PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data

Francisco Hítalo de Sousa Luz

APLICAÇÃO DE MODELOS DE PREVISÃO DE ÓBITOS POR COVID-19 EM CASOS DE FORTALEZA-CE

Belo Horizonte
2021

Francisco Hítalo de Sousa Luz

APLICAÇÃO DE MODELOS DE PREVISÃO DE ÓBITOS POR COVID-19 EM CASOS DE FORTALEZA-CE

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Ciência de Dados e Big Data como requisito parcial à obtenção do título de especialista.

Belo Horizonte
2021

SUMÁRIO

1. Introdução

1.1. Contextualização

A COVID-19, que consiste em uma síndrome respiratória aguda grave, foi identificada em Wuhan, província da China, que se disseminou rapidamente, se tornando pandemia em poucos meses, foi identificada em janeiro de 2020 e em março do mesmo ano considerada uma pandemia, mudando a forma de viver de todo um povo, com o uso de mascaras, fechamento de comércios, escolas, e demais serviços, com a disseminação da necessidade do distanciamento social. Desde que foi descoberta a COVID-19 se espalhou de forma preocupante por todo o planeta, até agosto de 2021 temos, 20.528.09 casos confirmados no Brasil, e 573.511 óbitos confirmados. No mundo passam de 210 milhões de casos confirmados e mais de 4 milhões de óbitos, em destaque o Estados Unidos da América (EUA), com mais de 37 milhões de casos. Diversos esforços foram feitos por pesquisadores de todo o mundo para encontrar uma maneira de diminuir o impacto causado pela doença, na busca de remédios, e mais importante, no desenvolvimento de vacinas, em que, temos no Brasil, até agosto de 2021, 121.263.020 pessoas que receberam pelo menos a primeira dose da vacina.

Diante de tantos números, e de uma doença que gerou inúmeros danos, a técnica de machine learning pode ser usada para criação de modelos de previsão auxiliando assim o desenvolvimento de planos de ações. Analisaremos a disseminação da doença no município de Fortaleza, capital do Estado do Ceará, que possui, população estimada de 2.703.391 pessoas, possuindo densidade demográfica de 7.786,44 hab/km2. Portanto, utilizando técnicas de machine learning e com o uso dos dados retirados do SUS (2021), até o dia 30 de abril no município de Fortaleza, teremos como foco principal um modelo de previsão de óbitos.

1.2. O problema proposto

Nessa seção serão respondidas perguntas referentes ao problema pontuado anteriormente.

1.2.1. Por que esse problema é importante?

Dado o alto contagio, internações e por sequência o óbito, e o número de vagas de internações serem finitos acredita-se ser necessário todo auxilio na hora de decidir quem ficará com a vaga e tem mais prioridade.

1.2.2. De quem são os dados analisados?

Os dados são retirados do Integra SUS e também da prefeitura de Fortaleza.

1.2.3. Quais os objetivos com essa análise?

Acredita-se que conseguindo prever os casos com maior probabilidade de o quadro evoluir para óbito planos de ações poderá ser desenvolvido dando prioridade a pacientes que indicarem mais riscos.

1.2.4. Quais os aspectos geográficos e logísticos de sua análise.

A análise se dar em casos de Covid-19 registrados em Fortaleza - CE.

1.2.5. Qual o período está sendo analisado?

O período analisado será de março de 2020 até novembro do mesmo ano.

1.3. Objetivos

Este trabalho tem como objetivo apresentar modelos de previsão de óbitos por Covid-19 e compara-los.

2. Coleta de Dados

Os dados referentes a Covid-19 foram obtidos a partir da plataforma do Governo do Estado do Ceará, Integra SUS, e contém dados do período de fevereiro de 2020 a novembro de 2020, na Capital Fortaleza. A base de dados foi extraída no formato CSV, tendo em cada linha a notificação de um paciente, e as seguintes colunas: https://integrasus.saude.ce.gov.br/#/indicadores/indicadores-coronavirus/coronavirus-ceara

Nome da coluna/campo	Descrição	Tipo
bairro	Nomes dos bairros de Fortaleza - CE	String
sexoPaciente	Gênero do paciente	String
idadePaciente	Idade do paciente	Numeric
resultadoFinalExame	Resultado do exame de Covid-19	String
profissionalSaude	Descrição caso o paciente seja profissional da saúde	String
racaCorPaciente	Raça do paciente	String
comorbidadePuerperaSivep	Paciente com comorbidade puérpera	String
comorbidadeCardiovascularSivep	Paciente com comorbidade cardiovascular	String
comorbidadeHematologiaSivep	Paciente com comorbidade hematologia	String
comorbidadeSindromeDownSivep	Paciente com comorbidade síndrome down	String
comorbidadeAsmaSivep	Paciente com comorbidade asma	String
comorbidadeDiabetesSivep	Paciente com comorbidade diabetes	String

comorbidadeNeurologiaSivep	Paciente com comorbidade neurologia	String
comorbidadePneumopatiaSivep	Paciente com comorbidade pneumopatia	String
comorbidadeImunodeficienciaSivep	Paciente com comorbidade imunodeficiência	String
comorbidadeRenalSivep	Paciente com comorbidade renal	String
comorbidadeObesidadeSivep	Paciente com comorbidade obesidade	String
comorbidadeHiv	Paciente com comorbidade HIV	String
comorbidadeNeoplasias	Paciente com comorbidade neoplasias	String

Os dados referentes a cidade de Fortaleza e seus bairros, foram obtidos no site de mapas da capital, contendo as informações de IDH (Índice de desenvolvimento humano), número de habitantes por bairro e média de habitantes por casa. Os dados são referentes ao último censo de 2010, foram extraídos em CSV, sendo as informações organizadas por bairro, possibilitando fazer o incremento da base de dados da Covid-19.
https://mapas.fortaleza.ce.gov.br/#/

Nome da coluna/campo	Descrição	Tipo
Media_habitantes_por_casa	Média de habitantes por casa	Numeric
IDH2010	índice de desenvolvimento humano de acordo com o censo de 2010	Numeric
qtd_habitantes	Quantidade de habitantes por bairro	Numeric

3. Processamento/Tratamento de Dados

Nessa seção será apresentado o tratamento feito no dataset escolhido para ser utilizado. Inicialmente a base de dados tinha 44.098 registros contendo 899 linhas com dados faltantes e zero duplicadas. O tratamento feito sobre os dados faltantes pela quantidade ser de apenas 2% foi a exclusão dos registros. Resultando assim em uma base com 43.199 registros.

4. Análise e Exploração dos Dados

Nesta seção o intuito é conhecer os dados através da análise exploratória.

1. **Quantidade de bairros presentes na base:**

   São 117 de 121 bairros ao todo.

2. **Quantidade e proporção dos sexos presentes na base:**

Sexo	Quantidade	Proporção
Feminino	22.974	53,20%
Masculino	20.225	46,80%

3. **Distribuição das idades:**

Mínimo	1° Quartil	Mediana	Média	3° Quartil	Máximo
0	34	46	47,92	61	364

4. **Profissional de saúde:**

Profissional de saúde	Quantidade	Proporção
Sim	6.000	86,10%
Não	37.199	13,9%

5. **Raça/Cor do Paciente:**

Raça/Cor	Quantidade	Proporção
Amarela	2.391	5,53%
Branca	4.694	10,90%
Ignorado	14.970	34,70%
Indígena	41	0,09%
Parda	20.468	47,40%
Preta	635	1,47%

6. Comorbidades:

Puérpera	Quantidade	Proporção
Sim	29	0,06%
Não	43.170	99,94%

Cardiovascular	Quantidade	Proporção
Sim	2060	4,80%
Não	41.139	95,20%

Hematologia	Quantidade	Proporção
Sim	33	0,07%
Não	43.166	99,93%

Síndrome de Down	Quantidade	Proporção
Sim	10	0,02%
Não	43.189	99,98%

Asma	Quantidade	Proporção
Sim	107	0,20%
Não	43.092	99,80%

Diabetes	Quantidade	Proporção
Sim	1764	4,10%
Não	41.435	95,90%

Neurologia	Quantidade	Proporção
Sim	242	0,60%
Não	42.957	99,40%

Pneumopatia	Quantidade	Proporção
Sim	155	0,35%
Não	43.044	99,65%

Imunodeficiência	Quantidade	Proporção
Sim	153	0,35%
Não	43.046	99,65%

Renal	Quantidade	Proporção
Sim	259	0,60%
Não	42.940	99,40%

Obesidade	Quantidade	Proporção
Sim	129	0,29%
Não	43.070	99,71%

HIV	Quantidade	Proporção
Sim	0	0%
Não	43.199	100%

Neoplasias	Quantidade	Proporção
Sim	0	0%
Não	43.199	100%

7. Distribuição da média de habitantes por casa:

Mínimo	1° Quartil	Mediana	Média	3° Quartil	Máximo
2,87	3,31	3,43	3,40	3,55	4,02

8. Distribuição do IDH dos bairros:

Mínimo	1° Quartil	Mediana	Média	3° Quartil	Máximo
0,11	0,25	0,36	0,41	0,50	0,95

9. Distribuição do número de habitantes por bairro:

Mínimo	1° Quartil	Mediana	Média	3° Quartil	Máximo
1.470	17.128	29.271	31.157	43.218	79.346

5. Criação de Modelos de Machine Learning e Regressão Logística

5.1 Modelo Logístico Múltiplo

Nesta Subseção são apresentados os resultados da regressão logística múltipla com as variáveis contidas na Tabela. Primeiro a base foi amostrada todos os defaults iguais a 1 e a quantidades de defaults iguais a 0 é o dobro da quantidade de 1. Em seguida dividiu-se a base em treino (70%) e teste (30%).

Na execução e construção, o modelo foi ajusto na base de treino, aceitando variáveis com o valor-p ≤ 0, 05. Ao construir o modelo, foram obtidos os seguintes resultados apresentados.

Variáveis	Estimativa	Erro-Padrão	Estatística Z	Valor-p
Idade do Paciente	0,07	< 0,01	34,93	< 0,01
Cardiovascular	1,81	0,12	14,66	< 0,01
Asma	1,46	0,49	2,93	< 0,01
Neurologia	2,22	0,45	4,94	< 0,01
Pneumopatia	1,59	0,42	3,75	< 0,01
Imunodeficiência	3,02	0,43	6,99	< 0,01
Renal	1,85	0,38	4,80	< 0,01
Obesidade	2,50	0,44	5,59	< 0,01
Puérpera	-0,03	1,00	-0,03	0,97
Hematologia	1,91	1,11	1,71	0,08
Síndrome Down	2,86	1,55	1,84	0,06
Diabetes	2,16	0,13	15,67	< 0,01
Média habitantes	-0,09	0,28	-0,34	0,72
IDH	-2,11	0,31	-6,79	< 0,01
Qtd. habitantes	< 0.01	< 0,01	-2,35	0,01

Nota-se que em um primeiro momento com todas as variáveis possíveis as variáveis comorbidade Puérpera, Hematologia, Síndrome de Down e média de habitantes por casa deixaram de ser significativas na presença de outras, para valor-p > 0, 05. Com isso retira-se as variáveis não significativas e ajusta o modelo novamente. Obtêm-se os seguintes resultados.

Variáveis	Estimativa	Erro-Padrão	Estatística Z	Valor-p
Idade do Paciente	0,07	< 0,01	34,95	< 0,01
Cardiovascular	1,82	0,12	14,69	< 0,01
Asma	1,46	0,49	2,92	< 0,01
Neurologia	2,33	0,45	5,17	< 0,01
Pneumopatia	1,59	0,42	3,75	< 0,01
Imunodeficiência	3,11	0,43	7,28	< 0,01
Renal	1,90	0,39	4,86	< 0,01
Obesidade	2,50	0,44	5,61	< 0,01
Diabetes	2,16	0,13	15,67	< 0,01
IDH	-2,03	0,18	-10,76	< 0,01
Qtd. habitantes	< 0.01	< 0,01	-2,37	0,01

Os valores para o valor-p estão significativos, indicando que as variáveis tem influência na resposta. Indicando que se pode dar sequência a análise do modelo ajustado como a seguir:

$$\mathbb{P}(S) = \frac{1}{1 + e^{-(-4{,}47+0{,}07(X1)+1{,}82(X2)+1{,}45(X3)+2{,}16(X4)+2{,}33(X5)+1{,}58(X6)+3{,}11(X7)+1{,}90(X8)+2{,}50(X9)-2{,}03(X10)-0{,}001(X11)}}$$

Em que:

X1: Idade do Paciente X2: Cardiovascular X3: Asma

X4: Diabetes X5: Neurologia X6: Pneumopatia

X7: Imunodeficiência X8: Renal X9: Obesidade

X10: IDH X11: Qtd. Habitantes

5.1.1 Avaliação treino

	Referência	**Sim**	**Não**
Predição	**Sim**	3.134	789
	Não	530	4.637

Acurácia: 83,7% **Sensibilidade:** 80,11% **Especificidade:** 85,46%

5.1.2 Avaliação teste

	Referência	**Sim**	**Não**
Predição	**Sim**	966	353
	Não	222	1.925

Acurácia: 83,41% **Sensibilidade:** 81,31% **Especificidade:** 84,50%

5.2 Random Forest

5.2.1 Avaliação treino

	Referência	Sim	Não
Predição	Sim	2.330	160
	Não	334	5.266

Acurácia: 93,89% **Sensibilidade:** 87,46% **Especificidade:** 97,05%

5.2.2 Avaliação teste

	Referência	Sim	Não
Predição	Sim	878	307
	Não	310	1.971

Acurácia: 82,2% **Sensibilidade:** 73,91% **Especificidade:** 86,52%

5.3 XGBoost

5.3.1 Avaliação treino

	Referência	Sim	Não
Predição	Sim	1.940	480
	Não	724	4.946

Acurácia: 85,12% **Sensibilidade:** 72,82% **Especificidade:** 91,15%

5.3.2 Avaliação teste

	Referência	Sim	Não
Predição	**Sim**	867	227
	Não	321	2.051

Acurácia: 84,19% **Sensibilidade:** 72,98% **Especificidade:** 90,04%

6. Interpretação dos Resultados

Os modelos foram ajustados de maneira simplista com o intuito já em um primeiro instante fornecer indicativos dos melhores caminhos a ser seguidos. Observando os resultados de treino e teste para as três metodologias analisadas acredita-se que os modelos, logístico e o XGBoost tiveram ajustes satisfatórios tanto no treino como no teste mantiveram a qualidade, considerando a acurácia, a sensibilidade e a especificidade dos modelos. Acredita-se que pela capacidade de interpretação dos coeficientes do modelo logístico o mesmo deverá ser escolhido para dar continuidade nos trabalhos, pois, interpretar os coeficientes é de fundamental importância para entender os fatores que mais implicam para ocorrência do evento de interesse no estudo, aqui o óbito por Covid-19.

7. Apresentação dos Resultados

8. Links

Link para o vídeo: https://youtu.be/Cq-LAegAhUo

Link para o repositório: https://github.com/hitalusousa/tcc-predicao-obitos-covid-19

APÊNDICE

Programação/Scripts

Código desenvolvido em R

```
library(readxl)
library(dplyr)
library(caret)
library(forecast)
options(scipen = 9999)
# base de casos
base_covid = read_excel("D:/Hitalo/Área de Trabalho/pos/tcc/base/casos_covid09_11_20_fortal_fechado 19jan2021.xlsx",
                        col_types = c("text", "text", "text",
                                      "numeric", "text", "text", "text",
                                      "text", "numeric", "numeric", "numeric",
                                      "numeric", "numeric", "text", "text",
                                      "numeric", "numeric", "text", "text",
                                      "text", "text", "text", "text", "text",
                                      "text", "text", "text", "text", "text",
                                      "text", "text", "text", "text", "text",
                                      "text", "numeric", "text", "text",
                                      "text", "text"))

#base info adicionais
inform_adicionais <- read_excel("D:/Hitalo/Área de Trabalho/pos/tcc/base/inform_adicionais.xlsx")


base_covid_tratada = base_covid %>% dplyr::mutate(default = case_when(obitoConfirmado == "True" ~ "1",TRUE ~ "0"),
                                                  comorbidadePuerperaSivep = case_when(comorbidadePuerperaSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeCardiovascularSivep = case_when(comorbidadeCardiovascularSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeHematologiaSivep = case_when(comorbidadeHematologiaSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeSindromeDownSivep = case_when(comorbidadeSindromeDownSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeAsmaSivep = case_when(comorbidadeAsmaSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeDiabetesSivep = case_when(comorbidadeDiabetesSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeNeurologiaSivep = case_when(comorbidadeNeurologiaSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadePneumopatiaSivep = case_when(comorbidadePneumopatiaSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeImunodeficienciaSivep = case_when(comorbidadeImunodeficienciaSivep == "Sim" ~ "1",TRUE ~
"0"),
                                                  comorbidadeRenalSivep = case_when(comorbidadeRenalSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeObesidadeSivep = case_when(comorbidadeObesidadeSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeHiv = case_when(comorbidadeObesidadeSivep == "Sim" ~ "1",TRUE ~ "0"),
                                                  comorbidadeNeoplasias = case_when(comorbidadeNeoplasias == "Sim" ~ "1",TRUE ~ "0"),
                                                  racaCorPaciente = case_when(is.na(racaCorPaciente) ~ "Ignorado",TRUE ~ racaCorPaciente),
                                                  resultadoFinalExame = case_when(resultadoFinalExame == "Positivo" ~ "1",TRUE ~ "0"),
```

```
                        bairro = `bairro 1`) %>%

dplyr::select(default,bairro,sexoPaciente,idadePaciente,resultadoFinalExame,profissionalSaude,racaCorPaciente,starts_with("comorbidade
"))

base_default = base_covid_tratada %>% dplyr::left_join(inform_adicionais, by = c("bairro" = "Bairros")) %>%
dplyr::filter(resultadoFinalExame == "1") %>% na.omit()
dim(base_default)

table(base_default$default)

str(base_default)

names(base_default)
# exploratoria ------------------------------------------------------------

### quantidade de bairros
length(unique(base_default$bairro))

### Proporção de Sexo

base_default %>% dplyr::group_by(sexoPaciente) %>%
  dplyr::summarise(qtd = n(),
            prop = n()/nrow(base_default))

### sumario das idades

summary(base_default$idadePaciente)

### profissional de saude

base_default %>% dplyr::group_by(profissionalSaude) %>%
  dplyr::summarise(qtd = n(),
            prop = n()/nrow(base_default))

### raça do paciente

base_default %>% dplyr::group_by(racaCorPaciente) %>%
  dplyr::summarise(qtd = n(),
            prop = n()/nrow(base_default))

### Comorbidades

base_default %>% dplyr::group_by(comorbidadeNeoplasias) %>%
  dplyr::summarise(qtd = n(),
            prop = n()/nrow(base_default))

### Media de habitantes
```

```
summary(base_default$Media_habitantes_por_casa)

#### IDH 2012

summary(base_default$IDH2010)

#### Quantidade de habitantes

summary(base_default$qtd_habitantes)

# modelo logistico ----------------------------------------------------------



base_0 = base_default %>% dplyr::filter(default == "0")

base_1 = base_default %>% dplyr::filter(default == "1")

amostra_0 = sample(1:length(base_0$default),length(base_1$default)*2)

base_0_fim = base_0[amostra_0,]

base_desenvolvimento_ = rbind(base_1,base_0_fim) %>% as.data.frame() %>% mutate_at(.vars = c("default","comorbidadePuerperaSivep",
                                                                                              "comorbidadeCardiovascularSivep","comorbidadeHematologiaSivep",
                                                                                              "comorbidadeSindromeDownSivep","comorbidadeAsmaSivep",
                                                                                              "comorbidadeDiabetesSivep","comorbidadeNeurologiaSivep",
                                                                                              "comorbidadePneumopatiaSivep","comorbidadeImunodeficienciaSivep",
                                                                                              "comorbidadeRenalSivep","comorbidadeObesidadeSivep"),
                                                                                            as.factor)
base_desenvolvimento = base_desenvolvimento_[,!apply(base_desenvolvimento_,2,is.constant)]
set.seed(123654)

split1<- sample(c(rep(0, 0.7 * nrow(base_desenvolvimento)), rep(1, 0.3 * nrow(base_desenvolvimento))))

train <- base_desenvolvimento[split1 == 0, ]

test <- base_desenvolvimento[split1== 1, ]

chisq.test(base_default$default,base_default$sexoPaciente)

mod = glm(data = base_default,formula = as.factor(default) ~ idadePaciente,family = "binomial")
summary(mod)

chisq.test(base_default$default,base_default$resultadoFinalExame)

chisq.test(base_default$default,base_default$profissionalSaude)
```

```
chisq.test(base_default$default,base_default$comorbidadeObesidadeSivep)


###### Preparando base treino


base_default_treino = train


### Treinando modelo

mod1 = glm(data = base_default_treino,formula = default ~ .,family = "binomial")

summary(mod1)

base_retreino = base_default_treino %>% dplyr::select(-c(comorbidadePuerperaSivep,comorbidadeHematologiaSivep,
                                        comorbidadeSindromeDownSivep,
                                        Media_habitantes_por_casa))

mod2 = glm(data = base_retreino,formula = default ~ .,family = "binomial")

summary(mod2)

#### predizendo o treino

predito = predict(mod2, base_retreino[,-1], type="response")

pred_class <- as.factor(ifelse(predito > .35, "1", "0"))

cmtrx <- confusionMatrix(pred_class,base_retreino$default,positive = "1");cmtrx

library(ROSE)

roc.curve(response = base_retreino$default,predicted = pred_class)

library(DescTools)

DescTools::PseudoR2(mod2)

######### Teste

base_teste = test %>% dplyr::select(-c(comorbidadePuerperaSivep,comorbidadeHematologiaSivep,
                                        comorbidadeSindromeDownSivep,
                                        Media_habitantes_por_casa)) %>%
  dplyr::select(names(mod2$data))

predito = predict(mod2, base_teste[,-1], type="response")
```

```
exp(mod2$coefficients)

pred_class <- as.factor(ifelse(predito > .35, "1", "0"))

cmtrx <- confusionMatrix(pred_class,base_teste$default,positive = "1");cmtrx
```

```
library(randomForestSRC)
library(caret)

data(train, package = "randomForestSRC")
breast.obj <- rfsrc(default ~ ., data = train, nsplit = 10)
breast.pred <- predict(breast.obj, train)

#### Treino
pred_class <- as.factor(ifelse(breast.pred$predicted[,2] > .5, "1", "0"))

cmtrx <- confusionMatrix(pred_class,train$default,positive = "1");cmtrx

### teste
breast.pred <- predict(breast.obj, test)

pred_class <- as.factor(ifelse(breast.pred$predicted[,2] > .5, "1", "0"))

cmtrx <- confusionMatrix(pred_class,test$default,positive = "1");cmtrx
```

```
control <- trainControl(method='cv',
                        number=5)

set.seed(123654)
rf_random <- train(default ~ .,
                   data = train,
                   method = 'xgboost',
                   metric = 'Accuracy',
                   tuneLength  =  10,
                   trControl = control)
print(rf_random)

#### Treino

test_predict <- predict(rf_random, train)

pred_class <- as.factor(ifelse(test_predict > .5, "1", "0"))
```

```
cmtrx <- confusionMatrix(test_predict,train$default,positive = "1");cmtrx

##### Teste

test_predict <- predict(rf_random, test)

pred_class <- as.factor(ifelse(test_predict > .5, "1", "0"))

cmtrx <- confusionMatrix(test_predict,test$default,positive = "1");cmtrx
```