Clusteres Semi Supervisionados de Clientes

Andrew

20 de novembro de 2017

/ Introdução

Nesse report, detalho as escolhas e o processo de criação dos clusteres, tendo em mente a criação de clusteres que agrupam clientes em diferentes escalas de valor ao negócio, seja ela ele demonstrado (investimentos realizados) ou valor em potencial (perfil, profissão).

/ Input de Dados

Vamos começar carregando alguns pacotes e carregando o dataset. Por conveniência, o dataset foi transformado em csv usando o excel.

```
library(randomForest)
library(caTools)
library(biganalytics)
library(corrplot)
library(tm)
library(lubridate)
library(readr)
library(dplyr)
library(ggplot2)
library(plotly)
setwd("D:/Unimed")

# [1] Ler Dados
raw.data = read_csv2("DataSet.csv") %>% as.data.frame()
glimpse(raw.data)
```

```
## Observations: 4,972
## Variables: 11
## $ ID              <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14,...
## $ GEO_REFERENCIA  <int> 780, 35, 54, 35, 883, 78, 131, 97, 351, 51, 12...
## $ DATA_NASCIMENTO <chr> "15/08/1992 00:00", "24/02/1990 00:00", "17/07...
## $ PROFISSAO       <chr> "ANALISTA DE SISTEMAS", "SERVIDOR P\xdaBLICO E...
## $ GENERO          <chr> "M", "F", "M", "M", "M", "M", "M", "M", "M", "...
## $ ESTADO_CIVIL    <chr> "SOLTEIRO(A)", "SOLTEIRO(A)", "SOLTEIRO(A)", "...
## $ VALOR_01        <dbl> 342.86, 942.86, 2000.00, 857.14, 8615.39, 571....
## $ VALOR_02        <dbl> 342.86, 0.00, 0.00, 285.71, 0.00, 410.31, 0.00...
## $ VALOR_03        <dbl> 428.57, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0....
## $ VALOR_04        <dbl> 28.57, 0.00, 2857.14, 1428.57, 47471.79, 57.14...
## $ PERFIL          <chr> "A", "A", "A", "A", "A", "B", "A", "A", "A", "...
```

Antes de verificar as estatísticas descritivas do dataset, optei por verificar sua integridade, pois NAs mascarariam seus reais valores.

```r
# [2] Verificar integridade dos dados
colSums(is.na(raw.data)) %>% as.data.frame()
```

```
##                 .
## ID              0
## GEO_REFERENCIA  0
## DATA_NASCIMENTO 0
## PROFISSAO       0
## GENERO          0
## ESTADO_CIVIL    0
## VALOR_01        0
## VALOR_02        0
## VALOR_03        0
## VALOR_04        0
## PERFIL          0
```

```r
summary(raw.data)
```

```
##        ID       GEO_REFERENCIA  DATA_NASCIMENTO     PROFISSAO
##  Min.   :   1   Min.   : 10.0   Length:4972        Length:4972
##  1st Qu.:1244   1st Qu.: 70.0   Class :character   Class :character
##  Median :2486   Median :224.0   Mode  :character   Mode  :character
##  Mean   :2486   Mean   :336.8
##  3rd Qu.:3729   3rd Qu.:607.0
##  Max.   :4972   Max.   :999.0
##     GENERO          ESTADO_CIVIL          VALOR_01
##  Length:4972        Length:4972        Min.   :     0.0
##  Class :character   Class :character   1st Qu.:   628.6
##  Mode  :character   Mode  :character   Median :  1371.4
##                                        Mean   :  2022.7
##                                        3rd Qu.:  2571.4
##                                        Max.   :400000.0
##     VALOR_02          VALOR_03          VALOR_04         PERFIL
##  Min.   :      0   Min.   :      0   Min.   :     0   Length:4972
##  1st Qu.:      0   1st Qu.:      0   1st Qu.:     0   Class :character
##  Median :      0   Median :      0   Median :     0   Mode  :character
##  Mean   :  18638   Mean   :   4246   Mean   :  5041
##  3rd Qu.:   6006   3rd Qu.:      0   3rd Qu.:  1429
##  Max.   :2857143   Max.   :1428571   Max.   :685714
```

Uma das exclusões feitas logo após o sumário, é a variável de geolocalização. A razão por trás de não considera-la é:

- Desconhecimento de seu significado (É impossível representar posições geográficas com apenas 1 coordenada, caso sejam ids de localidades, não há garantia de que sejam contínuos)
- Desnecessário para o objetivo estipulado

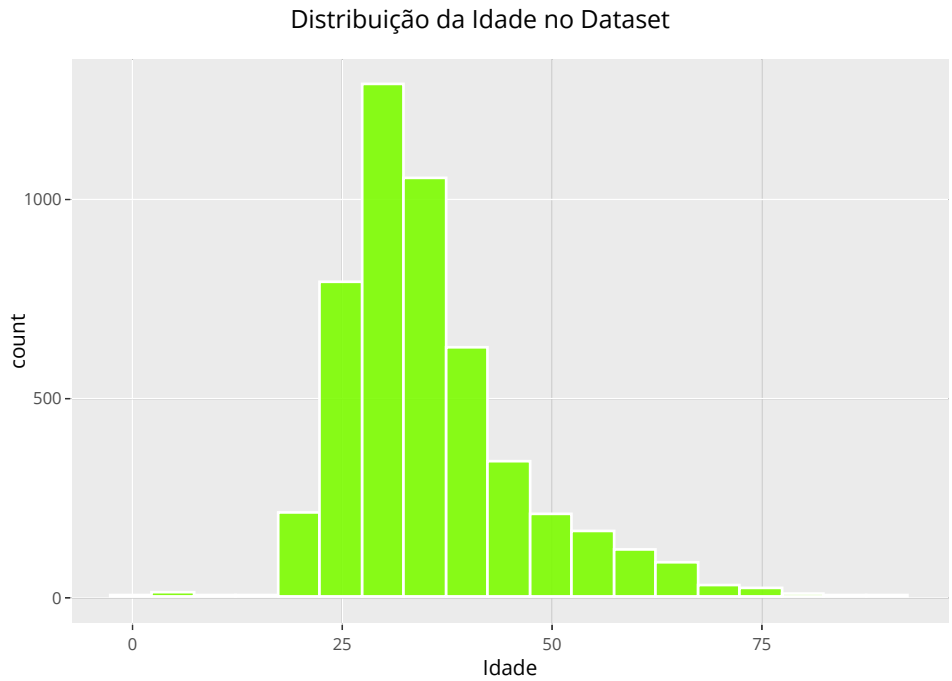
O dataset parece estar compelto, para prosseguir, criaremos algumas features para facilitar sua visualização e clusterização à frente.

/ Análise Exploratória

Para iniciar a análise exploratória, criamos algumas features:

```
# [3] Feature Engineering
raw.data$DATA_NASCIMENTO = as.Date(raw.data$DATA_NASCIMENTO, format="%d/%m/%Y")
raw.data$Idade = 2017 - year(raw.data$DATA_NASCIMENTO)
raw.data$Valor_total = raw.data$VALOR_01 + raw.data$VALOR_02 + raw.data$VALOR_03 + raw
```

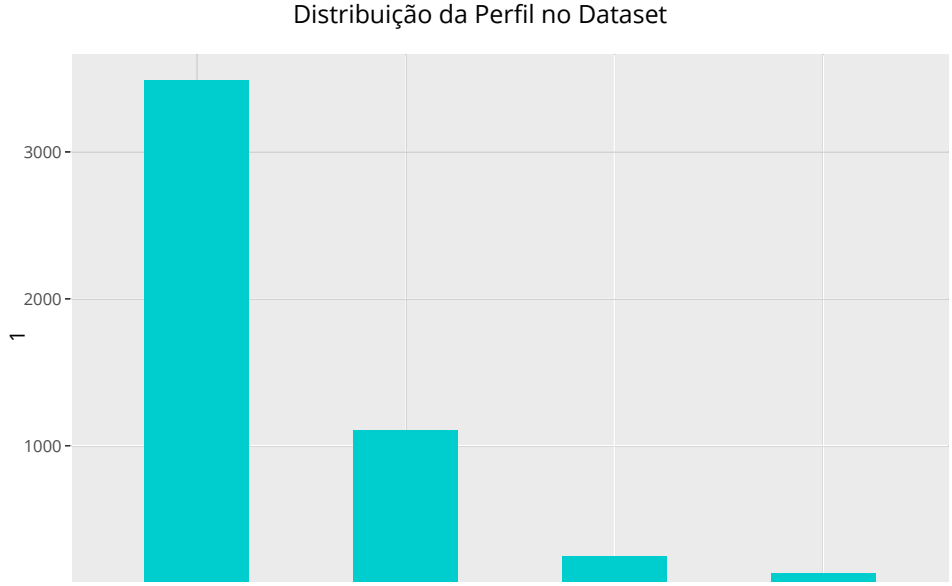
Em seguida, podemos verificar a distribuição da Idade dos clientes.

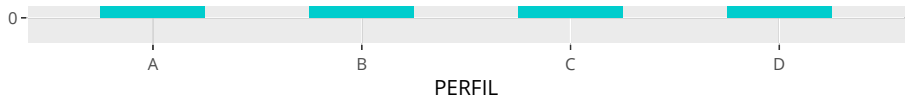


Pela distribuição de idade, optei por excluir a variável `PERFIL`, uma vez que ela representa a renda familiar (critério IBGE) e não a renda individual, o que é especialmente enviesador para um dataset com metade das pessoas jovens (< 30 anos).

Podemos verificar que a idade se concentra entre 20 e 50 anos, em seguida veremos a distribuição dos perfis:

```
## We recommend that you use the dev version of ggplot2 with `ggplotly()`
## Install it with: `devtools::install_github('hadley/ggplot2')`
```



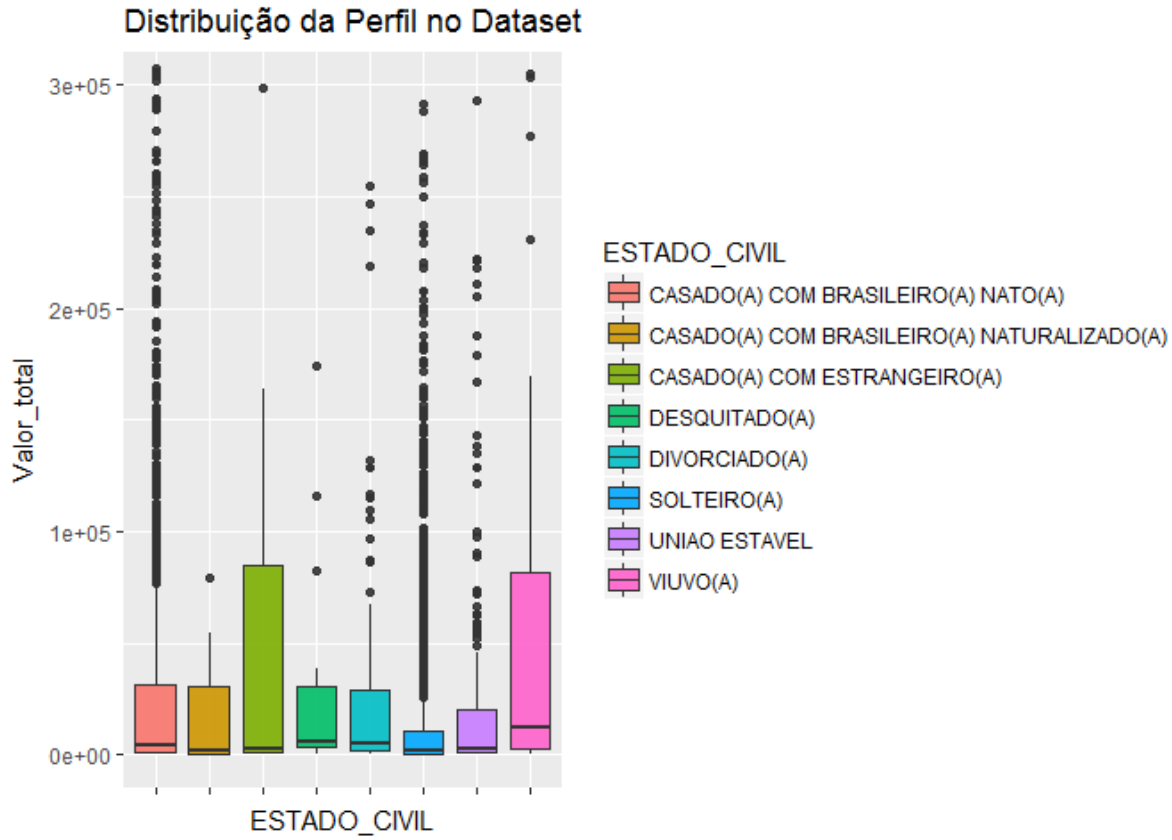


Há uma maior quantidade de clientes A e B, como esperado pelo fato de se tratar de investimentos.

```
# Valor_Total
summary(raw.data$Valor_total)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##       0.0     914.3    2714.3   29948.1   18083.4 2874285.7
```

Podemos verificar a presença de outliers no valor dos investimentos, é importante retira-los da clusterização, uma vez que os métodos de clusterização se apoiam sobre métricas de distância. Estes serão tratados como **excessões**.



Pela variação nos quartis, a variável estado civil aparenta ter algum impacto sobre o valor investido, e portanto será incluida na clusterização.

/ Objetivo

A questão chave que podemos definir nesse ponto é que resultado esperamos da clusterização. Na minha experiência, utilizei k means para facilitar entendimento / descobrir padrões e para quebrar grandes problemas em casos menores e então prever de maneira mais assertiva.

Pelo descrito na plataforma, creio que o mais adequado seja o primeiro caso, sendo o valor do cluster proporcional ao entendimento que ele traz.

Me colocando nos pés da corretora do mercado financeiro, creio que o maior ganho seria entender **quais são os clientes mais valiosos**, tendo em mente que o valor seria consequência do quanto foi investido mas também do potencial de investimento de cada cliente.

Na prática isso significaria olhar para as variáveis que delimitam o **perfil** do cliente (capital potencial) e os **investimento** (capital investido) realizados por ele.

Por uma questão de incerteza, optei por atribuir um peso maior ao capital já investido, pois ele possui menor margem para erro quando comparado com a estimativa do potencial de investimento.

Ilustrando os últimos parágrafos em forma matemática temos:

$$Valor_{cliente} = 0.4 * Capital_{potencial} + 0.6 * Capital_{investido}$$

Essa equação é apenas para ilustrar, não será seguida a risca. Uma vez que a questão de valor é subjetiva não há uma única métrica capaz de validar os clusters, o que coloca em foco o *business sense* na seleção dos clusters.

A variável `PERFIL` será reservada para auxiliar a validar os clusters, junto com variáveis relativas ao valor e ocupação dos clientes.

Objetivo da Clusterização: Agrupar clientes de acordo com seu valor demonstrado e valor potencial.

Uma vez que os *clusters* serão gerados com variáveis pré selecionadas, ponderadas e terão uma label para comparação, mesmo que subjetiva, trata-se de *semi supervised clustering*.

/ Pré Processamento

Para satisfazer o objetivo, busquei criar features em torno do perfil e do comportamento de investimento de cada cliente.

Iniciaremos removendo outliers e preparando a feature Profissão

```
# [5] Filtrar outliers e remover acentos da variável profissão (para viabilizar text m
clean.data = filter(raw.data, Valor_total <= 100000)
remover.acentos = function(x) iconv(x, to = "ASCII//TRANSLIT")
clean.data$Profissao = remover.acentos(clean.data$PROFISSAO)
```

Um aspecto chave de investidores é sua exposição à risco, que na prática se reflete na concentração / dispersão de ações.

Para tentar capturar esse efeito, utilizei o desvio padrão das variáveis `VALOR_01` até `VALOR_04`. Como o incremento desse desvio está associado positivamente a grandes valores de dinheiro investidos em apenas um produto financeiro, utilizarei o nome concentração de investimentos.

```
# Criar variável para representar a Concetração / Dispersão de Investimentos
clean.data$Concentracao_investimento = 0
for (i in 1:nrow(clean.data)){
  clean.data$Concentracao_investimento[i] = sd(c(clean.data$VALOR_01[i], clean.data$VA
                                                 clean.data$VALOR_03[i], clean.data$VA
}
```

A variável `PROFISSAO` é interessante para a predição de renda e logo de investimento, no entanto ela sofre por alta cardinalidade.

Na realidade, a variável `PROFISSAO` guarda dentro de si duas variáveis, o nível hierárquico (ex: analista, gerente) e a área de atuação (ex: sistemas, processos).

Realizar one hot encoding dessa variável tornaria os dados extremamente esparsos, uma vez que temos 79 classes únicas, além de não representar as similaridades entre níveis hierárquicos.

Ao invés de criar dummies, optei por aplicar text mining para gerar uma DTM e usa-la como feature na clusterização, gerando menos classes e capturando a similaridade de nível hierarquico.

```
# Criar Tf Idf com base na profissão para uso como feature
corpus = VCorpus(VectorSource(clean.data$Profissao))

# Remover Pontuação, Stopwords, Números
limpar.dados = function(corpus){
  corpus = tm_map(corpus, stripWhitespace)
  corpus = tm_map(corpus, removePunctuation)
  corpus = tm_map(corpus, removeWords, c(stopwords("pt-br")))
  corpus = tm_map(corpus, removeNumbers)
  corpus = tm_map(corpus, content_transformer(tolower))
}

bag.words = limpar.dados(corpus)

# Transformar em DTM e remover termos esparsos (presentes em menos de 1% das observaçõ
bag.words = DocumentTermMatrix(bag.words, control = list(weighting = weightTfIdf))
bag.words = removeSparseTerms(bag.words, sparse = 0.99)
bag.words = as.data.frame(as.matrix(bag.words))

# Unir resultados ao dataframe
clean.data = bind_cols(clean.data,bag.words)
```

Das várias maneiras de se criar uma bag of words, optei por um `tf idf`, pois ela contempla a raridade de termos e não apenas sua contagem bruta, assim representando melhor a profissão 'médicos' por exemplo.

Para tornar o processo de limpeza, filtragem de termos esparsos ainda mais rápida, utilizei a escrutura `corpus` do pacote `tm`, a idéia central de eficiência desse formato é não representar valores nulos, que são caracteristicos de dados esparsos como texto.

Para incluir o estado civil e o gênero na clusterização temos de realizar seu one hot encoding. Nessa caso há apenas 8 classes únicas.

```
# Criar dummies do Estado Civil
ES_dummy = as.data.frame(model.matrix(~clean.data$ESTADO_CIVIL))
ES_dummy[,1] = NULL
colnames(ES_dummy) = c("ES_Casado_naturalizado", "ES_Casado_estrangeiro", "ES_Desquita
                       "ES_Divorciado", "ES_Solteiro", "ES_Uniao_estavel", "ES_Viuvo")

# Unir resultados ao dataframe
clean.data = bind_cols(clean.data,ES_dummy)

# Recodificar variável genero
clean.data$Genero_F = ifelse(clean.data$GENERO == "F",1,0)
```

/ Clustering

Antes de iniciar o algoritmo de clusterização, é importante normalizar os dados, isto é, subtrair a média e dividir pelo desvio padrão de cada coluna. Na prática isso significa atribuir igual peso para

todas as features.

Isso se faz necessário, na medida em que não há diferenciação de variáveis pelo algoritmo - a distância de 10 anos de idade seria interpretada como a mesma de 10 reais a mais ou a menos investidos.

Além da normalização, temos que reponderar uma vez que todas as variáveis tenham igual peso. A razão para isso é que temos 46 features de perfil e apenas 6 de investimento. Sendo todas tratadas com igual peso, a clusterização não seria feita em torno do objetivo que estipulamos.

Para manter, aproximadamente, o respeito à fórmula, iremos multiplicar as variáveis de investimento em um fator de 60x no total, de modo a equivaler à aproximadamente 60% da distância

```
set.seed(16)
# Normalizar Dados
scaled.data = as.data.frame(scale(clean.data[,c(7:10,12,13,15:61)]))

# Ponderar variáveis de investimento
scaled.data$Valor_total = 30 * scaled.data$Valor_total
scaled.data$VALOR_01 = 5 * scaled.data$VALOR_01
scaled.data$VALOR_02 = 5 * scaled.data$VALOR_02
scaled.data$VALOR_03 = 5 * scaled.data$VALOR_03
scaled.data$VALOR_04 = 5 * scaled.data$VALOR_04
scaled.data$Concentracao_investimento = 10 * scaled.data$Concentracao_investimento

# Executar K-means
k.clust = bigkmeans(as.matrix(scaled.data), centers = 4, iter.max=100, dist="euclid")
clean.data$cluster = as.factor(k.clust$cluster)
```

Na escolha do algoritmo, considerei K means e Hierarchical Clustering (usando método ward.d2: foco na minização da variância intracluster).

DBScan e OPTICs não foram considerados, uma vez que sua grande falha está em dados esparsos, com baixa densidade. A introdução de one hot encoding e da bag of words tornaram o dataset bem mais esparso do que de início.

A escolha de K means foi basicamente pelo critério **performance**, uma vez que Hclust escala de maneira inferior a K means, pelo fato de sempre calcular todas as distâncias individuais de cada ponto, enquanto K means aproxima centróides.

Sobre a função `bigkmeans` em específico, trata-se de uma implementação com mais de 2x a velocidade do algoritmo base do R.

Pela "regra de 7" de George A. Miller, o número de *clusters* deve estar entre 3 a 7 unidades para ser inteligível a pessoas. Ao final das iterações, o melhor valor obtido foi 4.

/ Avaliação do Cluster

O Primeiro passo para avaliar uma clusterização é verificar a quantidade de clusteres gerados.

```
table(clean.data$cluster)
```

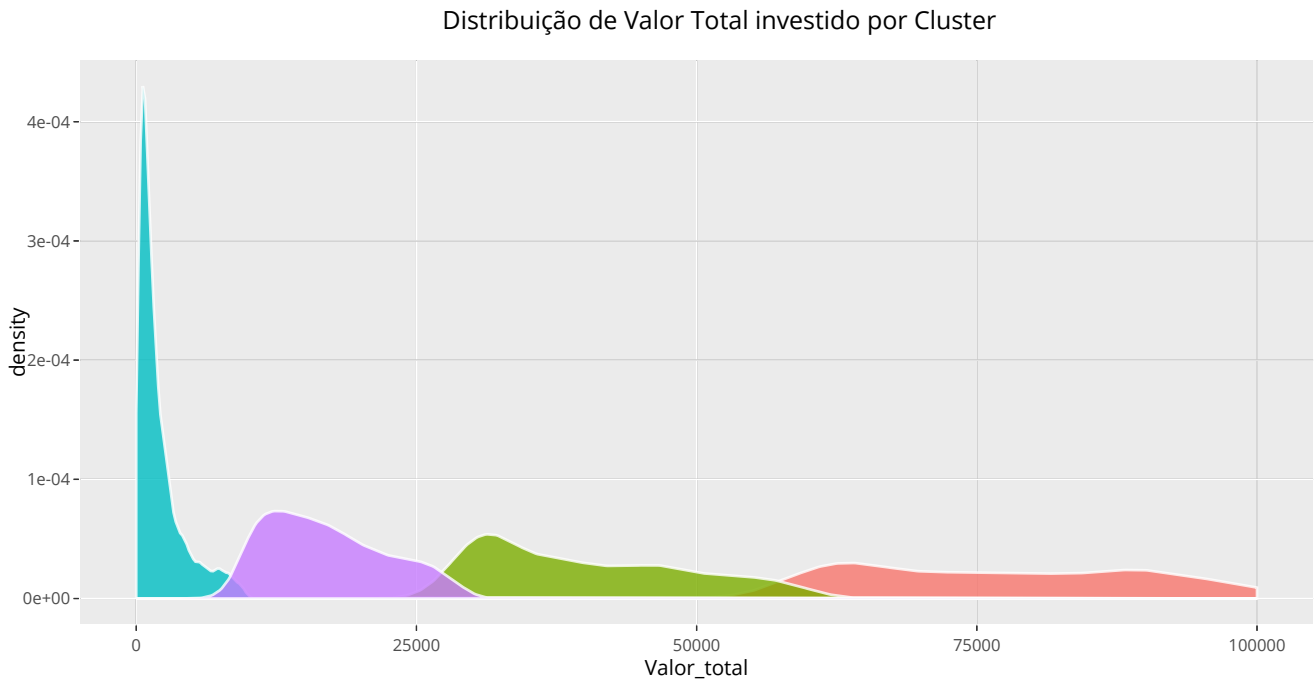
```
##
##    1    2    3    4
```

```
##  252  400 3321  658
```

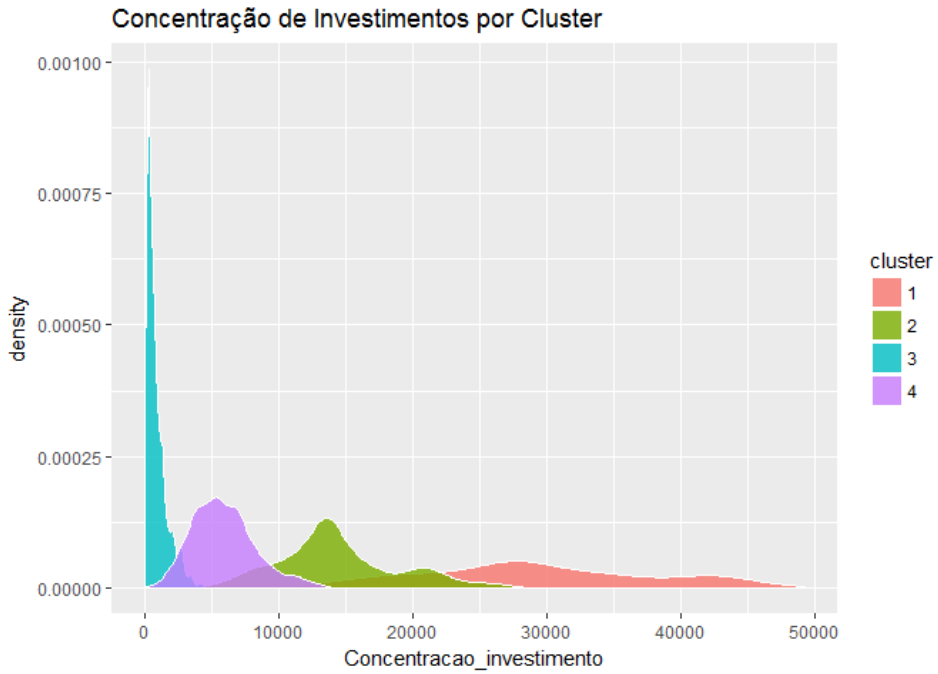
De modo geral, os clusteres tem quantidades de dados bem distribuidas. A escolha de 4 clusteres em específico ocorreu por conta da má distribuição entre 5 a 7 clusteres, na qual clusters traço apareciam com menos de 60 observações cada.

Para validar nossa escolha de cluster, vamos observar o quanto nossos clusters cumprem o objetivo estipulado:

```
## We recommend that you use the dev version of ggplot2 with `ggplotly()`
## Install it with: `devtools::install_github('hadley/ggplot2')`
```

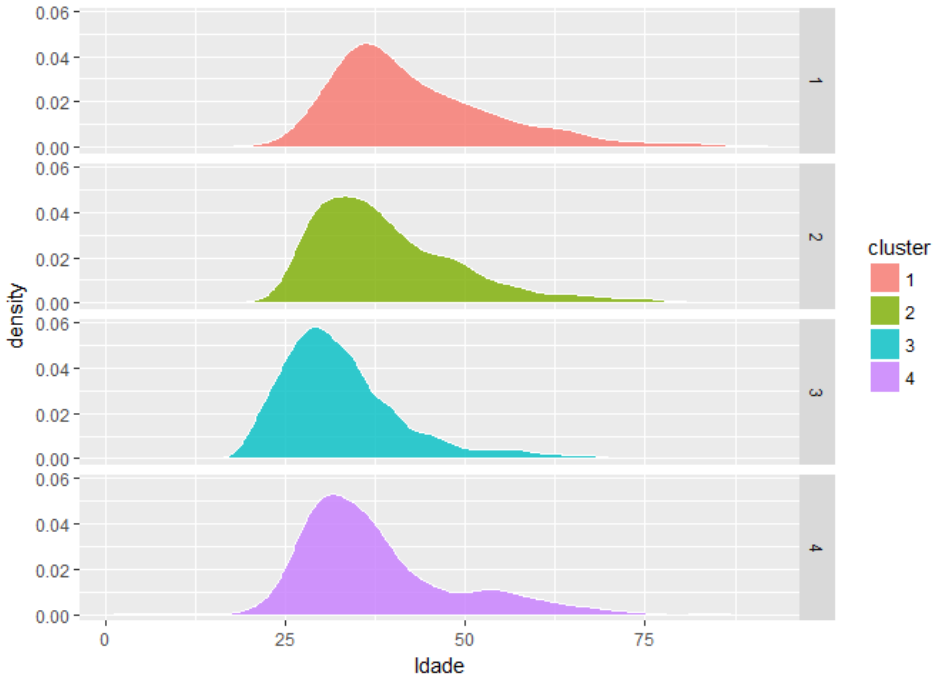


O agrupamento separou de maneira ótima os valores investidos.

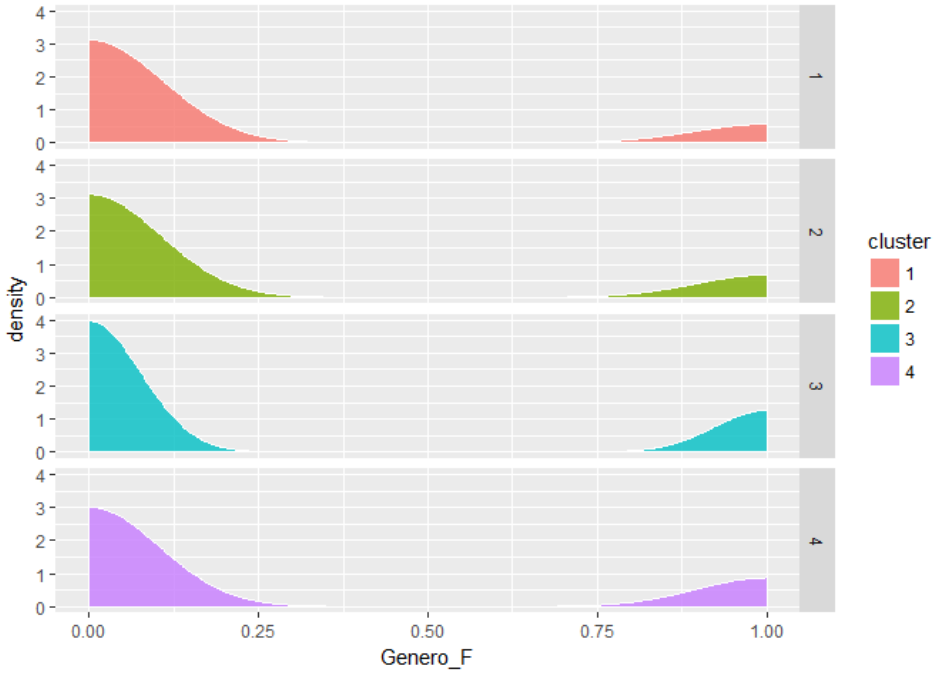


Ao observar o cluster de maior valor, pode-se perceber que nele há uma grande variação da concentração de investimentos, que levanta a hipótese de dois grupos dentro dele: 'investidores profissionais' (que dispersam seus investimentos) e 'investidores com alto capital'.

No que tange as variáveis de perfil, a Idade e Gênero são pouco determinantes, assim como o esperado.

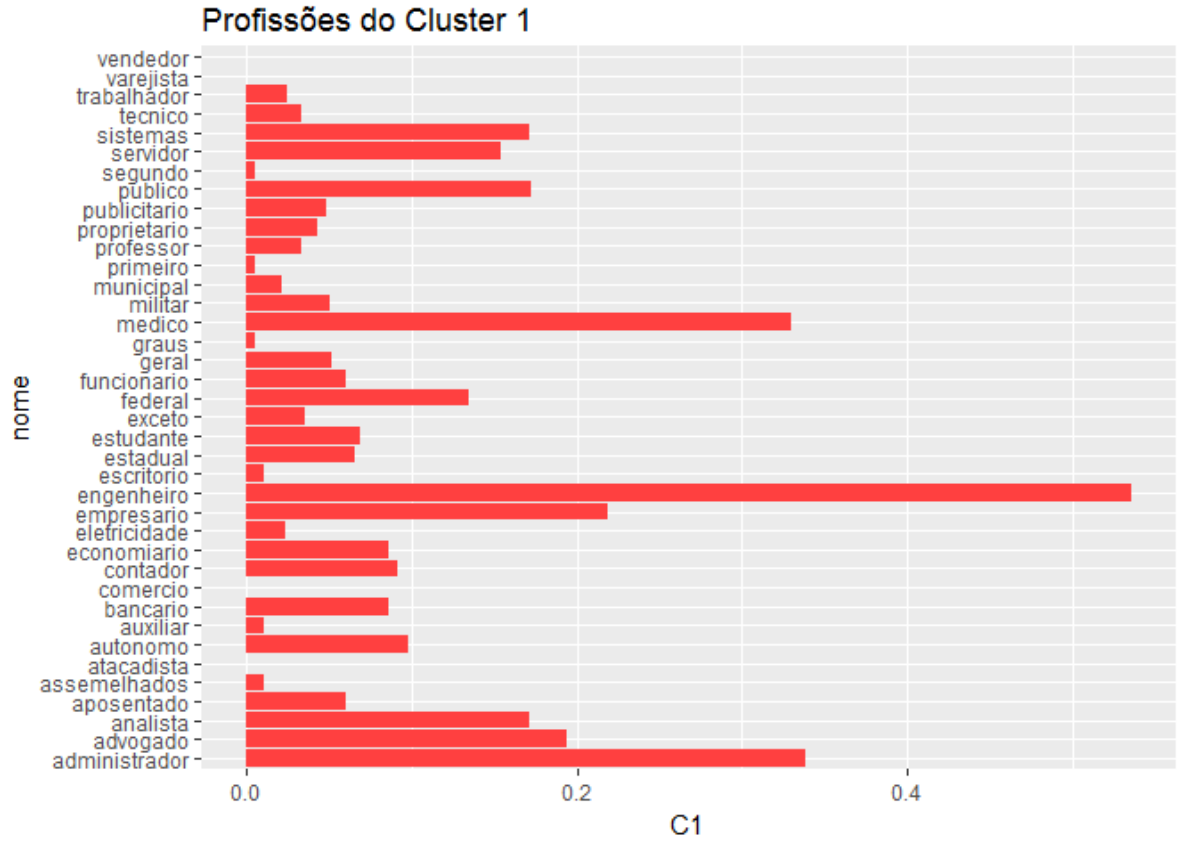


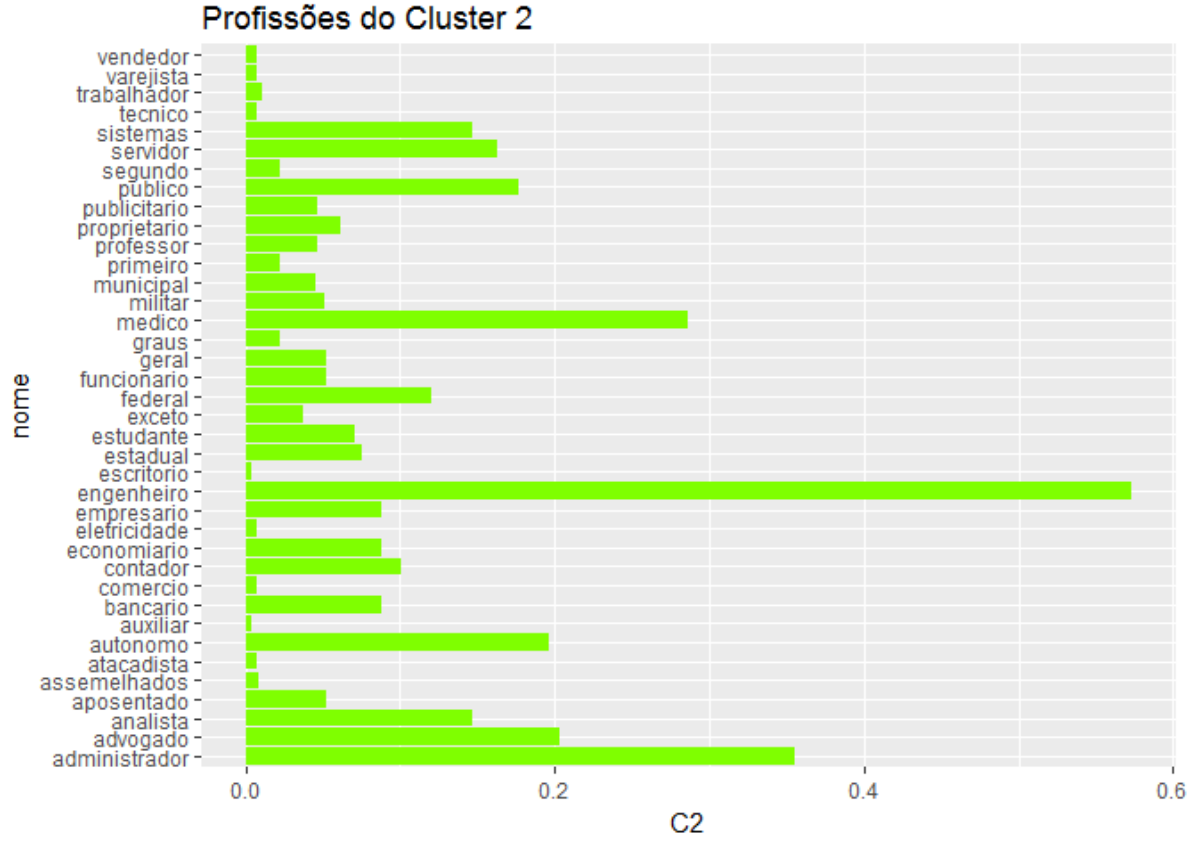
A idade apresenta uma leve tendência a maiores investimentos, vista no cluster 1 (maior valor) e nos cluster 3 e 4 com idades menores.

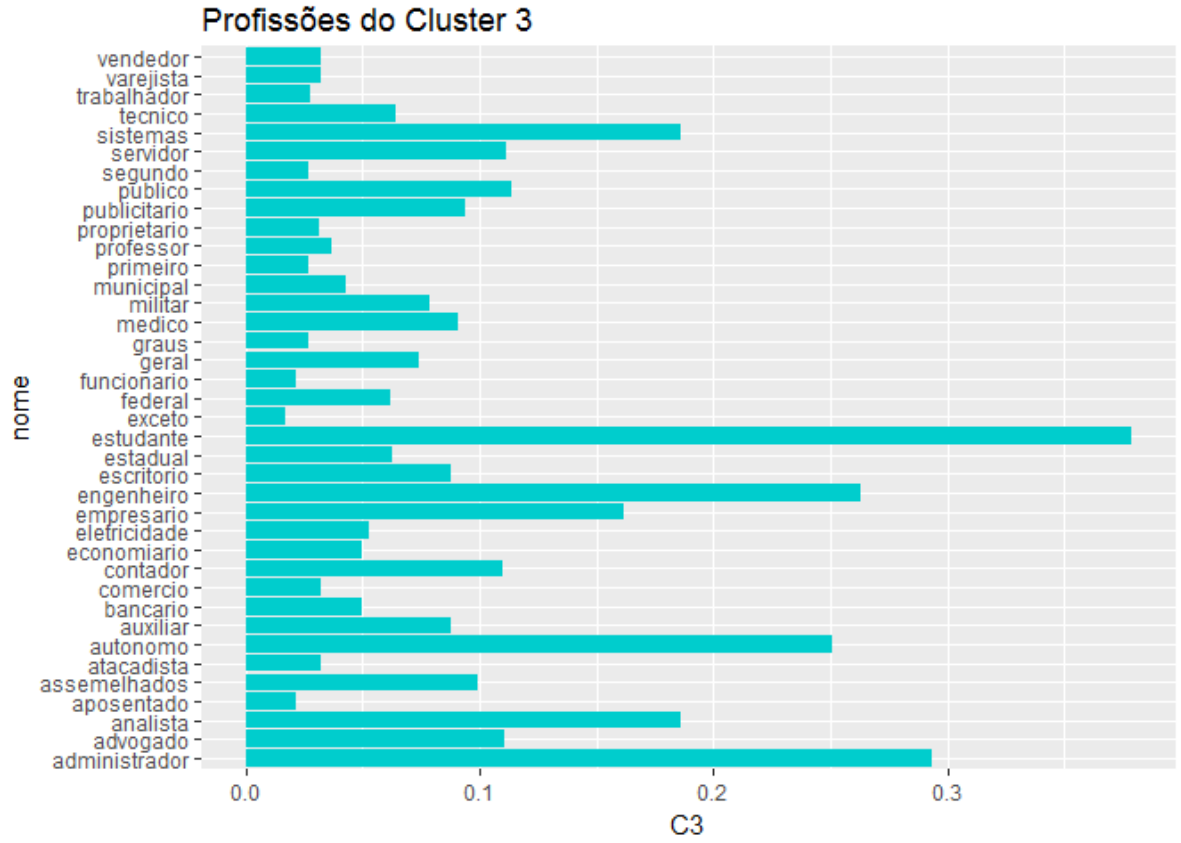


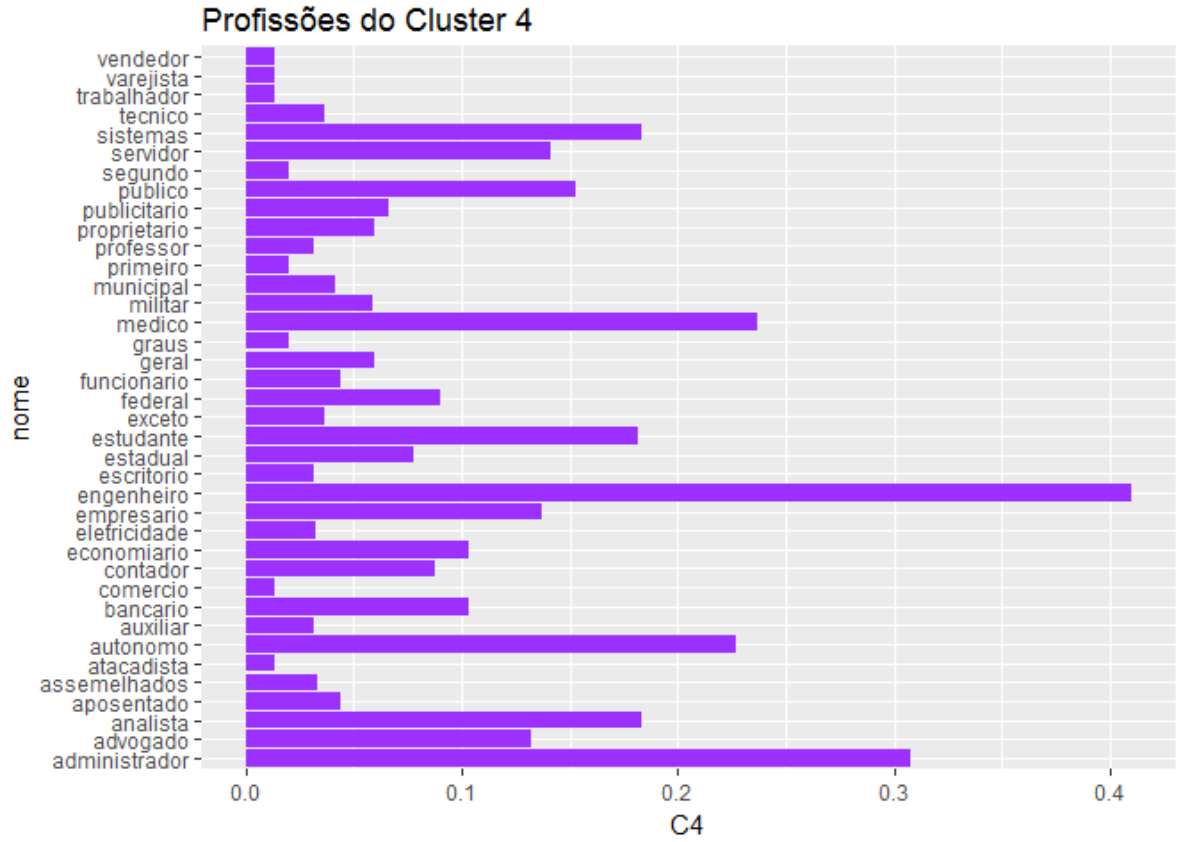
O gênero não tem demonstrado grande efeito sobre os clusters.

Ao verificar a composição de profissões em cada cluster, é possível verificar que o cluster de maior valor possui mais empresários, enquanto o cluster de menor valor possui mais estudantes e cargos de nível hierarquico menor (vendedor, auxiliar, tecnico.


Profissões do Cluster 1
vendedor
varejista
trabalhador
tecnico
sistemas
servidor
segundo
publico
publicitario
proprietario
professor
primeiro
municipal
militar
medico
graus
geral
funcionario
federal
exceto
estudante
estadual
escritorio
engenheiro
empresario
eletricidade
economiario
contador
comercio
bancario
auxiliar
autonomo
atacadista
assemelhados
aposentado
analista
advogado
administrador
nome
0.0
0.2
0.4
C1


Profissões do Cluster 2
vendedor
varejista
trabalhador
tecnico
sistemas
servidor
segundo
publico
publicitario
proprietario
professor
primeiro
municipal
militar
medico
graus
geral
funcionario
federal
exceto
estudante
estadual
escritorio
engenheiro
empresario
eletricidade
economiario
contador
comercio
bancario
auxiliar
autonomo
atacadista
assemelhados
aposentado
analista
advogado
administrador
nome
0.0
0.2
0.4
0.6
C2





Por fim, os clusteres estão consistentes com o objetivo de separar clientes de acordo com o valor que demonstram e que apresentam em potencial.

Essa afirmação é suportada pela ótima separação por valor total investido e pelas profissões observadas na composição dos clusteres.

Os clusteres formados seriam interessantes para a àrea de CRM, marketing, entre outros. Para criar ainda mais valor ao negócio, seria interessante criar um modelo preditivo que com base nas mesmas

variáveis prevesse o cluster do cliente, dessa forma, seria possível identificar e tratar de maneira personalizada os clientes de diferentes escalas de valor ao negócio.

Há mais do que eu gostaria de fazer, mas com 6 horas não posso tentar tudo que gostaria, qualquer sugestão de melhoria é bem vinda e obrigado por visualizar esse trabalho.