LBB Classification1

Enlik

21 February 2019

Summmary

This is `Learning by Building` project for Classification I in Machine Learning. We will use `wholesale.csv` dataset for customer segment prediction case.

We will use two models:

1. Logistic Regression Model using `glm()` function
2. KNN Model using `knn()` function

Following Question That I've tried to answer:

- If you use a logistic regression, how do we correctly interpret the negative coefficients obtained from your logistic regression?
- What is your accuracy? Was the logistic regression better than kNN in terms of accuracy? (recall the lesson on obtaining an unbiased estimate of the model's accuracy)
- Was the logistic regression better than our kNN model at explaining which of the variables are good predictors?
- What are some strategies to improve your model?
- List down 1 disadvantage and 1 strength of each of the approach (kNN and logistic regression)

Read Data

```
# Read the dataset in, drop the "Region" feature because it's not interesting
wholesale <- read.csv("data_input/wholesale.csv", header=TRUE)
wholesale <- wholesale[,-2]
str(wholesale)
```

```
## 'data.frame':    440 obs. of  7 variables:
##  $ Channel         : int  2 2 2 1 2 2 2 2 1 2 ...
##  $ Fresh           : int  12669 7057 6353 13265 22615 9413 12126 7579 5963 6006 ...
##  $ Milk            : int  9656 9810 8808 1196 5410 8259 3199 4956 3648 11093 ...
##  $ Grocery         : int  7561 9568 7684 4221 7198 5126 6975 9426 6192 18881 ...
##  $ Frozen          : int  214 1762 2405 6404 3915 666 480 1669 425 1159 ...
##  $ Detergents_Paper: int  2674 3293 3516 507 1777 1795 3140 3321 1716 7425 ...
##  $ Delicassen      : int  1338 1776 7844 1788 5185 1451 545 2566 750 2098 ...
```

Identify Label

```
wholesale$Industry <- factor(wholesale$Channel, levels = c(1, 2), labels = c("horeca", "retail"))

# After doing that we can remove the original Channel feature
wholesale <- wholesale[,-1]
table(wholesale$Industry)
```

```
##
## horeca retail
##    298    142
```

Identify Feature And Scaling (Normalize)

```
wholesale.z <- as.data.frame(scale(wholesale[,-7]))
summary(wholesale.z)
```

```
##      Fresh              Milk            Grocery            Frozen
##  Min.   :-0.9486   Min.   :-0.7779   Min.   :-0.8364   Min.   :-0.62763
##  1st Qu.:-0.7015   1st Qu.:-0.5776   1st Qu.:-0.6101   1st Qu.:-0.47988
##  Median :-0.2764   Median :-0.2939   Median :-0.3363   Median :-0.31844
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.: 0.3901   3rd Qu.: 0.1889   3rd Qu.: 0.2846   3rd Qu.: 0.09935
##  Max.   : 7.9187   Max.   : 9.1732   Max.   : 8.9264   Max.   :11.90545
##  Detergents_Paper    Delicassen
##  Min.   :-0.6037   Min.   :-0.5396
##  1st Qu.:-0.5505   1st Qu.:-0.3960
##  Median :-0.4331   Median :-0.1984
##  Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.2182   3rd Qu.: 0.1047
##  Max.   : 7.9586   Max.   :16.4597
```

```
wholesale.n <- as.data.frame(cbind(wholesale.z, Industry = wholesale$Industry))
summary(wholesale.n)
```

```
##      Fresh              Milk            Grocery            Frozen
##  Min.   :-0.9486   Min.   :-0.7779   Min.   :-0.8364   Min.   :-0.62763
##  1st Qu.:-0.7015   1st Qu.:-0.5776   1st Qu.:-0.6101   1st Qu.:-0.47988
##  Median :-0.2764   Median :-0.2939   Median :-0.3363   Median :-0.31844
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.: 0.3901   3rd Qu.: 0.1889   3rd Qu.: 0.2846   3rd Qu.: 0.09935
##  Max.   : 7.9187   Max.   : 9.1732   Max.   : 8.9264   Max.   :11.90545
##  Detergents_Paper    Delicassen        Industry
##  Min.   :-0.6037   Min.   :-0.5396   horeca:298
##  1st Qu.:-0.5505   1st Qu.:-0.3960   retail:142
##  Median :-0.4331   Median :-0.1984
##  Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.2182   3rd Qu.: 0.1047
##  Max.   : 7.9586   Max.   :16.4597
```

```
prop.table(table(wholesale.n$Industry))
```

```
##
##    horeca    retail
## 0.6772727 0.3227273
```

Split Train and Test Set

```r
set.seed(9999)
intrain <- sample(nrow(wholesale.n), nrow(wholesale.n) * 0.8)
wholesale.train <- wholesale.n[intrain, ]
wholesale.test <- wholesale.n[-intrain, ]
table(wholesale.train$Industry)
```

```
##
## horeca retail
##    236    116
```

Train with Logistic Regression Model

```r
logistic.model <- glm(Industry ~ Fresh + Milk + Grocery + Frozen + Detergents_Paper + Delicassen, whole
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(logistic.model)
```

```
##
## Call:
## glm(formula = Industry ~ Fresh + Milk + Grocery + Frozen + Detergents_Paper +
##     Delicassen, family = "binomial", data = wholesale.n)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8094  -0.3163  -0.2285   0.0395   3.1918
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -0.44511    0.22435  -1.984   0.0473 *
## Fresh             0.08161    0.21503   0.380   0.7043
## Milk              0.54409    0.39772   1.368   0.1713
## Grocery           1.11829    0.56122   1.993   0.0463 *
## Frozen           -0.81140    0.44698  -1.815   0.0695 .
## Detergents_Paper  4.02219    0.63599   6.324 2.54e-10 ***
## Delicassen       -0.19081    0.30625  -0.623   0.5333
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 553.44  on 439  degrees of freedom
## Residual deviance: 203.91  on 433  degrees of freedom
## AIC: 217.91
##
## Number of Fisher Scoring iterations: 7
```

```r
wholesale.test$horeca_pred_logistic <- predict(logistic.model, wholesale.test, type = "response")
```

```r
predict(logistic.model, head(wholesale.test), type = "response")
```

```
##          3          9         12         19         20         24
## 0.48206374 0.20786317 0.05691216 0.45651094 0.40770621 0.96850846
```

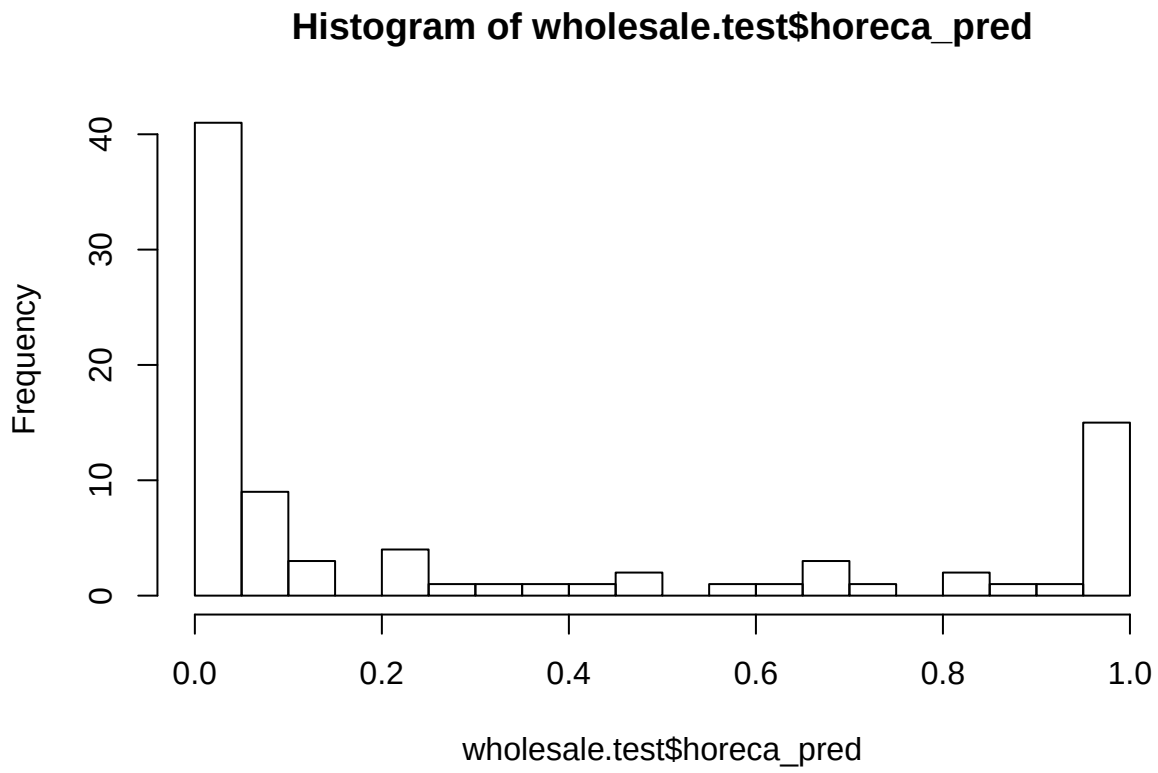
```r
predict(logistic.model, head(wholesale.test), type = "link")
```

```
##           3           9          12          19          20          24
## -0.07177584 -1.33785412 -2.80765045 -0.17439690 -0.37345593  3.42603823
```

```r
predict(logistic.model, head(wholesale.test))
```

```
##           3           9          12          19          20          24
## -0.07177584 -1.33785412 -2.80765045 -0.17439690 -0.37345593  3.42603823
```

```r
hist(wholesale.test$horeca_pred, breaks = 20)
```



Create Confusion Matrix for Logistic Regression

```r
wholesale.test$pred.isHoreca <- ifelse(wholesale.test$horeca_pred_logistic <= 0.5, 1, 0)
table("predicted" = wholesale.test$pred.isHoreca, "actual" = wholesale.test$Industry)
```

```
##          actual
## predicted horeca retail
##         0      5     20
##         1     57      6
```

Calculate matrices

```
accu.log <- round((57+20)/nrow(wholesale.test), 2)
reca.log <- round(57/(5+57), 2)
prec.log <- round(57/(6+57), 2)
spec.log <- round(20/(20+6), 2)

paste("Accuracy:", accu.log)
```

```
## [1] "Accuracy: 0.88"
```

```
paste("Recall:", reca.log)
```

```
## [1] "Recall: 0.92"
```

```
paste("Precision:", prec.log)
```

```
## [1] "Precision: 0.9"
```

```
paste("Specificity:", spec.log)
```

```
## [1] "Specificity: 0.77"
```

Train with KNN Model

```
require("class")
```

```
## Loading required package: class
```

```
horeca_pred_knn <- knn(train = wholesale.train[,1:6],
                       test = wholesale.test[,1:6],
                       cl = wholesale.train$Industry,
                       k = 21)
```

Create Confusion Matrix for Logistic Regression

```
table("actual" = wholesale.test$Industry, "predicted" = horeca_pred_knn)
```

```
##         predicted
## actual   horeca retail
##   horeca     55      7
##   retail      4     22
```

Calculate matrices

```
accu.knn <- round((55+22)/nrow(wholesale.test), 2)
reca.knn <- round(55/(4+55), 2)
prec.knn <- round(55/(7+55), 2)
spec.knn <- round(22/(22+7), 2)

paste("Accuracy:", accu.knn)
```

```
## [1] "Accuracy: 0.88"
```

```
paste("Recall:", reca.knn)
```

```
## [1] "Recall: 0.93"
```

```
paste("Precision:", prec.knn)
```

```
## [1] "Precision: 0.89"
```

```
paste("Specificity:", spec.knn)
```

```
## [1] "Specificity: 0.76"
```

Conclusion

Question: If you use a logistic regression, how do we correctly interpret the negative coefficients obtained from your logistic regression?

Answer: The coefficient tells us about how much the dependent variable is expected to increase (if the coefficient is positive) or decrease (if the coefficient is negative) when that independent variable increases by one

Question: What is your accuracy? Was the logistic regression better than kNN in terms of accuracy? (recall the lesson on obtaining an unbiased estimate of the model's accuracy)

Answer: In my case, it's same. Logistic Regression: "Accuracy: 0.88"

kNN: "Accuracy: 0.88"

Question: What are some strategies to improve your model?

Answer:

- Testing multiple models
- Applying feature engineering
- Selecting features and examples
- Looking for more data

Reference

Question: List down 1 disadvantage and 1 strength of each of the approach (kNN and logistic regression)

Answer: kNN advantages:

- Simple technique that is easily implemented
- Building model is cheap
- Extremely flexible classification scheme

kNN disadvantages:

- Classifying unknown records are relatively expensive
- Accuracy can be severely degraded by the presence of noisy or irrelevant features

Logistic Regression advantages:

- It's very efficient and doesn't require too many computational resources
- It's highly interpretable
- It's easy to regularize

Logistic Regression disadvantages:

- It can't solve non-linear problems

- Its high reliance on a proper presentation of your data, it means that logistic regression is not a useful tool unless you have already identified all the important independent variables
- Can only predict a categorical outcome

Reference_1 Reference_2