simple\_reg\_1.R

SANGHOOJEFFREY

Tue Jun 26 19:41:53 2018

# 회귀분석 기초와 상관분석  
  
# 회귀분석의 이해  
# http://www.shodor.org/interactivate/activities/Regression/  
  
# 회귀분석 기초1(Simple regression analysis 1)  
# 가장 간단한 형태의 회귀분석은 한 개의 설명변수와 한 개의 반응변수 간의 관계식을 찾는 문제  
# 이러한 경우를 단순회귀분석이라고 한다.   
  
# 단순회귀분석을 위해선 가장 먼저 산점도를 통해 설명변수와 반응변수 간 선형 관계가 있는지 확인해야한다.  
# 산점도는 plot()함수를 이용  
  
data(cars) # 속도에 따른 정지거리(dist)에 관한 데이터  
  
cars

## speed dist  
## 1 4 2  
## 2 4 10  
## 3 7 4  
## 4 7 22  
## 5 8 16  
## 6 9 10  
## 7 10 18  
## 8 10 26  
## 9 10 34  
## 10 11 17  
## 11 11 28  
## 12 12 14  
## 13 12 20  
## 14 12 24  
## 15 12 28  
## 16 13 26  
## 17 13 34  
## 18 13 34  
## 19 13 46  
## 20 14 26  
## 21 14 36  
## 22 14 60  
## 23 14 80  
## 24 15 20  
## 25 15 26  
## 26 15 54  
## 27 16 32  
## 28 16 40  
## 29 17 32  
## 30 17 40  
## 31 17 50  
## 32 18 42  
## 33 18 56  
## 34 18 76  
## 35 18 84  
## 36 19 36  
## 37 19 46  
## 38 19 68  
## 39 20 32  
## 40 20 48  
## 41 20 52  
## 42 20 56  
## 43 20 64  
## 44 22 66  
## 45 23 54  
## 46 24 70  
## 47 24 92  
## 48 24 93  
## 49 24 120  
## 50 25 85

plot(cars$speed, cars$dist)  
  
# 산점도를 그려보면 속도(speed)와 정지거리(dist) 간 선형관계가 있어보인다.  
# 설명변수와 반응변수 간 상관정도를 정량적으로 확인하기 위해 상관분석을 실시한다.  
# R에서는 cor()함수를 이용  
  
cor(cars$speed, cars$dist)

## [1] 0.8068949

cor.test(cars$speed, cars$dist) # 가설검정까지 원하는 경우

##   
## Pearson's product-moment correlation  
##   
## data: cars$speed and cars$dist  
## t = 9.464, df = 48, p-value = 1.49e-12  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.6816422 0.8862036  
## sample estimates:  
## cor   
## 0.8068949

# 그러면 최적 선형식은 어떻게 구할까? (모수추정)  
# 최적 선형식을 위해선 dist = beta\_0 + beta\_1 \* speed의 beta\_0와 beta\_1을 추정해야 한다.   
  
# 만약 beta\_0= -2, beta\_1= 4 를 넣는다면  
# dist = -2 + 4\* speed로 예측된다.   
  
hat.dist = -2 + 3.8 \* cars$speed  
  
# 실제값과 예측값의 차이를 만들어보자.   
diff = cars$dist - hat.dist  
  
# 실제값과 예측값이 차이의 제곱 합을 계산하면 다음과 같다.   
sum(diff^2)

## [1] 20544.12

# 그러면 위의 값을 최소로 하는 식이 최적식이 아닐까? (최소제곱법)  
f.out <- function(x) {  
 hat.dist = x[1] + x[2] \* cars$speed  
 diff = cars$dist - hat.dist  
 return(sum(diff^2))  
}  
  
f.out(x=c(-2,4))

## [1] 25171

f.out(x=c(-17,4))

## [1] 11491

f.out(x=c(-18,4))

## [1] 11379

f.out(x=c(-19,4))

## [1] 11367

f.out(x=c(-20,4))

## [1] 11455

f.out(x=c(-21,4))

## [1] 11643

# 대충 최적값을 짐작하면 beta\_0 = -19, beta\_1 =4 이다.   
# 하지만 이 값이 최적점일까? NO  
  
optim(c(0,1), f.out)$par # 수치적 방법을 이용하여 최적값을 찾을 수 있다.

## [1] -17.578151 3.932216

# 이러한 모수 추정방법을 최소제곱법이라 한다.   
# 하지만 위의 방법으로 수치적 방법을 이용하지 않더라도 lm()함수를 이용하면 결과 확인 가능  
  
out <- lm(cars$dist~cars$speed) # 반응변수 ~ 설명변수 식을 lm()에 넣어주면 된다.  
out <- lm(dist~speed, data=cars) # 또 다른 표현   
  
summary(out) # 결과 확인

##   
## Call:  
## lm(formula = dist ~ speed, data = cars)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -29.069 -9.525 -2.272 9.215 43.201   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -17.5791 6.7584 -2.601 0.0123 \*   
## speed 3.9324 0.4155 9.464 1.49e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 15.38 on 48 degrees of freedom  
## Multiple R-squared: 0.6511, Adjusted R-squared: 0.6438   
## F-statistic: 89.57 on 1 and 48 DF, p-value: 1.49e-12

# 모수의 추정방법으로 우도함수를 이용한 최대우도법이 있으나, 모수추정치는 동일하여 설명은 생략한다.  
  
# 모형평가  
out <- lm(dist~speed, data=cars) # 또 다른 표현   
summary(out)

##   
## Call:  
## lm(formula = dist ~ speed, data = cars)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -29.069 -9.525 -2.272 9.215 43.201   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -17.5791 6.7584 -2.601 0.0123 \*   
## speed 3.9324 0.4155 9.464 1.49e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 15.38 on 48 degrees of freedom  
## Multiple R-squared: 0.6511, Adjusted R-squared: 0.6438   
## F-statistic: 89.57 on 1 and 48 DF, p-value: 1.49e-12

# summary() 함수의 가장 처음에는 함수식을 나타낸다.  
# Residuals 부분은 실제 데이터에서 관측된 잔차를 보여준다.  
# Residual = 관측값 - 예측값  
  
obs <- cars$dist # 실제 정지거리  
pred <- -17.5791+3.9324\*cars$speed # 추정된 선형식을 통한 예측값  
resd <- obs- pred  
  
summary(resd) # summary(out)의 Residuals값과 동일

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -29.06890 -9.52520 -2.27170 0.00014 9.21490 43.20150

out$residuals # 잔차를 불러오기

## 1 2 3 4 5 6   
## 3.849460 11.849460 -5.947766 12.052234 2.119825 -7.812584   
## 7 8 9 10 11 12   
## -3.744993 4.255007 12.255007 -8.677401 2.322599 -15.609810   
## 13 14 15 16 17 18   
## -9.609810 -5.609810 -1.609810 -7.542219 0.457781 0.457781   
## 19 20 21 22 23 24   
## 12.457781 -11.474628 -1.474628 22.525372 42.525372 -21.407036   
## 25 26 27 28 29 30   
## -15.407036 12.592964 -13.339445 -5.339445 -17.271854 -9.271854   
## 31 32 33 34 35 36   
## 0.728146 -11.204263 2.795737 22.795737 30.795737 -21.136672   
## 37 38 39 40 41 42   
## -11.136672 10.863328 -29.069080 -13.069080 -9.069080 -5.069080   
## 43 44 45 46 47 48   
## 2.930920 -2.933898 -18.866307 -6.798715 15.201285 16.201285   
## 49 50   
## 43.201285 4.268876

residuals(out)

## 1 2 3 4 5 6   
## 3.849460 11.849460 -5.947766 12.052234 2.119825 -7.812584   
## 7 8 9 10 11 12   
## -3.744993 4.255007 12.255007 -8.677401 2.322599 -15.609810   
## 13 14 15 16 17 18   
## -9.609810 -5.609810 -1.609810 -7.542219 0.457781 0.457781   
## 19 20 21 22 23 24   
## 12.457781 -11.474628 -1.474628 22.525372 42.525372 -21.407036   
## 25 26 27 28 29 30   
## -15.407036 12.592964 -13.339445 -5.339445 -17.271854 -9.271854   
## 31 32 33 34 35 36   
## 0.728146 -11.204263 2.795737 22.795737 30.795737 -21.136672   
## 37 38 39 40 41 42   
## -11.136672 10.863328 -29.069080 -13.069080 -9.069080 -5.069080   
## 43 44 45 46 47 48   
## 2.930920 -2.933898 -18.866307 -6.798715 15.201285 16.201285   
## 49 50   
## 43.201285 4.268876

out$fitted.values # 추정된 선형식으로 예측된 값을 선형식을 세우지 않고 불러올 수 있다.

## 1 2 3 4 5 6 7   
## -1.849460 -1.849460 9.947766 9.947766 13.880175 17.812584 21.744993   
## 8 9 10 11 12 13 14   
## 21.744993 21.744993 25.677401 25.677401 29.609810 29.609810 29.609810   
## 15 16 17 18 19 20 21   
## 29.609810 33.542219 33.542219 33.542219 33.542219 37.474628 37.474628   
## 22 23 24 25 26 27 28   
## 37.474628 37.474628 41.407036 41.407036 41.407036 45.339445 45.339445   
## 29 30 31 32 33 34 35   
## 49.271854 49.271854 49.271854 53.204263 53.204263 53.204263 53.204263   
## 36 37 38 39 40 41 42   
## 57.136672 57.136672 57.136672 61.069080 61.069080 61.069080 61.069080   
## 43 44 45 46 47 48 49   
## 61.069080 68.933898 72.866307 76.798715 76.798715 76.798715 76.798715   
## 50   
## 80.731124

fitted(out)

## 1 2 3 4 5 6 7   
## -1.849460 -1.849460 9.947766 9.947766 13.880175 17.812584 21.744993   
## 8 9 10 11 12 13 14   
## 21.744993 21.744993 25.677401 25.677401 29.609810 29.609810 29.609810   
## 15 16 17 18 19 20 21   
## 29.609810 33.542219 33.542219 33.542219 33.542219 37.474628 37.474628   
## 22 23 24 25 26 27 28   
## 37.474628 37.474628 41.407036 41.407036 41.407036 45.339445 45.339445   
## 29 30 31 32 33 34 35   
## 49.271854 49.271854 49.271854 53.204263 53.204263 53.204263 53.204263   
## 36 37 38 39 40 41 42   
## 57.136672 57.136672 57.136672 61.069080 61.069080 61.069080 61.069080   
## 43 44 45 46 47 48 49   
## 61.069080 68.933898 72.866307 76.798715 76.798715 76.798715 76.798715   
## 50   
## 80.731124

# Coefficients 에서는 회귀모형의 계수와 이 계수의 통계적 유의성을 보여준다.  
# Estimate 열은 절편과 계수의 추정치  
# dist = -17.5791+3.9324\*speed  
# Pr(>|t|)는 t 분포를 이용하여 각 변수가 유의한지 판단. 기준은 일반적으로 0.05  
# 만약, p-value가 0.05보다 크면 혜당 계수가 0이라는 귀무가설을 기각할 수 없으므로 0으로 봐야한다.  
  
  
# 마지막으로 결정계수 (Multiple R-squared) 와 회귀모형의 유의성을 의미하는 F통계량이 제시됨  
# 여기서 결정계수란?? 선형모형의 설명력으로 해석   
  
var(cars$dist) # Var(관측값)

## [1] 664.0608

var(cars$dist-fitted(out)) + var(fitted(out)) # Var(관측값-예측값)+Var(예측값)

## [1] 664.0608

SST = sum((cars$dist - mean(cars$dist))^2)  
SSE = sum((cars$dist - fitted(out))^2)  
SSR = sum((fitted(out)-mean(cars$dist))^2)  
SST == SSE+ SSR # 논리 확인

## [1] FALSE

# 모형이 잘 맞는다는건 관측값과 예측값이 비슷하다고 볼 수 있다. 즉 SSE가 0에 가까워짐  
# SSR/SST는 전제분산 중 예측값으로 설명되는 분산의 비  
# Multiple R-squared로 의미는 반응변수의 분산 중 설명변수로 설명되는 분산의 비율  
  
SSR/SST

## [1] 0.6510794

# R^2 는 0과 1범위에 존재하며 단순회귀모형의 경우 상관계수의 제곱과 같다.  
cor(cars$speed, cars$dist)^2

## [1] 0.6510794

# F 통계량은 full model : dist = beta\_0 + beta\_1 \* speed  
# Reduced model : dist = beta\_0   
# 간 차이를 비교한 값. 즉 통계적으로 유의미한다는건 설명변수가 반응변수에 영향을 미침  
  
model1 <- lm(dist~speed, data=cars)  
model2 <- lm(dist~1, data=cars)  
anova(model1, model2)

## Analysis of Variance Table  
##   
## Model 1: dist ~ speed  
## Model 2: dist ~ 1  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 48 11354   
## 2 49 32539 -1 -21186 89.567 1.49e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# 즉 speed가 유의미한 설명변수이다.   
  
anova(out) #회귀모형에서의 분산분석결과

## Analysis of Variance Table  
##   
## Response: dist  
## Df Sum Sq Mean Sq F value Pr(>F)   
## speed 1 21186 21185.5 89.567 1.49e-12 \*\*\*  
## Residuals 48 11354 236.5   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# 새로운 값이 있을 때 예측은? predict()함수 이용  
out <- lm(dist~speed, data=cars)  
predict(out, newdata=data.frame(speed=c(3,4,5)))

## 1 2 3   
## -5.781869 -1.849460 2.082949

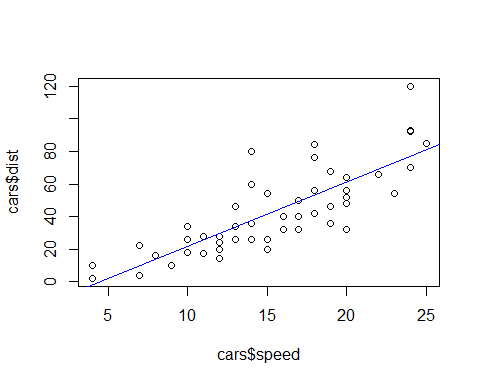
predict(out, newdata=data.frame(speed=c(3,4,5)), interval="confidence")

## fit lwr upr  
## 1 -5.781869 -17.02659 5.462853  
## 2 -1.849460 -12.32954 8.630624  
## 3 2.082949 -7.64415 11.810048

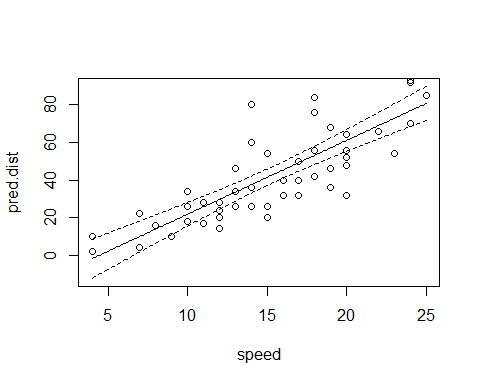
# 신뢰구간의 하한과 상한 제시, 평균적인 차량에 대한 신뢰구간  
  
predict(out, newdata=data.frame(speed=c(3,4,5)), interval="prediction")

## fit lwr upr  
## 1 -5.781869 -38.68565 27.12192  
## 2 -1.849460 -34.49984 30.80092  
## 3 2.082949 -30.33359 34.49948

# 특정 속도를 가잔 차량 한대의 제동거리는 평균적인 차량에 비해 오차가 크므로 범위가 더 넓어짐  
  
# 단순회귀모형의 시각화  
plot(cars$speed, cars$dist)  
abline(coef(out), col="blue")



speed <- seq(min(cars$speed), max(cars$speed),.1)  
pred.dist <- predict(out, newdata=data.frame(speed=speed), interval="confidence")  
matplot(speed, pred.dist, type='n')  
matlines(speed, pred.dist, lty=c(1,2,2), col=1) # 선형 회귀식은 직선, 신뢰구간은 점선으로 표현  
matpoints(cars$speed, cars$dist, pch=1)



###########################################################  
# 단순회귀분석 Summary 1  
  
# 1. 설명변수와 반응변수의 산점도를 그린다. plot()  
# - 설명변수와 반응변수 간 1차 선형관계가 있는지 확인한다.  
  
# 2. 상관분석을 통해 설명변수와 반응변수의 1차 선형관계를 확인한다. cor()  
# - p-value<0.05 이하 이면 유의미한 관계가 있다고 판단.  
  
# 3. lm()함수를 이용하여 1차 선형식을 추정한다.  
  
# 4. F통계량으로 설명변수의 회귀모형의 유의성을 확인한다.  
  
# 5. 결정계수를 통해 선형회귀모형의 설명력을 정량적으로 계산한다.  
  
# 6. 추정된 회귀계수를 통해 선형식을 구한다.  
  
# 7. predict()함수로 새로운 값의 예측값을 계산한다.   
  
alligator = data.frame(  
 lnLength = c(3.87, 3.61, 4.33, 3.43, 3.81, 3.83, 3.46, 3.76,  
 3.50, 3.58, 4.19, 3.78, 3.71, 3.73, 3.78),  
 lnWeight = c(4.87, 3.93, 6.46, 3.33, 4.38, 4.70, 3.50, 4.50,  
 3.58, 3.64, 5.90, 4.43, 4.38, 4.42, 4.25)  
)  
  
# Q.1 다음은 악어의 길이와 무게로 구성된 자료이다.  
# 연구자가 악어의 길이로 무게를 예측하기 위한 선형식을 구한다고 한다.   
# 적합한 선형식은?  
# Q.2. summary()함수를 이용하여 회귀분석 결과에 대해 해석하세요.  
# Q.3 길이가 4.5인 악어가 잡혔다고 한다. 이 악어의 예상 무게는?   
# Q.4 회귀분석 결과를 시각화 하세요   
  
plot(alligator$lnWeight, alligator$lnLength) # 산점도

