KNN em algoritmos de recomendação

Baseado em: “*Machine Learning for Hackers*”

Objetivo:

Utilizar o método KNN para gerar uma recomendação de um produto para um usuário baseado no seu histórico de consumo de outros produtos.

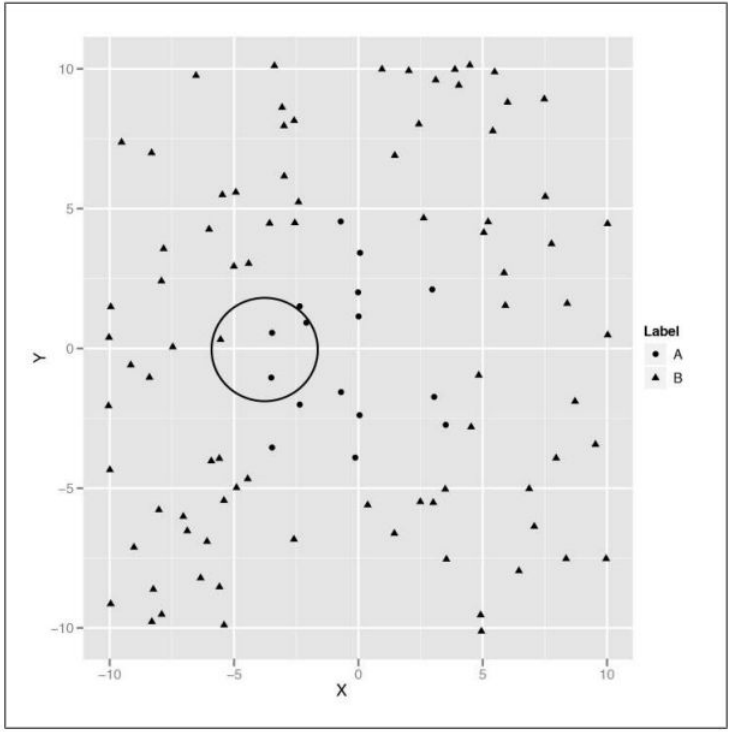
Considerações:

- Dataset usado: Filmes
- Bibliotecas do R Studio usadas:

R studio

Modelo de Classificação *K-Nearest-Neighbours*

Método de classificação que utiliza a distância entre observações como uma medida de semelhança entre elas, diferente dos estudos anteriores não usaremos KNN para classificar uma observação, mas sim para realizar um estudo dela baseado em seus vizinhos e de seus vizinhos baseada nela.



Machine learning for hackers - capítulo 10

Tratamento para a aplicação do KNN

	Nome.de.usuário	Filme	Nota
1	victorborin123@gmail.com	Batman..O.Cavaleiro.das.Trevas.	3
2	victorborin123@gmail.com	A.Origem	0
3	victorborin123@gmail.com	Pulp.Fiction	0
4	victorborin123@gmail.com	Senhor.dos.Anéis..A.Sociedade.do.Anel	0
5	victorborin123@gmail.com	O.Poderoso.Chefão	0
6	victorborin123@gmail.com	O.Senhor.dos.Anéis..O.Retorno.dos.Reis	0
7	victorborin123@gmail.com	Interestelar	0
8	victorborin123@gmail.com	O.Senhor.dos.Anéis..O.Retorno.dos.Reis.1	0
9	victorborin123@gmail.com	Se7en...Os.Sete.Crimes.Capitais	0
10	victorborin123@gmail.com	Django.Livre	0
11	victorborin123@gmail.com	Gladiador	0
12	victorborin123@gmail.com	Batman.Begins	0
13	victorborin123@gmail.com	Bastardos.Inglorios	5
14	victorborin123@gmail.com	O.Silêncio.dos.Inocentes	0
15	victorborin123@gmail.com	Os.Vingadores	5
16	victorborin123@gmail.com	O.Lobo.de.Wall.Street	5

R studio

Tratamento para a aplicação do KNN

```r
matriz <- cast(filmes, Nome.de.usuário ~ Filme, value = 'Nota')
row.names(matriz) <- matriz[, 1]
matriz <- matriz[, -1]
matriz[is.na(matriz)] <- 0

similarities <- cor(matriz)

distances <- -log((similarities / 2) + 0.5)
```

R studio

Tratamento para a aplicação do KNN

	À.Espera.de.um.Milagre	A.Origem	A.Outra.História.Americana	A.Viagem.de.Chihiro	Alien
123bruna321@gmail.com	0	3	0	0	5
ag735117@gmail.com	5	4	0	0	3
annajuliafigueiraa@gmail.com	4	0	0	0	4
bbeeaattrriizz96@gmail.com	0	0	0	0	0
beatrizluan1@gmail.com	0	0	0	0	0
brunodesouzamelo@gmail.com	5	5	0	0	0
caioabud1105@gmail.com	5	4	0	5	4
contatomiriamcsilva@gmail.com	5	3	0	0	0
diegogomes520@gmail.com	5	0	0	0	0
dilsibxyz@gmail.com	5	5	0	0	0
dinizyuridias@gmail.com	5	0	0	5	0
diogosouza1407@gmail.com	5	4	0	5	0
dogit51190@gmail.com	3	0	0	0	5
douglaslemos9975@gmail.com	0	4	0	5	0
edinilsonoelfo@gmail.com	0	0	0	5	0

R studio

Tratamento para a aplicação do KNN

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	1.000000000	0.354229198	0.15538338	-0.0435517640	0.170197364	-0.077290854	0.323264793	0.234292513	0.406236312	0.483960810
2	0.354229198	1.000000000	0.25595701	0.0411168652	0.339638610	0.254843981	0.310158294	0.474713740	0.430302542	0.475607104
3	0.155383380	0.255957005	1.00000000	0.0931236570	0.224849448	0.519101559	-0.058357938	0.318314375	0.054654628	0.183366144
4	-0.043551764	0.041116865	0.09312366	1.0000000000	0.218579405	-0.022845837	-0.133952022	0.112480346	0.020901346	0.119587279
5	0.170197364	0.339638610	0.22484945	0.2185794051	1.000000000	0.102933928	0.219062752	0.095597929	0.027668803	0.262970546
6	-0.077290854	0.254843981	0.51910156	-0.0228458365	0.102933928	1.000000000	0.009879403	0.326720439	-0.027191042	0.057666372
7	0.323264793	0.310158294	-0.05835794	-0.1339520223	0.219062752	0.009879403	1.000000000	0.114487428	0.352977094	0.295949115
8	0.234292513	0.474713740	0.31831437	0.1124803458	0.095597929	0.326720439	0.114487428	1.000000000	0.353198829	0.338482890
9	0.406236312	0.430302542	0.05465463	0.0209013462	0.027668803	-0.027191042	0.352977094	0.353198829	1.000000000	0.586109340
10	0.483960810	0.475607104	0.18336614	0.1195872791	0.262970546	0.057666372	0.295949115	0.338482890	0.586109340	1.000000000
11	0.027029315	0.241150726	0.45027820	0.2167987520	0.391638647	0.241559945	-0.050908488	0.255986679	0.201006521	0.302697238
12	0.235712901	0.076939875	-0.16723015	-0.0437619687	-0.027172673	-0.261504348	0.151630857	-0.123822177	0.254392795	0.235508974
13	0.292354101	0.144193149	-0.21598652	-0.0557875249	-0.010364977	-0.212318253	0.207871394	0.010474651	0.394515584	0.365673170
14	0.220475133	0.211030865	-0.10238692	-0.0995715719	0.158969733	-0.133452965	0.197986111	-0.056617387	0.347678500	0.356066006
15	0.196500654	0.396516200	0.06102615	-0.1006610058	-0.070359223	0.063462536	0.161118245	0.319558513	0.410827416	0.208364475

R studio

Tratamento para a aplicação do KNN

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	0.0000000	0.3899147	0.5487150	0.7376758	0.5359748	0.7735884	0.4130452	0.4826492	0.3522303	0.2984324
2	0.3899147	0.0000000	0.4652493	0.6528531	0.4007473	0.4661359	0.4229992	0.3046833	0.3352612	0.3040777
3	0.5487150	0.4652493	0.0000000	0.6041078	0.4903292	0.2750281	0.7532772	0.4167932	0.6399338	0.5247841
4	0.7376758	0.6528531	0.6041078	0.0000000	0.4954614	0.7162580	0.8369622	0.5865551	0.6724613	0.5801871
5	0.5359748	0.4007473	0.4903292	0.4954614	0.0000000	0.5951733	0.4950649	0.6018469	0.6658542	0.4596807
6	0.7735884	0.4661359	0.2750281	0.7162580	0.5951733	0.0000000	0.6833163	0.4104371	0.7207147	0.6370822
7	0.4130452	0.4229992	0.7532772	0.8369622	0.4950649	0.6833163	0.0000000	0.5847526	0.3908398	0.4339038
8	0.4826492	0.3046833	0.4167932	0.5865551	0.6018469	0.4104371	0.5847526	0.0000000	0.3906759	0.4016104
9	0.3522303	0.3352612	0.6399338	0.6724613	0.6658542	0.7207147	0.3908398	0.3906759	0.0000000	0.2318631
10	0.2984324	0.3040777	0.5247841	0.5801871	0.4596807	0.6370822	0.4339038	0.4016104	0.2318631	0.0000000
11	0.6664767	0.4771082	0.3213918	0.4969237	0.3626652	0.4767786	0.7453972	0.4652257	0.5099872	0.4287103
12	0.4814991	0.6190236	0.8761451	0.7378956	0.7206959	0.9962872	0.5519681	0.8253334	0.4664956	0.4816642
13	0.4366817	0.5584475	0.9364762	0.7505512	0.7035662	0.9318083	0.5042875	0.6827270	0.3606001	0.3814997
14	0.4939069	0.5016752	0.8011633	0.7980318	0.5456157	0.8363861	0.5125053	0.7514305	0.3947637	0.3885593
15	0.5137460	0.3591665	0.6339107	0.7992424	0.7661042	0.6316171	0.5437636	0.4158500	0.3489708	0.5038794

R studio

Tratamento para a aplicação do KNN: por que log?

Correlação entre uma variável e si mesma = 1

	V1	V2
1	1.000000000	0.354229198
2	0.354229198	1.000000000

Distância entre uma observação e si mesma = 0

	V1	V2
1	0.0000000	0.3899147
2	0.3899147	0.0000000

```
-log((similarities / 2) + 0.5)
```

R studio

Aplicação do KNN: método produto - produto

Consiste de calcular a semelhança entre filmes baseado nos usuários que os assistiram e as notas que deram. Tendo feito o tratamento anterior e dado um usuário para quem queremos recomendar filmes, para todo filme que ele não assistiu devemos analisar seus K vizinhos mais próximos e o filme cuja somatória das notas dos vizinhos for mais alta será recomendado.

Como desejamos recomendar mais de um filme, selecionamos os 10 filmes com maior somatória em ordem.

Aplicação do KNN: método produto - produto

```r
knn <- function(i, distances, k){
  return(order(distances[i, ])[2:(k + 1)])
}

filme.prob1 <- function(usu, matriz, distances, k){
  filmes <- which(matriz[usu,]==0)
  prox <- c()
  for(filme in filmes){
  prox <- c(prox,sum(matriz[usu,knn(filme, distances, k)]))
  }
  return(filmes[order(prox,decreasing = T)])
}
```

R studio

Aplicação do KNN: método produto - produto

```
> lista <- filme.prob1(1,matriz,distances,10)
> colnames(matriz)[lista[1:10]]
 [1] "À.Espera.de.um.Milagre"                    "Guardiões.da.Galáxia"
 [3] "Perdido.em.Marte"                          "Ratatouille"
 [5] "Guerra.nas.Estrelas...O.Império.Contra.Ataca" "Mad.Max"
 [7] "Prenda.Me.se.For.Capaz"                    "Pulp.Fiction"
 [9] "Senhor.dos.Anéis..A.Sociedade.do.Anel"     "Indiana.Jones.e.a.Última.Cruzada"
```

R studio

Aplicação do KNN: método usuário - usuário

Consiste de calcular a semelhança entre usuários baseado nos filmes que assistiram e as notas que deram. Será realizado um tratamento diferente do anterior.

Dado um usuário para quem queremos recomendar filmes, para seus K vizinhos mais próximos contabilizamos quais filmes o vizinho viu e o usuário não, em seguida para todo filme contabilizado somamos suas notas para os K vizinhos e o filme com maior soma será recomendado.

Como desejamos recomendar mais de um filme, selecionamos os 10 filmes com maior somatória em ordem.

Aplicação do KNN: método usuário - usuário

```r
matriz2 <- cast(filmes, Filme ~ Nome.de.usuário, value = 'Nota')
row.names(matriz2) <- matriz2[, 1]
matriz2 <- matriz2[, -1]
matriz2[is.na(matriz2)] <- 0

similarities2 <- cor(matriz2)

distances2 <- -log((similarities2 / 2) + 0.5)

lista <- filme.prob2(1,matriz2,distances2,10)
rownames(matriz2)[lista[1:10]]
```

R studio

Aplicação do KNN: método usuário - usuário

```
filme.prob2 <- function(usu, matriz2, distances2, k){
  neighbors <- knn(usu, distances2, k)
  candidatos <- c()
  for(i in neighbors){
    temp <- matriz2[matriz2[,usu]==0 & matriz2[,i]!=0,i]
    temp2 <- which(matriz2[,usu]==0 & matriz2[,i]!=0)
    candidatos <- c(candidatos,temp2[order(temp, decreasing = T)])
  }
  filmes <- as.integer(names(table(candidatos)))
  notas <-c()
  resul <-c()
  for(j in filmes){
  notas <- c(notas,sum(matriz2[j,neighbors]))
  }
  return(filmes[order(notas,decreasing = T)])
}
```

R studio

Aplicação do KNN: método usuário - usuário

```
> lista <- filme.prob2(1,matriz2,distances2,10)
> rownames(matriz2)[lista[1:10]]
 [1] "Guardiões.da.Galáxia"                  "À.Espera.de.um.Milagre"
 [3] "Ratatouille"                           "Senhor.dos.Anéis..A.Sociedade.do.Anel"
 [5] "Guerra.nas.Estrelas...O.Império.Contra.Ataca" "Perdido.em.Marte"
 [7] "Ilha.do.Medo"                          "Indiana.Jones.e.a.Última.Cruzada"
 [9] "Mad.Max"                               "O.Pianista"
```

R studio

Comparação entre os métodos

```
> lista <- filme.prob1(1,matriz,distances,10)
> colnames(matriz)[lista[1:10]]
 [1] "À.Espera.de.um.Milagre"                    "Guardiões.da.Galáxia"
 [3] "Perdido.em.Marte"                          "Ratatouille"
 [5] "Guerra.nas.Estrelas...O.Império.Contra.Ataca" "Mad.Max"
 [7] "Prenda.Me.se.For.Capaz"                    "Pulp.Fiction"
 [9] "Senhor.dos.Anéis..A.Sociedade.do.Anel"     "Indiana.Jones.e.a.Última.Cruzada"
> lista <- filme.prob2(1,matriz2,distances2,10)
> rownames(matriz2)[lista[1:10]]
 [1] "Guardiões.da.Galáxia"                      "À.Espera.de.um.Milagre"
 [3] "Ratatouille"                               "Senhor.dos.Anéis..A.Sociedade.do.Anel"
 [5] "Guerra.nas.Estrelas...O.Império.Contra.Ataca" "Perdido.em.Marte"
 [7] "Ilha.do.Medo"                              "Indiana.Jones.e.a.Última.Cruzada"
 [9] "Mad.Max"                                   "O.Pianista"
```

R studio

Comparação entre os métodos: qual escolher?

Essa decisão será tomada baseado no conjunto de dados a ser estudado, especificamente a proporção entre usuários e produtos ((lembrando que é obrigatória a existência de usuários para aplicação de ambos métodos)).

A lógica é que com muitos produtos e poucos usuários, é computacionalmente mais barato analisar os usuários, e vice versa. Porém também é verdade que muitas observações produzem um resultado mais acurado, então é importante balancear o custo computacional e a acurácia.

Testando a eficiência do método usuário - usuário

Para este método, um forma de testar sua eficiência é remover um filme que tanto um usuário quanto seus vizinhos gostaram ((nota =5)) e gerar as recomendações para este usuário, se o filme removido estiver incluso é um indicativo de que o modelo funciona bem.

Esta verificação também é válido para produto - produto

```
#removi harry potter e a pedra filosofal para verificar se aparecia nas recomendações
matriz2[28,1] <- 0
> lista <- filme.prob2(1,matriz2,distances2,10)
> rownames(matriz2)[lista[1:10]]
 [1] "Guardiões.da.Galáxia"                    "À.Espera.de.um.Milagre"
 [3] "Ratatouille"                             "Harry.Potter.e.a.Pedra.Filosofal."
```

R studio

Testando a eficiência do método usuário - usuário

Uma outra forma envolve gerar a matriz de distâncias e na hora da criação da recomendação do usuário, substituir todas suas notas de filme por zero e verificar se seus filmes preferidos ((nota=5)) aparecem nas recomendações.

```
        preferidos <- rownames(matriz2[matriz2[,1]==5,])

        matriz2[,1] <- 0

        lista <- filme.prob2(1,matriz2,distances2,10)
        recs <-rownames(matriz2)[lista[1:length(preferidos)]]

> print(paste0(sum(preferidos %in% recs),"/",length(preferidos)))
[1] "23/35"
```

R studio

Conclusão:

- É possível aplicar KNN em um contexto de recomendação baseada na análise de semelhança entre objetos
- Há dois métodos de realizar essa aplicação e a escolha entre eles se baseia na proporção entre usuários e produtos
- É obrigatório que o conjunto de dados contenha informações sobre usuários
- É difícil verificar com certeza absoluta a eficiência desses métodos, porém é possível tentar

FIM.