MEDICAL APPOINTMENTS

PREDIZER SE UM DETERMINADO PACIENTE IRÁ FALTAR OU NÃO A UMA CONSULTA

Motivação

Quem somos

- Felipe Pereira - Engenheiro de Dados
- Alexandre Guidin - Analista de Sistemas
- Anderson Rocha - Analista de Sistemas

ANÁLISE DE DADOS

Dimensionalidade:

- 110527 registros
- 14 atributos

Qualidade dos dados:

- Nenhum atributo nulo
- Nenhum atributo corrompido

Balanceamento do target:

- 80% de pessoas que comparecem
- 20% de pessoas que faltam

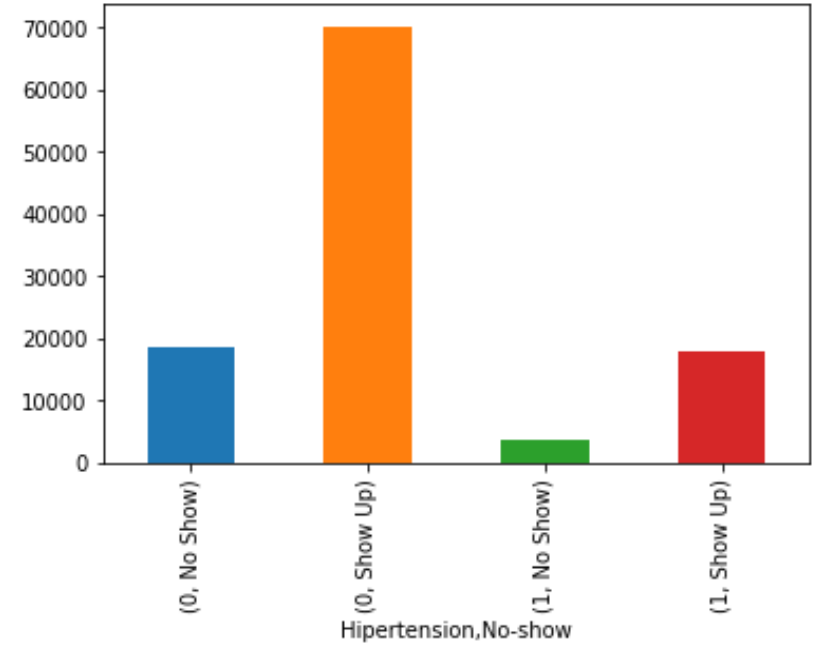
ANÁLISE DE DADOS

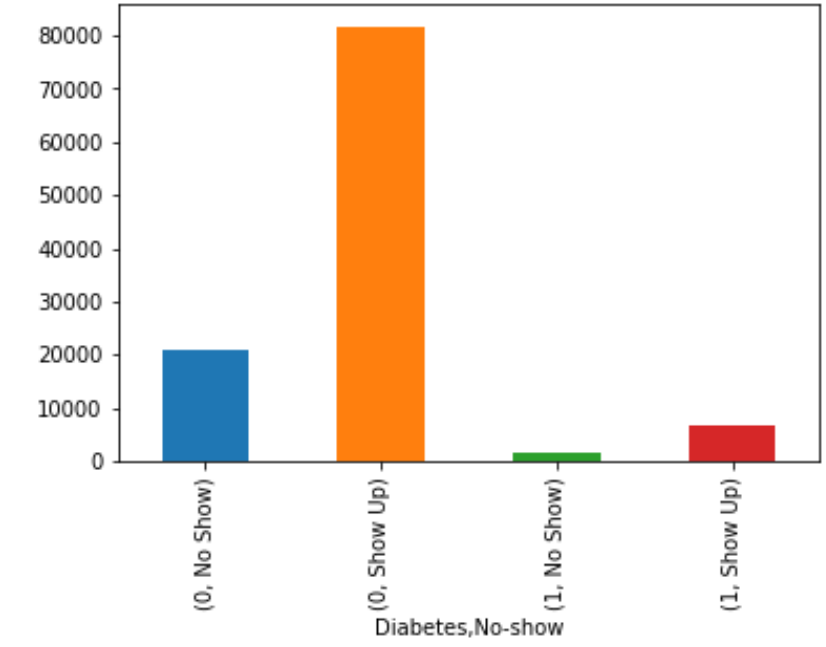
Correlação

	PatientId	AppointmentID	Age	Scholarship	Hipertension	Diabetes	Alcoholism	Handcap	SMS_received	No-show
PatientId	1.000000	0.004039	-0.004139	-0.002880	-0.006441	0.001605	0.011011	-0.007916	-0.009749	-0.001461
AppointmentID	0.004039	1.000000	-0.019126	0.022615	0.012752	0.022628	0.032944	0.014106	-0.256618	-0.162602
Age	-0.004139	-0.019126	1.000000	-0.092457	0.504586	0.292391	0.095811	0.078033	0.012643	-0.060319
Scholarship	-0.002880	0.022615	-0.092457	1.000000	-0.019729	-0.024894	0.035022	-0.008586	0.001194	0.029135
Hipertension	-0.006441	0.012752	0.504586	-0.019729	1.000000	0.433086	0.087971	0.080083	-0.006267	-0.035701
Diabetes	0.001605	0.022628	0.292391	-0.024894	0.433086	1.000000	0.018474	0.057530	-0.014550	-0.015180
Alcoholism	0.011011	0.032944	0.095811	0.035022	0.087971	0.018474	1.000000	0.004648	-0.026147	-0.000196
Handcap	-0.007916	0.014106	0.078033	-0.008586	0.080083	0.057530	0.004648	1.000000	-0.024161	-0.006076
SMS_received	-0.009749	-0.256618	0.012643	0.001194	-0.006267	-0.014550	-0.026147	-0.024161	1.000000	0.126431
No-show	-0.001461	-0.162602	-0.060319	0.029135	-0.035701	-0.015180	-0.000196	-0.006076	0.126431	1.000000

ANÁLISE DE DADOS

Hipertensão e Diabetes





TRANSFORMAÇÃO DE DADOS

- Extração do mês da data agendada;
- Extração do dia da data agendada;
- Extração e cálculo de diferença de dias entre o agendamento e a consulta;
- Drop de colunas;
- Mapeamento de variáveis categóricas para features discretas;
- Normalização.

Preparação dos Datasets

```
#Separação do dataset
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix

X = data.drop('Noshow',1)
y = data.Noshow

#Separação de conjunto de testes
X_model, X_test, y_model, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

#Separação de conjunto de validação
X_train, X_val, y_train, y_val = train_test_split(X_model, y_model, test_size=0.2, random_state=0)
```

```
#Imbalancement treatment com over sampling
from imblearn.over_sampling import SMOTE

balancer = SMOTE(kind='regular')
x_resampled, y_resampled = balancer.fit_sample(X_train, y_train)

print('Normal Data: ', collections.Counter(y_train))
print('Resampled: ', collections.Counter(y_resampled))
```

```
Normal Data:  Counter({0: 28707, 1: 6115})
Resampled:  Counter({0: 28707, 1: 28707})
```

Preparação dos Modelos

```
#Criação dos modelos
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier

rf = RandomForestClassifier(n_estimators=100, criterion='entropy', max_features='auto', random_state=0)
lr = LogisticRegression(random_state=0, penalty='l2', C=1, fit_intercept=True, solver='liblinear')
dt = DecisionTreeClassifier(random_state=0, criterion='entropy', splitter='best')

algs = []
algs.append(rf)
algs.append(lr)
algs.append(dt)

rf1 = RandomForestClassifier(n_estimators=100, criterion='entropy', max_features='auto', random_state=0)
lr1 = LogisticRegression(random_state=0, penalty='l2', C=1, fit_intercept=True, solver='liblinear')
dt1 = DecisionTreeClassifier(random_state=0, criterion='entropy', splitter='best')

algsSampled = []
algsSampled.append(rf1)
algsSampled.append(lr1)
algsSampled.append(dt1)

for alg in algs:
    print('Fitting: ', type(alg).__name__)
    alg.fit(X_train, y_train)

for alg in algsSampled:
    print('Fitting: ', type(alg).__name__, ' resampled')
    alg.fit(x_resampled, y_resampled)
```

```
Fitting:  RandomForestClassifier
Fitting:  LogisticRegression
Fitting:  DecisionTreeClassifier
Fitting:  RandomForestClassifier  resampled
Fitting:  LogisticRegression  resampled
Fitting:  DecisionTreeClassifier  resampled
```

Acurácia

	Name	Type	Resampled	ACC
0	RandomForestClassifier	Training	N	0.995175
1	LogisticRegression	Training	N	0.823388
2	DecisionTreeClassifier	Training	N	0.995233
3	RandomForestClassifier	Training	Y	0.997109
4	LogisticRegression	Training	Y	0.610200
5	DecisionTreeClassifier	Training	Y	0.997109
6	RandomForestClassifier	Validation	N	0.813347
7	LogisticRegression	Validation	N	0.821617
8	DecisionTreeClassifier	Validation	N	0.744199
9	RandomForestClassifier	Validation	Y	0.778544
10	LogisticRegression	Validation	Y	0.644843
11	DecisionTreeClassifier	Validation	Y	0.738571
12	RandomForestClassifier	Test	N	0.818783
13	LogisticRegression	Test	N	0.824848
14	DecisionTreeClassifier	Test	N	0.741684
15	RandomForestClassifier	Test	Y	0.784047
16	LogisticRegression	Test	Y	0.652086
17	DecisionTreeClassifier	Test	Y	0.743981

CONSIDERAÇÕES FINAIS

OBRIGADO PELA ATENÇÃO

Até a próxima!