

Análise preditiva - Trabalho final

Bruno Santos Wance de Souza

Lucas de Jesus Matias
Luiz Cesar Costa Raymundo

21 de novembro de 2018

Contents

Pagamento de Empréstimo	3
Leitura dos dados	3
Criação do modelo	3
Análise das variáveis	3
Predição do modelo	3
Verificação da predição	4
Conclusão	4
 Default de crédito	 5
Leitura dos dados	5
Criação do modelo	5
Análise das variáveis	5
Modelo final	6
Predição do modelo	6
Verificação da predição	6
Conclusão	7
 Estudo de caso Customer Churn	 8
Leitura dos dados	8
Preparação de variáveis	8
Criação do modelo	8
Predição do modelo	8
Verificação da predição	9
Conclusão	9

Pagamento de Empréstimo

Leitura dos dados

Os dados do csv gerado a partir da planilha foram carregados para a variável “pagamentoEprestimo”.

```
pagamentoEmprestimo <-  
  read.csv2("./dados/pagamento_emprestimo.csv", stringsAsFactors = FALSE)
```

Criação do modelo

A funcionalidade glm foi utilizada para geração do modelo de regressão e este vinculado à variável glmPagamento.

```
glm(data = pagamentoEmprestimo,  
    formula = pagamento ~ estadocivil + idade + sexo, family = binomial) ->  
  glmPagamento
```

Análise das variáveis

Os valores Ps das variáveis rejeitam a hipótese inicial de que são irrelevantes para o modelo, portanto foram consideradas úteis todas as variáveis para a predição.

```
summary(glmPagamento)  
  
##  
## Call:  
## glm(formula = pagamento ~ estadocivil + idade + sexo, family = binomial,  
##      data = pagamentoEmprestimo)  
##  
## Deviance Residuals:  
##      Min       1Q   Median       3Q      Max   
## -2.4892  -0.4015   0.4166   0.5905   2.1662   
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)      
## (Intercept) -1.96591    1.12267  -1.751  0.07993 .      
## estadocivil -2.95095    0.58293  -5.062 4.14e-07 ***   
## idade       0.11614    0.04432   2.621  0.00877 **    
## sexo        1.30123    0.43861   2.967  0.00301 **    
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
##    Null deviance: 212.70  on 179  degrees of freedom  
## Residual deviance: 146.65  on 176  degrees of freedom  
## AIC: 154.65  
##  
## Number of Fisher Scoring iterations: 5
```

Predição do modelo

Para testar o modelo, foi realizada a predição.

```
glmprobsPagamento <- predict(glmPagamento, type="response")
```

A predição acima de 0,5 foi considerada para o pagamento do empréstimo e menor ou igual a 0,5 como não pagamento. Foi testado pontos de corte menores e maiores, mas nenhum trouxe maior predição que o ponto de corte 0,5.

```
nLinhasPagamento <- nrow(pagamentoEmprestimo)
glmpredPagamento <- rep(0, nLinhasPagamento)
glmpredPagamento[ glmprobsPagamento > 0.5 ] <- 1
```

Verificação da predição

Aplicando a predição para os dados já possuídos, obtiveram-se 24 True Negatives, 125 True Positives, de um total de 180 registros. Os pagamentos forma previstos com aproximadamente 82,8% de sucesso.

```
table(glmpredPagamento, pagamentoEmprestimo$pagamento) -> tabelaPagamentoEmprestimo
tabelaPagamentoEmprestimo
```

```
##
## glmpredPagamento    0    1
##                   0  24   5
##                   1  26 125
```

```
(as.vector(tabelaPagamentoEmprestimo)[1] + as.vector(tabelaPagamentoEmprestimo)[4]) / nLinhasPagamento
## [1] 0.8277778
```

Conclusão

O modelo gerado obteve um sucesso de predição de 82,8% de sucesso sobre os dados já possuídos.

Default de crédito

Leitura dos dados

Os dados do csv gerado a partir da planilha foram carregados para a variável “defaultCredito”.

```
defaultCredito <-  
  read.csv2("./dados/default_de_credito.csv", stringsAsFactors = FALSE)
```

Criação do modelo

A funcionalidade glm foi utilizada para a geração do modelo de regressão e este vinculado à variável glmDefaultCredito

```
glm(data = defaultCredito,  
     formula = default ~ idade + educacao + t_emprego +  
                       t_endereco + renda + divida + divida_cc +  
                       outras_div, family = binomial) ->  
  glmDefaultCredito
```

Análise das variáveis

Após análise inicial do modelo, verificamos que algumas variáveis não rejeitaram a hipótese original, por possuir o valor P muito elevado, não acrescentando relevância ao modelo.

```
summary(glmDefaultCredito)
```

```
##  
## Call:  
## glm(formula = default ~ idade + educacao + t_emprego + t_endereco +  
##      renda + divida + divida_cc + outras_div, family = binomial,  
##      data = defaultCredito)  
##  
## Deviance Residuals:  
##      Min       1Q   Median       3Q      Max   
## -2.2989  -0.6653  -0.3230   0.1586   2.8708   
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)      
## (Intercept) -1.7619012  0.7422673  -2.374 0.017612 *      
## idade        0.0305786  0.0204313   1.497 0.134483      
## educacao     0.0830897  0.1440116   0.577 0.563963      
## t_emprego    -0.2504407  0.0387710  -6.459 1.05e-10 ***   
## t_endereco   -0.0967593  0.0270678  -3.575 0.000351 ***   
## renda       -0.0003825  0.0111299  -0.034 0.972585      
## divida       0.0737017  0.0380499   1.937 0.052748 .      
## divida_cc    0.5574310  0.1286410   4.333 1.47e-05 ***   
## outras_div   0.0491476  0.0966352   0.509 0.611040      
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
##      Null deviance: 570.95  on 499  degrees of freedom  
## Residual deviance: 401.37  on 491  degrees of freedom  
## AIC: 419.37
```

```
##  
## Number of Fisher Scoring iterations: 6
```

Modelo final

Removendo as variáveis não relevantes ao modelo, uma a uma, e reexecutando o modelo após a retirada de cada uma foi possível chegar a um modelo com variáveis relevantes.

```
glm(data = defaultCredito,  
     formula = default ~ t_emprego + divida + divida_cc, family = binomial) ->  
glmDefaultCredito  
  
summary(glmDefaultCredito)
```

```
##  
## Call:  
## glm(formula = default ~ t_emprego + divida + divida_cc, family = binomial,  
##      data = defaultCredito)  
##  
## Deviance Residuals:  
##      Min       1Q   Median       3Q      Max   
## -2.2752  -0.6731  -0.3738   0.2857   2.5518   
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)      
## (Intercept) -1.25358    0.27709  -4.524 6.07e-06 ***  
## t_emprego    -0.22966    0.03090  -7.434 1.06e-13 ***  
## divida        0.08066    0.02210   3.651 0.000262 ***  
## divida_cc     0.50322    0.09776   5.148 2.64e-07 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
##      Null deviance: 570.95  on 499  degrees of freedom  
## Residual deviance: 416.05  on 496  degrees of freedom  
## AIC: 424.05  
##  
## Number of Fisher Scoring iterations: 5
```

Predição do modelo

Para testar o modelo, foi criada a predição.

```
glmprobsDefaultCredito <- predict(glmDefaultCredito, type="response")
```

A predição acima de 0,5 foi considerada como positiva para a resposta e menor ou igual a 0,5 como negativa.

```
nLinhasDefaultCredito <- nrow(defaultCredito)  
glmPredDefaultCredito <- rep(0, nLinhasDefaultCredito)  
glmPredDefaultCredito[ glmprobsDefaultCredito > 0.5 ] <- 1
```

Verificação da predição

A predição foi comparada com os dados já possuídos, obtiveram-se 350 True Negatives, 60 True Positives, de um total de 500. Foi possível prever os resultados com 82% de sucesso.

```

table(glmpredDefaultCredito, defaultCredito$default) -> tabelaDefaultCredito
tabelaDefaultCredito

##
## glmpredDefaultCredito    0    1
##                0 350  69
##                1  21  60

(as.vector(tabelaDefaultCredito)[1] + as.vector(tabelaDefaultCredito)[4]) / nLinhasDefaultCredito

## [1] 0.82

```

Conclusão

O modelo gerado obteve um sucesso de predição de 82% de sucesso sobre os dados já possuídos.

Estudo de caso Customer Churn

Leitura dos dados

Os dados do csv gerado a partir da planilha foram carregados para a variável “customerChurn”.

```
customerChurn <-  
  read.csv2("./dados/estudo_caso_customer_churn.csv", stringsAsFactors = FALSE)
```

Preparação de variáveis

O tempo de utilização dos serviços dos clientes foram segmentados de acordo com informações do cliente. Os clientes são considerados novos possuem menos de 6 meses de utilização dos serviços. Entre 6 meses e 14 foram considerados de maiores riscos. Por algum motivo foi obtido melhores resultados classificando os grupo de clientes com risco até 18 meses.

```
customerChurn$clientes_novos <- 0  
customerChurn$clientes_novos[customerChurn$customer_age < 6] <- 1  
customerChurn$clientes_risco <- 0  
customerChurn$clientes_risco[customerChurn$customer_age > 6 &  
  customerChurn$customer_age <= 18] <- 1
```

Criação do modelo

A funcionalidade glm foi utilizada para geração do modelo de regressão e este vinculado à variável glmCustomer. Removendo as variáveis não relevantes ao modelo, uma a uma, e reexecutando o modelo após a retirada de cada uma foi possível chegar a um modelo com variáveis relevantes. Foi feito alguns testes de remoção e adição de variáveis, respeitando o valor P, de forma que otimizasse a predição e chegamos ao seguinte modelo:

```
glm(data = customerChurn,  
  formula = churn ~ clientes_risco + chi_score_month_0 + support_cases_month_0 +  
    days_since_last_login_0_1 + support_cases_0_1 , family = binomial) ->  
  glmCustomer
```

Predição do modelo

Para testar o modelo, foi realizada a predição.

```
glmprobsCostumer <- predict(glmCustomer, type="response")
```

Para definir o ponto de corte foi utilizado o algoritmo de curvas ROC(Receiving Operating Characteristic), pelo método youden. A metodologia busca um maior resultado possível para a sensibilidade e especificidade.

```
rocobj <- roc(customerChurn$churn, glmprobsCostumer)  
coords(rocobj, x="best", input="threshold", best.method="youden")[1] -> pontoCorte  
pontoCorte
```

```
## threshold  
## 0.05147994
```

Com o ponto de corte definido, podemos preparar a predição.

```
nLinhasCostumer <- nrow(customerChurn)  
glmPredCostumer <- rep(0, nLinhasCostumer)  
glmPredCostumer[ glmprobsCostumer > pontoCorte ] <- 1
```


Verificação da predição

Aplicando a predição para os dados já possuídos, obtiveram-se 4213 True Negatives, 212 True Positives, de um total de 6347 registros. Os pagamentos forma previstos com aproximadamente 69,7% de sucesso (sensitividade) e 65,6% de especificidade . Como a probabilidade de churn é bem pequena, cerca de 5%, é importante que a especificidade esteja alta também, pois é possível conseguir uma alta taxa de sucesso de predição se o ponto de corte for acima do ideal, mas a especificidade é prejudicada e a predição não traria informações úteis.

```
dadosReais <- customerChurn$churn
table(glmpredCostumer, dadosReais) -> tabelaCostumerChurn
tabelaCostumerChurn

##              dadosReais
## glmpredCostumer    0     1
##                0 4213   111
##                1 1811   212

(as.vector(tabelaCostumerChurn)[1] + as.vector(tabelaCostumerChurn)[4]) / nLinhasCostumer

## [1] 0.6971798

quantChurns = nrow(customerChurn[customerChurn$churn==1,])
#Percentual de Churns presentes nos nossos dados:
quantChurns/nLinhasCostumer

## [1] 0.05089018

sensitivity(tabelaCostumerChurn)

## [1] 0.6993692

specificity(tabelaCostumerChurn)

## [1] 0.6563467
```

Conclusão

A partir do modelo foi possível gerar a probabilidade de cada cliente deixar o serviço. Os parâmetros utilizados foram:

- **cliente_risco**: Variável binária, onde 0 = falso e 1 = verdadeiro, que representa a presença do cliente no grupo de clientes acima de 6 meses de contrato até 18. Têm um peso relevante em aumento de chance de churn caso verdadeiro.
- **chi_score_mont_0**: Representa o Chi-score em dezembro. Quanto maior, menor a chance de churn.
- **support_cases_month_0**: Representa a quantidade de casos abertos no mês de dezembro. Quanto maior, menor a chance, o que pode indicar que clientes que utilizam mais o serviço, abrem mais chamados e cancelam menos.
- **days_since_last_login_0_1**: Representa a diferença entre os dias desde o último login enter o mês de dezembro e novembro. O valor negativo significa que em dezembro os dias foram menores que de dezembro. Então uma quantidade maior aumenta as chances de churn.
- **support_cases_0_1**: Representa a diferença dos suportes abertos entre dezembro e novembro. O valor negativo significa que em dezembro a abertura de suportes foi inferior a novembro. Então uma quantidade maior aumenta as chances de churn.

```
##              (Intercept)              clientes_risco
##                -2.943803456                1.117847560
```

```
##          chi_score_month_0      support_cases_month_0
##          -0.006757281          -0.152530512
## days_since_last_login_0_1      support_cases_0_1
##          0.012072215          0.119399356
```

Acrescentamos também se os clientes realmente deixaram o serviço, coluna “churn”, visto que estamos aplicando para os dados já conhecidos.

```
customerChurn$probs = glmprobsCostumer
head(customerChurn[order(-customerChurn$probs),], 100) -> clientesMaisProvaveis

select(clientesMaisProvaveis, id, probs, churn) %>%
  mutate(id = id) %>%
  kable(caption = "Lista clientes mais prováveis - Churn")
```

Table 1: Lista clientes mais prováveis - Churn

id	probs	churn
1672	0.2491660	1
227	0.1897374	1
257	0.1897374	1
272	0.1897374	0
278	0.1897374	0
279	0.1897374	0
317	0.1897374	1
346	0.1897374	0
363	0.1897374	1
371	0.1897374	1
413	0.1897374	0
416	0.1897374	0
423	0.1897374	0
427	0.1897374	0
440	0.1897374	0
444	0.1897374	0
475	0.1897374	0
488	0.1897374	0
523	0.1897374	1
543	0.1897374	1
548	0.1897374	1
551	0.1897374	0
583	0.1897374	0
604	0.1897374	0
622	0.1897374	0
625	0.1897374	0
645	0.1897374	0
678	0.1897374	0
689	0.1897374	0
761	0.1897374	0
775	0.1897374	0
787	0.1897374	1
788	0.1897374	0
798	0.1897374	0
891	0.1897374	1
896	0.1897374	1
926	0.1897374	0

id	probs	churn
945	0.1897374	1
947	0.1897374	1
948	0.1897374	1
979	0.1897374	1
991	0.1897374	0
994	0.1897374	0
1152	0.1897374	0
1214	0.1897374	1
1468	0.1897374	0
1563	0.1897374	1
1593	0.1897374	0
1617	0.1897374	0
1693	0.1897374	0
1706	0.1897374	0
1711	0.1897374	1
1760	0.1897374	1
1767	0.1897374	0
1774	0.1897374	0
1808	0.1897374	0
1809	0.1897374	0
2361	0.1897374	0
2501	0.1897374	0
2586	0.1897374	0
2985	0.1897374	0
3139	0.1897374	0
3152	0.1897374	0
3163	0.1897374	1
3177	0.1897374	0
3186	0.1897374	0
3235	0.1897374	1
3265	0.1897374	0
3269	0.1897374	0
3290	0.1897374	0
3312	0.1897374	1
3313	0.1897374	1
3320	0.1897374	0
3349	0.1897374	1
3363	0.1897374	1
3417	0.1897374	0
3418	0.1897374	0
3437	0.1897374	0
3449	0.1897374	0
3491	0.1897374	0
3526	0.1897374	0
3536	0.1897374	0
3542	0.1897374	0
3545	0.1897374	0
3548	0.1897374	0
3598	0.1897374	0
3600	0.1897374	0
3613	0.1897374	0
3623	0.1897374	0

id	probs	churn
3655	0.1897374	0
3671	0.1897374	0
3714	0.1897374	0
3723	0.1897374	0
3734	0.1897374	0
3767	0.1897374	0
3772	0.1897374	1
3780	0.1897374	0
3799	0.1897374	0
3824	0.1897374	0
3846	0.1897374	0