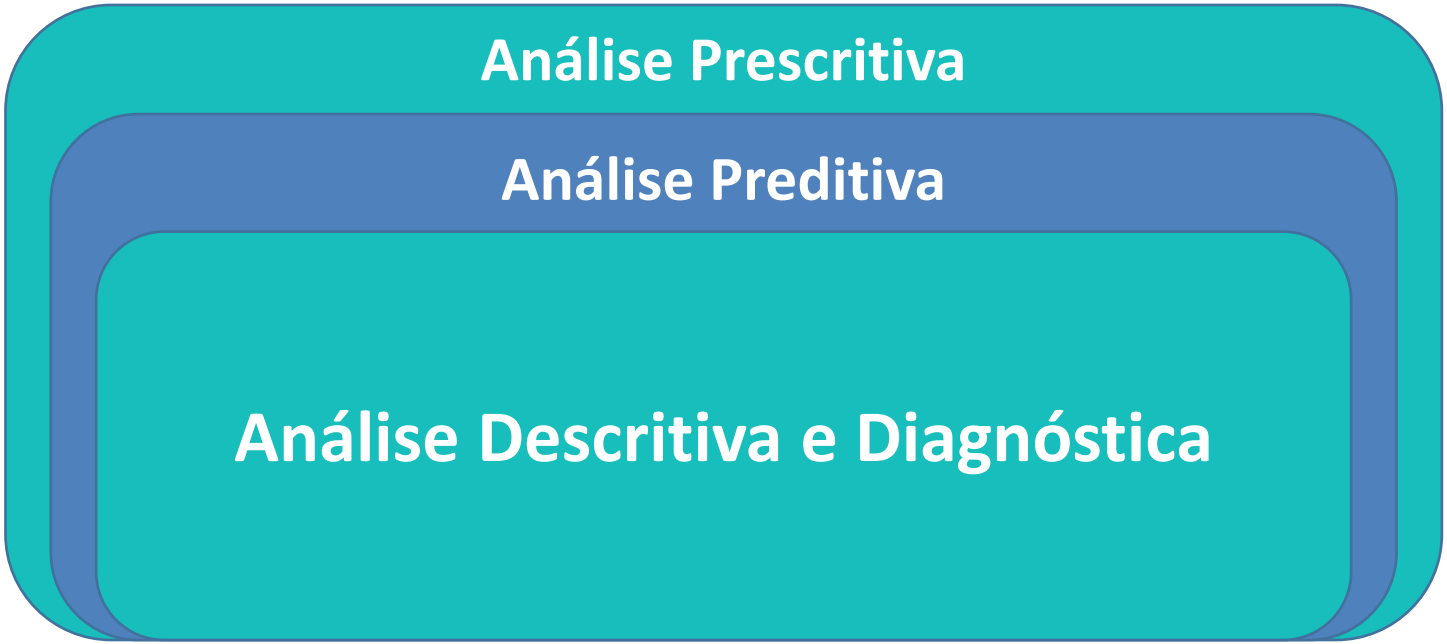
DATA DISCOVERY E ANALYTICS

Rodrigo Moravia

PUC Minas
Virtual

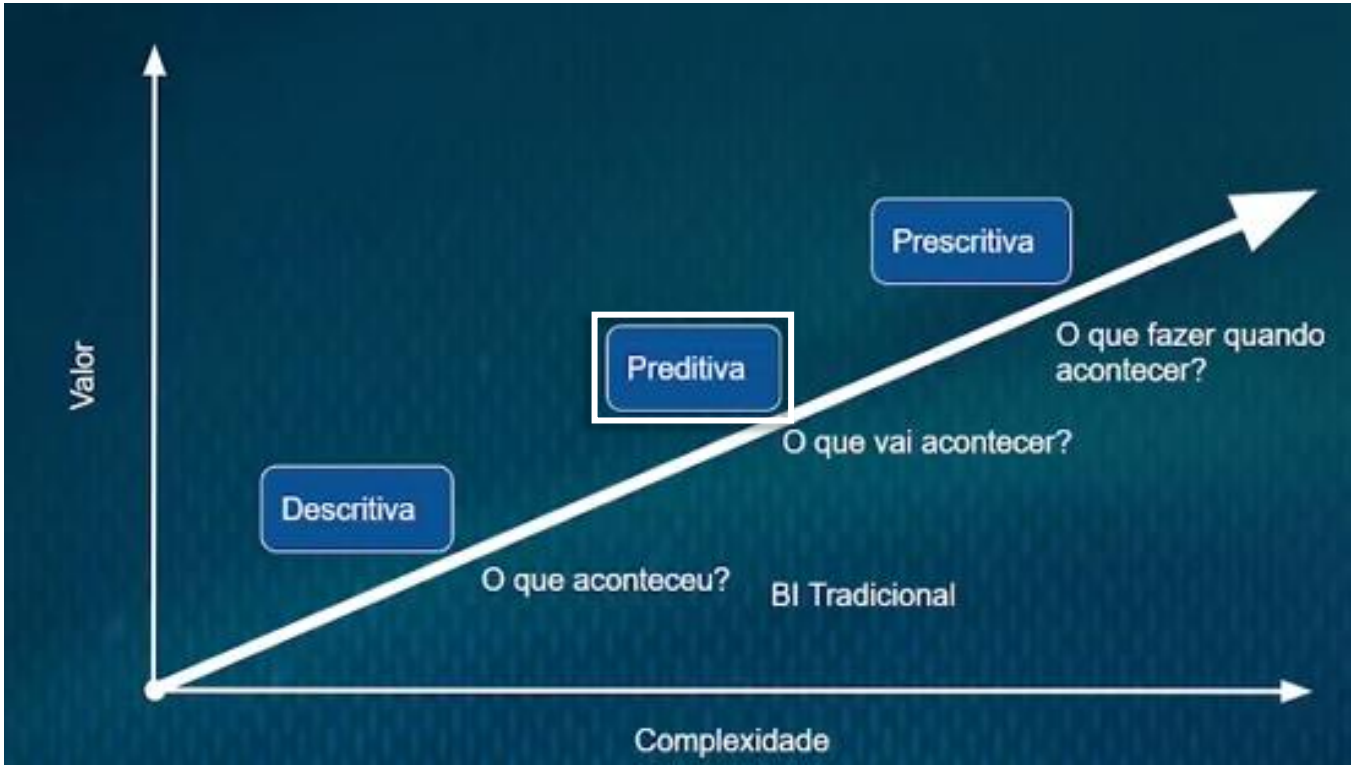
Análise Preditiva

Etapas de Data Discovery



Fonte: Imagem do Autor (2022)

Etapas de Data Discovery



Fonte: Imagem do Autor (2022)

O que é?

Prever o futuro sempre foi um grande desejo da humanidade.

A análise de dados **preditiva** significa tomar como referência dados do passado e do presente para, a partir deles, formular prognósticos.

Esse tipo de análise serve para trazer respostas tanto para questões já conhecidas quanto para aquelas que possam vir a acontecer.

O que é?

- Por conhecer os dados do passado é possível prever o comportamento futuro

- Em um mercado cada vez mais competitivo a empresa que aplica a análise preditiva (data-driven) costuma estar um passo a frente da concorrência.

O que é?

- Segundo o Gartner, a análise **preditiva** é uma forma de análise avançada que verifica dados ou conteúdos para responder à pergunta: o que é provável que aconteça no futuro?
- Graças ao Big Data, os dados obtidos através de todos os sistemas conectados podem ser interpretados para conseguir previsões sobre como uma pessoa ou um grupo de população irá se comportar, algo que também é aplicável aos negócios ou processos.

continuação

- O nome é autoexplicativo. Trata-se de uma modalidade de processamento e interpretação de dados que tem como objetivo prever diferentes cenários no futuro.
- Em uma economia cada vez mais competitiva, a utilização dessa ferramenta serve como um diferencial. Afinal de contas, qual empresa não gostaria de ter maior certeza dos possíveis desfechos para uma decisão?
- Não existe mágica. A análise preditiva pode antecipar eventos com base em tendências identificadas a partir de situações similares do passado.
- Cabe ao sistema, portanto, indicar a probabilidade de algum cenário se repetir com base em cálculos estatísticos e algoritmos sofisticados.

continuação

- Tão importante quanto obter dados é saber aproveitá-los. Não que esse tipo de sistema seja à prova de falhas. Mas os números traçados nas análises costumam ser consistentes por terem como base o histórico de dados do próprio negócio.

- É como se a tecnologia revelasse o futuro lançando um olhar sobre o passado. E não estamos falando de um recurso para poucas empresas: praticamente todos os ramos de atividade podem se beneficiar da análise preditiva.

continuação

- Há várias abordagens possíveis, mas, via de regra, o conceito se baseia na criação de um **modelo preditivo**, ou seja, de uma função matemática que, quando aplicada sobre os dados, vai dar uma previsão sobre um problema.

- Com a integração do BI à análise preditiva, é possível, por exemplo, entender quais são as necessidades de seu público-alvo antes de levar um novo produto ao mercado, aumentando as chances de sucesso.

Exemplo

- Uma pessoa de 30 anos, sexo masculino, solteiro, desempregado, possui uma probabilidade de atrasar o pagamento de uma fatura de 32%.
- Já uma senhora de 65 anos, viúva, aposentada, possui uma probabilidade de atrasar o pagamento de uma fatura de 4%.
- Ou seja, as alterações nas características alteram as probabilidades.

Importante

- A validação dos modelos preditivos são um tópico à parte, porém, em simples termos, é necessário que o modelo preditivo seja capaz de acertar pelo menos de 70% a 90% em das tentativas.
- Caso ele acerte menos de 50%, seria o mesmo que competir com a predição de cara ou coroa. Já se ele conseguir atingir 100% ou acima de 95%, pode ser que exista alguma variável no modelo comprometida.

continuação

- Algumas limitações típicas de modelos preditivos são a dificuldade de fazer predições sobre categorias múltiplas ou invés de prever se o cliente vai pagar não. Digamos que os clientes podem: “Pagar à vista”, “Pagar a prazo”, “Pagar por cartão”, “Pagar por boleto”, “Não vai pagar”.
- As taxas de assertividade por categoria podem ser muito discrepantes. Para superar essa limitação é preciso construir algoritmos específicos para o problema além de efetuar transformações na fase de preparação de dados.

Onde se aplica

Áreas	Exemplo
Detecção de fraudes	Cartão de crédito. Em busca de compras fora do normal das compras de um cliente.
Otimização de compras	Criar similaridades de perfil de clientes que compram em seu Website
Alocação de recursos	Prever venda. Quanto que uma filial venderá para você ter fornecimento de estoque suficiente
Churn	Perfil de perda de clientes. Com a capacidade de antever o momento em que clientes já não estão mais satisfeitos com as soluções que a eles são oferecidos, a empresa pode se planejar melhor ou se preparar melhor para a perda do cliente.
Tratamentos médicos	Qual a eficácia de uma nova abordagem terapêutica? Laboratórios e instituições especializadas precisam fazer estudos e testes exaustivos para encontrar a resposta, mas é possível agilizar esse processo com análises preditivas que consideram parâmetros genéticos, condições ambientais, fatores comportamentais, faixa etária, entre outros
Estratégia de Marketing	Avaliar as chances de uma campanha dar o retorno esperado, estimar a eficácia de um projeto de fixação de marca, identificar o momento certo para realizar uma ação.

Onde se aplica

Áreas	Exemplo
Upsell e cross-sell	Ao contrário da previsão de churn, neste ponto a empresa pode perceber a disposição do cliente em se interessar por um novo produto. Então, é possível abordá-lo de forma mais precisa para oferecer um upgrade que seja não só mais vantajoso para o cliente, mas também mais rentável para a empresa.
Segmentação de Leads	A prática de segmentar leads permite uma maior taxa de acertos para as empresas, já que proporciona a capacidade fazer apostas mais certeiras. Isso permite abordar clientes em potencial com ofertas precisas, e no momento mais adequado.

PUC Minas
Virtual

3 V’s da Análise Preditiva

3 V’s

Variedade

- É importante ter uma boa diversidade de fontes e formatos de dados, que permitirão uma análise mais profunda.

- Apostar em variedade também ajuda a ter resultados menos “viciados”, que podem ser causados por uma base de dados única.

Veracidade

- De nada adianta ter em mãos um volume enorme de dados se as informações que eles trazem não são confiáveis.

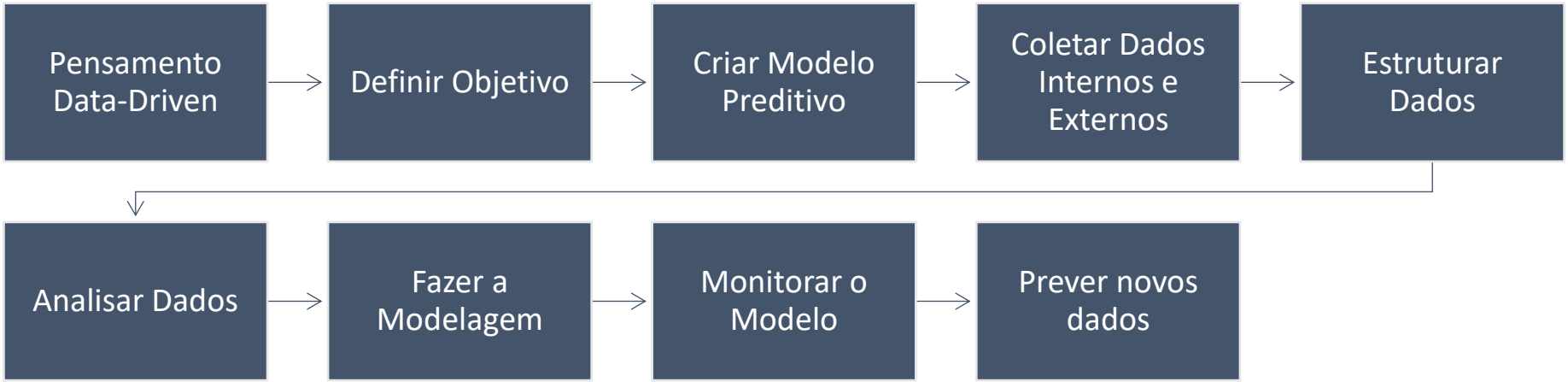
Velocidade

- Tão importante quanto ter dados confiáveis e diversificados é ter agilidade para processá-los, já que muitos insights podem não ser mais úteis se o timing relevante já tiver passado.

PUC Minas
Virtual

Etapas

Como estruturar?



Definir objetivo

- Antes de tudo, é preciso definir: qual é a finalidade da iniciativa? A utilização de um modelo deve ter objetivos claros e alinhados com a estratégia da empresa.
- O passo seguinte é buscar o benefício esperado com a análise preditiva, seja ele a compreensão do consumidor, seja observar tendências e oportunidades.
- Existem incontáveis aplicações da tecnologia nos negócios de uma empresa, por isso é importante determinar a abordagem antes de qualquer coisa.

Criar um modelo

- Definido o objetivo, é importante criar o modelo preditivo que será utilizado para alcançá-lo.
- O modelo define o modo como os dados selecionados para o projeto serão trabalhados, então ele é crucial para o sucesso.
- Esse passo envolve preparar os dados para que eles possam ser analisados de forma apropriada, realizar a amostragem experimental e testar qual formato de análise apresenta os melhores resultados.

Coleta

- Tão fundamentais quanto o modelo são os dados com os quais ele será alimentado.

Estruturar

- Uma vez que as fontes foram definidas e os dados coletados, o passo seguinte envolve a estruturação. Ela ajudará a organizar as informações para viabilizar análises mais eficientes.

- Esse processo inclui realizar a limpeza dos dados e também organizá-los em conjuntos que facilitem o processo.

Analisar

- Após os dados estarem prontos, resta realizar a análise propriamente dita, o que deve ser feito com cuidado para alcançar os melhores desfechos.
- É nesta fase que são produzidos os insights. Para isso, é necessário ter noções estatísticas a fim de avaliar e interpretar gráficos e as tendências que eles apontam.
- Existem três tipos de análise. A primeira é a univariada, em que cada variável é analisada isoladamente antes do cruzamento; outra é a bivariada, que estabelece relações entre duas variáveis, e a última é a multivariada, que estabelece relações entre mais variáveis.

Fazer Modelagem

- Após a condução da análise, é hora de criar o modelo preditivo utilizado para interpretar as informações.
- Aqui, utilizando técnicas estatísticas, é possível visualizar as relações estabelecidas entre as informações extraídas do banco de dados.
- Isso é feito por meio de técnicas estatísticas e matemáticas, que podem retornar os insights desejados.

Monitorar o Modelo

- Nem sempre o modelo utilizado é eterno. É importante acompanhar os resultados que ele produz para fazer os ajustes necessários com o tempo para garantir a qualidade da análise.

Cuidado

- O mercado já comprovou a utilização de técnicas avançadas de análise, como as de machine learning, como ferramentas poderosas para apoiar as organizações nos desafios de retenção de clientes.
- Mas estes não devem, no entanto, ser considerados como a solução para este problema. Apesar dos modelos preditivos fornecerem informações valiosas para direcionar as ações de retenção, a fidelidade dos clientes será realmente mantida por boas experiências e uma real vantagem competitiva enxergada sobre os demais concorrentes.

PUC Minas
Virtual

Técnicas de Regressão

Regressão

- Técnicas de Regressão
- Regressão Linear
- Regressão Logística e multinomial
- Regressão para variáveis categóricas

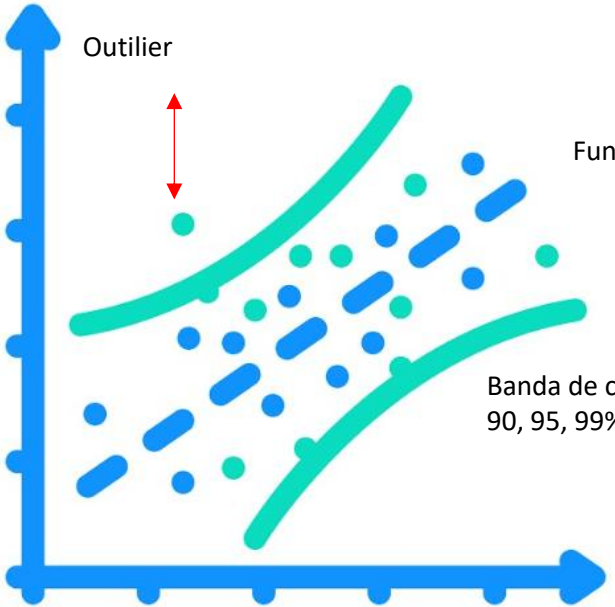
Técnicas de Regressão

- O foco principal das técnicas de regressão é descobrir na sua base de dados, qual o comportamento de uma determinada variável **dependente** no eixo **Y** em relação a outras variáveis **independentes** presentes no modelo, que será usada no eixo **X**.
- A Regressão usa estatística para descobrir funções e intervalos de distribuição que representam um determinado conjunto de dados.

Regressão Linear e Polinomial

- Usa-se quando a variável dependente é de natureza contínua
- Ela descobre uma função linear ou polinomial que descreve qual o comportamento dos dados
- Aplica-se para predizer onde os dados futuros estarão
- Linear Vs Polinomial
  - Linear ➔ quando há apenas uma variável independente
  - Polinomial ➔ variáveis múltiplas

Regressão Linear e Polinomial

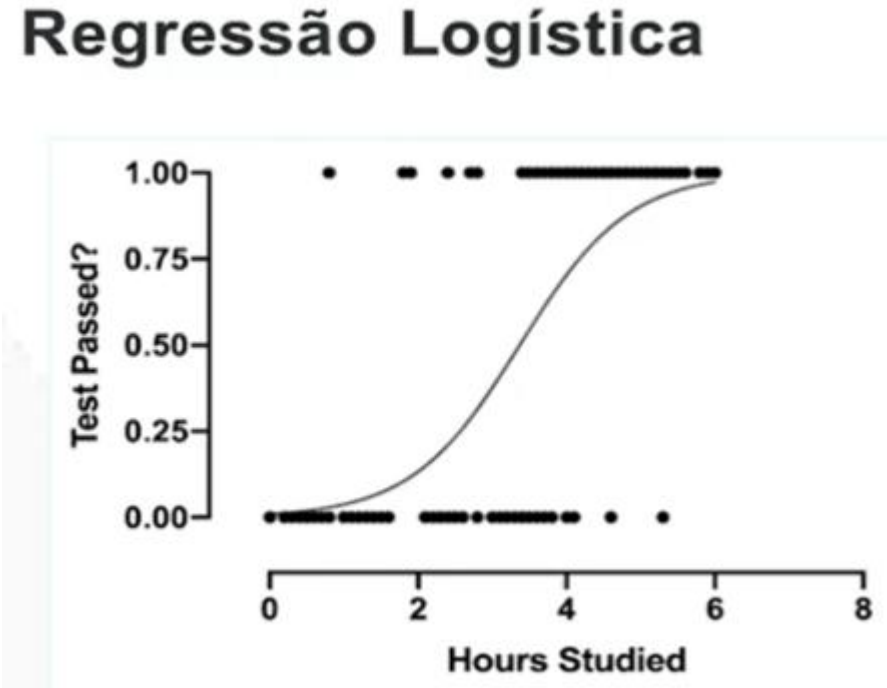


Regressão Logística

- Usa-se quando a variável dependente é **Binária**
- As variáveis independentes podem ser binárias ou contínuas
- A regressão logística multinomial permite que a variável dependente (Y) possua mais de duas categorias

Exemplo: saber se um determinado cliente da classe social A, B, C, D ou E (eixo X) comprará ou não (eixo Y) um determinado produto.

Regressão Logística



Como aplicar Regressão para Variáveis Categóricas

- O jeito mais comum é transformar as variáveis categóricas em variáveis Dummy
- Em estatística, uma variável **Dummy** é aquela que torna o valor de "zero" ou "um" indicando a ausência ou presença de qualidades ou atributos. Essas variáveis são usadas como dispositivos para classificar dados em categorias mutuamente exclusivas.
- Transforma as variáveis nominais em quantitativas, pois as ordinais você já consegue ordená-las naturalmente.

PUC Minas
Virtual

Estudo de Caso

Base de Dados Usando COLAB (Google)

VIVO_CHURN.csv

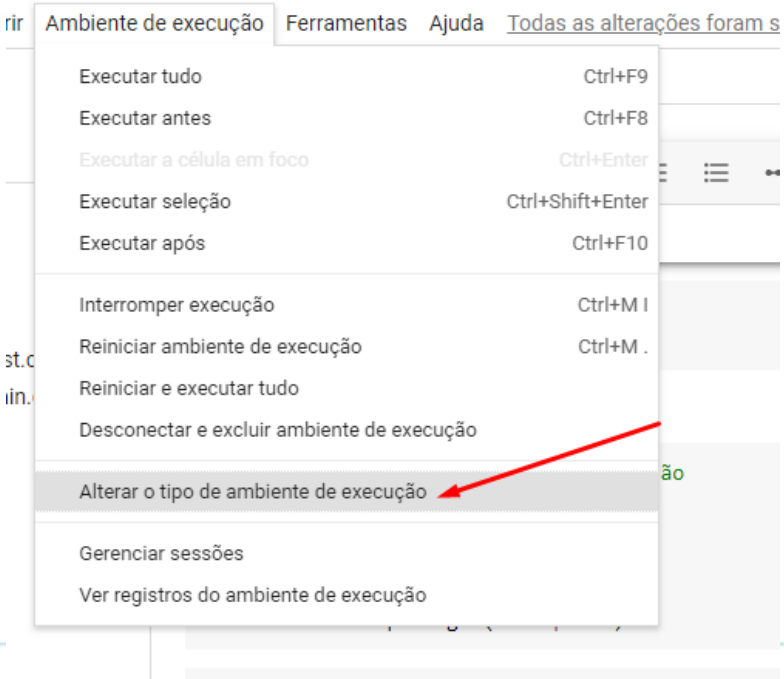
Fonte: https://www.kaggle.com/datasets/blastchar/telco-customer-churn?resource=download

Clientes que saíram no último mês (coluna CHURN)

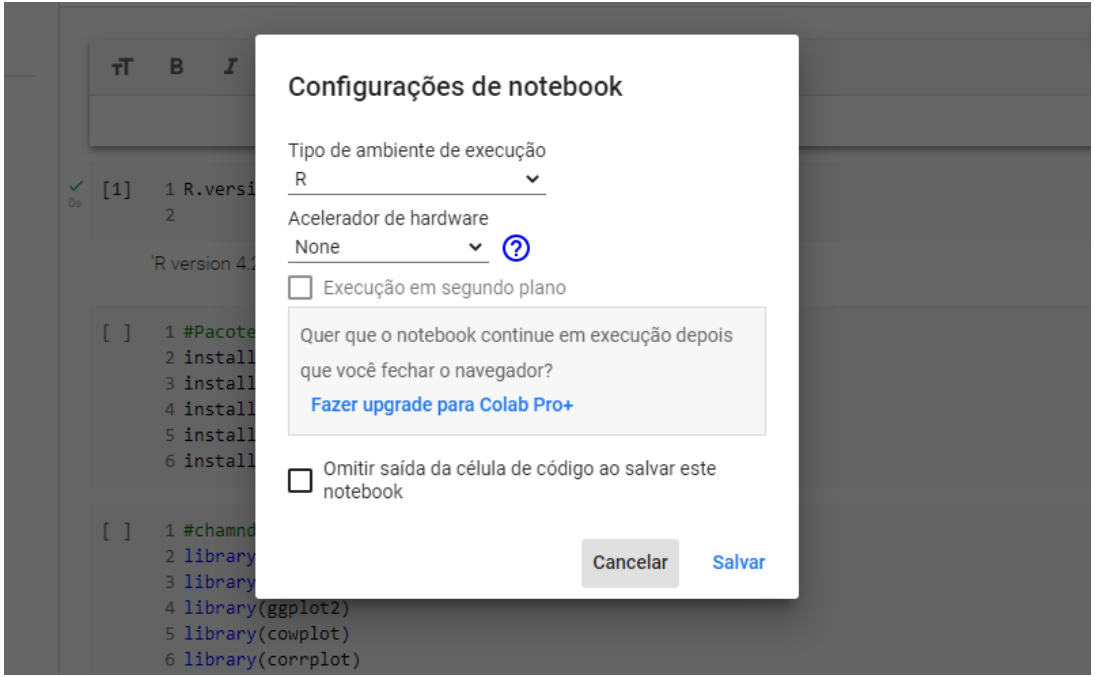
Serviços que cada cliente assinou – telefone, várias linhas, internet, segurança, online, backup online, proteção de dispositivo, suporte técnico e streaming de TV e filmes

Informações da conta do cliente – há quanto tempo ele é cliente, vigência do contrato, método de pagamento, faturamento sem papel, cobranças mensais e cobranças totais; informações demográficas sobre clientes: sexo, faixa etária (idoso ou não), dependentes e parceiros.

Colab usando R ou Python



Usando COLAB (Google)



Versão e pacotes

```
#Versão do R
R.version.string

#Pacotes necessarios - instalação
install.packages('tidyverse') #manipulação de dados
install.packages('ggplot2') #visualização
install.packages('cowplot') #visualização - unir gráficos
install.packages('caret') #modelos estatísticos
install.packages('corrplot') #matriz de correlação

#chamndo os pacotes ja instalados
library(caret)
library(tidyverse)
library(ggplot2)
library(cowplot)
library(corrplot)
```

Carregando

```
#carregando dados para o Data Frame (dados)
dados <-read.csv("VIVO_CHURN.csv", stringsAsFactors = T)

#Visualização dos dados
glimpse(dados)
summary(dados)

#Check dados faltantes e retirar
colSums(is.na(dados)) #se nulo
dados_1 <- dados[!is.na(dados$TotalCharges),]
           #diferente de nulo para o novo data frame (dados_1)
colSums(is.na(dados_1))
glimpse(dados_1)
```

Outliers

```
#outliers - visualizar dispersão dos dados via boxplot
dados_1 %>%
  ggplot(aes(x=Churn,y=tenure, fill=Churn)) +
  geom_boxplot() + geom_jitter(width=0.1,alpha=0.2)

dados_1 %>%
  ggplot(aes(x=Churn,y=MonthlyCharges, fill=Churn)) +
  geom_boxplot() + geom_jitter(width=0.1,alpha=0.2)

  dados_1 %>%
  ggplot(aes(x=Churn,y=TotalCharges, fill=Churn)) +
  geom_boxplot() + geom_jitter(width=0.1,alpha=0.2)
```

Conhecendo a base de dados

```
#verificando a classificação de cada variável - categórica(fatores)
ggplot(dados, aes(y = gender, x = gender, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = Partner, x = Partner, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = Dependents, x = Dependents, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = PhoneService, x = PhoneService, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = MultipleLines, x = MultipleLines, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = InternetService, x = InternetService, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = OnlineSecurity, x = OnlineSecurity, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = OnlineBackup, x = OnlineBackup, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = DeviceProtection, x = DeviceProtection, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = TechSupport, x = TechSupport, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = StreamingTV, x = StreamingTV, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = StreamingMovies, x = StreamingMovies, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = Contract, x = Contract, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = PaperlessBilling, x = PaperlessBilling, fill = Churn)) +  geom_bar(stat = "identity")
ggplot(dados, aes(y = PaymentMethod, x = PaymentMethod, fill = Churn)) +  geom_bar(stat = "identity")
```

Classificação das variáveis numéricas

```
#verificando a classificação de cada variável - numérica

ggplot(dados,
       aes(x = MonthlyCharges,
           fill = Churn)) +
  geom_density(alpha = 0.4) +
  labs(title = "")

ggplot(dados,
       aes(x = TotalCharges,
           fill = Churn)) +
  geom_density(alpha = 0.4) +
  labs(title = "")

ggplot(dados,
       aes(x = SeniorCitizen,
           fill = Churn)) +
  geom_density(alpha = 0.4) +
  labs(title = "")
```

Transformação de dados

```
#transformação de dados (qualitativo para quantitativo) - variáveis binárias
dados_quant <- dados_1
colnames (dados_quant)
dados_quant %>%
    mutate(customerID = NULL,
           PhoneService = as.factor(ifelse(PhoneService == "Yes",1,0)),
           Partner = as.factor(ifelse(Partner == "Yes",1,0)),
           Dependents = as.factor(ifelse(Dependents == "Yes",1,0)),
           PaperlessBilling = as.factor(ifelse(TechSupport == "Yes",1,0)),
           Dependents = as.factor(ifelse(Dependents == "Yes",1,0)),
           Churn = as.factor(ifelse(Churn == "Yes",1,0)),
            ) -> dados_quant
glimpse(dados_quant)
```

Utilização de variáveis dummy

```
#utilização de variáveis dummy
dummy_dados <- dados_quant %>% select(InternetService, Contract, PaymentMethod,
                                      MultipleLines, OnlineBackup, OnlineSecurity,
                                      DeviceProtection, StreamingTV,
                                      StreamingMovies, TechSupport)

dummy <- dummyVars(~ ., data = dummy_dados, fullRank = T)

dummy_dados <- predict(dummy, dummy_dados)
dados_quant1 <- bind_cols(dados_quant,dummy_dados)

dados_quant1 %>%
    rename( InternetService.Fiberoptic =`InternetService.Fiber optic`,
            Contract.Oneyear = `Contract.One year`,
            Contract.Twoyear = `Contract.Two year`,
            PaymentMethod.Creditcard = `PaymentMethod.Credit card (automatic)`,
            PaymentMethod.Electronic = `PaymentMethod.Electronic check`,
            PaymentMethod.Mailed = `PaymentMethod.Mailed check`,
            MultipleLines.NoService = `MultipleLines.No phone service`,
            OnlineBackup.NoService = `OnlineBackup.No internet service` ,
            OnlineSecurity.NoService = `OnlineSecurity.No internet service`,
            DeviceProtection.NoService = `DeviceProtection.No internet service`,
            StreamingTV.NoService = `StreamingTV.No internet service`,
            StreamingMovies.NoService = `StreamingMovies.No internet service`,
            TechSupport.NoService = `TechSupport.No internet service`) -> dados_quant1
```

Exclusão variáveis qualitativas

```
#exclusao de variaveis qualitativas
dados_quant1 %>%
    mutate(gender = NULL,
           InternetService = NULL,
           Contract = NULL,
           PaymentMethod = NULL,
           PaymentMOnlineBackupethod = NULL,
           OnlineSecurity = NULL,
           StreamingTV = NULL,
           StreamingMovies = NULL,
           MultipleLines = NULL,
           OnlineBackup = NULL,
           DeviceProtection = NULL,
           TechSupport = NULL) -> dados_quant1

glimpse(dados_quant1)
```

Utilização de variáveis dummy

Lembrado:

Em estatística, particularmente na análise de regressão, uma **variável Dummy** é aquela que toma o valor de "zero" ou "um" indicando a ausência ou presença de qualidades ou atributos. Essas **variáveis** são usadas como dispositivos para classificar dados em categorias mutuamente exclusivas.

Correlação dos dados

```
#Correlação dos dados
# todas que estão com <fct> transformar para numérico
dados_quant1 %>%
    mutate( Partner = as.numeric(Partner),
            Dependents = as.numeric(Dependents),
            PhoneService = as.numeric(PhoneService),
            PaperlessBilling = as.numeric(PaperlessBilling),
            Churn = as.numeric(Churn)) -> dados_num
glimpse(dados_num)

corrplot(cor(dados_num), method = "circle")
```

```
dados_num %>%
    mutate( Dependents = NULL) -> dados_num_1
#glimpse(dados_num_1)

corrplot(cor(dados_num_1),order = "hclust", method = "circle")
```

```
dados_num_1 %>%
    mutate(StreamingMovies.NoService = NULL,
            StreamingTV.NoService = NULL,
            DeviceProtection.NoService = NULL ,
            OnlineSecurity.NoService = NULL,
            InternetService.No = NULL,
            OnlineBackup.NoService = NULL ) -> dados_num_1
```

Balanceamento da base

```
#balanceamento de base de dados
dados_quant1 %>%
    select(Churn) %>%
    group_by(Churn) %>%
    summarise(n = n())
dados_quant1 %>%
    filter(Churn == 0) %>%
    sample_n(1869) -> dados_quant_0
dados_quant1 %>%
  filter(Churn == 1)  -> dados_quant_1
dados_quant_balanc <- bind_rows(dados_quant_0,dados_quant_1)
dados_num_1 %>%
    select(Churn) %>%
    group_by(Churn) %>%
    summarise(n = n())
dados_num_1 %>%
    mutate(Churn = ifelse(Churn == 1,0,1))%>%
    filter(Churn == 0) %>%
    sample_n(1869) -> dados_num_0
dados_num_1 %>%
  mutate(Churn = ifelse(Churn == 1,0,1))%>%
  filter(Churn == 1)  -> dados_num_2
dados_num_balanc <- bind_rows(dados_num_2,dados_num_0)
dados_num_balanc %>%
    select(Churn) %>%
    group_by(Churn) %>%
    summarise(n = n())
dados_quant_balanc %>%
    select(Churn) %>%
    group_by(Churn) %>%
    summarise(n = n())
```

Regressão por Stepwize

```
#stepwize
#verificar quais as variaveis agregam ao modelo
modelo_teste1 <- glm(data = dados_num_balanc, Churn  ~ .,family=binomial)
step(modelo_teste1)
summary(modelo_teste1)
```

https://pt.wikipedia.org/wiki/Regress%C3%A3o_por_stepwise

Validação

```
# Validação cruzada - avalia a capacidade de generalização do modelo
#O método de validação cruzada denominado k-fold consiste em dividir o conjunto
#total de dados em k subconjuntos mutuamente exclusivos do mesmo tamanho e,
#a partir daí, um subconjunto é utilizado para teste e os k-1 restantes são
#utilizados para estimação dos parâmetros, fazendo-se o cálculo da acurácia do modelo.
#Este processo é realizado k vezes alternando de forma circular o subconjunto de teste.

trainIndex <- createDataPartition(dados_num_balanc$Churn, p = .8, #80 por cento
                                  list = FALSE,
                                  times = 1)


vivoTrain <- dados_num_balanc[ trainIndex,]
vivoTest  <- dados_num_balanc[-trainIndex,]
```

Treinando a base

```
#treinando a base
set.seed(150)

glm_model = train(Churn ~ SeniorCitizen + tenure + PaperlessBilling +
                    MonthlyCharges + TotalCharges + InternetService.Fiberoptic +
                    Contract.Contract.Oneyear + Contract.Twoyear + PaymentMethod.Electronic +
                    MultipleLines.Yes + OnlineSecurity.Yes + StreamingTV.Yes +
                    StreamingMovies.Yes + TechSupport.NoService,
                    data= vivoTrain,
                    method= "glm",
                    trControl = trainControl(method = "cv"),
                    family = "binomial")

summary(glm_model)
varImp(glm_model)
```

Testando a base

```
#testando a base

reg_log_pred <- predict(glm_model,vivoTrain)
reg_log_pred1 <- data.frame(reg_log_pred)
reg_log_pred1$reg_log_pred <- as.factor(ifelse(reg_log_pred1$reg_log_pred >= 0.5,1,0))
reg_log_pred1$reg_log_pred <- as.factor(ifelse(reg_log_pred1$reg_log_pred == 1,"evadido","cliente"))
vivoTrain$Churn <- as.factor(vivoTrain$Churn)
vivoTrain$Churn <- as.factor(ifelse(vivoTrain$Churn == 1,"evadido","cliente"))
glimpse(reg_log_pred1)

matrix_reg <-
 confusionMatrix(data = reg_log_pred1$reg_log_pred, reference = vivoTrain$Churn, positive = "evadido")
matrix_reg$table

metricas <- data.frame(matrix_reg$byClass)
```

Excluindo o ID do Ciente

```
glimpse(dados_1)
#excluir o ID
#transformar em qualitativa as quantitativas
#Adicionar classes as variáveis numericas

dados_1 %>%
        mutate (SeniorCitizen = as.factor(ifelse(SeniorCitizen == 1, "Yes", "No")),
                customerID = NULL) -> dados_quali
```

Criando as classes

```
#criando classes para as variaveis quantitativas
tenure <- summary(dados_quali$tenure)
tenure
```

Criando os Quartis

```
TotalCharges <- summary(dados_quali$TotalCharges)

min_TotalCharges <- TotalCharges[[1]]-5
q1_TotalCharges <- TotalCharges[[2]]
q2_TotalCharges <- TotalCharges[[3]]
q3_TotalCharges <- TotalCharges[[5]]
max_TotalCharges <- TotalCharges[[6]]+5


dados_quali %>%
  mutate(TotalCharges = cut(TotalCharges, breaks = c(min_TotalCharges,
                                                     q1_TotalCharges,
                                                     q2_TotalCharges,
                                                     q3_TotalCharges,
                                                     max_TotalCharges))) -> dados_quali

summary(dados_quali$TotalCharges)
```

Balanceamento da Base

```
#balanceamento da base
glimpse(dados_quali)
dados_quali %>%
    select(Churn) %>%
    group_by(Churn) %>%
    summarise(n = n())

dados_quali %>%
    filter(Churn == "No") %>%
    sample_n(1869) -> dados_quali_no

dados_quali %>%
  filter(Churn == "Yes")  -> dados_quali_yes

dados_quali_balanc <- bind_rows(dados_quali_no,dados_quali_yes)

dados_quali_balanc %>%
    select(Churn) %>%
    group_by(Churn) %>%
    summarise(n = n())
```

Separação da Base – Teste e Treino

```
# #separação da base para teste e treino
trainIndex_quali <- createDataPartition(dados_quali_balanc$Churn, p = .8,
                                        list = FALSE,
                                        times = 1)


vivoTrain_quali <- dados_quali_balanc[ trainIndex_quali,]
vivoTest_quali <- dados_quali_balanc[-trainIndex_quali,]
```

Treinamento – Árvore de Decisão

```
#treinando o modelo de árvore de decisão

vivo.tree = train(Churn ~ .,
                  data= vivoTrain_quali,
                  method="rpart",
                  trControl = trainControl(method = "cv"))


vivo.tree


fancyRpartPlot(vivo.tree$finalModel)

vivo.pred = predict(vivo.tree, newdata = vivoTrain_quali)

matrix_tree <- confusionMatrix(data = vivo.pred, reference = vivoTrain_quali$Churn, positive = "Yes")
matrix_tree$table

metricas_tree <- data.frame(matrix_tree$byClass)
metricas_tree
```

PUC Minas
Virtual