Projeto Data Mining 2

Ângelo Gomes up201703990

Simão Cardoso up201604595

Sónia Rocha up201704679

05/06/2021

Introduction and Exploratory Data Analysis

O objetivo deste projeto é comparar diferentes estratégias de recomendação da data set users_brands.csv originado pela farfetch.

Inicialmente analisamos e tratamos a informação presente no data set que é constituído por 846490 observação, cada com 7 variáveis distintas. Durante este processo descobrimos que nenhuma observação tinha uma variável com valor NA, através de `any(is.na(users_brands))`.

Através do summary da dataset users_brands, verificamos que as variáveis platform,sequence_id e brand_id deveriam ser do tipo factor, e por isso foram mudadas para tal, assim como a variavel brand_id na data set brand_features.

```
##    user_id            brand_id          platform         country
##  Length:846490      Min.   :  102532   Min.   :15766995   Length:84649
0
##  Class :character   1st Qu.:27968276   1st Qu.:35279833   Class :chara
cter
##  Mode  :character   Median :51215738   Median :39857033   Mode  :chara
cter
##                     Mean   :52775914   Mean   :47381157
##                     3rd Qu.:80735391   3rd Qu.:59713904
##                     Max.   :99855154   Max.   :96462699
##  user_segment        sequence_id     perc_sale
##  Length:846490      Min.   :  1.0   Length:846490
##  Class :character   1st Qu.: 67.0   Class :character
##  Mode  :character   Median :129.0   Mode  :character
##                     Mean   :119.2
##                     3rd Qu.:170.0
##                     Max.   :231.0

##     brand_id          features
##  Min.   :  102532   Length:1815
##  1st Qu.:27223510   Class :character
##  Median :50399755   Mode  :character
##  Mean   :51084932
```

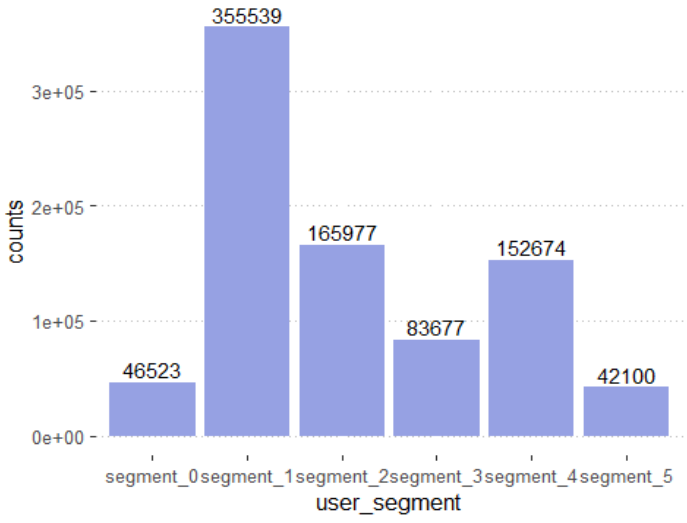
```
##  3rd Qu.:76268310
##  Max.   :99855154

## [1] FALSE

## [1] FALSE
```

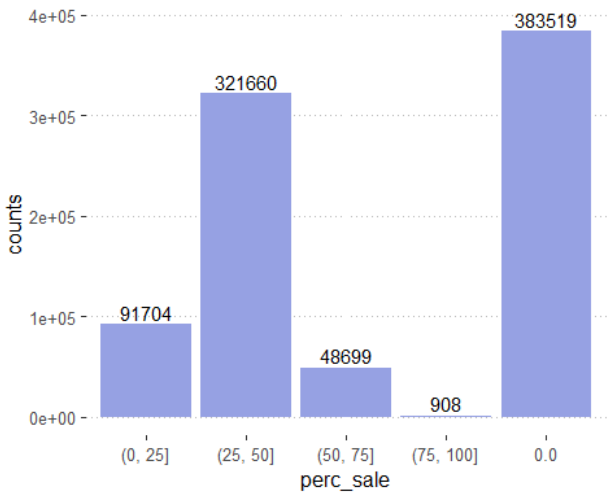
É também pertinente verificar quantos users distintos, brands,plataformas, países e sequências estão representadas no data set.

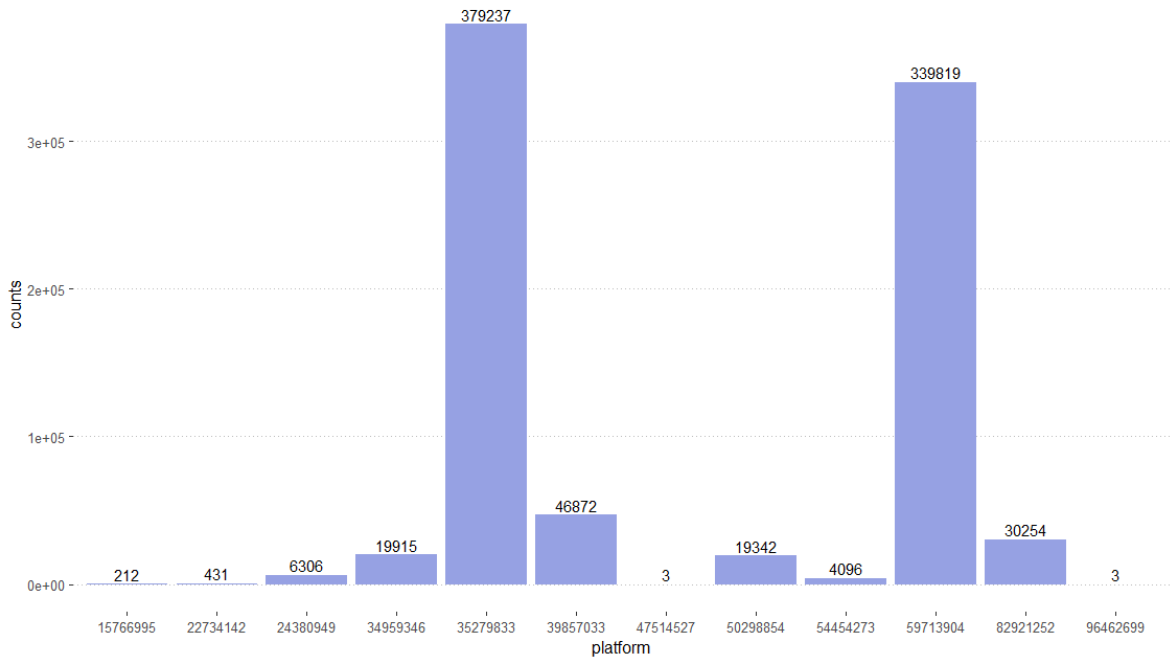
```
## [1] 329213

## [1] 1771

## [1] 175

## [1] 231

## [1] 12
```

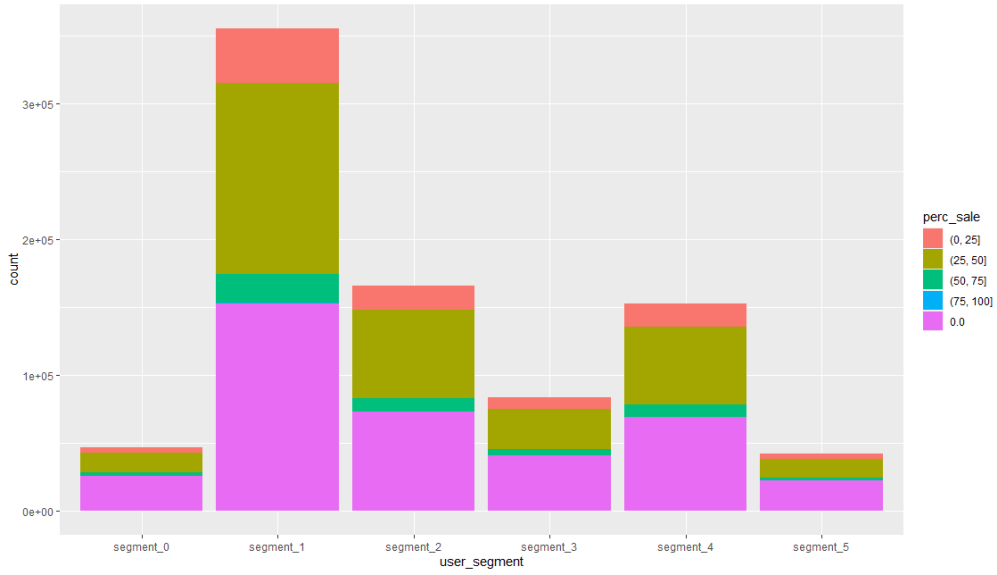
Assim, concluímos que, existem 329213 id's de users , 1771 marcas, 175 países, 231 sequências e 12 plataformas.

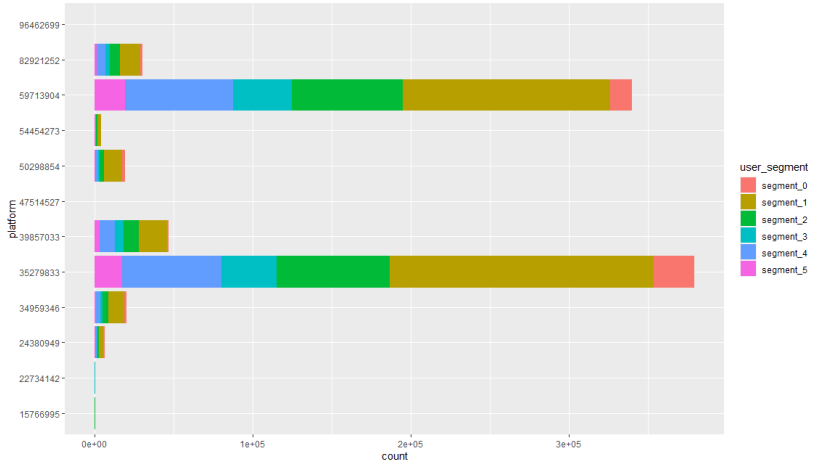
Por último visualizamos os gráficos que se seguem de forma a visualizar melhor alguns parâmetros desta data set.

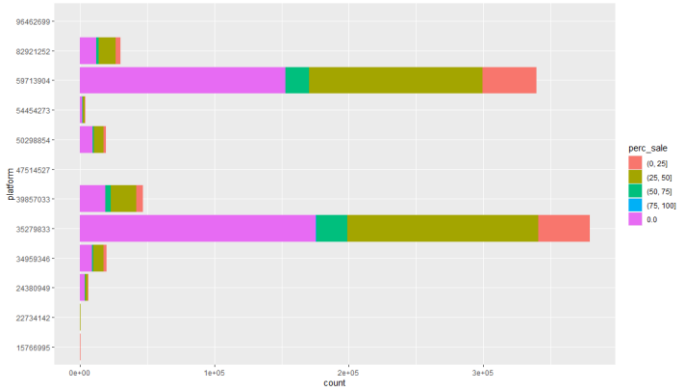





379237
339819
46872
30254
19915
19342
6306
4096
212
431
3
3
3e+05
2e+05
1e+05
0e+00
counts
15766995
22734142
24380949
34959346
35279833
39857033
47514527
50298854
54454273
59713904
82921252
96462699
platform


3e+05
2e+05
1e+05
0e+00
count
segment_0
segment_1
segment_2
segment_3
segment_4
segment_5
user_segment
perc_sale
(0, 25]
(25, 50]
(50, 75]
(75, 100]
0.0

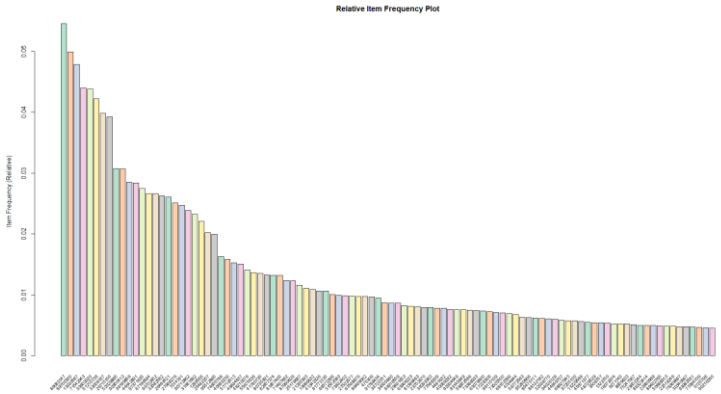


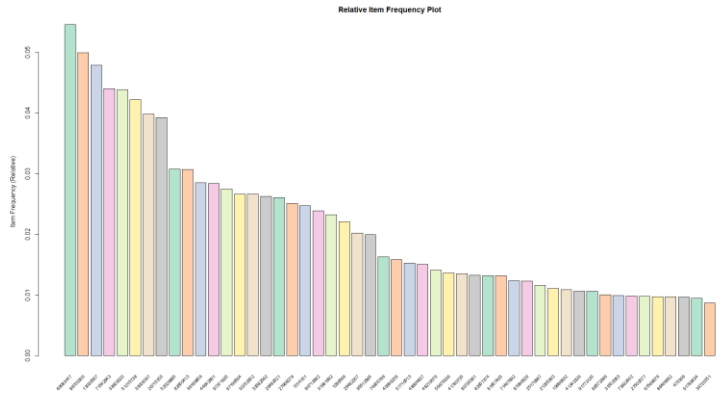


Recommender Systems

Nesta fase de desenvolvimento do projeto, foi importante diminuir a data set para user_id e brand_id com maior interação.

Assim, antes de escolher esses tais user_id, fizemos uma lista com os brand_id e user_id, à qual transformamos numa variável do tipo transaction, que utilizando a função itemFrequencyPlot() com top(100) das brand_id, conseguimos visualizar melhor quais as brands com maior influencia, assim verificamos que talvez fosse mais eficiente estudar as top 50 brands pois apresentam valores de interação relativamente altos (gráficos abaixo é top(100) e top(50) respetivamente).





De seguida, foi então criada uma data set com essas tais restrições, contendo 436788 observações e 2 variáveis (brand_id, user_id). Foi criada com esta data, uma variável brm do tipo "binaryRatingMatrix", começar assim a task 2 de Recommender Systems. Além disso, foi criada outra variável utilizada para training data, com cerca de aproximadamente 1/4 do tamanho da data brm, para testar os diferentes métodos e familiarizar aos outputs de cada método primeiramente vamos utilizar o user 500 com as diferentes top N recomendações indicadas (1,2,6).

Começando com o método Popular estas foram as predições:

```
## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "89083147"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "89083147" "89700300"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "89083147" "89700300" "13030587" "33603020" "51215738" "33008187"
```

Relativamente ao método Association Rules com suporte = 0.003 e nível de confiança = 0.05 (ficando com 69 regras):

```
## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "33008187"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "33008187" "89700300"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "33008187" "89700300" "13030587" "93702652" "7014181"  "33603020"
```

Relativamente ao método User-Based e Item-Based Collaborative Filtering utilizamos o método cosine para a similaridade entre os items (brands), e com n (neighborhood) 100 (pois mais baixo que este valor ou não fazia predições ou fazia um número inferior ao requerido).

Começando com o método User-Based CF estas foram as predições:

```
## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "7014181"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "7014181"  "25176687"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "7014181"  "25176687" "25482207" "26715155" "27503577" "27968276"
```

Para o método Item-Based CF estas foram as predições:

```
## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "93702652"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "93702652" "7014181"

## $`00a13c1219f23a2b0d584f18e26c574391b3c1e6f2c7e4c7f96b4ce9ea7f1faa`
## [1] "93702652" "7014181"  "33008187" "13030587" "89700300" "44642891"
```

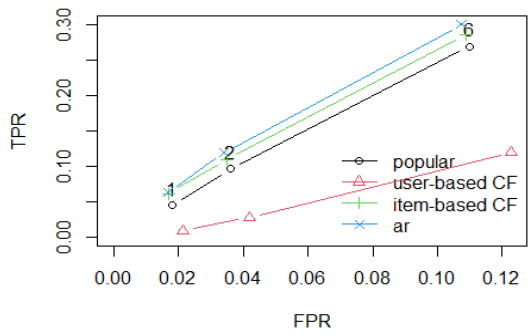
Após fazer as testagens dos diferentes métodos com a offline data, iremos começar por fazer a lista de métodos, que se irá encontrar na variável methods, com as características utilizadas anteriormente:

```
methods <- list(
  "popular" = list(name="POPULAR", param = NULL),
  "user-based CF" = list(name="UBCF", param = list(method="cosine",nn=100
)),
  "item-based CF" = list(name="IBCF", param = list(method="cosine",k=100)
),
  "ar" = list(name="AR", param = list(supp=0.003, conf=0.05))
)
```

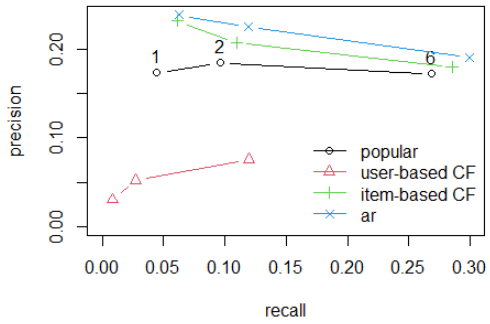
Assim, para seguidamente utilizar o método evaluationScheme, inicialmente procuramos utilizar o valor de given que nos fizesse obter os melhores resultados, chegando à conclusão de que obtivemos os melhores resultados com given = 1 pois foram observados os average results para cada um destes valores, e comparados.

Além disso, fizemos algumas observações mais pormenorizadas de cada método com da data known e train para n= 1, fazendo o seguinte para todos, " model <- Recommender(getData(e3,“train”),...), predict(model,getData(e3,“known”),..".

Dado estas informações este foi o resultado do plot de ROC onde o TPR ficou com no máximo 30%, onde o melhor método foi o Association Rules, de seguida o Popular, o Item-based e o pior foi User-based CF:



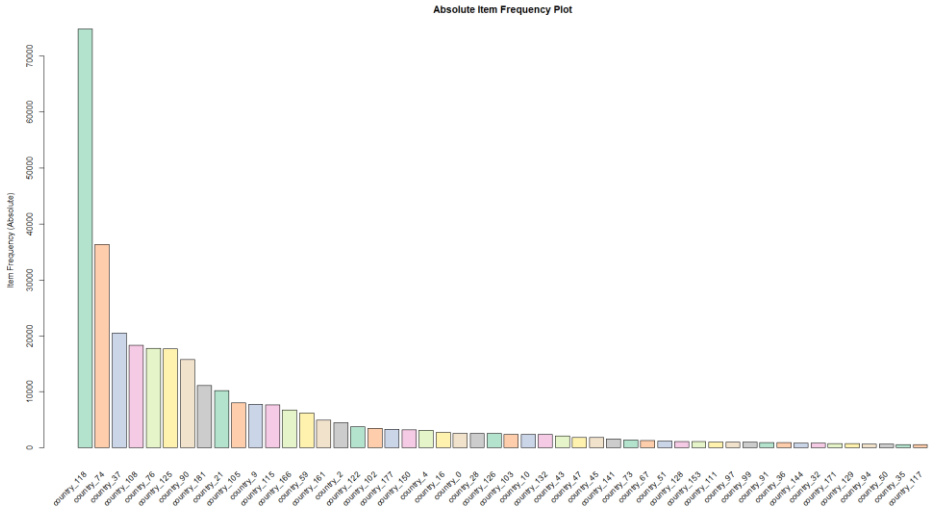
E este é o resultado do plot das curvas de precision/recall, onde os métodos Associaton Rules e Item-based CF com um valores de recall mais baixos quase alcançaram valores de 25% de precision mas ambos descem esta precision ao longo que o recall aumenta para valores próximos de 20% , o método mais estável no entanto é o Popular que ao longo dos valores de recall mantêm-se por volta dos 18% e por fim uma vez mais o método User-based CF é o que tem pior precisão com cerca de 12% no geral aproximadamente.



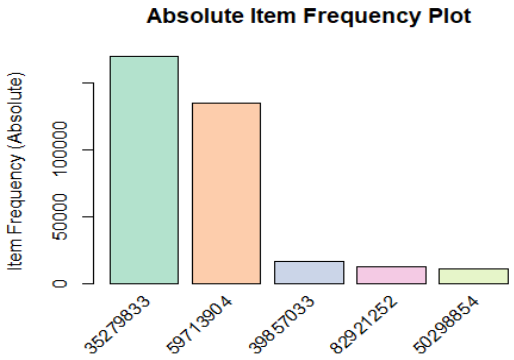
New Contextual Approach and Conclusions

Por fim, de forma a melhorar estas percentagens obtidas, decidimos fazer uma abordagem contextual diferente, utilizando algumas das restantes variáveis na dataset. Esta abordagem consiste em limitar a database relativamente ao país, sales e plataformas com mais influência, e só depois fazer as predições de brand_id sobre user_id, além disso, limitamos ainda mais o número de brand para em vez de top50 para top20 para ficar com valores ainda mais influentes. Para cada uma destas variáveis fizemos o mesmo que fizemos na primeira task, colocamos numa lista o user_id e a variável em questão, passando para uma variável do tipo transaction, e por fim fazer um itemFrequencyPlot para cada uma destas relações (user_id <-> country / user_id <-> perc_sales / user_id <-> plataformas). De seguida iremos mostrar os Frequency plot para cada uma destas relações, começando por user_id e country, de seguida user_id e perc_sales e por fim user_id e platfrom.

Após a visualização deste plot, verificamos que no top 50 country tem vários países com menos influencia e com uma frequência absoluta relativamente baixa, assim iremos utilizar na data que iremos fazer as predições, o top 10 dos países.

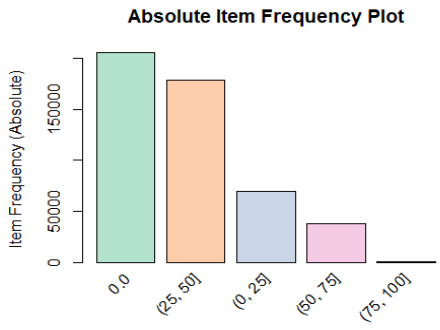


Este é o plot para top5 de perc_sales:



Como se pode verificar, as primeiras duas colunas de 0.0 e (25,50] têm muita mais aferência, e por isso decidimos utilizar na data o top2 de perc_sale.

Por fim este é o gráfico de platforms:

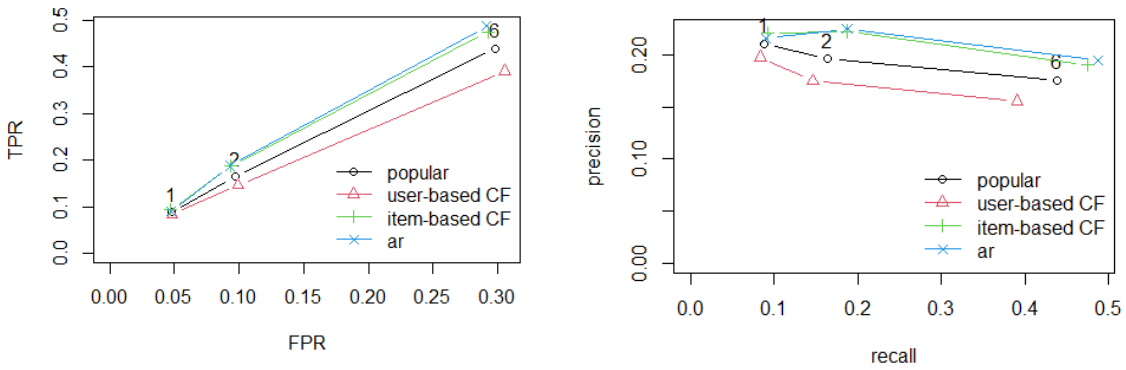


Assim, concluímos que as 2 primeiras plataformas são as mais influentes e fazem uma grande diferencia a níveis de frequência em relação às restantes, iremos utilizar assim o top2 de plataformas.

Após fazer a data reduction de uma variável que continha user_id, brand_id, country, perc_sales e plataformas com a seguinte ordem primeiro limitar esta data para top(20) de brand_id -> top(10) country -> top(2) perc_sale -> top(2) platfrorm.

Fizemos então uma variável contendo os métodos com as seguintes características, em User-based CF e Item-based CF , o método cosine com nn/k = 100, e em Association Rules, support = 0.003 e confidence level de 0.06. De seguida, antes de fazer o ROC plot e precision/recall plot, foi importante encontrar qual o valor de given ideal na função de Evaluation Scheme com train = 0.80, e por isso foi testado os resultados com given=3,2,1 e comparados os average result. O que obteve melhores resultados foi given=1.

Estes foram os resultados dos plots ROC e precision/recall respetivamente:



Assim, concluímos que houve melhoria relativamente aos plots feitos sem estas limitações, todos os métodos no plot de ROC melhoraram as suas percentagens, atingindo todos valores acima de 40% com a exceção de User-based CF que ficou perto mas não ultrapassou, o User-based CF apesar de continuar o pior, melhorou as suas percentagens de aproximadamente 12% para 40%.

No precision/recall também houve melhoria, nos métodos Association Rules e Item-based CF que inicialmente desciam muito os valores de precisão ao longo que o recall aumentava, agora essa descida foi atenuada e por isso mantendo valores de no geral sempre acima ou igual a 20%. No método Popular agora com uma precisão que chega a 20% e mantém-se em valores próximos a este, anteriormente não chegava a estes valores. Por fim, User-based CF obteve a maior melhoria, agora com percentagens a chegar aos 20% uma melhoria de aproximadamente no 12% (no máximo) relativamente ao calculado antes.

Conlcuímos assim, que o método que se adequa mais ao que foi feito é o Association rules, manteve tanto na task2 como na task3, valores de resultados sempre melhores aos dos restantes métodos. Assim concluímos este trabalho, o objetivo no futuro era conseguir utilizar a dataset de brand_features para uma nova abordagem contextual e obter melhores resultados.