Exercício Aplicado 03 - MAE 0501

Vítor Garcia Comissoli

2024-10-29

13)

a)

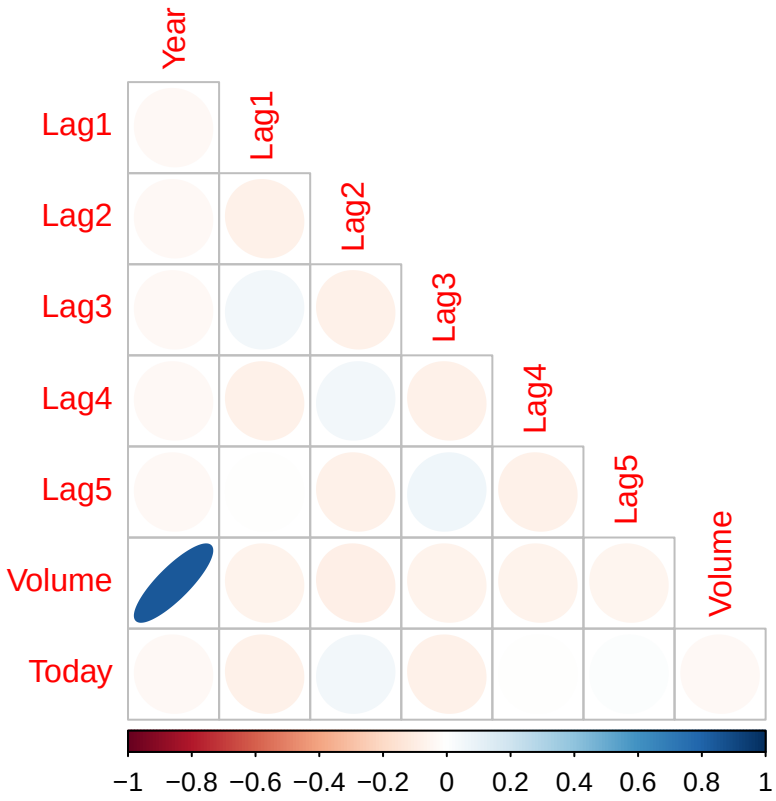
```
head(Weekly)
```

```
##   Year   Lag1   Lag2   Lag3   Lag4   Lag5    Volume  Today Direction
## 1 1990  0.816  1.572 -3.936 -0.229 -3.484 0.1549760 -0.270      Down
## 2 1990 -0.270  0.816  1.572 -3.936 -0.229 0.1485740 -2.576      Down
## 3 1990 -2.576 -0.270  0.816  1.572 -3.936 0.1598375  3.514        Up
## 4 1990  3.514 -2.576 -0.270  0.816  1.572 0.1616300  0.712        Up
## 5 1990  0.712  3.514 -2.576 -0.270  0.816 0.1537280  1.178        Up
## 6 1990  1.178  0.712  3.514 -2.576 -0.270 0.1544440 -1.372      Down
```

```
summary(Weekly)
```

```
##       Year           Lag1               Lag2               Lag3         
##  Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.1950  
##  1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.1580  
##  Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.2410  
##  Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.1472  
##  3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.4090  
##  Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.0260  
##       Lag4               Lag5              Volume           Today         
##  Min.   :-18.1950   Min.   :-18.1950   Min.   :0.08747   Min.   :-18.1950  
##  1st Qu.: -1.1580   1st Qu.: -1.1660   1st Qu.:0.33202   1st Qu.: -1.1540  
##  Median :  0.2380   Median :  0.2340   Median :1.00268   Median :  0.2410  
##  Mean   :  0.1458   Mean   :  0.1399   Mean   :1.57462   Mean   :  0.1499  
##  3rd Qu.:  1.4090   3rd Qu.:  1.4050   3rd Qu.:2.05373   3rd Qu.:  1.4050  
##  Max.   : 12.0260   Max.   : 12.0260   Max.   :9.32821   Max.   : 12.0260  
##  Direction 
##  Down:484  
##  Up  :605  
##            
##            
##            
##
```

```
corrplot(cor(Weekly[, -9]), type = "lower", diag = FALSE, method = "ellipse")
```



A variável Volume se correlaciona fortemente (positivamente) com a variável Year. De resto, as outras variáveis não apresentam correlações fortes entre si.

b)

```
fit <- glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = Weekly, family = binomial)

summary(fit)
```

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +
##     Volume, family = binomial, data = Weekly)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.26686    0.08593   3.106   0.0019 **
## Lag1        -0.04127    0.02641  -1.563   0.1181
## Lag2         0.05844    0.02686   2.175   0.0296 *
## Lag3        -0.01606    0.02666  -0.602   0.5469
## Lag4        -0.02779    0.02646  -1.050   0.2937
## Lag5        -0.01447    0.02638  -0.549   0.5833
## Volume      -0.02274    0.03690  -0.616   0.5377
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 1496.2  on 1088  degrees of freedom
## Residual deviance: 1486.4  on 1082  degrees of freedom
## AIC: 1500.4
##
## Number of Fisher Scoring iterations: 4
```

Fora o intercepto, somente a variável Lag2 se mostrou estatisticamente significante (à um α de 5%)

c)

```
contrasts(Weekly$Direction)
```

```
##      Up
## Down  0
## Up    1
```

```
pred <- predict(fit, type = "response") > 0.5

(t <- table(ifelse(pred, "Up (pred)", "Down (pred)"), Weekly$Direction))
```

```
##
##                Down  Up
##   Down (pred)    54  48
##   Up (pred)     430 557
```

```
sum(diag(t)) / sum(t)
```

```
## [1] 0.5610652
```

A fração de acertos do modelo é de aproximadamente 56,1%.

Observa-se que, mesmo que a regressão logística preve bem movimentos para cima, ela prevê incorretamente grande parte dos movimentos para baixo como movimentos para cima.

d)

```
train <- Weekly$Year < 2009

fit <- glm(Direction ~ Lag2, data = Weekly[train, ], family = binomial)

summary(fit)
```

```
##
## Call:
## glm(formula = Direction ~ Lag2, family = binomial, data = Weekly[train,
##     ])
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.20326    0.06428   3.162  0.00157 **
## Lag2         0.05810    0.02870   2.024  0.04298 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 1354.7  on 984  degrees of freedom
## Residual deviance: 1350.5  on 983  degrees of freedom
## AIC: 1354.5
##
## Number of Fisher Scoring iterations: 4
```

```
pred <- predict(fit, Weekly[!train, ], type = "response") > 0.5

(t <- table(ifelse(pred, "Up (pred)", "Down (pred)"), Weekly[!train, ]$Direction))
```

```
##
##                Down Up
##   Down (pred)    9  5
##   Up (pred)     34 56
```

```
sum(diag(t)) / sum(t)
```

```
## [1] 0.625
```

A fração de acertos do modelo é de 62,5%, o que é melhor que o modelo anterior, porém ainda não é um modelo considerado adequado.

Observa-seèo mesmo problema detectado no modelo anterior, onde a regressão logística prevê incorretamente grande parte dos movimentos para baixo como movimentos para cima.

g)

```
fit <- knn(Weekly[train, "Lag2", drop = FALSE], Weekly[!train, "Lag2", drop = FALSE],Weekly$Direction[t

summary(fit)
```

```
## Down   Up
##   50   54
```

```
(t <- table(fit, Weekly[!train, ]$Direction))
```

```
##
## fit    Down Up
##   Down   21 29
##   Up     22 32
```

```
sum(diag(t)) / sum(t)
```

```
## [1] 0.5096154
```

A fração de acertos do modelo é de aproximadamente 51%, o que é pior que os modelos anteriores.

Vale ressaltar também que, ao contrário dos modelos anrteriores, o modelo KNN não preve bem os movimentos em geral, tanto para cima como para baixo, onde a frequência de classificação para os erros e acertos para tanto movimentos para cima, quanto movimentos para baixo se mostra equilibrada (ver tabela plotada acima).

h)

```
fit <- naiveBayes(Direction ~ Lag2, data = Weekly, subset = train)

fit
```

```
##
## Naive Bayes Classifier for Discrete Predictors
```

```
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##      Down        Up
## 0.4477157 0.5522843
##
## Conditional probabilities:
##       Lag2
## Y             [,1]     [,2]
##   Down -0.03568254 2.199504
##   Up    0.26036581 2.317485
```

```
pred <- predict(fit, Weekly[!train, ], type = "class")

(t <- table(pred, Weekly[!train, ]$Direction))
```

```
##
## pred   Down Up
##   Down    0  0
##   Up     43 61
```

```
sum(diag(t)) / sum(t)
```

```
## [1] 0.5865385
```

A fração de acertos do modelo é de aproximadamente 58,6%, o que é melhor que o modelo KNN e o primeiro modelo de regressão logístico, porém ainda não é um modelo considerado adequado.

Observa-se que o modelo Naive Bayes prevê bem movimentos para cima, porém prevê incorretamente todos os movimentos para baixo como movimentos para cima (já que classifica todos os movimentos como movimentos para cima), o que surpreendemente produz um classificador melhor que o gerado por KNN e pelo primeiro modelo de regressão logística.

i)

O melhor classificador dentre os testados foi o modelo de regressão logística com a variável Lag2, que obteve uma fração de acertos de 62,5%. Este modelo foi o melhor dentre os testados, porém ainda não é um modelo considerado adequado.

j)

Realizando algumas experimentações com os modelos testados anteriormente, temos:

```
fit <- glm(Direction ~ Lag1, data = Weekly[train, ], family = binomial)

pred <- predict(fit, Weekly[!train, ], type = "response") > 0.5

mean(ifelse(pred, "Up", "Down") == Weekly[!train, ]$Direction)
```

```
## [1] 0.5673077
```

A fração de acertos do modelo é de aproximadamente 56,7%.

```
fit <- glm(Direction ~ Lag3, data = Weekly[train, ], family = binomial)
```

```
pred <- predict(fit, Weekly[!train, ], type = "response") > 0.5

mean(ifelse(pred, "Up", "Down") == Weekly[!train, ]$Direction)
```

```
## [1] 0.5865385
```

A fração de acertos do modelo é de aproximadamente 58,6%.

```
fit <- glm(Direction ~ Lag4, data = Weekly[train, ], family = binomial)

pred <- predict(fit, Weekly[!train, ], type = "response") > 0.5

mean(ifelse(pred, "Up", "Down") == Weekly[!train, ]$Direction)
```

```
## [1] 0.5865385
```

A fração de acertos do modelo é de aproximadamente 58,6%.

```
fit <- glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4, data = Weekly[train, ], family = binomial)

pred <- predict(fit, Weekly[!train, ], type = "response") > 0.5

mean(ifelse(pred, "Up", "Down") == Weekly[!train, ]$Direction)
```

```
## [1] 0.5865385
```

A fração de acertos do modelo é de aproximadamente 58,6%.

```
fit <- glm(Direction ~ Lag1 * Lag2 * Lag3 * Lag4, data = Weekly[train, ], family = binomial)

pred <- predict(fit, Weekly[!train, ], type = "response") > 0.5

mean(ifelse(pred, "Up", "Down") == Weekly[!train, ]$Direction)
```

```
## [1] 0.5961538
```

A fração de acertos do modelo é de aproximadamente 59,6%.

```
fit <- naiveBayes(Direction ~ Lag1 + Lag2 + Lag3 + Lag4, data = Weekly[train, ])

pred <- predict(fit, Weekly[!train, ], type = "class")

mean(pred == Weekly[!train, ]$Direction)
```

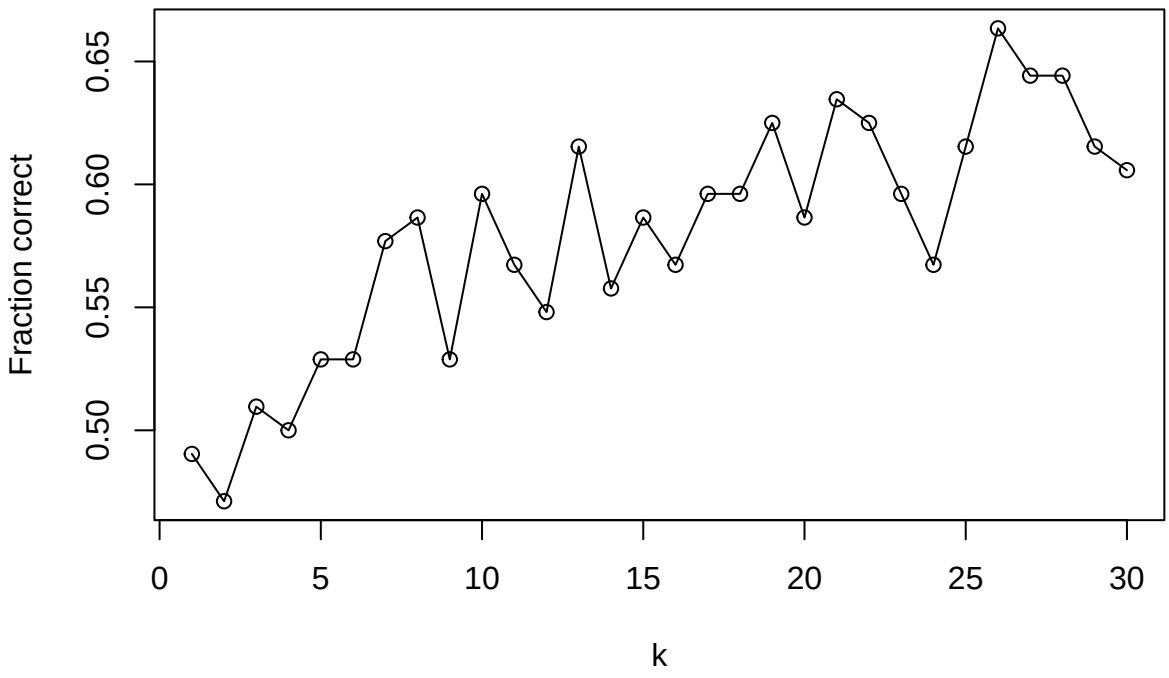
```
## [1] 0.5096154
```

A fração de acertos do modelo é de aproximadamente 51%.

```
set.seed(1181041)

res <- sapply(1:30, function(k) {fit <- knn(Weekly[train, 2:4, drop = FALSE], Weekly[!train, 2:4, drop =
  mean(fit == Weekly[!train, ]$Direction)})

plot(1:30, res, type = "o", xlab = "k", ylab = "Fraction correct")
```




```
(k <- which.max(res))
```

```
## [1] 26
```

Temos que o K que maximiza a fração de acertos é K = 26.

```
fit <- knn(Weekly[train, 2:4, drop = FALSE], Weekly[!train, 2:4, drop = FALSE], Weekly$Direction[train]

table(fit, Weekly[!train, ]$Direction)
```

```
##
## fit    Down Up
##   Down   26 18
##   Up     17 43
```

```
mean(fit == Weekly[!train, ]$Direction)
```

```
## [1] 0.6634615
```

A fração de acertos do modelo é de aproximadamente 66,3%, tornando o classificador KNN com K=26 o melhor modelo dentre os testados.