Basic statistics functions

Haewon Lee

2024-07-17

### t-test

R function : t.test -

option arguments : alternative = c(“two.sided”, “less”, “greater”), formula (종속변수~ 독립변수)

help files : ?t.test 를 치면 함수의 argument, values(results), detail에 대해서 설명이 나옴

library(survival)  
data("rotterdam")

Warning in data("rotterdam"): data set 'rotterdam' not found

group1 <- rotterdam[ rotterdam$grade == 2, "age"]  
group2 <- rotterdam[ rotterdam$grade != 2, "age"]  
t.test(group1, group2) ## unmatched 임의의 두개의 vector로 비교

Welch Two Sample t-test  
  
data: group1 and group2  
t = -1.7437, df = 1444.4, p-value = 0.08143  
alternative hypothesis: true difference in means is not equal to 0  
95 percent confidence interval:  
 -1.959901 0.115264  
sample estimates:  
mean of x mean of y   
 54.38161 55.30393

t.test(age~meno,data=rotterdam) ## matched 한개의 데이터프레임에서 paired t-test

Welch Two Sample t-test  
  
data: age by meno  
t = -76.545, df = 2972.8, p-value < 2.2e-16  
alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0  
95 percent confidence interval:  
 -21.53192 -20.45636  
sample estimates:  
mean in group 0 mean in group 1   
 43.30107 64.29521

### 𝜒2 (chi-square) test

R function : chisq.test

table(rotterdam[,c("hormon","size")])

size  
hormon <=20 20-50 >50  
 0 1283 1119 241  
 1 104 172 63

chisq.test(table(rotterdam[,c("hormon","size")]), correct = TRUE)

Pearson's Chi-squared test  
  
data: table(rotterdam[, c("hormon", "size")])  
X-squared = 51.92, df = 2, p-value = 5.317e-12

chisq.test(rotterdam$hormon, rotterdam$chemo)

Pearson's Chi-squared test with Yates' continuity correction  
  
data: rotterdam$hormon and rotterdam$chemo  
X-squared = 29.771, df = 1, p-value = 4.862e-08

x <- matrix(c(12, 5, 7, 7), ncol = 2) ## matrix를 만들어서 검정하는 방법  
x

[,1] [,2]  
[1,] 12 7  
[2,] 5 7

chisq.test(x)$p.value ## chisq test의 결과물은 list이다 여기서 p.value 부분만 출력

[1] 0.4233054

chisq.test(x, simulate.p.value = TRUE, B = 10000)$p.value

[1] 0.2925707

### Generalized linear regression and Loess smoothing (LOcal regrESSion)

R function : glm (generalized linear models) 다중 선형회귀

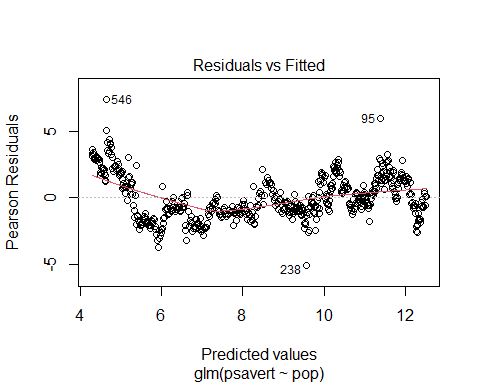
suppressMessages(library(ggplot2))  
data(economics, package="ggplot2")  
economics$index <- 1:nrow(economics) # create index variable  
glm\_model1 <- glm(psavert~pop, data = economics)  
summary(glm\_model1)

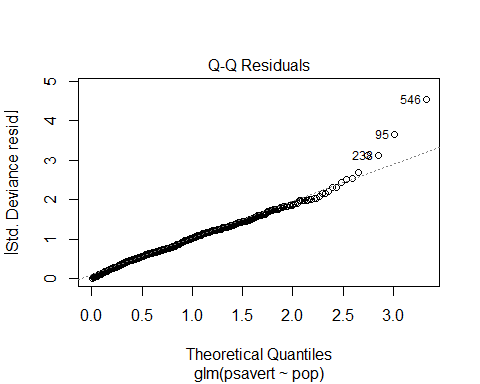
Call:  
glm(formula = psavert ~ pop, data = economics)  
  
Coefficients:  
 Estimate Std. Error t value Pr(>|t|)   
(Intercept) 2.595e+01 4.812e-01 53.92 <2e-16 \*\*\*  
pop -6.758e-05 1.852e-06 -36.48 <2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
(Dispersion parameter for gaussian family taken to be 2.645601)  
  
 Null deviance: 5034.6 on 573 degrees of freedom  
Residual deviance: 1513.3 on 572 degrees of freedom  
AIC: 2191.4  
  
Number of Fisher Scoring iterations: 2

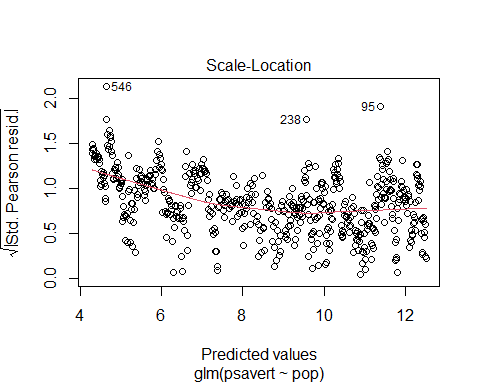
anova(glm\_model1)

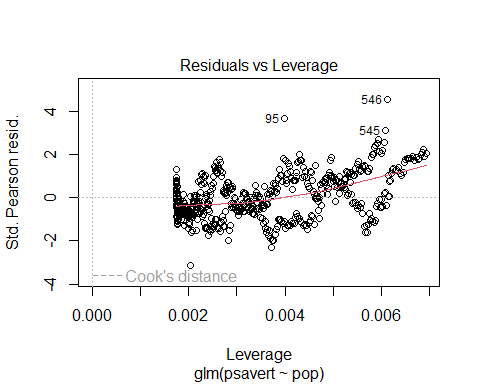
Analysis of Deviance Table  
  
Model: gaussian, link: identity  
  
Response: psavert  
  
Terms added sequentially (first to last)  
  
 Df Deviance Resid. Df Resid. Dev F Pr(>F)   
NULL 573 5034.6   
pop 1 3521.3 572 1513.3 1331 < 2.2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

plot(glm\_model1)







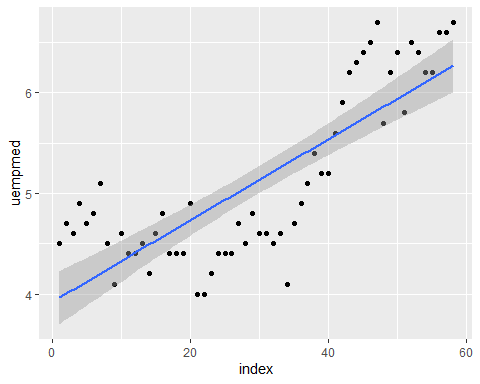


# 데이터 선택  
economics <- economics[100:180, ] # 좁은 범위 선택  
  
# Loess 모델 생성  
loessMod10 <- loess(uempmed ~ index, data=economics, span=0.10) # 10% smoothing span  
loessMod25 <- loess(uempmed ~ index, data=economics, span=0.25)  
loessMod50 <- loess(uempmed ~ index, data=economics, span=0.50)  
  
# 예측값 계산  
smoothed10 <- predict(loessMod10)  
smoothed25 <- predict(loessMod25)  
smoothed50 <- predict(loessMod50)  
  
# 그래프 그리기  
plot(economics$date, economics$uempmed, type="l", main="Loess Smoothing and Prediction", xlab="Date", ylab="Unemployment Median")  
  
# 예측된 smoothed 라인 추가  
lines(economics$date, smoothed10, col="red")  
lines(economics$date, smoothed25, col="green")  
lines(economics$date, smoothed50, col="blue")

위 코드가 제 개발환경에서 실행시 오류가 발생하므로 추후 수정해서 올려 드리겠습니다.

economics <- economics[1:58,]  
library(ggplot2)  
ggplot(data=economics, aes(x=index, y=uempmed))+  
geom\_point()+  
geom\_smooth(method = "lm")

`geom\_smooth()` using formula = 'y ~ x'

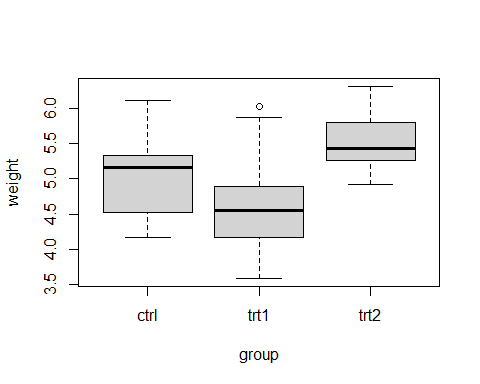


### One-way ANOVA

suppressMessages(library(psych))  
PlantGrowth ## 내장 dataset

weight group  
1 4.17 ctrl  
2 5.58 ctrl  
3 5.18 ctrl  
4 6.11 ctrl  
5 4.50 ctrl  
6 4.61 ctrl  
7 5.17 ctrl  
8 4.53 ctrl  
9 5.33 ctrl  
10 5.14 ctrl  
11 4.81 trt1  
12 4.17 trt1  
13 4.41 trt1  
14 3.59 trt1  
15 5.87 trt1  
16 3.83 trt1  
17 6.03 trt1  
18 4.89 trt1  
19 4.32 trt1  
20 4.69 trt1  
21 6.31 trt2  
22 5.12 trt2  
23 5.54 trt2  
24 5.50 trt2  
25 5.37 trt2  
26 5.29 trt2  
27 4.92 trt2  
28 6.15 trt2  
29 5.80 trt2  
30 5.26 trt2

plot(weight~group, data = PlantGrowth) ## Boxplot으로 자동으로 그려준다.



with(PlantGrowth, describeBy(weight,group))

Descriptive statistics by group   
group: ctrl  
 vars n mean sd median trimmed mad min max range skew kurtosis se  
X1 1 10 5.03 0.58 5.15 5 0.72 4.17 6.11 1.94 0.23 -1.12 0.18  
------------------------------------------------------------   
group: trt1  
 vars n mean sd median trimmed mad min max range skew kurtosis se  
X1 1 10 4.66 0.79 4.55 4.62 0.53 3.59 6.03 2.44 0.47 -1.1 0.25  
------------------------------------------------------------   
group: trt2  
 vars n mean sd median trimmed mad min max range skew kurtosis se  
X1 1 10 5.53 0.44 5.44 5.5 0.36 4.92 6.31 1.39 0.48 -1.16 0.14

bartlett.test(PlantGrowth$weight ~ PlantGrowth$group) ## 등분산 가정을 체크함

Bartlett test of homogeneity of variances  
  
data: PlantGrowth$weight by PlantGrowth$group  
Bartlett's K-squared = 2.8786, df = 2, p-value = 0.2371

aov\_model <- aov(weight~group, data = PlantGrowth)  
summary(aov\_model)

Df Sum Sq Mean Sq F value Pr(>F)   
group 2 3.766 1.8832 4.846 0.0159 \*  
Residuals 27 10.492 0.3886   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

### Correlation tests

Pearson correlation formula

Spearman correlation formula : non-parametric

where 𝑥′ = 𝑟𝑎𝑛𝑘(𝑥) and 𝑦′ = 𝑟𝑎𝑛𝑘(𝑦)

Kendall correlation formula : non-parametric

where 𝑛𝑐 : number of concordant pairs, 𝑛𝑑 : number of concordant pairs, 𝑛 : size of 𝑥 + 𝑦

res <- cor.test(economics$index, economics$uempmed, method = "pearson")  
res

Pearson's product-moment correlation  
  
data: economics$index and economics$uempmed  
t = 10.152, df = 56, p-value = 2.639e-14  
alternative hypothesis: true correlation is not equal to 0  
95 percent confidence interval:  
 0.6901407 0.8802298  
sample estimates:  
 cor   
0.8049464

res$p.value ## res는 리스트형태로 나오는 결과물이다. 여기에서 필요한 값만 골라냄

[1] 2.639201e-14

res$estimate

cor   
0.8049464

res2 <- cor.test(economics$index, economics$uempmed, method = "spearman")

Warning in cor.test.default(economics$index, economics$uempmed, method =  
"spearman"): Cannot compute exact p-value with ties

res2

Spearman's rank correlation rho  
  
data: economics$index and economics$uempmed  
S = 9033.3, p-value = 1.578e-10  
alternative hypothesis: true rho is not equal to 0  
sample estimates:  
 rho   
0.7221299

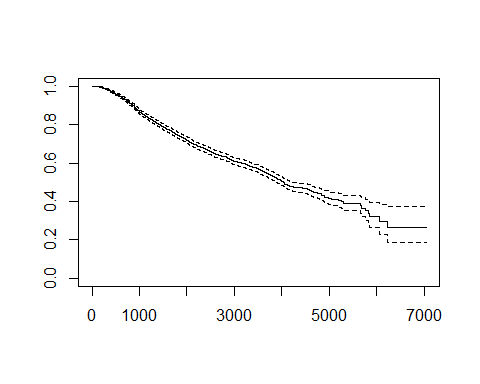
res3 <- cor.test(economics$index, economics$uempmed, method = "kendall")  
res3

Kendall's rank correlation tau  
  
data: economics$index and economics$uempmed  
z = 6.0101, p-value = 1.854e-09  
alternative hypothesis: true tau is not equal to 0  
sample estimates:  
 tau   
0.5541495

### Survival analysis

* Kaplan Meier Analysis - Basic survival model survival::Surv

km <- Surv(rotterdam$dtime, event = rotterdam$death) ## default type : "right"  
plot(km) ## km - Surv class (time, status) 가지고 있는 리스트



median(km); mean(km) ## Surv 객체에 대한 method 함수들이 있다. plot.Surv포함

$quantile  
 50   
4033   
  
$lower  
 50   
3888   
  
$upper  
 50   
4309

[1] 1302.883

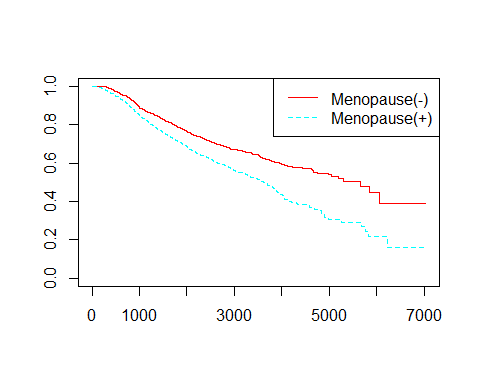
* Kaplan Meier Analysis - survfit model km\_

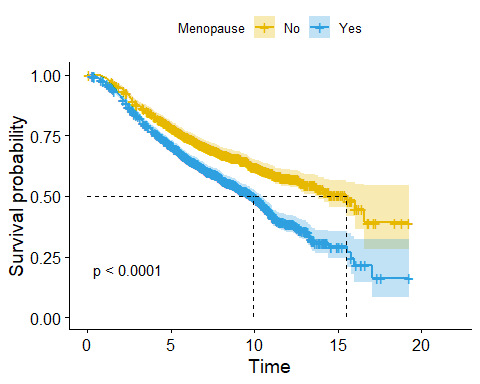
km\_fit <- survfit(km~rotterdam$meno)  
summary(km\_fit, c(365\*1:19)) ### 정해진 time에 맞는 생존테이블표를 만든다.

Call: survfit(formula = km ~ rotterdam$meno)  
  
 rotterdam$meno=0   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 365 1298 13 0.990 0.00274 0.985 0.995  
 730 1236 56 0.947 0.00617 0.935 0.959  
 1095 1140 90 0.878 0.00905 0.861 0.896  
 1460 1071 59 0.833 0.01035 0.813 0.853  
 1825 973 59 0.786 0.01141 0.764 0.809  
 2190 865 50 0.745 0.01222 0.721 0.769  
 2555 754 43 0.706 0.01292 0.681 0.732  
 2920 611 31 0.675 0.01354 0.649 0.702  
 3285 480 15 0.656 0.01397 0.629 0.684  
 3650 345 21 0.623 0.01505 0.594 0.653  
 4015 217 13 0.595 0.01631 0.563 0.627  
 4380 138 6 0.575 0.01760 0.542 0.611  
 4745 88 4 0.554 0.02000 0.516 0.594  
 5110 54 3 0.530 0.02334 0.487 0.578  
 5475 29 2 0.506 0.02783 0.455 0.564  
 5840 14 1 0.481 0.03617 0.415 0.558  
 6205 5 2 0.391 0.06658 0.280 0.546  
 6570 3 0 0.391 0.06658 0.280 0.546  
 6935 1 0 0.391 0.06658 0.280 0.546  
  
 rotterdam$meno=1   
 time n.risk n.event survival std.err lower 95% CI upper 95% CI  
 365 1616 46 0.972 0.00402 0.9645 0.980  
 730 1508 103 0.910 0.00701 0.8966 0.924  
 1095 1366 129 0.832 0.00919 0.8143 0.850  
 1460 1245 111 0.764 0.01046 0.7440 0.785  
 1825 1111 87 0.710 0.01122 0.6883 0.732  
 2190 944 82 0.655 0.01186 0.6326 0.679  
 2555 819 58 0.614 0.01231 0.5901 0.638  
 2920 642 45 0.577 0.01272 0.5530 0.603  
 3285 474 42 0.536 0.01334 0.5104 0.563  
 3650 342 31 0.496 0.01418 0.4685 0.524  
 4015 188 35 0.430 0.01614 0.3998 0.463  
 4380 113 17 0.386 0.01772 0.3531 0.423  
 4745 62 6 0.358 0.01989 0.3210 0.399  
 5110 28 7 0.309 0.02431 0.2652 0.361  
 5475 14 1 0.293 0.02796 0.2431 0.353  
 5840 8 3 0.217 0.04323 0.1469 0.321  
 6205 4 0 0.217 0.04323 0.1469 0.321  
 6570 1 1 0.163 0.05710 0.0819 0.324  
 6935 1 0 0.163 0.05710 0.0819 0.324

suppressMessages(library(survminer))  
plot(km\_fit, col = rainbow(2), lty=1:2)  
legend("topright", legend = c("Menopause(-)","Menopause(+)"),  
 col= rainbow(2), lty=1:2)  
library(survminer)  
ggsurvplot(km\_fit, data = rotterdam,  
 conf.int = T, xscale = 365.2425, ## xscale can be "d\_y"  
 break.x.by = 5\*365.2425,  
 pval = T, pval.size =4, surv.median.line = "hv",  
 risk.table = FALSE, ## if TRUE, risk table is displayed under graph  
 legend.title="Menopause", legend.labs=c("No","Yes"),  
 palette = c("#E7B800", "#2E9FDF"),)

Warning in geom\_segment(aes(x = 0, y = max(y2), xend = max(x1), yend = max(y2)), : All aesthetics have length 1, but the data has 2 rows.  
ℹ Please consider using `annotate()` or provide this layer with data containing  
 a single row.  
All aesthetics have length 1, but the data has 2 rows.  
ℹ Please consider using `annotate()` or provide this layer with data containing  
 a single row.





## ggplot + survminer package

Cox Proportional model

args(coxph)

function (formula, data, weights, subset, na.action, init, control,   
 ties = c("efron", "breslow", "exact"), singular.ok = TRUE,   
 robust, model = FALSE, x = FALSE, y = TRUE, tt, method = ties,   
 id, cluster, istate, statedata, nocenter = c(-1, 0, 1), ...)   
NULL

library(carData) ## Rossi data set 이용하기 위해서 사용  
suppressMessages(library(car)) ## Anova function  
colnames(Rossi) # emp1-52 : factor (yes or no)

[1] "week" "arrest" "fin" "age" "race" "wexp" "mar" "paro"   
 [9] "prio" "educ" "emp1" "emp2" "emp3" "emp4" "emp5" "emp6"   
[17] "emp7" "emp8" "emp9" "emp10" "emp11" "emp12" "emp13" "emp14"   
[25] "emp15" "emp16" "emp17" "emp18" "emp19" "emp20" "emp21" "emp22"   
[33] "emp23" "emp24" "emp25" "emp26" "emp27" "emp28" "emp29" "emp30"   
[41] "emp31" "emp32" "emp33" "emp34" "emp35" "emp36" "emp37" "emp38"   
[49] "emp39" "emp40" "emp41" "emp42" "emp43" "emp44" "emp45" "emp46"   
[57] "emp47" "emp48" "emp49" "emp50" "emp51" "emp52"

cox\_model1 <- coxph(Surv(week, arrest) ~  
fin + age + race + wexp + mar + paro + prio,  
data = Rossi)  
summary(cox\_model1)

Call:  
coxph(formula = Surv(week, arrest) ~ fin + age + race + wexp +   
 mar + paro + prio, data = Rossi)  
  
 n= 432, number of events= 114   
  
 coef exp(coef) se(coef) z Pr(>|z|)   
finyes -0.37942 0.68426 0.19138 -1.983 0.04742 \*   
age -0.05744 0.94418 0.02200 -2.611 0.00903 \*\*  
raceother -0.31390 0.73059 0.30799 -1.019 0.30812   
wexpyes -0.14980 0.86088 0.21222 -0.706 0.48029   
marnot married 0.43370 1.54296 0.38187 1.136 0.25606   
paroyes -0.08487 0.91863 0.19576 -0.434 0.66461   
prio 0.09150 1.09581 0.02865 3.194 0.00140 \*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
 exp(coef) exp(-coef) lower .95 upper .95  
finyes 0.6843 1.4614 0.4702 0.9957  
age 0.9442 1.0591 0.9043 0.9858  
raceother 0.7306 1.3688 0.3995 1.3361  
wexpyes 0.8609 1.1616 0.5679 1.3049  
marnot married 1.5430 0.6481 0.7300 3.2614  
paroyes 0.9186 1.0886 0.6259 1.3482  
prio 1.0958 0.9126 1.0360 1.1591  
  
Concordance= 0.64 (se = 0.027 )  
Likelihood ratio test= 33.27 on 7 df, p=2e-05  
Wald test = 32.11 on 7 df, p=4e-05  
Score (logrank) test = 33.53 on 7 df, p=2e-05

Anova(cox\_model1)

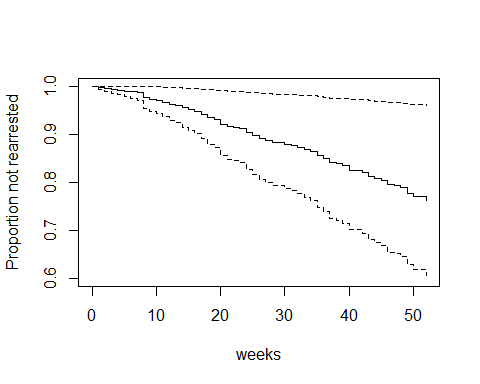
Analysis of Deviance Table (Type II tests)  
 LR Chisq Df Pr(>Chisq)   
fin 3.9862 1 0.045874 \*   
age 7.9880 1 0.004709 \*\*  
race 1.1252 1 0.288812   
wexp 0.5003 1 0.479352   
mar 1.4312 1 0.231572   
paro 0.1870 1 0.665450   
prio 8.9766 1 0.002735 \*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(cox\_model1)

Analysis of Deviance Table  
 Cox model: response is Surv(week, arrest)  
Terms added sequentially (first to last)  
  
 loglik Chisq Df Pr(>|Chi|)   
NULL -675.38   
fin -673.46 3.8371 1 0.0501315 .   
age -666.24 14.4526 1 0.0001437 \*\*\*  
race -665.84 0.7887 1 0.3745021   
wexp -664.22 3.2472 1 0.0715467 .   
mar -663.58 1.2841 1 0.2571359   
paro -663.24 0.6798 1 0.4096690   
prio -658.75 8.9766 1 0.0027346 \*\*   
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

모델의 전체적인 생존곡선을 알고 싶으면 survfit 함수를 이용해서 생존곡선을 그릴 수 있다

plot(survfit(cox\_model1), ylim = c(0.6,1),xlab = "weeks",  
ylab = "Proportion not rearrested")



Rossi.fin <- with(Rossi, data.frame(fin=c(0, 1),  
age=rep(mean(age), 2), race=rep(mean(race == "other"), 2),  
wexp=rep(mean(wexp == "yes"), 2), mar=rep(mean(mar == "not married"), 2),  
paro=rep(mean(paro == "yes"), 2), prio=rep(mean(prio), 2)))  
## fin = 0,1 이것을 두그룹으로 나누고 나머지 변수들은 평균적인 값으로 고정해 버림  
plot(survfit(cox\_model1, newdata = Rossi.fin), conf.int = T,  
lty = c(1,2), ylim = c(0.6,1),xlab = "weeks",  
ylab = "Proportion not rearrested", col = c("blue","red")  
)

Warning in model.frame.default(data = structure(list(fin = c(0, 1), age =  
c(24.5972222222222, : variable 'fin' is not a factor

Warning in model.frame.default(data = structure(list(fin = c(0, 1), age =  
c(24.5972222222222, : variable 'race' is not a factor

Warning in model.frame.default(data = structure(list(fin = c(0, 1), age =  
c(24.5972222222222, : variable 'wexp' is not a factor

Warning in model.frame.default(data = structure(list(fin = c(0, 1), age =  
c(24.5972222222222, : variable 'mar' is not a factor

Warning in model.frame.default(data = structure(list(fin = c(0, 1), age =  
c(24.5972222222222, : variable 'paro' is not a factor

legend("bottomleft", legend=c("fin = no", "fin = yes"), lty=c(1 ,2),  
col=c("blue","red") , inset=0.02)

