감독학습 - supervised learning

Jinseog Kim
Dongguk University
jskim1986@gmail.com

2018-03-14

Contents

1 감독학습: Supervised learning

1. generalized linear models
    1. linear regression
    2. logistic regression
    3. poisson, gamma, ···
2. non-parametric models
    1. decision tree, knn
3. neural network(1 hidden layer)
4. penalized regression models
    1. ridge - l_2 penalty
    2. lasso - l_1 penalty
    3. elastic net - $l_1 + l_2$ penalty
5. ensembles
    1. bagging
    2. various boosting
    3. random forest
6. deep learning : tensorflow, h2o

1.1 모델링 절차 및 R methods(functions)

1. 모형 적합 `fit <- knn(trainingData, outcome, k = 5)`
2. 모형 결과 `print`, `plot`, `summary`
3. 모형과 검증자료를 이용한 예측 `predict(fit, newdata)`
4. 모형 평가 `ROCR`, 'caret' packages

2 회귀분석모형

2.1 R패키지 및 관련함수

1. R패키지: stats
2. R함수

function	descriptions
lm, glm	fit linear or generalized linear model
update()	update model
anova()	ANOVA table
summary()	model summary, parameter estimates
predict()	predict with new data
step()	variable selection, eg. step-wise

2.2 회귀분석 예제: 난방비 데이터

 데이터

```
rd <- read.csv("data/난방비.csv", header=T)
dim(rd)
```

```
## [1] 20  5
```

```
head(rd, 3)
```

```
##   난방비 외부온도 단열재두께 창문수 아파트나이
## 1    252     1.65          3     10          6
## 2    382    -1.65          4      1         10
## 3    167     2.20          7      9          3
```

 모형적합

```
gfit <- lm(난방비~.,data=rd)
gfit
```

```
##
## Call:
## lm(formula = 난방비 ~ ., data = rd)
##
## Coefficients:
## (Intercept)     외부온도    단열재두께        창문수    아파트나이
##     294.460       -8.442      -14.907       -1.165        6.263
```

 추정 모형 결과

```
summary(gfit)
```

```
##
## Call:
## lm(formula = 난방비 ~ ., data = rd)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -79.221 -36.600  -1.548  28.339  87.483
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  294.460     64.615   4.557 0.000378 ***
## 외부온도      -8.442      1.553  -5.436 6.89e-05 ***
## 단열재두께   -14.907      5.307  -2.809 0.013214 *
## 창문수        -1.165      5.114  -0.228 0.822815
## 아파트나이     6.263      4.310   1.453 0.166843
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 54.43 on 15 degrees of freedom
## Multiple R-squared:  0.7975, Adjusted R-squared:  0.7435
## F-statistic: 14.77 on 4 and 15 DF,  p-value: 4.385e-05
```

2.2.1 회귀분석 예제: Boston housing data

- 데이터

```
library(MASS)
dim(Boston)
```

```
## [1] 506  14
```

```
head(Boston, 2)
```

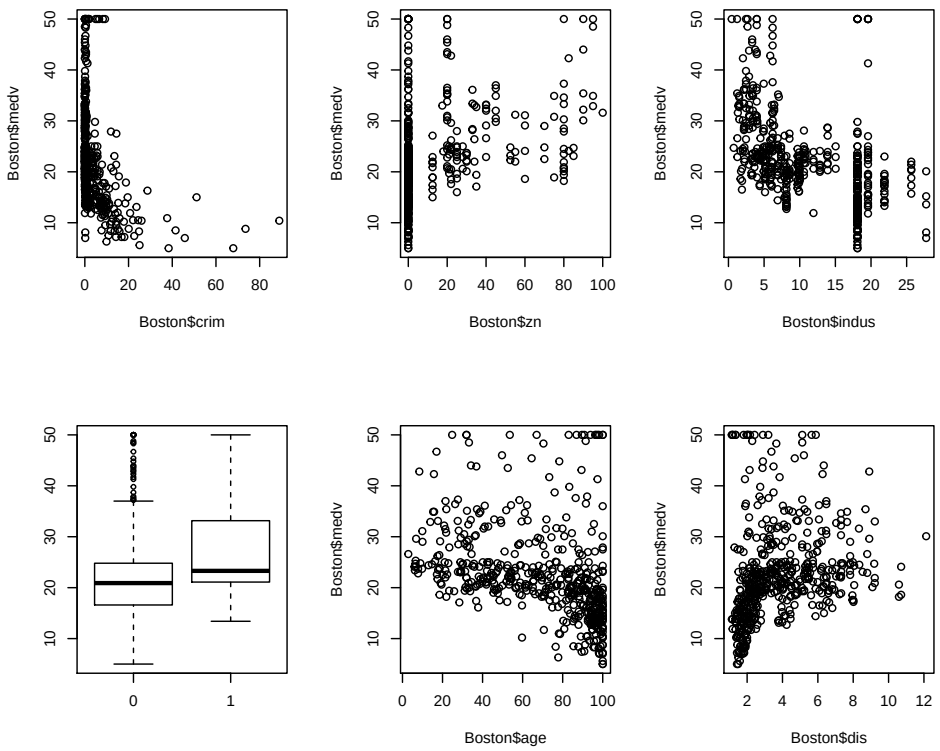
```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio black
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.9
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.9
##   lstat medv
## 1  4.98 24.0
## 2  9.14 21.6
```

 변수 설명

variables	description
crim	지역의 범죄율
zn	25,000 sq.ft.당 주거지 비율
indus	비소매사업의 면적비율
chas	Charles강유역 여부, dummy
nox	질소 산화물 농도
rm	주택 당 평균 방의 수
age	1940년 이전에 건설된 자가 주택 비율
dis	5개의 보스턴 고용 센터까지의 평균 거리
rad	고속도로 접근성 지수
tax	1만 달러당 재산세율
ptratio	학생당 교사비율
black	1000(흑인비율 - 0.63)^2
lstat	저소득층 비율
medv	자가소유 주택 가격의 중앙값

 탐색적자료분석 (EDA)

```
par(mfrow=c(2,3))
plot(Boston$crim, Boston$medv)
plot(Boston$zn, Boston$medv)
plot(Boston$indus, Boston$medv)
boxplot(medv~chas, data=Boston)
plot(Boston$age, Boston$medv)
plot(Boston$dis, Boston$medv)
```


Boston$medv
Boston$crim
Boston$zn
Boston$indus
Boston$age
Boston$dis

 모형적합

```
lm.fit <- lm(medv~.,data=Boston)
#summary(lm.fit)
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
zn           4.642e-02  1.373e-02   3.382 0.000778 ***
indus        2.056e-02  6.150e-02   0.334 0.738288
chas         2.687e+00  8.616e-01   3.118 0.001925 **
nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
age          6.922e-04  1.321e-02   0.052 0.958229
dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
black        9.312e-03  2.686e-03   3.467 0.000573 ***
lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.745 on 492 degrees of freedom
Multiple R-squared:  0.7406,  Adjusted R-squared:  0.7338
F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

 모형 갱신(업데이트)

```r
# model update
lm.fit1 <- update(lm.fit, ~.-indus-age)
summary(lm.fit1)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat, data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.5984  -2.7386  -0.5046   1.7273  26.2373
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  36.341145   5.067492   7.171 2.73e-12 ***
## crim         -0.108413   0.032779  -3.307 0.001010 **
## zn            0.045845   0.013523   3.390 0.000754 ***
## chas          2.718716   0.854240   3.183 0.001551 **
## nox         -17.376023   3.535243  -4.915 1.21e-06 ***
```

```
## rm            3.801579   0.406316   9.356  < 2e-16 ***
## dis          -1.492711   0.185731  -8.037 6.84e-15 ***
## rad           0.299608   0.063402   4.726 3.00e-06 ***
## tax          -0.011778   0.003372  -3.493 0.000521 ***
## ptratio      -0.946525   0.129066  -7.334 9.24e-13 ***
## black         0.009291   0.002674   3.475 0.000557 ***
## lstat        -0.522553   0.047424 -11.019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.736 on 494 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7348
## F-statistic: 128.2 on 11 and 494 DF,  p-value: < 2.2e-16
```

2.3 변수변환 후 회귀모형

- 입력변수의 변환

  1. polynomial : I(x^2), poly(x, k)
  2. log(x), exp(x)

- 변환된 변수에 대한 모형적합

```
lm.fit3 <- lm(medv~.+poly(lstat,10), data=Boston)
lm.fit4 <- lm(medv~.+log(rm),data=Boston)
```

- 잔차분석

```
par(mfrow=c(2,2)); plot(lm.fit1)
```

3 로지스틱회귀를 이용한 UniversalBank 자료의 분석

3.1 데이터

- UniversalBank

```
loan <- read.csv("data/UniversalBank.csv")
```

```
loan[1:3,]
```

```
##   Age Experience Income Family CCAvg Education Mortgage PersonalLoan
## 1  25          1     49      4   1.6         1        0            0
## 2  45         19     34      3   1.5         1        0            0
## 3  39         15     11      1   1.0         1        0            0
##   SecuritiesAccount CDAccount Online CreditCard
## 1                 1         0      0          0
## 2                 1         0      0          0
## 3                 0         0      0          0
```

3.2 자료 탐색

 반응변수 및 입력변수

```
loan$Education <- factor(loan$Education)
# binary: PersonalLoan, SecuritiesAccount, CDAccount, Online, CreditCard
t1 <- table(PersonalLoan=loan$PersonalLoan, CreditCard=loan$CreditCard)
sweep(t1, 1, apply(t1, 1, sum), "/")
```

```
##             CreditCard
## PersonalLoan         0         1
##            0 0.7064159 0.2935841
##            1 0.7020833 0.2979167
```

3.3 훈련자료 및 검증자료의 준비

 loan자료를 분할

```
# 전체자료의 수
(nsample <- nrow(loan))
```

```
## [1] 5000
```

```
# 자료의 분할 : tr.idx는 훈련자료의 번호(인덱스)
#             : 전체 자료의 인덱스에서 tr.idx를 제외하면 검증자료가 됨
set.seed(1234)
tr.idx <- sample(nsample, 2500, replace=F)
```

3.4 모형 적합

- 함수: glm(formula, data, family=)
  - glm(formula, data, family="binomial"): 로지스틱회귀

```
fit <- glm(PersonalLoan~., data=loan, subset=tr.idx, family="binomial")
#summary(fit)
```

```
Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -1.424e+01  2.708e+00  -5.259 1.45e-07 ***
Age                1.829e-02  9.878e-02   0.185 0.853138
Experience        -1.512e-02  9.797e-02  -0.154 0.877324
Income             6.450e-02  4.632e-03  13.925  < 2e-16 ***
Family             6.645e-01  1.160e-01   5.728 1.02e-08 ***
CCAvg              1.466e-01  6.633e-02   2.209 0.027142 *
Education2         4.148e+00  4.074e-01  10.182  < 2e-16 ***
Education3         4.230e+00  3.978e-01  10.634  < 2e-16 ***
Mortgage           1.812e-03  8.815e-04   2.056 0.039783 *
SecuritiesAccount -1.088e+00  4.451e-01  -2.443 0.014549 *
CDAccount          3.728e+00  5.395e-01   6.909 4.87e-12 ***
```

```
Online            -8.387e-01  2.428e-01  -3.455 0.000551 ***
CreditCard        -1.327e+00  3.429e-01  -3.869 0.000109 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1512.69  on 2499  degrees of freedom
Residual deviance:  540.77  on 2487  degrees of freedom
AIC: 566.77

Number of Fisher Scoring iterations: 8
```

3.5 변수선택

 단계적 선택법을 이용한 변수 선택

```
fit2 <- step(fit)
```

```
## Start:  AIC=566.77
## PersonalLoan ~ Age + Experience + Income + Family + CCAvg + Education +
##     Mortgage + SecuritiesAccount + CDAccount + Online + CreditCard
##
##                     Df Deviance    AIC
## - Experience         1   540.80 564.80
## - Age                1   540.81 564.81
## <none>                   540.77 566.77
## - Mortgage           1   544.92 568.92
## - CCAvg              1   545.69 569.69
## - SecuritiesAccount  1   547.64 571.64
## - Online             1   553.02 577.02
## - CreditCard         1   558.38 582.38
## - Family             1   577.22 601.22
## - CDAccount          1   595.57 619.57
## - Education          2   748.95 770.95
```

```
## - Income             1   932.76 956.76
##
## Step:  AIC=564.8
## PersonalLoan ~ Age + Income + Family + CCAvg + Education + Mortgage +
##     SecuritiesAccount + CDAccount + Online + CreditCard
##
##                     Df Deviance    AIC
## - Age                1   540.89 562.89
## <none>                   540.80 564.80
## - Mortgage           1   545.05 567.05
## - CCAvg              1   545.73 567.73
## - SecuritiesAccount  1   547.73 569.73
## - Online             1   553.03 575.03
## - CreditCard         1   558.40 580.40
## - Family             1   577.26 599.26
## - CDAccount          1   595.58 617.58
## - Education          2   755.55 775.55
## - Income             1   937.39 959.39
##
## Step:  AIC=562.89
## PersonalLoan ~ Income + Family + CCAvg + Education + Mortgage +
##     SecuritiesAccount + CDAccount + Online + CreditCard
```

```
##
##                       Df Deviance    AIC
## <none>                     540.89 562.89
## - Mortgage             1   545.10 565.10
## - CCAvg                1   545.74 565.74
## - SecuritiesAccount    1   547.76 567.76
## - Online               1   553.10 573.10
## - CreditCard           1   558.75 578.75
## - Family               1   577.32 597.32
## - CDAccount            1   595.93 615.93
## - Education            2   755.68 773.68
## - Income               1   937.47 957.47
```

```
#summary(fit2)
```

```
Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)       -1.371e+01  8.823e-01 -15.536  < 2e-16 ***
Income             6.446e-02  4.608e-03  13.987  < 2e-16 ***
Family             6.646e-01  1.161e-01   5.726 1.03e-08 ***
CCAvg              1.438e-01  6.563e-02   2.192 0.028413 *
Education2         4.148e+00  4.070e-01  10.191  < 2e-16 ***
```

```
Education3         4.234e+00  3.944e-01  10.734  < 2e-16 ***
Mortgage           1.815e-03  8.762e-04   2.072 0.038296 *
SecuritiesAccount -1.084e+00  4.434e-01  -2.446 0.014463 *
CDAccount          3.729e+00  5.390e-01   6.920 4.53e-12 ***
Online            -8.373e-01  2.427e-01  -3.450 0.000561 ***
CreditCard        -1.334e+00  3.425e-01  -3.895 9.82e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1512.69  on 2499  degrees of freedom
Residual deviance:  540.89  on 2489  degrees of freedom
AIC: 562.89

Number of Fisher Scoring iterations: 8
```

3.6 예측

- confusion matrix (오분류표)

		Predicted($\hat{y}$)	
		true($\hat{y}=1$)	false($\hat{y}=0$)
Actural(y)	positive (1)	True Positive (TP)	False Negative (FN) (type II error)
	negative (0)	False Positive (FP) (Type I error)	True Negative (TN)

  - True positive rate (Recall:재현율, Sensitivity: 민감도): $P(\hat{Y}=+|Y=+)$
  - Error rate(오분류율): $P(\hat{Y}!=Y)$
  - Precision (정확률): $P(Y=+|\hat{Y}=+)$
  - Specificity (특이도) : $P(\hat{Y}=-|Y=-)$

- 민감도(sensitivity)와 특이도(specificity)
  - 민감도(sensitivity) - 질병에 걸린 환자 중에서, 진단 결과가 양성으로 나올 확률
  - 특이도(specificity) - 질병에 걸리지 않은 환자 중, 진단 결과가 음성으로 나올 확률

 예측 및 오분류표

  predict(fit, newdata, type=c("link", "response", "terms"))

```
pred <- predict(fit2, newdata=loan[-tr.idx,], type="response")
(t1 <- table(y=loan$PersonalLoan[-tr.idx], pred=pred>0.5))
```

```
##    pred
## y   FALSE TRUE
##   0  2211   34
##   1    79  176
```

```
(err <- 1- sum(diag(t1))/sum(t1))
```

```
## [1] 0.0452
```

3.7 ROC(Receiver Operator Charastics)를 이용한 모형 평가

- R 패키지 및 함수

```
library(ROCR)
```

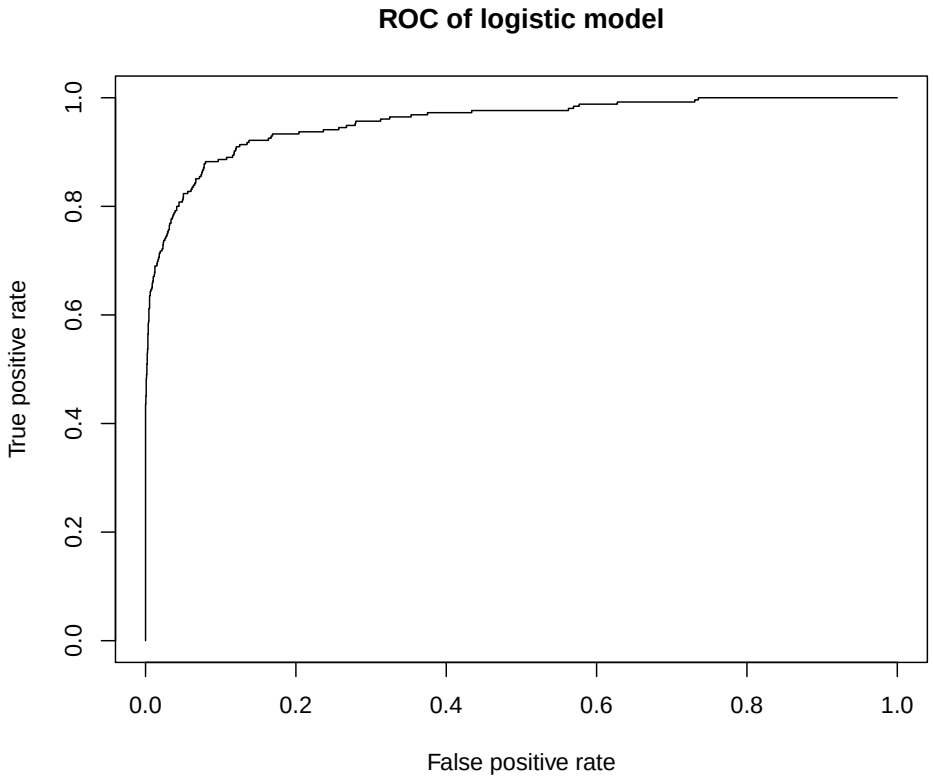
1. prediction(predicted, y)
2. performance(pred, measure, x.measure)
   - "tpr" = "rec" = "sens": $P(\hat{Y} = +|Y = +)$
   - "fpr": $P(\hat{Y} = +|Y = -)$
   - "err": $P(\hat{Y} ! = Y)$
   - "prec": $P(Y = +|\hat{Y} = +)$
   - "spec"="tnr" : $P(\hat{Y} = -|Y = -)$
3. plot()

- ROC: "fpr"(1-spec) vs "tpr"(sens)

- AUC: area under the ROC curve

 Plot ROC

```
p <- prediction(pred, loan$PersonalLoan[-tr.idx])
perf <- performance(p, measure = "tpr", x.measure = "fpr")
plot(perf, main="ROC of logistic model")
```



- Plot ROC curve and add AUC (Area Under the Curve)

```
plot(perf, colorize=TRUE, print.cutoffs.at=seq(0,1,by=0.1), text.adj=c(-0.2,1.7))
abline(0,1)
auc = as.numeric(performance(p, "auc")@y.values)
text(0.5, 1, paste("AUC:",round(auc,4)))
```