Previsão de resultados escolares de alunos de ensino médio

1st Vinicius Costa Secundino
Departamento de Teleinformática
Universidade Federal do Ceará
Fortaleza, Brasil
viniciussec@alu.ufc.br

2nd Caio dos Santos Nascimento
Departamento de Teleinformática
Universidade Federal do Ceará
Fortaleza, Brasil
caio.santos@alu.ufc.br

3rd Paulo Douglas Melo da Silva
Departamento de Teleinformática
Universidade Federal do Ceará
Fortaleza, Brasil
douglascomp@alu.ufc.br

***Resumo*—O uso de dados socioeconômicos de estudantes pode ser útil na determinação de sucesso deles. Essa determinação pode ser feita através dos métodos de classificação disponíveis na literatura que aborda esse tipo de predição. O presente trabalho busca realizar um estudo acerca das relações entre o comportamento e características de alunos de ensino médio de escolas de Portugal e suas relações com a aprovação escolar a partir dados coletados pelo professor universitário Paulo Cortez [1]. Os estudantes com notas finais maiores ou igual a 10, de um máximo de 20, foram considerados aprovados. Para realizar a classificação foram aplicados métodos lineares (regressão logística e linear discriminant analysis) e não-lineares (quadratic discriminant analysis, k-nearest neighbours e support vector machine) obtendo resultados entre 80% à 90% de acurácia.**

***Index Terms*—Classificação, Regressão Logística, Linear Discriminant Analysis(LDA), K-Nearest Neighbours(KNN), Quadratic Discriminant Analysis(QDA), Student, Performance.**

I. Introdução

No setor de educação atual, a retenção de diversas informações socioeconômicas sobre os estudantes permite analisar o impacto desses aspectos no aprendizado, que é medido geralmente pelas notas das avaliações. Diante disso, podem ser estudadas as relações de alguns fatores com a forma que o aluno absorve o conteúdo e, consequentemente, garante (ou não) a aprovação. Mas, antes de irmos direto para a explicação dos métodos, vamos investigar alguns conceitos relacionados ao objetivo deste trabalho.

A. Entendendo Classificação

Os métodos de classificação são, por sua essência, técnicas de realizar predições em variáveis qualitativas (rótulos ou categorias), diferentemente de regressões, onde são realizadas predições em variáveis quantitativas (números decimais). A ferramenta utilizada para aplicar esses métodos foi a liguagem de programação R [2], que já apresenta modelos de classificação implementados. As etapas, enumeradas a seguir, desse processo são parecidas com as de regressão:

1) O conjunto de dados será dividido em conjunto de treino e conjunto de teste;
2) Cada modelo de classificação será aplicado a esse conjunto de treino;
3) Com o modelo já treinado, aplicamos no conjunto de teste para obtermos as predições de cada entrada;
4) Por fim, calculamos algumas métricas, como acurácia e plotamos a curva ROC.

Ainda, dependendo da distribuição dos dados utilizados, podemos utilizar métodos de classificação lineares ou não lineares. Para exemplificar, algumas aplicações úteis da classificação são:

- Determinar doenças em uma emergência com base nos sintomas
- Determinar se uma transação é fraudulenta com base em dados de transações anteriores
- Determinar time de futebol apoiado com base nas cores da camisa

B. Objetivo

Nosso objetivo principal é analisar os dados de vários alunos, fornecidos por duas escolas portuguesas nas matérias de português e matemática, e tentar prever se eles foram aprovados ou não. Para tal, faremos uso dos métodos citados no tópico "Modelos de classificação".

C. Modelos de classificação

Nesta seção, vamos entender os mecanismos matemáticos por trás de cada um dos modelos que serão utilizados, baseando-se principalmente no exposto do livro Introduction to Statistical Learning [3].

- Regressão logística
  Esse método busca estimar a probabilidade da variável dependente (saída, que vai ser predita) sabendo o valor dos demais p preditores escolhidos, e essa estimativa é feita usando o método da máxima verossimilhança, o qual consiste em encontrar um estimador θ que maximize a equação 2 utilizando a equação 1 como fórmula para calcular as probabilidades:

$$P(Y=1) = \frac{1}{1+e^{-}\beta_0+\beta_1 X_1+...+\beta_p X_p} \quad (1)$$

  e os coeficientes $(\beta_0, \beta_1, ..., \beta_p)$ são os parâmetros que serão ajustados de acordo com o processo de treino.

$$\hat{\theta}_{mle} = argmax \sum_{i=1}^{n} f(x_i|\theta) \quad (2)$$

As vantagens desse método se resumem ao elevado acerto do modelo em problemas linearmente separáveis, é de fácil implementação e eficiente para treinar, enquanto as desvantagens se dão ao empregar essa técnica em problemas em que não há relação direta (linear) entre os preditores, a variável a ser predita e a separação das classes é feita tomando a média da variável dependente, o que nem sempre é interessante e pode sofrer overfitting em conjunto de dados muito extensos.

- Linear Discriminant Analysis (LDA)
  Esse método simples assume que os dados têm distribuição gaussiana e que toda variável tem amostras que variam nas mesmas quantidades em torno da média. Em seguida, o que o método busca é uma forma de separar as classes por meio de um limite linear de forma a maximizar a distância entre as classes analisadas.
  Dessa forma, os dados são colocados em um eixo e busca-se ajustar o limite linear, porém, esse método falha uma vez que os dados das classes compartilhem a mesma média.
  Assim, os cálculos de probabilidade de cada classe são feitos a partir da densidade gaussiana multivariada [4] mostrada na equação 3 e os limites entre as classes são definidos pelo classificador Bayesiano de acordo com a equação 4, em que "p"é o número de preditores, μ é a média e $\sum$ é a matriz de covariância p x p para um vetor com p entradas.

$$f(x) = \frac{1}{(2\pi)^{\frac{p}{2}}|\sum|^{\frac{1}{2}}} exp(-\frac{1}{2}(x-\mu)^T(\sum)^{-1}(x-\mu)) \tag{3}$$

  e, sendo k o número da classe observada:

$$\delta_k(x) = x^T(\sum)^{-1}\mu_k - \frac{1}{2}\mu_k^T(\sum)^{-1}\mu_k + log\pi_k \tag{4}$$

  Como vantagem desse método podemos citar que ele é simples e rápido na implementação e no treino (mesmo comparado com regressão logística). Como desvantagem, esse método precisa que os dados tenham uma distribuição normal (gaussiana) e pode não ajustar tão bem para algumas variáveis categóricas.
- Quadratic Discriminant Analysis (QDA)
  De forma semelhante ao método LDA, o QDA assume que a função de probabilidade é uma distribuição gaussiana multivariada, porém, para o QDA, cada classe terá sua própria matriz de covariância, ou seja, agora o classificador Bayesiano será da forma apresentada na equação 5.

$$\begin{aligned}\delta_k(x) = &-\frac{1}{2}x^T(\sum)_k^{-1}x + x^T(\sum)_k^{-1}\mu_k \\ &-\frac{1}{2}\mu_k^T(\sum)_k^{-1}\mu_k - \frac{1}{2}log|(\sum)_k| + log\pi_k\end{aligned} \tag{5}$$

  As vantagens desse método é que a classificação é feita de forma rápida e os resultados são geralmente mais precisos que os métodos KNN e LDA, enquanto como desvantagens podemos citar que assume-se que os dados possuem distribuição gaussiana e o tempo de treino pode ser bastante elevado.
- K-Nearest Neighbours (KNN)
  O método KNN, como o nome sugere, busca classificar o novo dado a partir da proximidade dele com os dados ao seu redor. Informalmente, se há mais dados da classe A do que de outra classe perto da amostra, o método sugere que a classe da amostra é também A.
  Desse modo, a ideia intuitiva simples também se estende para a abordagem puramente matemática, ou seja, o cálculo da proximidade é um emprego do cálculo da distância euclidiana entre pontos do espaço aprendida no ensino fundamental e, após isso, a classificação é feita olhando para a classe a que os mais próximos pertencem.
  Como vantagens desse método podemos citar que é fácil de implementar, não requer treinamento antes de fazer as previsões e a inserção de novos dados é de maneira instantânea já que não é necessário treinar. Como desvantagens podemos citar que o método é lento para dados com elevada dimensão e com um conjunto de dados muito grande.
- Support Vector Machines (SVM)
  Essa técnica [6] consiste em encontrar o hiperplano que melhor separa, de forma a maximizar a margem, as classes, ou seja, o problema a ser otimizado, explicitado na equação 6, é encontrar os coeficientes do hiperplano que maximize a distância entre ele e as amostras de cada classe, sendo y_i o rótulo pertencente ao intervalo [-1,1] associado à classe, x_i as observações de treino p-dimensionais, β_j um dos p coeficientes do hiperplano sujeitos à restrição da equação 7 e M a margem maximal:

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip}) \geq M \tag{6}$$

$$\sum_{j=1}^{p} \beta_j^2 = 1 \tag{7}$$

  As vantagens desse método são que ele funciona em dados bem separados no espaço analisado mesmo que haja muitos preditores envolvidos, além de também ser eficiente em se ajustar a dados com distribuição não-linear, enquanto para as desvantagens, temos que o modelo não é tão eficiente quando há muitas amostras, pois o tempo de treino se torna elevado, ou ruídos no conjunto de dados e também esse método não retorna uma probabilidade diretamente o que pode ser interessante em alguns problemas.

D. Métricas

Para avaliar a eficiência dos métodos de classificação precisamos [5] verificar se o modelo consegue prever corretamente para amostras do conjunto de teste (acurácia), mas, de forma geral, a matriz de confusão resume o quanto o modelo acertou e errou para cada classe definindo um limiar de probabilidade. Para completar, plotamos também a curva característica do

receptor (ROC), que representa uma curva de probabilidade, no intervalo [0,1], criada ao comparar a taxa de verdadeiro positivos (classe positiva predita como positiva) com a taxa de falsos positivos (classe negativa predita como positiva), e a área sob essa curva, que exprime o quão bom o modelo é em prever adequadamente as classes.

II. Métodos

A. Visão geral sobre o conjunto de dados

O problema do trabalho consiste em prever a situação final de um aluno de ensino médio. Os dados analisados são de duas escolas de Portugal, utilizando as matérias de matemática e português. O conjunto de dados possui as seguintes variáveis:

- school - Escola do estudante ('GP' ou 'MS')
- sex - Sexo do estudante ('F' ou 'M')
- age - Idade do estudante
- address - Localização da casa do aluno ('R' (rural) ou 'U' (urbano))
- famsize - Tamanho da família
- Pstatus - Estado de relacionamento entre os pais ('T' (juntos) ou 'A' (separados))
- Medu - Nível de educação da mãe
- Fedu - Nível de educação do pai
- Mjob - Trabalho da mãe
- Fjob - Trabalho do pai
- reason - Razão pela escolha da escola
- guardian - Quem detem a guarda do estudante
- traveltime - Duração do tempo de transporte até a escola
- studytime - Duração do estudo
- Failures - Número de reprovações
- Schoolsup - Suporte adicional da escola
- paid - Aulas extra na matéria
- activities - Atividades extra-curriculares
- Nursery - Presença na enfermaria da escola
- Higher - Vontade de cursar ensino superior
- Internet - Acesso a internet em casa
- Romantic - Participa de um relacionamento romântico
- Famrel - Qualidade das relações familiares
- Freetime - Quantidade de tempo livre
- Goout - Quantidade de saídas com os amigos
- Dalc - Consumo diário de álcool
- Walc - Consumo de álcool aos fins de semana
- health - Estado de saúde atual
- absences - Número de faltas
- G1 - Nota do primeiro período
- G2 - Nota do segundo período
- G3 - Nota do terceiro período

B. Análise dos dados e escolha dos principais preditores

Para caracterizar a aprovação dos alunos, a variável G3 será a referência adotada para definir a situação final do aluno que será utilizada nos modelos de predição. Os estudantes com notas finais maiores ou igual a 10, de um máximo de 20, foram considerados aprovados. Dessa forma a variável G3 foi transformada na variável binária Approved.

Para prever a situação do aluno foram escolhidos quatro preditores com base na correlação com a variável G3 como é mostrado na matriz de correlação na Figura 1.

Apesar das variáveis G1 e G2 terem forte correlação com a variável G3 ela não traz informações a respeito das características dos alunos pois representam notas anteriores dos alunos o que implicam em uma redundância na predição. Tal redundância é melhor visualizada na matriz de correlação, onde é possível ver que as notas G1 e G2 têm valores de correlação com os demais proditores similares em relação aos valores de correlação de G3 com os mesmos preditores.

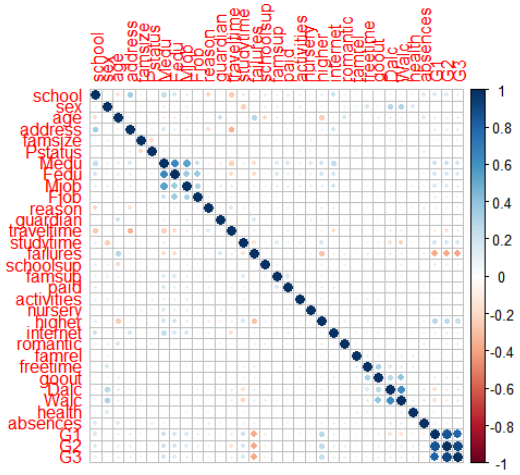
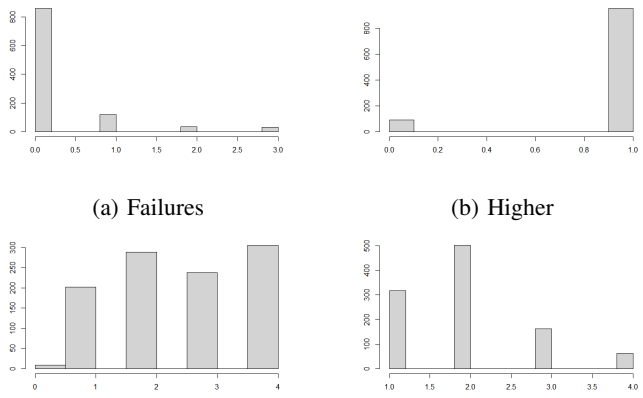


Figura 1: Matriz de correlação

Podemos notar uma maior influência das variáveis Medu, Fedu, higher, failures e studytime em G3. Porém, percebemos que Medu e Fedu têm basicamente a mesma influência na variável G3, além de serem bastante correlacionadas entre si. Assim, entre esses dois preditores, priorizamos Medu por ter maior correlação com G3, ou seja, os preditores escolhidos serão Medu, higher, failures e studytime.

Seguindo, realizamos as análises monovariadas dos 4 preditores, que inclui o mapeamento de seus histogramas, cálculo das médias, desvios padrão e obliquidade, utilizando todas as 1044 amostras. Tais análises estão representadas na Tabela I.



(a) Failures (b) Higher

(c) Medu (d) StdTime

Figura 2: Histogramas

Realizamos cálculos de média, desvio padrão e obliquidade e armazenamos todos os valores em uma mesma tabela, para facilitar visualização. Isso está representado na Tabela I:

	Média	Desvio Padrão	Obliquidade
Medu	2.60	1.12	-0.13
studytime	1.97	0.83	0.66
failures	0.26	0.65	2.77
higher	0.91	0.28	-2.96
Approved	0.77	0.41	-1.34

Tabela I: Analise monovariada

Ao realizar a análise bivariada mostrada na Figura 3, onde plotamos os preditores 2 a 2, assim como seus coeficientes de correlação, para podermos ver se há algum nível de correlação entre eles.

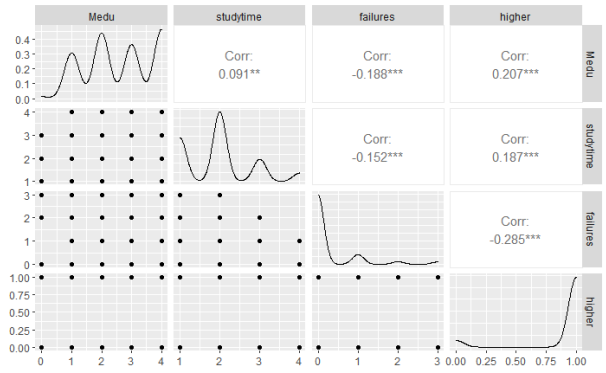


Figura 3: Analise bivariada

Para a análise dos componentes principais (PCA), útil para investigar a relação de cada preditor escolhido com a distribuição dos dados no espaço, geramos o gráfico biplot, uma vez que os dados ficam mais visíveis, como é mostrado na Figura 4. Dos elementos da PCA, destaca-se os números que representam o índice da amostra trabalhada em que os vermelhos correspondem ao valor 0 e os verdes correspondem ao valor 1 para ”Approved”, enquanto os vetores (setas azuis) representam os preditores escolhidos. Diante disso, vemos que o preditor ”failures”tem correlação negativa e o ”higher”possui certa correlação positiva.

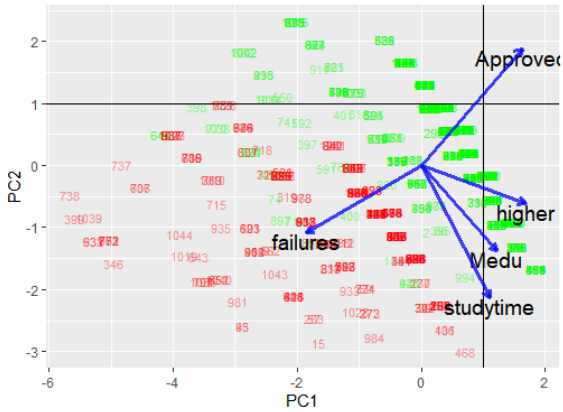


Figura 4: Biplot

O próximo passo consiste em separamos o conjunto de dados em um conjunto de treino (aproximadamente 70% das amostras) e um de validação (aproximadamente 30% das amostras). Em seguida é iniciada a etapa de treinamento (ajuste dos parâmetros) e cálculo e plots das métricas aplicadas a cada método.

III. Resultados

Das métricas discutidas na introdução, sabendo que há 245 ”1”s e 68 ”0”s relativos à variável dependente Approved no conjunto de treino, temos:

- a Tabela 2, resumindo os resultados de acurácia

Método	Acurácia
Regressão Logística	0.8051118
LDA	0.80830670
QDA	0.81150159
KNN	0.91054313
SVM	0.80511182

Tabela II: Acurácia por modelo

Utilizando 7 casas decimais com o intuito de diferenciar precisamente o resultado, podemos ver que os modelos possuem boa acurácia em geral (acima de 70%), mas destaca-se que os de probabilidade logística (Reg. Logística) e os com separadores de classe lineares (LDA e SVM) tiveram resultados semelhantes($\approx$ 80%). Enquanto que, em relação aos lineares, o QDA (separador quadrático) teve uma leve melhora ($\approx$ 1%) e o KNN teve um salto na acurácia de aproximadamente 10%.

- a Tabela 3, mostrando as matrizes de confusão

	FALSE	TRUE
0	21	56
1	9	227

(a) Regressão logística

	FALSE	TRUE
0	16	52
1	8	237

(b) LDA

	FALSE	TRUE
0	41	27
1	1	244

(c) KNN

	FALSE	TRUE
0	22	46
1	13	232

(d) QDA

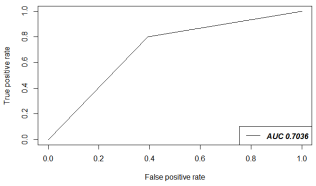
	FALSE	TRUE
0	11	57
1	4	241

(e) SVM

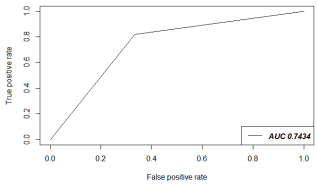
Tabela III: Matrizes de confusão

De acordo com o que foi explanado na introdução sobre a matriz de confusão, podemos, ao comparar os resultados extremos, identificar que o pior modelo foi a regressão logística, pois teve a menor quantidade de acertos da classe "1"ao comparar com os demais métodos, enquanto que o KNN obteve maior acerto dessa classe (e também da classe 0).

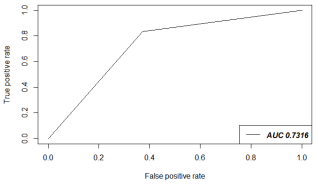
- e a Figuras 5, os plots das curvas ROC, junto à medida da área sob a curva



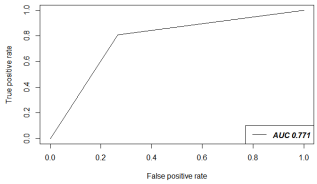
(a) Logistic Regression



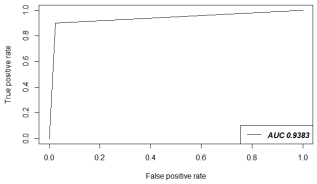
(b) LDA



(c) QDA



(d) SVM



(e) KNN

Figura 5: Curvas ROC

Como sabemos que, quanto maior a área sob a curva, melhor será o modelo, podemos avaliar nossos modelos com base nas curvas ROC. Assim, é fácil perceber que o modelo KNN é o que prevê melhor as classes trabalhadas, com uma AUC (area under curve) de 0,9383. Por outro lado, os outros 4 modelos tiveram um desempenho bem semelhante, com uma AUC na faixa de 0,70 a 0,77.

Concluindo, verificamos que todos os quatro métodos tiveram um desempenho aceitável, uma vez que os comprovamos, tanto pela curva ROC, como pelo próprio cálculo da acurácia, além da matriz de confusão, que pode ser considerada uma boa métrica para informações adicionais. Ademais, podemos perceber que dos métodos utilizados, a regressão logística possui o pior desempenho performático, devido, em grande parte, ao fato de as variáveis independentes serem zero ou um.

Referências

[1] P. Cortez and A. Silva. Using Data Mining to Predict Secondary School Student Performance. In A. Brito and J. Teixeira Eds., Proceedings of 5th FUture BUsiness TEChnology Conference (FUBUTEC 2008) pp. 5-12, Porto, Portugal, April, 2008, EUROSIS, ISBN 978-9077381-39-7.

[2] R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

[3] G. James, D. Witten, T. Hastie e R. Tibshirani. "An Introduction to Statistical Learning with Applications in R"2013, Springer.

[4] K. Al-jabery et al."Computational Learning Approaches to Data Analytics in Biomedical Applications"2020, Spring.

[5] Hossin, M. e Sulaiman, M.N. "A REVIEW ON EVALUATION METRICS FOR DATA CLASSIFICATION EVALUATIONS", IJDKP, Vol.5, No.2, March 2015

[6] M. Awad, R. Khanna. "Efficient Learning Machines"2015, Springer.

Lista de códigos

Importação de dados

```
d1=read.table("student-mat.csv",sep=";",header=TRUE)
d2=read.table("student-por.csv",sep=";",header=TRUE)

d3=merge(d1,d2,all = TRUE)
```

Transformação dos valores das classes em numéricos

```
library(dplyr)
d3$romantic=case_when(d3$romantic=="yes" ~ 1L, d3$
    romantic=="no" ~ 0L  );

d3$internet=case_when(d3$internet=="yes" ~ 1L, d3$
    internet=="no" ~ 0L  );

d3$nursery=case_when(d3$nursery=="yes" ~ 1L, d3$
    nursery=="no" ~ 0L  );
d3$schoolsup=case_when(d3$schoolsup=="yes" ~ 1L, d3$
    schoolsup=="no" ~ 0L  );
d3$famsup=case_when(d3$famsup=="yes" ~ 1L, d3$famsup
    =="no" ~ 0L  );
d3$higher=case_when(d3$higher=="yes" ~ 1L, d3$higher
    =="no" ~ 0L  );
d3$activities=case_when(d3$activities=="yes" ~ 1L,
    d3$activities=="no" ~ 0L  );
d3$paid=case_when(d3$paid=="yes" ~ 1L, d3$paid=="no"
     ~ 0L  );
d3$Fjob=case_when( d3$Fjob=="teacher" ~ 4L,d3$Fjob==
    "health" ~ 3L,d3$Fjob=="services" ~ 2L, d3$Fjob
    =="at_home" ~ 1L,d3$Fjob=="other" ~ 0L);
d3$Mjob=case_when( d3$Mjob=="teacher" ~ 4L, d3$Mjob
    =="health" ~ 3L, d3$Mjob=="services" ~ 2L, d3$
    Mjob=="at_home" ~ 1L, d3$Mjob=="other" ~ 0L);
d3$reason=case_when( d3$reason=="other" ~ 3L, d3$
    reason=="course" ~ 2L, d3$reason=="reputation" ~
     1L, d3$reason=="home" ~ 0L);
d3$guardian=case_when( d3$guardian=="other" ~ 2L, d3
    $guardian=="father" ~ 1L, d3$guardian=="mother"
    ~ 0L);
```

```
d3$famsize=case_when( d3$famsize=="GT3" ~ 1L, d3$
    famsize=="LE3" ~ 0L);
d3$Pstatus=case_when(d3$Pstatus=="A" ~ 1L, d3$
    Pstatus=="T" ~ 0L);
d3$sex=case_when(d3$sex=="M" ~ 1L, d3$sex=="F" ~ 0L)
    ;
d3$address=case_when(d3$address=="U" ~ 1L, d3$
    address=="R" ~ 0L);
d3$school=case_when(d3$school=="GP" ~ 1L, d3$school
    =="MS" ~ 0L);
```

Análise monovariada: média, desvio-padrão, obliquidade e boxplots

```
media_stu <- sapply(Filter(is.numeric, d3), mean)

#hist(d1$LSTAT)
desvpad_stu <- sapply(Filter(is.numeric, d3), sd)

#install.packages("timeSeries")
library(timeSeries)

obliq_stu <- colSkewness(d3,pvalue=true)

resume_stu <- data.frame(media=media_stu,
    desvioPadrao=desvpad_stu, skewness = obliq_stu)
resume_stu

for (i in 1:length(names(d3))){
  boxplot(d3[i], xlab=colnames(d3)[i], ylab="values"
    )
}


## Após analisar a relação entre os preditores e a
    saída

#install.packages("corrplot")
library(dplyr)
library(corrplot)
res <- cor(d3)
round(res, 2)

corrplot(res, method = "circle")

df <- data.frame(Medu=d3$Medu, studytime=d3$
    studytime, failures=d3$failures, higher=d3$
    higher, G3 =d3$G3)

df$G3=case_when(d3$G3 < 10 ~ 0L, d3$G3>=10 ~ 1L);

colnames(df)[5] <- "Approved"

corrplot(cor(df), method="circle")

for (i in 1:length(names(df))){
  boxplot(df[i], xlab=colnames(df)[i], ylab="values"
    )
}

# Histogramas dos preditores escolhidos

for (i in 1:length(names(df))){
  hist(df[,i], main=("histograma"), xlab=colnames(df
    )[i], ylab="Frequency" )
}
```

PCA

```
library(ggplot2)
PCbiplot <- function(PC, df, x="PC1", y="PC2",
    colors=c('black', 'black', 'red', 'red')) {
    # PC being a prcomp object
    #data <- data.frame(obsnames=row.names(PC$x), PC
    $x)

  data<-data.frame(obsnames= 1:1044, PC$x)

  plot <- ggplot(data, aes_string(x=x, y=y)) + geom_
    text(alpha=.4, size=3, aes(label=obsnames),color
    =ifelse(df$Approved == 1, 'green', 'red'))#Dados

    plot <- plot + geom_hline(aes(0), size=0.2,
    yintercept=1, xintercept=1) + geom_vline(aes(0),
     size=.2, color=colors[2], xintercept=1) #Nomes
    datapc <- data.frame(varnames=rownames(PC$
    rotation), PC$rotation)

    mult <- min(
        (max(data[,y]) - min(data[,y])/(max(datapc[,
    y])-min(datapc[,y]))),
        (max(data[,x]) - min(data[,x])/(max(datapc[,
    x])-min(datapc[,x])))
        )

    datapc <- transform(datapc,
            v1 = .7 * mult * (get(x)),
            v2 = .7 * mult * (get(y))
            )

    plot <- plot + coord_equal() + geom_text(data=
    datapc, aes(x=v1, y=v2, label=varnames), size =
    5, vjust=1, color=colors[1])

    plot <- plot + geom_segment(data=datapc, aes(x
    =0, y=0, xend=v1, yend=v2), size=1.1, arrow=
    arrow(length=unit(0.2,"cm")), alpha=0.75, color=
    colors[4])

    plot
}

pca.eg<-prcomp(df, scale=T)

summary(pca.eg)

PCbiplot(pca.eg, df, colors=c("black", "black", "red
    ", "blue"))
```

Separando o conjunto de dados em conjunto de treino e teste:

```
library(caret)
library(glmnet)

training.samples <- createDataPartition(df$Approved,
    p = 0.7, list = FALSE)
train.data  <- df[training.samples, ]
test.data <- df[-training.samples, ]
```

Aplicando os seguintes modelos de classificação: Regressão Logística

```
regLog <- glm(Approved ~ ., data=train.data)
summary(regLog)

probab_regLog <- predict(regLog,test.data, type = "
    response")
predicted.classes <- ifelse(probab_regLog > 0.5, "1"
    , "0")


print("Taxa de acerto:")
mean(predicted.classes == test.data$Approved)
```

```
print("Matriz de confusão:")
table(test.data$Approved, predicted.classes>0.5)

#install.packages("ROCR")
library(ROCR)

rocplot <- function(pred , truth)
  { predob <- prediction(pred , truth)
  perf <- performance(predob , "tpr", "fpr")
  plot(perf)
  }

rocplot(test.data$Approved, predicted.classes )
```

Linear Discriminant Analysis(LDA)

K-Nearest Neighbours(KNN)

```
#install.packages("class")
library(class)

perc.erro = seq(1,20)

for(i in 1:20){
    set.seed(1)
    previsoes = knn(train = train.data, test= test.
    data,cl= train.data$Approved,k=i)
    perc.erro[i] =mean(train.data$Approved !=
    previsoes)
}

library(ggplot2)

k.values <- 1:20
error.df <- data.frame(perc.erro,k.values)


ggplot(error.df,aes(x=k.values,y=perc.erro)) + geom_
    point()+ geom_line(lty="dotted",color='red')

index_min = which(perc.erro==min(perc.erro))

paste("K para o menor erro:",error.df$k.values[index
    _min[1]], collapse = " ")


preds_knn = knn(train = train.data, test= test.data,
    cl= train.data$Approved,k=18)

accur_knn = mean(preds_knn == test.data$Approved)
paste("Acurácia do knn:",accur_knn, collapse = " ")

preds_knn.classes <- ifelse(preds_knn == 1, "1", "0"
    )

table(test.data$Approved, preds_knn.classes > 0.5)


rocplot(test.data$Approved, preds_knn)
```

Quadratic Discriminant Analysis(QDA)

```
library(MASS)
library(klaR)
library(ROCR)
set.seed(101)

qda_stu <- qda(Approved ~ ., data=train.data)


preds_qda = predict(qda_stu,test.data)


accur_qda = mean(preds_qda$class == test.data$
    Approved)
paste("Acurácia da QDA:",accur_qda,collapse = " ")



preds_qda.classes <- ifelse(preds_qda$class == 1, "1
    ", "0")

print("Matriz de confusão:")
table(test.data$Approved, preds_qda.classes > 0.5)


rocplot(test.data$Approved, preds_qda$class)
```

Support Vector Machine(SVM)

```
#install.packages('e1071')
library(e1071)

classifier = svm(formula = Approved ~ .,
                 data = train.data,
                 type = 'C-classification',
                 kernel = 'linear')


preds_svm = predict(classifier, newdata=test.data )


accur_svm = mean(preds_svm == test.data$Approved)
paste("Acurácia do SVM:",accur_svm,collapse = " ")


print("Matriz de confusão")
preds_svm.classes <- ifelse(preds_svm == 1, "1", "0"
    )

table(test.data$Approved, preds_svm.classes > 0.5)


rocplot(test.data$Approved, preds_svm)
```