: $F = F_0$, where F_0 is known, except for some parameters.

Devide the range into a grid of m cells.

Let n_i be the count of observations fall in cell i,

Proceed as before.

Example 3.

(x = runif(100,0,4))

breaks = quantile(x)

0%	25%	50%	75%	100%
0.0318276	1.2288552	2.0762343	3.0105985	3.9955591

y=fitdistr(x,"weibull")

z=pweibull(breaks, y$e[1], y$e[2])

(u=z[2:5]-z[1:4])

u=c(z[1],u,1-z[5])

(q=c(0,25,25,25,25,0))

chisq.test(q,p=u)

X-squared = 35.477, df = 5, p-value = 1.208e-06 **Is it what you expected ?**

H_o: $x[1] : x[2] : \cdots : x[6] = u[1] : u[2] : \cdots : u[6]$. or $x \sim Weibull$.

ks.test(x, "pweibull", y$e[1],y$e[2])

D = 0.1052, p-value = 0.2183 **Is it what you expected ?**

Another run:

X-squared = 18.311, df = 5, p-value = 0.002581 which test ?

D = 0.088881, p-value = 0.4084 which test ?

What can you conclude ?

What is wrong with the ks.test ?

How can ks.test be properly applied ?

3. mcnemar.test. (McNemar's Chi-Square Test for Count Data).

Performs a McNemar's chi-square test on a 2-dimensional $R \times R$ contingency table.

Data X_i, $i = 1, ..., n$ (may be dependent).

$X_i = (x, y)$, $x \in A$ and $y \in B$, $||A|| = ||B|| = R$.

H_0: $P\{X_1 = (x, y)\} = P\{X_1 = (y, x)\}\ \forall\ (x, y)$ or

$p(x, y) = p(y, x)\ \forall\ (x, y)$.

Remark. Differences between mcnemar.test and chisq.test:

(1) $R \times C$ **allows** $R \neq C$ in chisq.test, but not allows in mcnemar.test.

(2) X_i's must be independent in chisq.test, but not necessary in mcnemar.test.

(3) chisq.test tests independence, but mcnemar.test tests symmetry.

Under H_0, McNemar's statistic approximately $\sim \chi^2_{R(R-1)/2}$ (similar to the LRT).

why R(R-1)/2 ?

df of Θ ? $\left(p_{ij}\right)_{R\times R}$

df of Θ_0 ? H_0: $p_{ij} = p_{ji}\ \forall\ (i, j)$

For $R = 2$: Let n_{ij} be the count in cell [i,j].

The test statistic is $T = Z^2$ with

$$Z = \frac{n_{12} - (n_{12} + n_{21})/2}{\sqrt{(n_{12} + n_{21})(\frac{1}{2})^2}} = \frac{n_{12} - n_{21}}{\sqrt{n_{12} + n_{21}}}.$$

	A_1	A_2
B_1	n_{11}	n_{12} .
B_2	n_{21}	n_{22}

> x = factor(c(1,1,2,1,2,1,1,2,2), labels=c("male", "female"))

> y = factor(c(1,1,1,2,1,2,2,1,1), labels=c("democrat", "republican"))
> mcnemar.test(x,y)

McNemar's chi-squared = 0, df = 1, p-value = 1

Conclusion ?

> (x=table(x,y))

	y	
x	$democrat$	$republican$
$male$	2	3
$female$	4	0

Ans. Proporpotion of the female democrats and male republicans are the same.

Homework 5.3.1.1: Prove or disprove in general: independent <=> symmetry ?

4. **mantelhaen.test.** Performs a Mantel-Haenszel chi-square test on a three-dimensional contingency table.

Data: independent X_1, ..., $X_n \in \left\{(x_1, x_2, x_3), x_h \in A_h, h \in \{1, 2, 3\}\right\}$,

where A_h are sets of sizes r, c and i (Row, Column and Item).

Example 1. X_1, ,..., X_4 are input by

$\vec{x}_1$=factor(c(1,2,1,2), labels=c("NoResponse","Response")),

$\vec{x}_2$=factor(c(1,2,2,1), labels=c("Male", "Female")),

$\vec{x}_3$=factor(c(1,2,1,1), labels=c("Nodular", "Diffuse")).

where $r = c = i = 2$.

In general, there are 3 factors, taking r, c and i values respectively.

Factor 1 takes values a_{11}, ..., a_{1r},

Factor 2 takes values a_{21}, ..., a_{2c},

Factor 3 takes values a_{31}, ..., a_{3i},

Contingency table:

		Ma		Fe
		a_{21}	$\cdots$	a_{2c}
NoR	a_{11}	w_{111}	$\cdots$	w_{1c1}
	$\vdots$	$\cdots$	$\cdots$	$\vdots$
Re	a_{1r}	w_{r11}	$\cdots$	w_{rc1}

a_{31} (??)

$\cdots$

	a_{21}	$\cdots$	a_{2c}
a_{11}	w_{11i}	$\cdots$	w_{1ci}
$\vdots$	$\cdots$	$\cdots$	$\vdots$
a_{1r}	w_{r1i}	$\cdots$	w_{rci}

a_{3i}

$n = \sum_{k,j,h} w_{kjh}$.

H_0: Conditional on $Iterm = h$, Column factor $\perp$ Row factor.

P$\{R = a_{1j}, C = b_{2k} | I = h\} = P\{R = a_{1j} | I = h\} P\{C = b_{2k} | I = h\}$ for each (j, k, h).

For example, suppose that we have a sequence of 2×2 tables from k different age groups, obtained from independent observations $X_h = (x, y)$, $h = 1$, ..., n, where x and y are the indicator functions that the h-th person belongs to the groups $\{R = 1\}$ and $\{C = 1\}$, respectively ($x = \mathbf{1}(R_h = 1)$ and $y = \mathbf{1}(C_h = 1)$). Here $\{R = u\} \cap \{C = v\}$ may not be empty (e.g, democratic and artist), called cross-classified.

$item\ 1$	$C = D$	$C = D^c$		$item\ k$	$C = D$	$C = D^c$	
$R = A$	w_{111}	w_{121}	n_{11}	$R = A$	w_{11k}	w_{12k}	n_{k1}
$R = A^c$	w_{211}	w_{221}	n_{12}	$R = A^c$	w_{21k}	w_{22k}	n_{k2}
	m_{11}	m_{12}	n_1		m_{k1}	m_{k2}	n_k

$\cdots$

H_o: $p_{11} = p_{12}$, ..., $p_{k1} = p_{k2}$, where

$p_{i1} = P(D|R = A,\ I = i)$ and $p_{i2} = P(D|R = A^c,\ I = i)$.
Is H_o the same as $P(AD|I = i) = P(A|I = i)P(D|I = i)$, $i = 1$, ..., k ?

$$A \perp B \text{ iff } P(AB) = P(A)P(B) \text{ iff } P(A|B) = P(AB)/P(B) = P(A) = \cdots = P(A|B^c) \text{ ?}$$

Test statistic is $MH = \frac{\sum_{j=1}^{k}(w_{11j} - E_0(w_{11j}))}{\sqrt{\sum_{j=1}^{k} Var_0(w_{11j})}}$, where $MH^2 \sim \chi^2(1)$.

```
> x=factor(rep(c(1,2,1,2),c(3,10,15,2)),labels=c("NoResponse","Response"))
> y=factor(rep(c(1,2,1,2,1,2,1,2), c(1,2,4,6,12,3,1,1)), labels=c("Male", "Female"))
> z=factor(rep(c(1,2), c(13,17)), labels=c("Nodular", "Diffuse"))
> mantelhaen.test(x,y,z)
> x=table(x,y,z)
> mantelhaen.test(x) # same answer
    Mantel-Haenszel X-squared = 0.15182, df = 1, p-value = 0.6968
```

How to generate simulation data for contingency table ?

```
> x=mvrnorm(90,c(0,0),matrix(c(4,0.4,0.4,3),2,2))        # dimension of x ?
> x=round(x/4)          # What does x represent ? decimal or integer ?
> fisher.test(x)        # what do you expect ?
    p-value = 1
> fisher.test(x[1,],x[2,])
    p-value = 1
> fisher.test(x[,1],x[,2])
    p-value = 0.3362
> chisq.test(x[,1],x[,2])
    X-squared = 4.4544, df = 4, p-value = 0.348
> chisq.test(factor(x[,1]),factor(x[,2]))
    X-squared = 4.4544, df = 4, p-value = 0.348
```

What is the conclusion ? Does it work ?

```
n=30
x=rbinom(2*n,1,0.5)
dim(x)=c(2,n)
fisher.test(x[1,],x[2,])          What do you expect for p-value ?
y=matrix(c(1,1,0,1),ncol=2)
for(i in 1:n)
x[,i]=x[,i]%*%y
fisher.test(x[1,],x[2,])          What do you expect for p-value ?
```

5. **ks.test.** Kolmogorov-Smirnov Goodness-of-Fit Test. Performs a one or two sample Kolmogorov -Smirnov test, which tests the relationship between two distributions.

5.1. One-sample. Suppose that X_1, ..., X_m are a random sample from F. To test against H_1: $F \neq F_o$, where F_o is given (upto a parameter). The test statistic is

$J = \sup\{|F_m(t) - F_o(t)| : t \in R\}$. P-value is given in R.

Remark. Most of the time, we do not know the parameters in F_o and has to estimate the parameters. The statistic is changed this way. For instance, under normal assumption,

for n large, the critical values (percentiles) for the ks.test with parameters known and for ks.test with estimated parameters (called Lilliefors' test, lillie.test) are

$F(t)$	0.90	0.95	0.99
with estimators	$1.22/\sqrt{n}$	$1.36/\sqrt{n}$	$1.52/\sqrt{n}$
with known parameters	$0.82/\sqrt{n}$	$0.89/\sqrt{n}$	$1.04/\sqrt{n}$

For other distributions, we can use the resampling method to estimate the percentiles.
5.2. Two-sample. Suppose that X_1, ..., X_m and Y_1, ..., Y_n are two independent samples with continuous cdfs F and G, respectively. To test against H_1: $F \neq G$, let F_m and G_n be the edf's of F and G, respectively, let d = greatest common divisor of m and n. (What is it if (m,n)=(12,14) ?) The test statistic is

$$J = \frac{mn}{d} \sup\{|F_m(t) - G_n(t)| : t \in R\}$$

P-value is given in R.
A simulation example. Test whether two random sample have the same distribution.
i.e. H_o: $F = G$, where $X_i \sim F$ and $Y_j \sim G$.

```
m=1000
n=90
fun1=function(){
     x = rnorm(n)
     y = rnorm(n, mean = 0.5, sd = 1)
     a=t.test(x,y)$p.value #                    what do you expect regarding F = G ?
     return(a)
}
fun2=function(){
     x = rnorm(n)
     y = rnorm(n, mean = 0.5, sd = 1)
     a=ks.test(x,y)$p.value #                   what do you expect regarding F = G ?
     return(a)
}
u=matrix(rep(0,m*n),m)
s=apply(u,1,fun2)
mean((s<0.05))
     [1] 0.775
mean((apply(u,1,fun1) <0.05))                   # > 0.775 or < 0.775 ? Why ?
     [1] 0.909
```

If the means are different and under NID, t-test is better (**what does it mean ?**)
Otherwise, the t.test may not be valid or may be misleading, but ks.test is always valid.

$$F_Y = F_X \begin{cases} => \\ <\neq \end{cases} \quad \mu_Y = \mu_X.$$

```
>x=rnorm(100,0,5)
>y=rnorm(100)
```

>as.numeric(t.test(x,y)$p.value<0.05) (What do you expect, 0 or 1 ?)
>as.numeric(ks.test(x,y)$p.vale<0.05) (What do you expect, 0 or 1 ?)

Remark. In the previous simulation study, we test H_o: $F = G$. fun2() is really test H_o, fun1() is to test H_o^*: $\mu_F = \mu_G$. Notice that

$$H_o => H_o^*, \text{ but not vice versa.} \tag{1}$$

How about mantelhaen.test ? (with H_o: conditional I, $R \perp C$, versus
H_o^*: $P(C = a_{2l}|R = a_{1h}, I = a_{3k}) = P(C = a_{2l}|R = a_{1j}, I = a_{3k})$ $\forall$ possible (l, h, j, k)).
Statement (1) also is applicable to cor.test with H_o: $X \perp Y$ and H_o^*: $\rho_{X,Y} = 0$.

§5.6. Density Estimation

Given a random sample, X_1, ..., X_n from X, denote its cdf by F where $F(t) = P(X \le t)$.
Its d.f. f is $f = \begin{cases} F' & \text{if } X \text{ is continuous} \\ F(t) - F(t-) & \text{if } X \text{ is discrete} \end{cases}$.
Q: F =? and f = ?
Two typical approaches for estimating F:

Parametric. $F(t) = F_o(t;\theta)$, $\theta \in \Theta \subset \mathcal{R}^p$. $\hat{F}(t) = F_o(t;\hat{\theta})$.

Non-parametric. $\hat{F}$ is the edf.

$$\hat{F}(t) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}(X_i \le t).$$

Q: Why do we need to know F ?

(1) Estimate $P(X \in A)$ by $\int \mathbf{1}(x \in A) d\hat{F}(x)$.

(2) Estimate $E(g(X))$ by $\int g(x) d\hat{F}(x)$ (the Lebesgue Stieltjes integral).

(3) Compare two distributions H_o: $F(t) \le G(t)$ $\forall$ t.

Note that $E(g(X)) = \int g(x)f(x)dx$ if X is continuous.

$\int g(x)d\hat{F}(x) = \sum_{i=1}^{n} g(x_i)\frac{1}{n}$ $(= \overline{X})$ if $g(x) = x$ and $\hat{F}$ is the edf.

Q: Why do we need to know f ?
One Example: If X_1 is continuous, the sample median $med(X)$ satisfies

$$\sqrt{n}(med(X) - m) \xrightarrow{D} N(0, \sigma^2), \text{ with } \sigma^2 = \tfrac{1}{4(f(m))^2}.$$

Q: f =?
Two approaches:

A. Parametric: $\hat{f}(x) = f_o(x;\hat{\theta})$, where $\hat{\theta}$ is an estimate.

B. Non-parametric:

B.1. If X is discrete, $\hat{f}(t) = \hat{F}(t) - \hat{F}(t-) = \sum_{i=1}^{n} \mathbf{1}(X_i = t)/n$.

B.2. If X is continuous, $\hat{f}$ in B.1 may not be desirable.

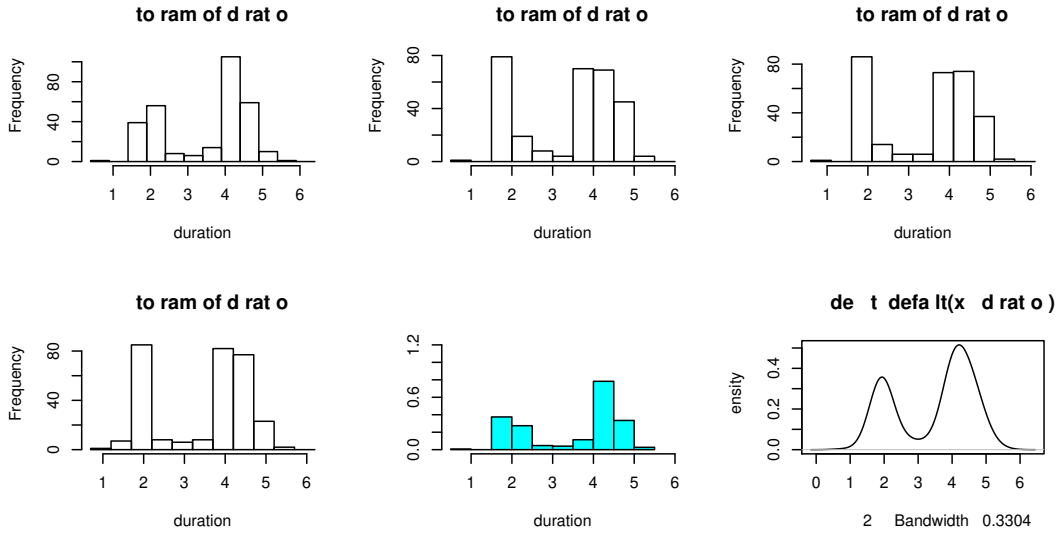
Possible estimators in case B.2:

(1) Histograms. (hist() (not really an estimator of f) or truehist()).

(2) Kernel estimators.

Drawbacks of histograms: It depends too much on the initial point and nclass and is not smooth. See display, two graphs below and their R programs

```
attach(geyser)
geyser[1:3,]
    waiting  duration
1        80  4.016667
2        71  2.150000
3        57  4.000000
length(geyser[,2])
[1] 299
hist(duration,breaks=seq(0.4,6.4,0.5))
hist(duration,breaks=seq(0.5,6,0.5))
hist(duration,breaks=seq(0.6,6.1,0.5))
hist(duration,breaks=seq(0.7,6.2,0.5))
truehist(duration,nbin=15,xlim=c(0.5,6),ymax=1.2)
                  # shaped area =1, # of block ≤ 15
plot(density(duration),lty=1,type="l")
```



Kernel estimators are as the form:

$$\hat{f}(t) = \frac{1}{b}\int K(\frac{x-t}{b})d\hat{F}(x) = \sum_{i=1}^{n} K(\frac{X_i - t}{b})\frac{1}{nb},$$

where $b = width$ (bandwidth), $K(\cdot)$ is a kernel, satisfying $\int K(x)dx = 1$ (and $K(x) \geq 0$). Examples of kernels:

$$g\ (gaussian): \quad K(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$$

$$r\ \ (rectangular): \quad K(x) = \frac{1}{2}\mathbf{1}(|x| \leq 1) \qquad constant$$

$$t\ \ (triangular): \quad K(x) = (1-|x|)\mathbf{1}(|x| \leq 1) \qquad linear$$

$$e\ \ (epanechnikov): \quad K(x) = \frac{3}{4}(1-x^2)\mathbf{1}(|x| \leq 1) \qquad quadratic$$

$$c\ \ (cosine): \quad K(x) = \frac{1}{2}(1+\cos(\pi x))\mathbf{1}(|x| \leq 1)$$

Bandwith selection:

Minimize the mean integrated squared error (MISE)

$MISE = E(\int (\hat{f}(x;b) - f(x))^2 dx)$ **Q:** Why not MSE $E((\hat{f}(x;b) - f(x))^2)$?

$=E(\int \hat{f}^2(x;b) - 2\hat{f}(x;b)f(x) + f^2(x)dx)$ $\hat{f}(x;b) = \frac{1}{n}\sum_{i=1}^{n} K(\frac{X_i - x}{b})\frac{1}{b}$

$=E(\int \hat{f}^2(x;b)dx) - 2\int E(\hat{f}(x;b))f(x)dx + E(\int f^2(x)dx)$

$=E(\int \hat{f}^2(x;b)dx) - 2\int E(K(\frac{X-x}{b}))\frac{1}{b}f(x)dx + E(\int f^2(x)dx)$ $as\ \hat{f}(x;b) = \sum_{i=1}^{n} K(\frac{X_i - x}{b})\frac{1}{nb}$

$=E(\int \hat{f}^2(x;b)dx) - 2E(\int K(\frac{X-x}{b})\frac{1}{b}f(x)dx) + \int f^2(x)dx$

$=E(\int \hat{f}^2(x;b)dx - 2E\hat{f}(X;b) + \int f^2(x)dx)$

$=\frac{1}{nb}\int K^2 + \frac{b^4}{4}\int (f'')^2\{\int x^2K\}^2 + O(\frac{1}{nb} + b^4) + \int f^2$

$\to\infty$ if $b \to 0+$ or $b \to \infty$.

The optimal bandwidth would be

$$b = (\frac{\int K^2}{n\int (f'')^2\{\int x^2K\}^2})^{1/5}$$

with f'' given. Since f'' needs to be estimated, a compromise is

$b = nrd = 1.06min(\hat{\sigma}, IQR/1.34)n^{-1/5}$, where IQR=3rd quantile -1st quantile

Another choice is width="SJ" (Sheather and Jones (1991)).

R code: density(x, ...)

Default S3 method:

density(x, bw = "nrd0", adjust = 1, kernel = c("gaussian", "epanechnikov", "rectangular", "triangular", "biweight", "cosine", "optcosine"), weights = NULL, window = kernel, width, give.Rkern = FALSE, n = 512, from, to, cut = 3, na.rm = FALSE, ...) $512 = 2^9$

One may only set window (or kernel), width and n.

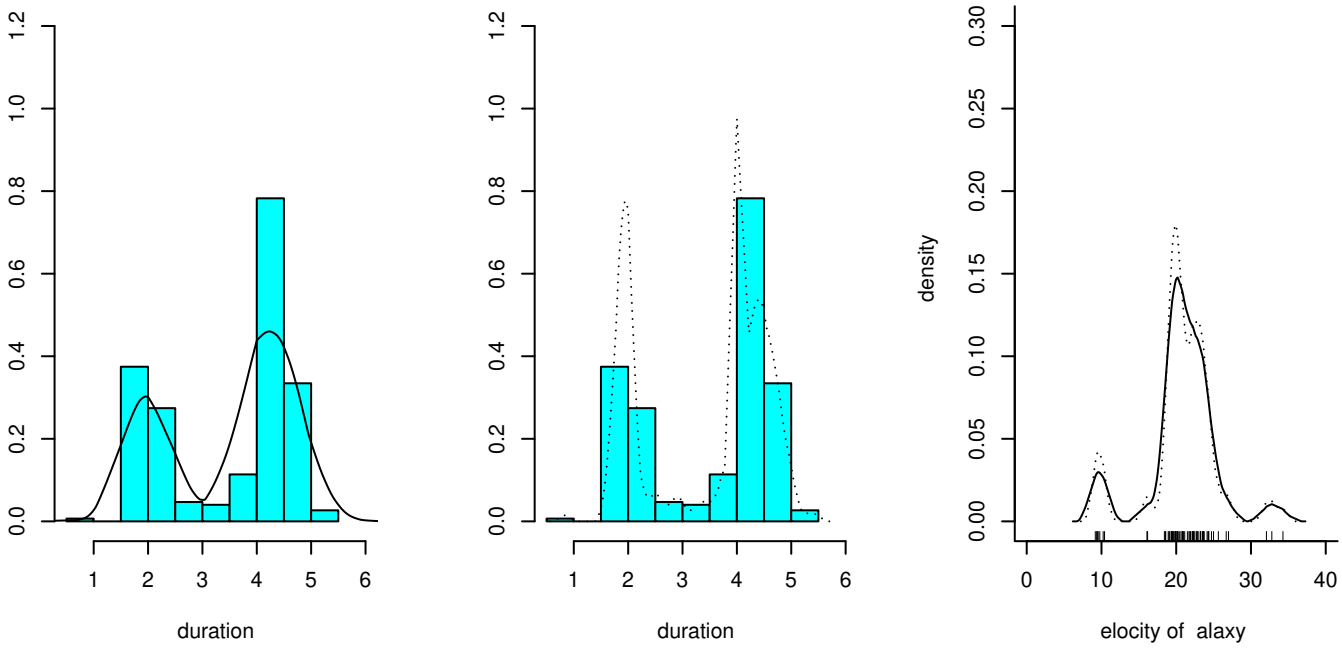
Example. Compute the SD of the sample median, using galaxy data (velocities in km/sec of 82 galaxies), where $\sigma^2 = \frac{1}{4(f(m))^2}$.

```
> min(galaxies)
[1] 9172
> gal=galaxies/1000 #                    due to 9172
> median(gal)
[1] 20.8335
```

> (u=density(gal, from=20.8335, to=20.8335))
Output: .....

x	y
$Min.$: 20.83	$Min.$: 0.1353
$1stQu.$: 20.83	$1stQu.$: 0.1369
⋮	⋮
$Max.$: 20.83	$Max.$: 0.1353

> u$x #≈median(gal)
[1] 20.83
> u$y
[1] 0.1353
> 1/(2 ∗ uy∗sqrt(length(gal))) # $\hat{\sigma}_m = \frac{1}{2\hat{f}(m)\sqrt{n}}$
[1] 0.4079768



Density estimation using geyser data and galaxy data

> par(mfrow =c(1,3))
> truehist(duration,nbin=15,xlim=c(0.5,6),ymax=1.2)
> lines(density(duration,window="triangular",width="nrd")) # n= ?
> truehist(duration,nbin=15,xlim=c(0.5,6),ymax=1.2)
> lines(density(duration,window="triangular",width="SJ",n=256),lty=3) # $n = 2^8$
> gal=galaxies/1000
> plot(x=c(0,40),y=c(0,0.3),type="n",bty="l",xlab="velocity of galaxy",ylab="density")
> rug(gal)
> lines(density(gal,window="triangular",width="SJ",n=256),lty=3)

> lines(density(gal,window="triangular", n=256),lty=1)

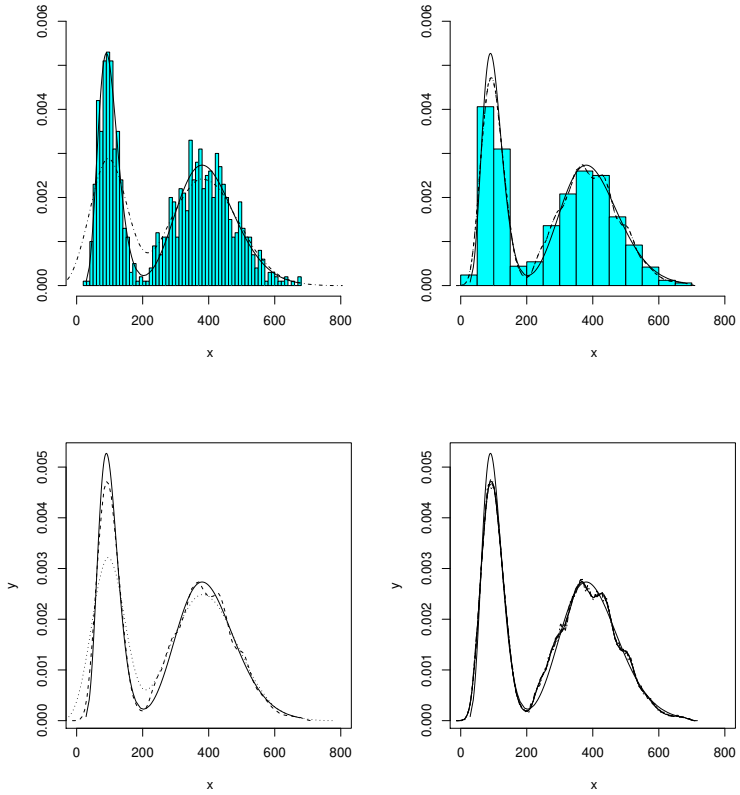
For a given data set, the density estimators vary.

Do we know the true density ?

A simulation study helps us see the difference between the real one and guesses (est).

A simulation study. Generate data from the mixture of two Gamma distributions $Gamma(shape, scale)$. The density is then $f(x) = f_W(x) = 0.4 * f_X(x) + 0.6 * f_Y(x)$, where f_X and f_Y are the densities of $Gamma(10, 10)$ and $Gamma(20, 20)$, respectively, or $W = \begin{cases} X & \text{if } Z = 0 \\ Y & \text{if } Z = 1 \end{cases}$, where $Z \sim bin(1, 0.6)$. Plot and compare the density, the histogram, various estimates of density in the graphs.

```
p=(rbinom(1000,1,0.6)+1)*10                    $ p= ? why ?
x=rgamma(1000,shape=p, scale=p)                      # x=rgamma(1000,p, 1/p)
x=sort(x)
y=0.4*dgamma(x,10,0.1)+0.6*dgamma(x,20,0.05) # x serves as x-axis
truehist(x,nbin=80,xlim=c(0,800),ymax=0.006)
lines(density(x,window="triangular",width="nrd",n=500),lty=4)
lines(x,y,lty=1)
truehist(x,nbin=15,xlim=c(0,800),ymax=0.006)
lines(density(x,window="triangular",width="SJ",n=100),lty=3)
lines(density(x,window="triangular",width="SJ",n=500),lty=2)      little difference
lines(x,y,lty=1)
plot(x,y,type="l",lty=1,xlim=c(0,800))
lines(density(x,window="triangular",width="SJ",n=100),lty=2)
lines(density(x,window="triangular", n=100),lty=3)                    # width=nrd0
plot(x,y,type="l",lty=1,xlim=c(0,800))
lines(density(x,window="g",width="SJ"),lty=4)
lines(density(x,window="c",width="SJ"),lty=5)
lines(density(x,window="r",width="SJ"),lty=6)
lines(density(x,window="t",width="SJ"),lty=7)
lines(density(x,window="e",width="SJ"),lty=2)
lines(x,y,lty=1)
```



Density estimation using simulation data

§5.7. Bootstraping

Q: Why bootstrapping?

Ans: To estimate (1) the variance of an estimator = ? and (2) confidence interval = ?

Under parametric approach, say $X \sim F_o(\cdot;\theta)$, the MLE $\hat{\theta}$ often satisfies $\frac{\hat{\theta}-\theta}{\sigma_{\hat{\theta}}} \xrightarrow{D} N(0,1)$. where $\sigma^2_{\hat{\theta}}$ can be estimated by the observed Fisher information

$$\hat{\sigma}^2_{\hat{\theta}} = -\left(\frac{\partial^2 \log \prod_{i=1}^n f_o(X_i;\theta)}{\partial \theta^2}\bigg|_{\theta=\hat{\theta}} \right)^{-1} \tag{1}$$

and a 95% CI of θ can be approximated by

$$\hat{\theta} \pm 1.96\sqrt{\hat{\sigma}^2_{\hat{\theta}}}. \tag{2}$$

However, Eq. (1) and Eq. (2) may not hold if

under non-parametric approach

or an estimator is not asymptotically normally distributed.

Bootstrap method may provide a solution in such cases.

Suppose we want to estimate θ, by a statistic $\hat{\theta}(\underline{X})$ based on observations $\underline{X} = (X_1, ..., X_n)$.

Method: Random samples with replacement of size n are taken from $\{X_1, ..., X_n\}$ m times,

denoted by $\underline{X}^{(i)} = (X_1^{(i)}, ..., X_n^{(i)})$., $i = 1, ..., m$. $(m \geq 100)$.

$\underline{X}^{(i)}$ is the i-th resampling sample.

One can compute $\hat{\theta}^{(i)} = \hat{\theta}(\underline{X}^{(i)})$ based on the i-th resampling sample.

Thus get m $\hat{\theta}(\underline{X}^{(i)})$s, $i = 1, 2, ..., m$, say $\hat{\theta}^{(i)}$s.

Estimator of $\sigma^2_{\hat{\theta}}$= the sample variance of $\hat{\theta}^{(i)}$'s

Example 1 (simulation study). Derive $\hat{\sigma}_{sample\ median}$.

```
> n=100
> m=1000
> da=rcauchy(n*n)
> dim(da)=c(n,n)
> y=rep(0,n)
> t=rep(0,n)
> res=numeric(m)                    # res=rep(0,m)
> for (j in 1:n) {
    y[j]=median(da[j,])
    for(i in 1:m) res[i] = median(sample(da[j,],replace=T))      # may use apply()
    t[j] = sd(res)                  # from 1000 bootstraped samples
    }
> sd(y)                  # not from bootstrap, not feasible in real data
    [1] 0.150803
> mean(t)
    [1] 0.1689971
> sd(t)
    [1] 0.03216013
> 1/(sqrt(n)*2/pi)                  # 1/(2f(m)√n)          f(x) = 1/(π(1+x^2))
    [1] 0.15708
> sd(y) [1] 0.150803                # not from bootstrap, not feasible in real data
> mean(t) [1] 0.1689971
> sd(t) [1] 0.03216013
> 1/(sqrt(n)*2/pi) [1] 0.15708      # 1/(2f(m)√n)          f(x) = 1/(π(1+x^2))
```

What do sd(y) and mean(t) mean ?

sd(y)= $SE_{sample\ median}$, or $\hat{\sigma}_{sample\ median}$,

mean(t) $\approx$ the average of the bootstraping estimates of $\sigma_{sample-median}$.

How to justify "$\approx$" ?

What does the simulation result suggest ?

Which of 0.150803, 0.1689971 0.15708 is the true value of $\sigma_{sample\ median}$?

The simulation study suggests

$$|\sigma_{median} - \hat{\sigma}_{median}| < \tfrac{1}{3} SD_{\hat{\sigma}_{median}}.$$

Another R function for bootstraping: boot() and boot.ci().

Example 2. Using Galaxies data.

First way:

```
> for(i in 1:m) res[i] = median(sample(gal,replace=T))
> sd(res)
    [1] 0.5254444
```

What is the answer next time ? (1)

The second way:

```
> temp=boot(gal,function(x,i) median(x[i]),R=1000)
> temp
```

Bootstrap Statistics :

	original	*bias*	*std.error*
$t1*$	20.8335	0.0808045	0.5317111

> summary(temp)

	Length	*Class*	*Mode*	
$t0$	1	$-none-$	*numeric*	
t	1000	$-none-$	*numeric*	$= res[1:1000]$
R	1	$-none-$	*numeric*	
data	82	$-none-$	*numeric*	
seed	626	$-none-$	*numeric*	
...				

> sd(temp$t) **=sd(res) ?**

[1] 0.5317111 **Why sd(res)=0.525444 ?** (see Eq.(1))

Recall $\hat{\sigma}_{median} = \frac{1}{2\hat{f}(median)} = 0.4079768.$ **=sd(res) ?**

Which of 0.5254444, 0.5317111, 0.4079768 is the true value ?

> temp$t[998:1000]

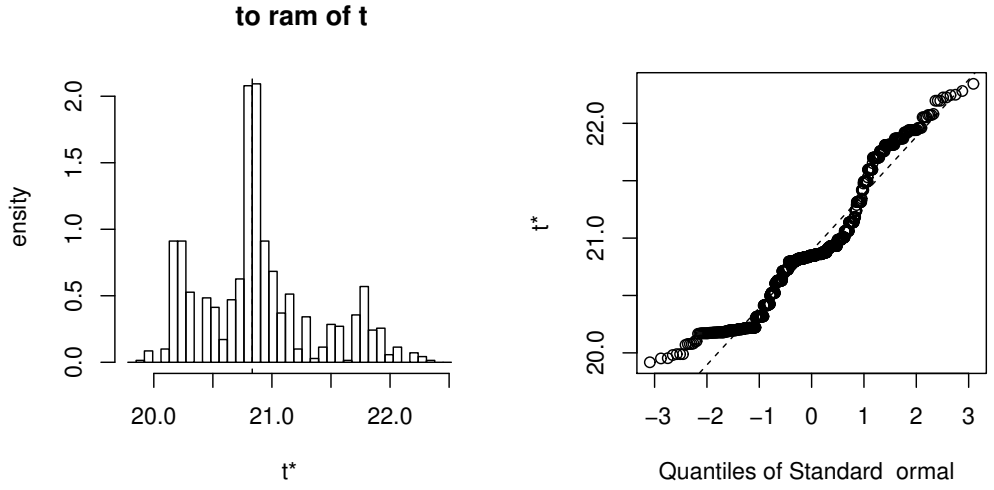
[998,] 20.7120

[999,] 20.7950

[1000,] 21.8675

> plot(temp) # yields the next figure

truehist() and qqplot based on the bootstraping sample of med($\underline{X}$) suggest that the cdf of the med($\underline{X}$) does not have a normal distribution.



Histogram and qqplot of bootstraping med(galaxies/1000)

Remark. The solution to the current homework is in the item "hw-solution" of my website.

Confidence interval of a parameter (L, R):

e.g. 95% approximate CI of θ satisfies $\mathrm{P}\{L < \theta < R\} \approx 0.95$.

If $\hat{\theta}$ is approximately $N(\theta, \sigma^2_{\hat{\theta}})$, then

$(L, R) = (\hat{\theta} - 1.96\hat{\sigma}_{\hat{\theta}},\ \hat{\theta} + 1.96\hat{\sigma}_{\hat{\theta}})$,

as $P\{\hat{\theta} - \theta > 1.96\hat{\sigma}_{\hat{\theta}}\} \approx 0.025$.

$\hat{\theta}$ may not be approximately normal, then there are several approaches:
(1) percentile CI, (2) basic CI and (3) bca.

(1) percentile CI: One expects that $\hat{\theta} \approx \theta$, the edf $\hat{F}_{\hat{\theta}^*}$ of $\hat{\theta}^* \approx$ the cdf of $F_{\hat{\theta}}$,
the empirical quantile $\hat{Q}_{\hat{\theta}^*}(0.025)$ and $\hat{Q}_{\hat{\theta}^*}(0.975)$ satisfies
$P\{\hat{Q}_{\hat{\theta}^*}(0.025) \le \theta \le \hat{Q}_{\hat{\theta}^*}(0.975)\} \approx 0.95$,
thus

$$(L, R) = (\hat{Q}(0.025), \hat{Q}(0.975)) \qquad \text{is called the percentile CI.}$$

L (R) is the 2.5th (97.5th) quantile of the edf of the m $\hat{\theta}_1^*$, ..., $\hat{\theta}_m^*$s (based on bootstrapping).

(2) Basic CI: One expects that
$0.95 = P(a \le \hat{\theta} - \theta \le b) \approx P(a \le \hat{\theta}^* - \hat{\theta} \le b)$.
Thus a and b are the 2.5th and 97.5th quantiles of the edf based on $\hat{\theta}_i^* - \hat{\theta}$, $i = 1$, ..., m.
A 95% CI is $(L, R) \approx (\hat{\theta} - b, \hat{\theta} - a)$. It can be shown that
$(L, R) \approx (2\hat{\theta} - \hat{Q}_{\hat{\theta}^*}(0.975), 2\hat{\theta} - \hat{Q}_{\hat{\theta}^*}(0.025))$.

(3) bca. R also gives another CI denoted by **bca** or **BCa** (see BC_a on page 136 of V&R).

Three programs for bootstraping galaxies data:

```
gal=galaxies/1000
(1) m=1000; res=numeric(m)            # res=rep(0,m)
    for(i in 1:m) res[i] = median(sample(gal,replace=T))
    s=sd(res) # sample SD of sample median
    x=median(gal)
    c(x-1.96*s,x+1.96*s) # normal CI
    (y=quantile(res,p=c(0.025,0.975))) # percentile CI   y=sort(res)[c(25,975)]
    2*x-y[2:1]                        # ??
(2) temp=boot(gal,function(x,i) median(x[i]),R=1000)
    boot.ci(temp, type = c("norm", "basic", "perc", "stud"))
(3) fun = function(d, i) {
        m = median(d[i])
        n = length(i)
        v = (n-1)*var(d[i])/n**2                    # var(x) = 1/(n-1) sum_{i=1}^n (x_i - xbar)^2
        c(m, v)
    }
    temp=boot(gal,fun, R=1000)
    boot.ci(temp, type = c("norm", "basic", "perc", "stud"))
    boot.ci(temp)
```

Output:

```
(1) > c(x-1.96*s,x+1.96*s)
    [1] 19.79584 21.87116
    > (y=quantile(res,p=c(0.025,0.975)))
        2.5%    97.5%
    20.17245 22.05300
    > 2*x-y[2:1]
        97.5%     2.5%
    19.61400 21.49455
(2) > temp=boot(gal,function(x,i) median(x[i]),R=1000)
    > boot.ci(temp, type = c("norm", "basic", "perc", "stud"))
```

BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates
CALL :
boot.ci(boot.out = temp, type = c("norm", "basic", "perc", "stud"))
Intervals :

Level	*Normal*	*Basic*	*Percentile*
95%	$(19.78, 21.81)$	$(19.62, 21.50)$	$(20.17, 22.05)$

(3) > boot.ci(temp, type = c("norm", "basic", "perc", "stud"))
Intervals :

Level	*Normal*	*Basic*
95%	$(19.78, 21.81)$	$(19.62, 21.50)$
Level	*Studentized*	*Percentile*
95%	$(19.57, 21.59)$	$(20.17, 22.05)$

> boot.ci(temp)
Intervals :

Level	Normal	Basic	Studentized
95%	(19.78, 21.81)	(19.62, 21.50)	(19.57, 21.59)
Level	Percentile	BCa	
95%	(20.17, 22.05)	(20.08, 21.92)	

§**5.5. Robust estimators.**
Suppose that X_1, ..., X_n are i.i.d. from a df $f = f(\cdot;\theta)$, $\theta \in \Theta \subset \mathcal{R}^p$.
θ =? (estimation of θ).
Several methods:

1. MLE. $\hat{\theta} = argmax_\theta \prod_{i=1}^n f(X_i;\theta)$.
2. MME. Solve θ through $\overline{X^k} = E_\theta(X^k),\ k = 1, ..., p$.
3. MDE. $\hat{\theta} = argmin_\theta \sum_{i=1}^n |X_i - \theta|$, or
$\hat{\theta} = argmin_\theta \sqrt{\sum_{i=1}^n |X_i - \theta|^2} = argmin_\theta \sum_{i=1}^n (X_i - \theta)^2$ (LSE).
4. Bayes estimator. $\hat{\theta} = argmin_{\tilde{\theta}} r(\pi, \tilde{\theta})$ (r is the Bayes risk of $\tilde{\theta}$),
π is a density function of θ, $r(\pi, \tilde{\theta}) = E((\tilde{\theta}(X) - \theta)^2 | \theta)$
5. Robust estimator ?

We shall introduce through **Location or scale parameter example.**
Suppose X_i are from a cdf F with median m or mean μ, and scale τ or SD σ.
Example 1. Exp(1) distribution. μ_X =1, median m =ln2 and σ_X =1.
The mean and median are both called location or center.
The SD is called a scale.
Exp(θ) distribution with d.f. $f = \frac{1}{\theta} e^{-x/\theta} \mathbf{1}(x > 0)$. $\mu_X = \theta$, median m =ln2/θ and $\sigma_X = \theta$.
The mean and median are now called location parameters.
σ_X is called a scale parameter.
Example 2. Cauchy Distribution $f(x;\theta,\tau) = \frac{1}{\tau} f_o(\frac{x-\theta}{\tau})$, where $f_o(x) = \frac{1}{\pi(1+x^2)}$.
θ is the median, a location parameter, the mean = ?
τ is a scale parameter, the standard deviation = ?
If the distribution is symmetric about the center (e.g., $N(\mu,\sigma)$ with df $f = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$),
the mean and median are the same (**Is it correct ??**)
$\overline{X}$ and med(X) are two location estimators.

$S = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2}$ is a scale estimator.

Note that μ and *median* are often quite close.

$\overline{X} \neq \mu$! (estimate $\hat{\theta} \neq \theta$!)

Example 3. (A simulation study on N(,) and Exp()).

> x=rnorm(10)

> x=c(x,100) an outlier to 10 X_i's from N(0,1)

The data x can be roughly viewed as a random sample from

$F_X(t) = \frac{1}{11}[\mathbf{1}(t \geq 100) + 10\Phi(t)]$, where $\Phi(t)$ is the cdf of $N(0,1)$.

$0.5 = F_X(m) = \frac{1}{11}[\mathbf{1}(m \geq 100) + 10\Phi(m)]$,

$$\frac{1}{11}[\mathbf{1}(t \geq 100) + 10\Phi(t)] \begin{cases} \geq \frac{1}{11} + \frac{10}{11}0.5 > 0.5 & \text{if } t \geq 100 \\ = \frac{10}{11}\Phi(m) & \text{if } t < 100 \end{cases} \quad (1)$$

> mean(x) =E(X) ?

[1] 9.138084 # compare to the mean of N(0,1)

> median(x) = median of X ?

[1] 0.1937377 # compare to the median of N(0,1)

m (= median of X)= ? $m < 100$ or m≥100 ?

$0.5 = F_X(m) = \frac{1}{11}[0 + 10\Phi(m)]$

$\Rightarrow \Phi(m) = 0.5 * 11/10 = 0.55$ (see Eq. (1)).

> qnorm(.55) # =median of X

[1] 0.1256613

> 100/11 # $E(X) = 0 + 100 \times \frac{1}{11}$

[1] 9.090909

	mean	*median*
$N(0,1)$	0	0
F_X	9.0909	0.1257
data x	9.138	0.1937

> x=rexp(40)

> x=c(x,100) an outlier to 40 data from Exp(1)

> mean(x)

[1] 3.254283 # compare to the mean of Exp(1) = ?

> median(x)

[1] 0.6262377 # compare to the median of Exp(1) *i.e.*, $\log(2) \approx 0.693$

In the above examples, 100 in x is called an outlier.

Observation:

med(X) is less sensitive to outliers than $\overline{X}$.

	$N(0,1)$	$Exp(1)$		$N(0,1)$	$Exp(1)$
$\overline{X}$:	9.1	3.3	$med(x)$:	0.19	0.63
mean	0	1	*median*	0	0.69

Outliers distort some estimators greatly. In fact, given an outlier X_1 in $X_1, ..., X_n$,

$$\lim_{X_1 \to \pm\infty} \overline{X} = \pm\infty$$

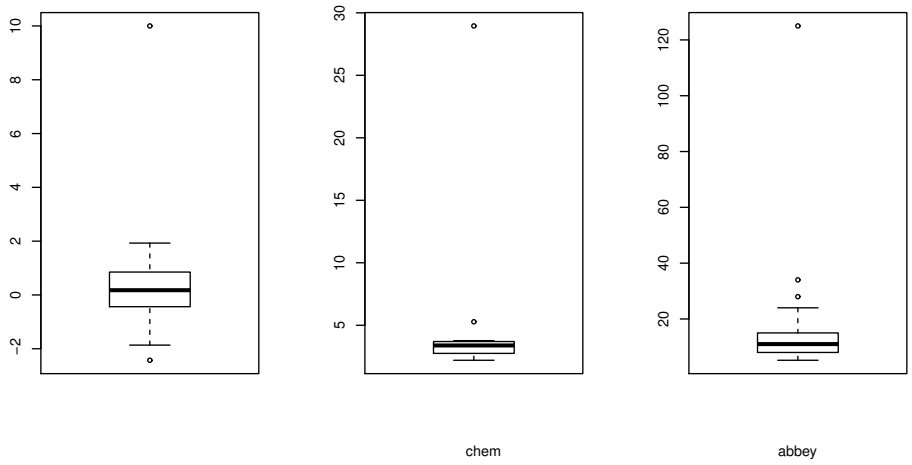
but $\lim_{X_1 \to \pm\infty}$ med(X) is finite (=med($\{X_2, ..., X_n\}$)).

Q: How to quantify outliers ?

Use boxplot for detecting outliers.

Example 4. Boxplots of data chem and abbey (in R).

```
> summary(chem)                    # (24 determinations of copper in wholemeal flour)
   Min.  1st Qu.  Median   Mean  3rd Qu.   Max.
  2.200   2.775   3.385   4.280   3.700  28.950
> summary(abbey)                   # (31 Daily Price Returns Of Abbey National Shares)
   Min.  1st Qu.  Median   Mean  3rd Qu.   Max.
  5.20    8.00    11.00   16.01   15.00  125.00
> x=c(rnorm(40),10)
> boxplot(x)
> boxplot(chem, sub="chem")
> boxplot(abbey, sub="abbey")
```



Def. If an observation is more than 4 IQR (inter-quartile-range or $Q_3 - Q_1$) away from the center of the data, it is called an outlier.

center —- median

lower and upper hinges —– 1st and 3rd quartiles

whiskers ———- 1.5 IQR from hinges (unless max or min $<$ 1.5 IQR).

Observations outside whiskers are suspected outliers.

It is definitely an ourlier if it is $\begin{cases} 3 \text{ IQR away from hinges, or} \\ 3 \text{ SD away from the center.} \end{cases}$

Outliers may due to typos, errors or maybe true observations (**why ?**)

The sample can be viewed as from $X = \begin{cases} U & \text{if } Y = 0 \\ V & \text{if } Y = 1, \end{cases}$ or

$F_X(x;\theta) = (1-\alpha)F_U(x;\theta) + \alpha F_V(x)$, $\alpha \in [0, \epsilon)$, and $Y \sim bin(1, \alpha)$.

Q: How to quantify that an estimator is less sensitive to outliers ?

Ans: Stability and resistancy.

Def. An estimator is **stable** if it does not change by a large amount when **one** outlier is added in. In particular, the estimator is bounded no matter what the outlier is equal to.

The **breakdown value** (or point) of an estimator is the supremum value of the proportion p of the sample that can be moved to ∞ without moving the statistic to ∞ (in the case that the sample size is as large as necessary).

An estimator with a large breakdown point is said to be "**resistant** (to gross errors)".

Remark. *Definition from "Robust Nonparametric Statistical Methods"*
by T.P. Hettmansperger and J.W. Mckean (1998) is as follows:

Asymptotic breakdown point.

Let $\mathbf{x} = (x_1, ..., x_n)$ *represent a realization of a sample and let*

$$\mathbf{x}^{(m)} \in \mathcal{R}^n \text{ represent the corruption of any } m \text{ of the } n \text{ observations,} \tag{1}$$

that is,

x_{i_1}, ..., $x_{i_{n-m}}$ *among* $\{x_1, ..., x_n\}$ *are fixed at their original values,*
but the rest m *observations are changing (possibly to* $\pm\infty$*).*

Let $\mathcal{X}$ *be the collection of all combination of choosing* m x_i*'s among* $\{x_1, ..., x_n\}$ *to be corrupted with the rest fixed. Of course,* $\mathcal{X}$ *depends on the sample and it contains sequences of elements that* $||\mathbf{x}^{(m)}||$ *(see EQ. (1)) tends to* ∞*. We define the bias of an estimator* $\hat{\theta}$ *by*

$$bias(m; \hat{\theta}, \mathbf{x}) = \sup\{|\hat{\theta}(\mathbf{x}^{(m)}) - \hat{\theta}(\mathbf{x})| : \ \mathbf{x}^{(m)} \in \mathcal{X}\} \qquad (see\ EQ.\ (1))$$

If the bias is infinite, we say the estimator has broken down and the

$$\textbf{sample breakdown value} = \min\{m/n : \ bias(m; \hat{\theta}, \mathbf{x}) = \infty\}$$

Its limit as $n \to \infty$*, if it exists, is called the (asymptotic) breakdown value,*

$$p = \textbf{breakdown value} = \lim_{n\to\infty} \min\{m/n : \ bias(m; \hat{\theta}, \mathbf{x}) = \infty\}$$

For med(X), p=50%.

e.g., if n is 5 and less than half of the sample are moved to $+\infty$,
med(X) is bounded by original values of $X_{(1)}$ and $X_{(n)}$.
However, if 3 observations are moved to $+\infty$, $med(X) \to \infty$.
Thus the sample breakdown value of med(X) is 3/5 if $n = 5$,
and is $\begin{cases} \frac{1+n/2}{n} & \text{if } n \text{ is even} \\ \frac{n+1}{2n} & \text{if } n \text{ is odd.} \end{cases}$

The limit or the (population) breakdown value is thus $p = 1/2$.

For $\overline{X}$, p=0 (asymptotically), as the sample breakdown value is $1/n$.

Remark.
Med(X) is stable, but $\overline{X}$ is not, as Med(X) is bounded no matter what the ourliers is.
Med(X) is resistant, but $\overline{X}$ is not, as the asymptotic breakdown point is p=0.5.

Q: Robustness ?

Def. Let $\theta \in \mathcal{R}$, the relative efficiency (RE) of $\tilde{\theta}$ to $\hat{\theta}$ is
$RE(\tilde{\theta}, \hat{\theta}) = \lim_{n\to\infty}(\sigma_{\hat{\theta}}/\sigma_{\tilde{\theta}}) = \lim_{n\to\infty}(\hat{\sigma}_{\hat{\theta}}/\hat{\sigma}_{\tilde{\theta}})$ a.e.,

where $\hat{\sigma}^2_{\hat{\theta}}$ – estimator of the asymptotic variance of $\hat{\theta}$ ($\lim_{n\to\infty} \hat{\sigma}_{\hat{\theta}}/\sqrt{Var(\hat{\theta})} = 1$), ...

Recall that the Fisher information matrix is $I = E(\frac{\partial \ln f(X;\theta)}{\partial \theta} \frac{\partial \ln f(X;\theta)}{\partial \theta'})$ $(= -E(\frac{\partial^2 \ln f(X;\theta)}{\partial \theta \partial \theta'})$ under certain conditions). Under the exponential family, if $\hat{\theta}$ is the MLE of $\theta \in \mathcal{R}^p$, then

$n\Sigma^2_{\hat\theta} \approx I^{-1} = (E(\frac{d\ln f(X;\theta)}{d\theta}\frac{d\ln f(X;\theta)}{d\theta'}))^{-1}$ $\Sigma^2_{\hat\theta}$ is the covariance matrix of $\hat\theta$.
(as $n\Sigma^2_{\hat\theta} \xrightarrow{P} (E(\frac{d\ln f(X;\theta)}{d\theta}\frac{d\ln f(X;\theta)}{d\theta'}))^{-1}$).
$\sum_{i=1}^n \frac{d\ln f(X_i;\theta)}{d\theta}\frac{d\ln f(X_i;\theta)}{d\theta'} \to nE(\frac{d\ln f(X;\theta)}{d\theta}\frac{d\ln f(X;\theta)}{d\theta'})$ a.s. ? **Are these 2 right ?**
$\hat\Sigma^2_{\hat\theta} = (\sum_{i=1}^n \frac{d\ln f(X_i;\theta)}{d\theta}\frac{d\ln f(X_i;\theta)}{d\theta'})^{-1}|_{\theta=\hat\theta}$? or $\hat\Sigma^2_{\hat\theta} \approx (\sum_{i=1}^n \frac{d\ln f(X_i;\theta)}{d\theta}\frac{d\ln f(X_i;\theta)}{d\theta'})^{-1}|_{\theta=\hat\theta}$?
Which is right ?
If $\theta \in \mathcal{R}$, then $\hat\sigma^2_{\hat\theta} = (\sum_{i=1}^n (\frac{d\ln f(X_i;\theta)}{d\theta})^2)^{-1}|_{\theta=\hat\theta}$?
$\hat\sigma^2_{\hat\theta} = (\sum_{i=1}^n \frac{-d^2\ln f(X_i;\theta)}{d\theta^2})^{-1}|_{\theta=\hat\theta}$?
$ARE(\tilde\theta, \hat\theta)$ — asymptotic RE of $\tilde\theta$ to $\hat\theta$.
Recall: How to quantify that an estimator is less sensitive to outliers ?
Ans: Stability and resistancy.
Def. An estimator is **stable** if it does not change by a large amount when **one** outlier is added in. In particular, the estimator is bounded no matter what the outlier is equal to.
The **breakdown value** (or point) of an estimator is the supremum value of the proportion p of the sample that can be moved to ∞ without moving the statistic to ∞ (in the case that the sample size is as large as necessary).
An estimator with a large breakdown point is said to be "**resistant** (to gross errors)".

Robust method studies how to find a stable or resistant estimator $\tilde\theta$
with a **large** ARE$(\tilde\theta, \hat\theta)$ $(\approx \frac{\sigma_{\hat\theta}}{\sigma_{\tilde\theta}})$ to a (possibly efficient or standard) estimator $\hat\theta$
under $F(\cdot;\theta) = (1-\alpha)F_o(\cdot;\theta) + \alpha F_1$.
The resulting estimator is called a **robust** estimator.
It is often that the standard situation is under the normal assumption.
Example 5. If X has the density $f(x) = f_o(x-\mu)$ (here $\alpha = 0$ see the definition above),

$$\text{ARE(med(X),}\overline{X}) = \begin{cases} 64\% & \text{if } f_o \text{ is } N(0,\sigma^2) \\ 96\% & \text{if } f_o \text{ is } t_5 \\ > 1 & \text{if } f_o = exp(-|x|)/2 \\ & \text{(the standard double exponential (DE) distribution)} \end{cases}$$

They are all symmetric distributions and thus $m = \mu$.
Homework 5.5.1. Recall the exponential family: $f(x;\theta) = h(x)c(\theta)exp(\sum_{j=1}^k w_j(\theta)t_j(x))$.
Prove or disprove the following two statements:
1. $f(x;\theta) = exp(-(x-\theta))$, $x > \theta$, belongs to the exponential family.
2. The double exponential distribution $f(x;\theta) = e^{-|x-\theta|}/2$ belongs to the exponential family.

Q: What are the candidates of robust estimators ? (of location, scale, or others)
A class of location estimators:
M-estimators (MLE-like estimators).
Consider a location parameter related to $f(x-\mu)$, where
$f(x)$ is a density with $\int f(x)dx = 1$.
The MLE of μ satisfies:

$$\hat\mu = \begin{cases} argmin_\mu(-\ln\prod_{i=1}^n f(X_i-\mu)) \\ zero.point_\mu \sum_{i=1}^n (\ln f(X_i-\mu))' \end{cases} \text{ if the zero point exists. } \textbf{always exists ?}$$

$$<=> \quad \hat\mu = argmin_\mu \sum_{i=1}^n \rho(X_i-\mu) \text{ with } \rho = -\ln f \tag{1}$$

$$\hat{\mu} = zero.point_{\mu} \sum_{i=1}^{n} \psi(X_i - \mu) \text{ with } \psi = \frac{f'(X_i - \mu)}{f(X_i - \mu)} = \rho' \text{ if the zero point exists.}$$

Examples of (ρ and ψ):

$$\begin{cases} \text{The MLE } \overline{X} \text{ under } N(\mu, \sigma^2): & \rho \propto x^2/2 \ \& \ \psi(x) = x. \\ \text{The MLE med(X) under the DE (double exponential)} & \rho \propto |x| \ \& \ \psi(x) = sign(x) \end{cases} \quad (2)$$

DE: $f(x) = \frac{1}{2\tau} e^{\frac{-|x-\mu|}{\tau}}$.

Example 6. med(X): $\rho = |x|$ and $\psi(x) = sign(x)$ (see Eq.s (1) and (2) above). That is,

$$med(X) = zero.point_{\mu} \sum_{i=1}^{n} \psi(X_i - \mu) = zero.point_{m} \sum_{i=1}^{n} \text{sign}(X_i - m)$$

$zero.point_{m} \sum_{i=1}^{n} \text{sign}(X_i - m) =$ **?? How to derive it ?**

If $n = 4$, $X_i = i$, Med(X)= $(X_{(2)} + X_{(3)})/2$ or $\in (X_{(2)}, X_{(3)})$.

$zero.point_{\mu} \sum_{i=1}^{n} \psi(X_i - \mu) = (X_{(2)} + X_{(3)})/2$, as

$$\sum_{i=1}^{n} \text{sign}(X_i - m) \begin{cases} > 0 & \text{if } m \le X_{(2)} \\ = 0 & \text{if } m \in (X_{(2)}, X_{(3)}) \\ < 0 & \text{if } m \ge X_{(3)} \end{cases} \textbf{ Is it right ?}$$

If $n = 3$, Med(X)= $X_{(2)} = zero.point_{m} \sum_{i=1}^{n} \text{sign}(X_i - m)$, as

$$\sum_{i=1}^{n} \text{sign}(X_i - m) \begin{cases} > 0 & \text{if } m < X_{(2)} \\ = 0 & \text{if } m = X_{(2)} \\ < 0 & \text{if } m > X_{(2)} \end{cases} \textbf{ Is it right ?}$$

Homework 5.5.2. Answer the previous two quetions.

If $n = 5$, X_i's are 1, 2, 2, 3, 4, Med(X)= **??**

$zero.point_{m} \sum_{i=1}^{n} \text{sign}(X_i - m) =$ **?? How to derive it ?**

$$\sum_{i=1}^{5} \text{sign}(X_i - m) \begin{cases} = 5 & \text{if } m < 1 \\ = 4 & \text{if } m = 1 \\ = 3 & \text{if } m \in (1, 2) \\ = 1 & \text{if } m = 2 \\ = -1 & \text{if } m \in (2, 3) \\ = -2 & \text{if } m = 3 \\ = -3 & \text{if } m \in (3, 4) \\ = -4 & \text{if } m = 4 \\ = -5 & \text{if } m > 4. \end{cases} \quad (1)$$

There is no solution in this case to $zero.point_{m} \sum_{i=1}^{n} \text{sign}(X_i - m)$

How to handle it ??

Interpret $zero.point_{m} \sum_{i=1}^{n} \text{sign}(X_i - m)$ as

$$zero.crossing.point \sum_{i=1}^{n} \text{sign}(X_i - m)$$

Remark. The solution in Eq.(1) above is $zero.crossing.point \sum_{i=1}^{n} \text{sign}(X_i - m) = 2$.

Remark: The MLE under $N(\mu, \sigma^2)$ is $\overline{X}$, which is not robust if $X \not\sim N(\mu, \sigma^2)$.

The MLE under the DE is med(X), which is robust even if X is no longer the DE.

But they motivate the MLE-like function ρ and the score function ψ ($= \rho'$) as follows, where the ρ does not have to related to $-\ln f$, and ψ does not need to related to $-(\ln f)'$.

Other M-estimators:

Metric trimming M-est: (by Huber) (robust) (bisquare)

$$\psi(x) = \begin{cases} x & |x|/c < 1 \\ 0 & \text{otherwise} \end{cases} \text{ (delete large outliers).} \qquad (v.s.\ \phi(x) = x)$$

Metric Winsorizing M-est: (by Huber) (robust) (attributed to C.P. Winsor)

$$\psi(x) = \begin{cases} -c & x < -c \\ x & |x|/c < 1 \\ c & x > c \end{cases}. \qquad (v.s.\ \phi(x) = x)$$

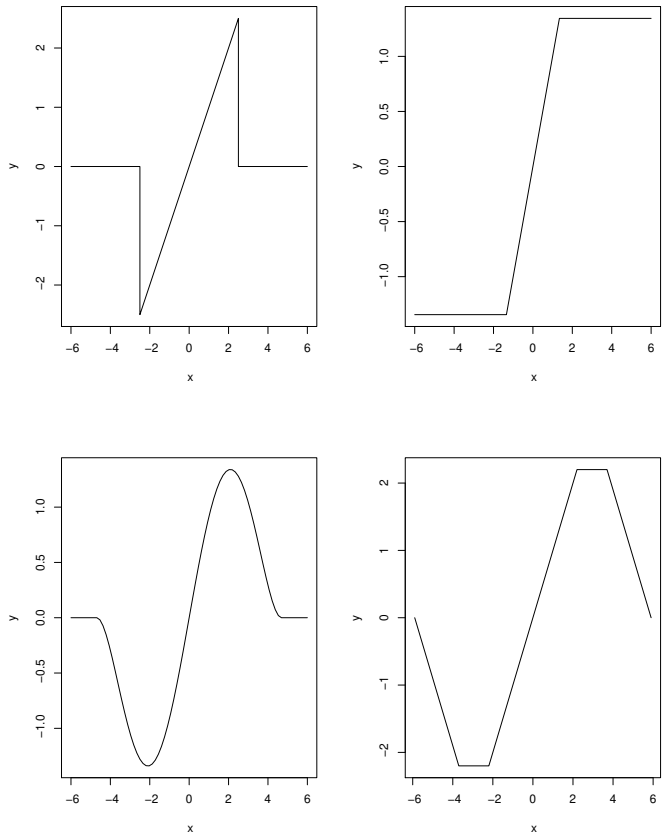
(bring large outliers to $\mu \pm c$). ARE to $\overline{X}$ is 95% under $N(\mu, \sigma^2)$ if $c = 1.345$.
Tukey's biweight M-est.:

$$\psi(x) = x\big[1 - (x/R)^2\big]_+^2$$

where $[x]_+ = x\mathbf{1}(x > 0)$. The value $R = 4.685$ gives 95% ARE at the normal.

Hampel's M-est.: $\psi(x) = \begin{cases} x & 0 < |x| < a \\ \text{sign}(x) \cdot a & a < |x| < b \\ \text{sign}(x)\frac{a(c-|x|)}{c-b} & b < |x| < c \\ 0 & \frac{|x|}{c} > 1 \end{cases}$ e.g., $a = 2.2s$, $b = 3.7s$, $c = 5.9s$.

Remark. There is a scaling problem above (c, R and s are unknown).
It can be replaced by an estimate of a scale parameter (introduced next).



Graph of the last 4 score functions.

Possible estimators of scale parameter:

$$\textbf{non-robust}\begin{cases} S=\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i-\overline{X})^2} & \text{(breakdown point } p=0\text{).}\\ \hat{\sigma}_m=\frac{1}{n}\sum_{i=1}^{n}|X_i-\overline{X}|\sqrt{\pi/2} & \text{(breakdown point } p=0\text{)}\end{cases}$$
$$\textbf{robust}\begin{cases} \text{mad(X)}=med(|X-med(X)|)/0.6745\\ \hat{\sigma}_q=IQR/1.35\end{cases} \qquad (1)$$

The coefficients are made so that they equal σ under the normal distribution.

> huber(y, k = 1.5, tol = 1e-06) #2 k: Winsorizes at k standard deviations
Finds the Huber M-estimator of location with mad scale (see EQ. (1)).

> hubers(y, k = 1.5, mu, s, initmu = median(y), tol = 1e-06) #1
Finds the Huber M-estimator for location with scale specified,
scale with location specified, or both if neither is specified.

```
> mad(x, center = median(x), constant = 1.4826)
> length(chem)
[1] 24
> x=sort(chem)
> mean(x)
[1] 4.280417
> mean(x[2:23])
[1] 3.253636
> mean(chem,trim=0.05)
[1] 3.253636
> mean(chem,trim=0.1)
[1] 3.205
> median(x)
[1] 3.385
> median(x[2:23]) = ?
> sd(chem)
[1] 5.297396
> mad(chem)
[1] 0.526323
> unlist(huber(chem))                    # 2 Huber Winsorizing M-est.
       mu         s
 3.206724  0.526323                       # compare to mad(chem) [1] 0.526323
> huber(chem)$mu
[1] 3.206724
> unlist(hubers(chem))                   # 1 Huber, robust estimates
       mu         s
 3.205498  0.673652                       # compare to mean(chem,trim=0.1) [1] 3.205
> fitdistr(chem,"t",list(m=3,s=0.5),df=5)
         m          s
 3.1853947  0.6422023                    MLE
(0.1474804) (0.1279530)
> fitdistr(chem,"t",df=5)                # same results
```

§6.9. A Comment on the MLE with regression data.
Given data X_1, ..., X_n, one can fit them to a certain parametric distribution and obtain the MLE of the parameter (using R program fitdist(), which contains 18 distributions).

Now suppose that (X_1, Y_1), ..., (X_n, Y_n) be i.i.d. observations from $F_{X,Y}$, where $X \in \mathcal{R}^p$, a column vector. Assume that

(*) (X_i, Y_i) satisfies the linear regression model:

$$Y_i = X_i'\beta + \alpha + \epsilon_i, \qquad (\mathbf{Y} = \mathbf{X}_{n\times(p+1)} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} + \mathbf{e}) \tag{1}$$

where ϵ_is are i.i.d. with $E(\epsilon_i|X_i) = 0$ and $\sigma^2 = Var(\epsilon_i|X_i)$?? or

ϵ_is are i.i.d., $E(\epsilon_i) = 0$ and $\sigma^2 = Var(\epsilon_i)$ **?? Which is a correct assumption ?**

The LSE minimizes (α, β) over $\sum_{i=1}^n (Y_i - \alpha - \beta' X_i)^2$,

$$\begin{pmatrix} \hat\alpha \\ \hat\beta \end{pmatrix} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Y}, \text{ where } \mathbf{X} = \begin{pmatrix} 1 & X_1' \\ \vdots & \vdots \\ 1 & X_n' \end{pmatrix} \text{ and } \mathbf{Y} = (Y_1, ..., Y_n)' \text{ **if ..... ?**} \tag{2}$$

$$\hat\alpha = \overline{Y} - \overline{X'}\hat\beta,$$

$$\hat\beta = \left(\overline{XX'} - \overline{X}(\overline{X'})\right)^{-1}(\overline{XY} - \overline{X}(\overline{Y})), \qquad \frac{\sum_i X_iY_i/n - \overline{X}\cdot\overline{Y}}{\sum_i X_i^2/n - (\overline{X})^2}$$

$$\hat\beta \overset{a.s.}{\to} \beta^* = \Sigma^{-1}(E(XY) - E(X)E(Y)) \quad (\beta^* = \beta\ ?)\ \ \Sigma = Cov(X), \tag{3}$$

if the expectations exist. This is due to **????**

Remark 1. Equations (2) and (3) do not rely on assumption (*).
It is often to further assume $\epsilon_i \perp X_i$ and/or $\epsilon_i \sim N(0, \sigma^2)$, as assumed hereafter in §6.9.

Remark 2. $Y = X'\beta + \alpha + \epsilon$

$<=>$ $Y - X'\beta - \alpha = \epsilon$ and $f_{Y|X}(t|x) = f_\epsilon(\underbrace{t - \alpha - x'\beta}_{u})$.

$<=>$ $Y - X'\beta = W$ and $f_{Y|X}(t|x) = f_W(t - x'\beta)$. W = ??

Example 1. If $Y|(X = x) \sim N(\beta' x + \alpha, \sigma^2)$, then $W \sim N(\alpha, \sigma^2)$. $Y = \beta' X + \overbrace{\alpha + \epsilon}^{W}$.

Example 2. If $Y = \beta' X + W$, $W \sim Exp(1)$, then

$$f_{Y|X}(t|x) = f_W(t - \beta' x) = e^{-(t-\beta' x)}, \quad t > ??$$

Does $Y|X$ have an Exponential distribution ? $(i.e.,\ f(t) = \theta t^{-\theta t},\ t > 0)$.

Abusing notations, write $X\beta = X'\beta$ etc.

Remark 3. $\ln Y = \beta X + \alpha + \epsilon$,

$<=>$ $Y = e^{\beta X} W$ $<=>$ $Y/e^{\beta X} = W$ $<=>$ $Y = e^{\alpha+\beta X} T$ $\quad (T = e^\epsilon)$

$<=>$ $f_{Y|X}(t|x) = f_W(\frac{t}{e^{\beta x}})$ $(= f_W(w))$, and $\ln W = \alpha + \epsilon$.

Example 3. If $S_{Y|X}(t|x) = \exp(-e^{-\beta x}t) = S_W(t/e^{\beta x})$, $t > 0$, then $W \sim$?

$Y|(X = x) \sim$??

$Y = W/e^{\beta X}$ **? or** $Y = We^{\beta X}$ **?**

$\ln Y = -\beta X + \alpha + \epsilon$? or $\ln Y = \beta X + \alpha + \epsilon$?

Remark 4. If $Y = \beta X + \alpha + \epsilon$, then the LSE
$\hat{\alpha} = \overline{Y} - \hat{\beta}\overline{X}$, where $\hat{\beta}$ is as in Eq. (2) *i.e.*, $(\hat{\alpha}, \hat{\beta}')' = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$,
The LSE satisfies $E(\hat{\beta}) = \beta$ and $V(\hat{\alpha}, \hat{\beta}'|\mathbf{X}) = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}$.
If $\epsilon \not\sim N(0, \sigma^2)$, then the anova table is not valid as it is based on F distribution; and the LSE is not an efficient estimator.

One can consider regression models under the parametric assumption.
Assume that $Y_i|(X_i = x) \sim F$, where $F = F_o(y|x, \beta)$ has a parametric form,
and F_o is known except β.

Then in order to find $\hat{F}$, it suffices to find $\hat{\beta}$.
A standard estimator is the MLE, that maximizes
Ł$(b) = \prod_{i=1}^n f_o(Y_i|\mathbf{X}_i, b)$,
where f_o is the density of F_o and $S_o = 1 - F_o$.

The MLE is efficient if the parametric assumption is valid and certain regularity conditions are satisfied.

1. Gaussian distribution

Common form $f(t) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(t-\mu)^2}{2\sigma^2})$.
With covariate in Splus or R, reparametrization:
$f_Y(y|\mathbf{x}, \beta) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(y-\beta'\mathbf{x})^2}{2\sigma^2})$ ($\beta'\mathbf{x}$ is a location)
or $Y = \beta'\mathbf{x} + \sigma Z$, $Z \sim N(0, 1)$. $E(Z) = 0$ **???**

2. Exponential distribution

Common form $S(t) = exp(-t/\theta)$, $t > 0$.
With covariate in Splus or R, reparametrization:
$S_Y(y|\mathbf{x}, \beta) = exp(-e^{-\beta'\mathbf{x}}y)$, $y > 0$. $E(Y|\mathbf{x}, \beta) = e^{\beta x}$.
$\ln Y = \beta'\mathbf{x} + \ln Z$, $Z \sim Exp(1)$. $E(\ln Z) = 0$ **???**
$E(\ln Z) = \int \cdots$??

$E(\ln(Z)) \approx -0.577$.
$Y = Z/e^{-\beta'\mathbf{x}}$, or $W = Y/e^{\beta'\mathbf{x}}$ ($\beta'\mathbf{x}$ is not a location of $\ln Y$)

3. Weibull distribution

Common form $S(t) = exp(-t^\gamma/\theta) = exp(-(t/\mu)^{1/\tau})$, $t > 0$.
With covariate in Splus or R, reparametrization:
$S_Y(y|\mathbf{x}, \beta) = exp(-(e^{-\beta'\mathbf{x}}y)^{1/\tau})$, $y > 0$.
$\ln Y = \beta'\mathbf{x} + \tau \ln Z$, $Z \sim Exp(1)$. $E(\tau \ln Z) = 0$ **???**
$Z = (e^{-\beta'\mathbf{x}}Y)^{1/\tau}$.

4. Logistic distribution

Common form $S(t) = \frac{1}{1+exp(t)}$,
With covariate in Splus or R, reparametrization:
$S_Y(y|\mathbf{x}, \beta) = \frac{1}{1+exp(\frac{y-\beta'\mathbf{x}}{\tau})}$,
$Y = \beta'\mathbf{x} + \tau Z$, $Z \sim logistic(0, 1)$. $E(Z) = 0$, $\sigma_Z = \pi/\sqrt{3}$.

5. Lognormal distribution

Assume $\ln Y = \beta'\mathbf{x} + \sigma Z$, where $Z \sim N(0, 1)$. $E(Z) = 0$ **???**

6. Loglogistic distribution

$\ln Y = \beta'\mathbf{x} + \tau Z$, $Z \sim logistic(0, 1)$. $E(Z) = 0$ **???**

Remark 5. About $\ln Y = \beta X + Z$. If $f_{Y|X}(t|x) = e^{-\beta x} f_o(te^{-\beta x})$, where f_o is a df.

Then $U = e^Z = Y/e^{\beta x}$ has d.f. f_o.

This is due to u-substitution.

$$\int_{-\infty}^{y} e^{-\beta x} f_o(t/e^{\beta x})dt = \int_{-\infty}^{y/e^{\beta x}} f_o(u)du, \qquad \text{where } u = t/e^{\beta x}.$$

R command:

The parametric MLE is efficient under certain regularity assumptions. In particular,

if the residual plot suggests that certain parametric family is plausible,

one can obtain the MLE by R codes to data (x,y): (1) survreg() (2) glm().

(1) survreg():

zz=survreg(Surv(y)∼x, dist="exponential")

dist: (default: weibull), gaussian, logistic, lognormal and loglogistic

(2) glm(): The generalized linear model includes a subset of the exponential family, which are also parametric distributions. The conditional distribution $f_{Y|X}$ may not satisfy the linear regression model or log linear regression model. We can compute the MLE of the parameters based on regression data. The GLM includes

$N(\mu, \sigma^2)$,

$G(\alpha, \beta)$,

$bin(m, p)$

Poisson(μ)

inverse-Gaussian $\qquad f(x) = \sqrt{\frac{\lambda}{2\pi x^3}} \exp(-\frac{\lambda(x-\mu)^2}{2\mu^2 x})$.

Only $N(\mu, \sigma^2)$ is considered in both glm() and survreg(). Here we only discuss $G(\alpha, \beta)$ under the glm. The glm is specified by

$$\ln f_{Y_i|X_i}(y|x) = \frac{A_i}{\phi}[y\theta + \gamma(\theta)] + \tau(y, \frac{A_i}{\phi}), \ ((\theta, \phi) \text{ is a function of } (\alpha, \beta, x)).$$

The gamma distribution family belongs to the generalized linear model (glm) but does not satisfy the ordinary linear regression model or the ordinary log linear regression model.

7. Gamma Distribution. $f_Y(y) = \frac{1}{\Gamma(\alpha)\eta^\alpha} y^{\alpha-1} e^{-y/\eta}$, $y, \alpha, \eta > 0$ (v.s. (α, β), why ?)

In glm(), just one parameter. If $\alpha \neq 1$, then treat α as known. Thus the mean $\mu = \alpha\eta$,

$$\begin{aligned}
\ln f_{Y_i|X_i}(y|x) =& \ln \frac{y^{\alpha-1}e^{-y/\eta}}{\Gamma(\alpha)\eta^\alpha} && \text{so the parameter is } \eta \text{ or } \mu, \text{ a function of } \beta' x \\
=& \alpha \ln y - \ln y - y/\eta - \ln\Gamma(\alpha) - \alpha \ln\eta \\
=& -y/\eta - \alpha\ln\eta - \ln\Gamma(\alpha) + \alpha\ln y - \ln y && \text{shuffle around} \\
=& -y\alpha/\mu - \alpha \ln\frac{\mu}{\alpha} - \ln\Gamma(\alpha) + (\alpha-1)\ln y && \eta = \mu/\alpha \\
=& \underbrace{\alpha}_{\frac{A_i}{\phi}}[y \underbrace{(-1/\mu)}_{\theta_i} - \underbrace{\ln\mu}_{\gamma(\theta_i)}] + \underbrace{[\alpha\ln\alpha - \ln\Gamma(\alpha) + (\alpha-1)\ln y]}_{\tau(y, \frac{A_i}{\phi})} && (l(\mu) = \beta' x) \\
=& \frac{A_i}{\phi}[y\theta_i + \gamma(\theta_i)] + \tau(y, \frac{A_i}{\phi}) \text{ (the standard form of the GLM for } f_{Y_i|X_i}
\end{aligned}$$

with regression data (Y_i, X_i)'s).
Then $\theta_i = \frac{-1}{\mu_i}$,
$A_i/\phi = \alpha$,
$\gamma(\theta_i) = -\ln(-\theta_i)$.
The link $l(\mu_i) = \beta' X_i$ (and $\mu_i = l^{-1}(\beta' X_i)$) specifies the relation between μ and $\mathbf{X}'\beta$.
Each link specifies a reparametrization for μ in terms of $\beta' X$.
The default link is $l(\mu_i) = -1/\mu_i \quad (= \beta' X_i) => \mu_i = -1/(\beta' X_i)$
The other links are $l(\mu) = \mu$ (identify link)
and $l(\mu) = \ln(\mu)$ (log link),
The identity link leads to $\mu_i = \beta' X_i$.
$=> E(Y|X) = \beta' X$ (by identity link).
$V(Y|X) = \frac{(\beta' X)^2}{\alpha}$ (why ?) $(\mu = \alpha\eta,\ \sigma^2 = \alpha\eta^2)$
The log link leads to $\mu_i = \exp(\beta' X_i)$
$=> E(Y|X) = e^{\beta' X}$
$E(\ln Y|X) = \beta' X$??
The inverse link leads to $\mu_i = -1/(\beta' X_i)$.

$$\ln f_{Y|X}(y|x_i) = \begin{cases} \ln \frac{y^{\alpha-1} e^{-y/\eta}}{\Gamma(\alpha)\eta^\alpha} = -y/\eta - \alpha\ln\eta + \alpha\ln y - \ln y - \ln\Gamma(\alpha) & \text{link:} \\ \underbrace{\alpha}_{\frac{A_i}{\phi}}[y\underbrace{(-1/\mu)}_{\beta x_i} - \underbrace{\ln\mu}_{\ln(\frac{-1}{\beta x_i})}] + \underbrace{[\alpha\ln\alpha - \ln\Gamma(\alpha) + (\alpha-1)\ln y]}_{\tau(y,\frac{A_i}{\phi})} & \text{inverse} \\ \underbrace{\alpha}_{\frac{A_i}{\phi}}[y\underbrace{(-1/\mu)}_{-1/\beta x_i} - \underbrace{\ln\mu}_{\ln(\beta x_i)}] + \underbrace{[\alpha\ln\alpha - \ln\Gamma(\alpha) + (\alpha-1)\ln y]}_{\tau(y,\frac{A_i}{\phi})} & \text{identity} \quad y > 0. \\ \underbrace{\alpha}_{\frac{A_i}{\phi}}[y\underbrace{(-1/\mu)}_{-1/e^{\beta x_i}} - \underbrace{\ln\mu}_{\ln(e^{\beta x_i})}] + \underbrace{[\alpha\ln\alpha - \ln\Gamma(\alpha) + (\alpha-1)\ln y]}_{\tau(y,\frac{A_i}{\phi})} & \text{log} \end{cases}$$

Example 4. Simulation study on the LSE and MLE of (β_o, β_1), under the model $Y = e^{2x}Z$, or $\ln Y = \beta_o + \beta_1 x + \tau lnZ$, where $Z \sim Exp(1)$. $(\beta_o, \beta_1, \tau) = ?$

> library(MASS); library(survival); n=500; x=sample(1:4,n,replace=T); b=2;
> y=rgamma(n,1,exp(-b∗x)) # (n,shape,rate), $(\alpha, \beta) = (1, exp(b * x))$
y=rexp(n,scale=exp(b∗x)) # y=rweibull(n,1,exp($-b * x$))
> z=lm(log(y)∼x); summary(z)
> z=survreg(Surv(y)∼x); summary(z) #weibull
> z=glm(y∼x,family=Gamma(link=log),maxit=50); summary(z)
> z=survreg(Surv(y)∼x, dist="exponential"); summary(z)
> z=survreg(Surv(y)∼x-1, dist="exponential"); summary(z)
> z=lm(log(y)∼x-1); summary(z) **Is the model correct ?**
> predict(z,data.frame(x=4),se=T) # estimate $E(Y|\mathbf{X} = 4)$ with SE

		Value	*Std. Error*	*z*	*p*	*true value*
$\sim x$						
LSE	(*Intercept*)	−0.59293	0.15615	−3.797	0.000164	−0.577
	x	1.96718	0.05792	33.964	$< 2e-16$	2
survreg(*weib*)	(*Intercept*)	0.0591	0.1219	0.485	0.628	0
	x	1.9563	0.0449	43.530	0.000	2
	Log(*scale*)	0.0721	0.0351	2.053	0.040	ln1
glm(*gamma*)	(*Intercept*)	0.09658	0.11475	0.842	0.4	0
	x	1.95446	0.04256	45.920	$< 2e-16$	2
survreg(*exp*)	(*Intercept*)	0.0966	0.1123	0.86	0.39	0
	x	1.9545	0.0419	46.66	0.01	2
$\sim x - 1$						
survreg(*exp*)	x	1.99	0.0168	119	0	2
LSE	x	1.76649	0.02401	73.57	$< 2e-16$	2

What is the true value of (β_o, β_1, τ) **?** $\ln Y = \beta_o + \beta_1 x + \tau lnZ$.

Remark 6. One can find some interesting facts from the table.

1. Notice that $\hat{\beta}_o \begin{cases} \approx 0 & \text{in the MLE of Exp or Weibull or Gamma,} \\ \approx -0.59 \pm 0.31 & \text{for the LSE (**Anything wrong ?**)} \end{cases}$
2. $\hat{\beta}_1 \approx 2$ under lm(y∼x-1) ? **What does it mean ?**
   $\hat{\beta}_1 \approx 2$ under other models ?
3. **Relation between the** $\hat{\sigma}_{\hat{\beta}_1}$ under the models:
   lm($y \sim x$), Gamma $y \sim x$, Weibull $y \sim x$, Exp $y \sim x$ and Exp $y \sim x - 1$.
   Which $\hat{\sigma}_{\hat{\beta}_1}$ **is smaller ?**
   Why such a relation ?
   Which estimator of β_1 **is better ?** (more efficient)

Answer to question in (1):

$\beta_o = 0$ in the MLE approach under Exp($e^{\beta x}$), or Weibull or gamma, but
$\beta_o = E(\ln(Z)) \approx -0.577$ in the LSE approach $\ln Y = \beta_1 X + \ln Z = \beta_1 X + \beta_o + \epsilon$.
>mean(log(rexp(1000000000))) [1] $-0.5772387 \approx \int_0^\infty \ln x e^{-x} dx$.

Answer to question in (3):

(a) Semi-parametric v.s. parametric approach; which is semi-pa ?
(b) The # of parameters goes from 3 to 1.

Remark 7. Example 4 illustrates that one should simplify the model as much as possible in data analysis. **How ?**
The data in Example 4 satisfy the Weibull distribution:
$\ln Y = \beta_o + \beta_1 x + \tau \ln Z$, with $Z \sim Exp(1)$, $(\beta_0, \beta_1, \tau) = (0.1, 1.95, e^{0.07})$? Or $(0, 2, 1)$?
The data fit Weibull() better than Exponential() (in CB), but one needs to
check whether the data are indeed from Weibull **(how ?)** and try to simplify it.
The more the parameter, the less the efficiency, thus the SD increases if # of parameters increases (it is why $Log(scale)$ is significant with p-value 0.02×2).

	weibull	*gamma or exp*
β_o	0.12	0.11
β_1	0.044	0.042

The simplified model is $\ln Y = \beta_o + \beta_1 x + \tau \ln Z$, with $(\beta_0, \beta_1, \tau) = (0, 2, 1)$ and $Z \sim Exp(1)$, and becomes $Exp(\mu)$ with mean e^{2x}). **How to test it ?**

Example 5. (Simulation study). Generate data from $Y \sim Exp(\mu)$, where
$S(t) = \exp(-\lambda t) = \exp(-\frac{t}{\mu})$, $t > 0$,
$Z = \frac{Y}{\mu} \sim Exp(1)$.
$\ln Y = \ln\mu + \ln Z$.
> library(MASS) ; library(survival) ; n=500 ; y=rexp(n,rate=5);
Then pretend we do not know it. Estimate $E(Y)$ by the MLE based on Weibull, Exp or Gamma and by the LSE, using varies codes:
non-regression:
fitdistr(y,"exponential"), fitdistr(y,"weibull"), fitdistr(y,"gamma")
regression and parametric:
survreg(Surv(y)∼ 1),
survreg(Surv(y)∼ 1, dist="exponential")
glm(y∼1,family=Gamma(link=log))
semi-parametric:
lm(log(y)∼1)
lm(y∼1)
Question: What is the true value that is estimated ?
How many distinct values of the estimates ?
> (z=fitdistr(y,"exponential")$e)
rate $\lambda = 1/scale$ $S(t) = \exp(-\lambda t)$
5.235007 $\hat\lambda = 1/\hat\mu$
> 1/z
0.1910217 $\hat\mu$
> (z=survreg(Surv(y)∼1, dist="exponential")$co) $Y = e^\alpha Z$
-1.655368 $\hat\alpha$
> exp(z)
0.1910217 $\hat\mu = e^{\hat\alpha}$
> predict(z,data.frame(x=0))
0.1910217 $\hat\mu$
> z=glm(y∼1,family=Gamma(link=log),maxit=50)
> z[[1]] (= link $\ln\mu = \alpha$)