UNIVERSIDADE ESTADUAL DO MARANHÃO
CENTRO DE CIÊNCIAS TECNOLÓGICAS - CCT
DISCIPLINA DE PROCESSAMENTO DE SINAIS BIOLÓGICOS

GLACY KELLY MOURA GOMES - 201709434

PREDIÇÃO DE DIABETES COM SVM

São Luís – MA
2021

GLACY KELLY MOURA GOMES - 201709434

PREDIÇÃO DE DIABETES COM SVM

Trabalho apresentado ao curso de Graduação em Engenharia da Computação na Universidade Estadual do Maranhão como pré-requisito para obtenção de nota na disciplina de Processamentos de Sinais Biológicos sob orientação do Prof. Lúcio Flávio de Albuquerque Campos.

São Luís – MA

2021

Sumário

1 INTRODUÇÃO

Segundo o site de medicina Hermes Pardini, a diabetes é uma doença crônica que faz com que o corpo não produza insulina e também não consegue empregar adequadamente a insulina que a pessoa produz. Por isso, é importante ter o diagnóstico precoce do diabetes pois pode evitar suas implicações. Logo, a utilização de algoritmos para predizer se uma pessoa tem tendência a se tornar diabética são necessários para o auxílio na área da saúde

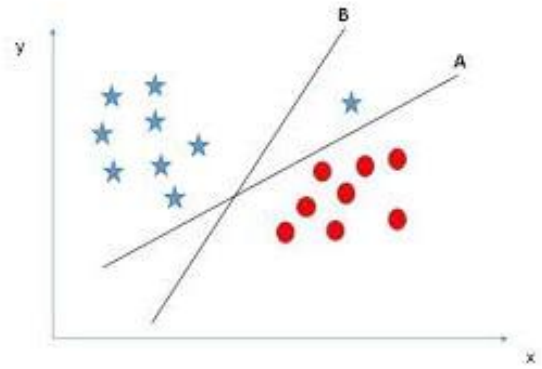
Este trabalho tem como objetivo apresentar a utilização da predição de diabetes através de algoritmos que utilizam as máquinas de vetores de suporte ou SVM (support vector machine). A metodologia escolhida para esse relatório foram as pesquisas de artigos científicos e códigos no repositório do github.

2 MÁQUINA DE VETORES DE SUPORTE

Segundo Kumari e Chitra (2013), máquinas de vetor de suporte ou SVM é uma técnica de aprendizado de máquina supervisionada que tem principalmente suas aplicações em classificação, ou também, em regressão. Por isso, ela é frequentemente utilizada em diagnóstico médico para classificação de doenças como por exemplo, a diabetes.

De acordo com Erivelton Guedes (2019), cada informação se transforma em um ponto dentro de um espaço n-dimensional, onde n é o número de variáveis do dataset e o valor de cada variável determina a coordenada. Além disso, ele cita que a criação do modelo de classificação gera um hiperplano ou hyperplane que diferencia as duas classes. Hiperplano é a reta que melhor separa as duas classes, maximizando as distâncias entre os pontos mais próximos. Esta distância é chamada de margem. Os vetores de suporte são justamente os pontos próximos à margem, como é ilustrado na figura 1.

Figura 1 - Hiperplano A e B



Fonte: (SVM)-erivelton | Kaggle

3 MÉTRICAS DE DESEMPENHO

3.1 Acurácia

Segundo Mariana González (2019), acurácia é a proximidade de um resultado com o seu valor de referência real. Dessa forma, quanto maior a acurácia, mais próximo da referência ou valor real é o resultado encontrado. Além disso, a acurácia é uma métrica importante nos algoritmos de classificação para validar os seus resultados e verificar se estão próximos de sua precisão.

3.2 Verdadeiro positivo/negativo e falso positivo/negativo

Os verdadeiros positivos são observações cujo valor real é positivo e o valor previsto é positivo, isto é, o modelo acertou. Já os verdadeiros negativos são observações cujo valor real é negativo e o valor previsto é negativo, isto é, o modelo acertou. Já os falsos positivos são casos em que o resultado correto é negativo entretanto o resultado obtido é positivo, isto é, o modelo errou. Por fim, os falsos negativos são casos em que o resultado correto é positivo entretanto o resultado obtido é negativo, isto é, o modelo errou. Essa representação é ilustrada na figura 2.

Figura 2 - Erros e Acertos do Modelo

Valor real / Valor previsto	Positivo	Negativo
Positivo	Verdadeiro Positivo (TP)	Falso Positivo (FP)
Negativo	Falso Negativo (FN)	Verdadeiro negativo (TN)

Fonte: datarisk.

3.3 Matriz de confusão

Segundo o site datarisk, a matriz de confusão é uma tabela onde facilmente identificamos todos os quatro tipos de classificação do modelo de classificação binário (isto é, com apenas dois valores distintos na variável resposta). Com ela, facilmente é possível calcular valores como a acurácia.

3.4 Curva ROC

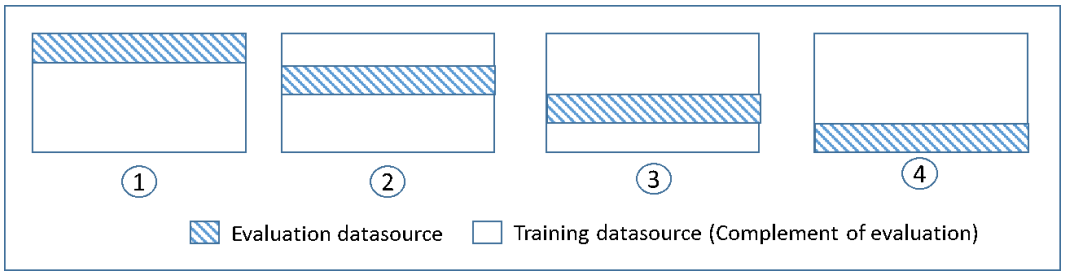
Segundo Ana Cristina da Silva Braga (2000), a curva ROC permite estudar a variação da sensibilidade e especificidade, para diferentes valores de corte. Neste trabalho, a curva ROC avalia a performance do algoritmo. A AUC (Area Under the ROC Curve) é a área sobre a curva e esse valor pode variar de 0 até 1 e o limiar entre a classe é 0,5.

3.5 Validação cruzada

Segundo o site aws, validação cruzada é uma técnica para avaliar modelos por meio de treinamento de vários modelos em subconjuntos de dados de entrada disponíveis e avaliação deles no subconjunto complementar dos dados. A validação cruzada é utilizada para detectar sobreajuste, ou seja, a não generalização de um padrão.

Na validação cruzada k-fold, os dados de entradas são divididos em subconjuntos de dados k (que podem ser chamados de folds). Você treina um modelo em todos, menos em um (k-1) dos conjuntos de dados e, em seguida, avalia o modelo no conjunto de dados que não foi usado para treinamento. Esse processo é repetido k vezes, com um subconjunto diferente reservado para avaliação (e excluído do treinamento) a cada vez. Na figura 3 é ilustrado esse processo.

Figura 3 - Processo de validação cruzada



Fonte: Site Amazon Machine Learning.

4 ESTUDO DE CASO

Um dos mais importantes processos metabólicos do organismo é a conversão de alimentos em energia e calor, dentro do corpo. Os alimentos são constituídos de três nutrientes principais:

• Carboidratos - (digestão) Glicose

• Proteínas - (digestão) Aminoácidos

• Gorduras - (digestão) Ácidos Graxos

Quando o organismo falha no processo de execução deste processo metabólico, ele é caracterizado como diabético. Diabete é uma anormalidade caracterizada por uma quantidade de açúcar em excesso no sangue e na urina. A diabete mata e não tem cura mas pode ser controlada, por isso é importante diagnosticá-la o quanto antes.

O objetivo é projetar e implementar em python, um algoritmo de classificação através de Máquinas de vetores de Suporte, que classifique os pacientes como diabético ou não diabético, a partir das 8 entradas listadas a seguir:

1. Número de vezes que ficou grávida

2. Concentração de glicose no Plasma em teste de tolerância de glicose oral

3. Pressão sangüínea Diastólica (mm Hg)

4. Dobras na pele do tríceps (mm)

5. 2-Horas de insulina de soro (mu U/ml)

6. Índice de massa corpórea (peso em kg/(altura em m2)

7. Função de genealogia de diabete

8. Idade (anos)

5 MATERIAIS E MÉTODOS

5.1 Base de dados

A base de dados da doença de diabetes utilizada tem as seguintes colunas conforme mostrado na figura 4.

1. Nº Gravidez: Número de vezes que ficou grávida

2. Glicose: Concentração de glicose no Plasma em teste de tolerância de glicose oral

3. PSanguínea: Pressão sangüínea Diastólica (mm Hg)

4. DobrasPele: Dobras na pele do tríceps (mm)

5. Insulina: 2-Horas de insulina de soro (mu U/ml)

6. IMC: Índice de massa corpórea (peso em kg/(altura em m2)

7. GenealogiaDiabética: Função de genealogia de diabete

8. Idade: Idade (anos)

9. Resultado: não diabético (0) ou diabético (1)

Figura 4 - Base de dados de diabetes

	NGravidez	Glicose	PSanguinea	DobrasPele	Insulina	IMC	GenealogiaDiabetica	Idade	Resultado
0	6.0	148.0	72.0	35.0	0.0	33.6	0.6	50.0	1
1	1.0	85.0	66.0	29.0	0.0	26.6	0.4	31.0	0
2	8.0	183.0	64.0	0.0	0.0	23.3	0.7	32.0	1
3	1.0	89.0	66.0	23.0	94.0	28.1	0.2	21.0	0
4	0.0	137.0	40.0	35.0	168.0	43.1	2.3	33.0	1

Fonte: Autoria Própria.

Essa base de dados tem cerca de 768 registros de pacientes, baseados nas colunas citadas anteriormente. Na coluna “Resultado”, é possível observar que existem dois valores: 0 e 1. O valor 0 representa que a pessoa testou negativo para a diabetes, portanto, o valor 1 representa positivo para a doença.

6 RESULTADOS E DISCUSSÕES

6.1 Aplicação da base de dados

Foi feita a separação das amostras para treino e teste, dividindo entre o x e o y, anteriormente separados, as amostras de testes e treino são representadas na figura 5 e 6.

Figura 5 - Amostras de teste

	NGravidez	Glicose	PSanguinea	DobrasPele	Insulina	IMC	GenealogiaDiabetica	Idade
285	7.0	136.0	74.0	26.0	135.0	26.0	0.6	51.0
101	1.0	151.0	60.0	0.0	0.0	26.1	0.2	22.0
581	6.0	109.0	60.0	27.0	0.0	25.0	0.2	27.0
352	3.0	61.0	82.0	28.0	0.0	34.4	0.2	46.0
726	1.0	116.0	78.0	29.0	180.0	36.1	0.5	25.0
...	...	...	...	...	...	...	...	...
247	0.0	165.0	90.0	33.0	680.0	52.3	0.4	23.0
189	5.0	139.0	80.0	35.0	160.0	31.6	0.4	25.0
139	5.0	105.0	72.0	29.0	325.0	36.9	0.2	28.0
518	13.0	76.0	60.0	0.0	0.0	32.8	0.2	41.0
629	4.0	94.0	65.0	22.0	0.0	24.7	0.1	21.0

192 rows × 8 columns

Fonte: Autoria Própria.

Figura 6 - Amostras de treino

	NGravidez	Glicose	PSanguinea	DobrasPele	Insulina	IMC	GenealogiaDiabetica	Idade
118	4.0	97.0	60.0	23.0	0.0	28.2	0.4	22.0
205	5.0	111.0	72.0	28.0	0.0	23.9	0.4	27.0
506	0.0	180.0	90.0	26.0	90.0	36.5	0.3	35.0
587	6.0	103.0	66.0	0.0	0.0	24.3	0.2	29.0
34	10.0	122.0	78.0	31.0	0.0	27.6	0.5	45.0
...	...	...	...	...	...	...	...	...
645	2.0	157.0	74.0	35.0	440.0	39.4	0.1	30.0
715	7.0	187.0	50.0	33.0	392.0	33.9	0.8	34.0
72	13.0	126.0	90.0	0.0	0.0	43.4	0.6	42.0
235	4.0	171.0	72.0	0.0	0.0	43.6	0.5	26.0
37	9.0	102.0	76.0	37.0	0.0	32.9	0.7	46.0

576 rows × 8 columns

Fonte: Autoria Própria.

A quantidade de amostras gerada depois desses passos foi de 576 amostras para o treino e 192 amostras para o teste, ambas com 8 colunas que representam as características de cada amostra individualmente.

Para realizar a parte de validação cruzada, foram utilizados 10 folds na Árvore de Decisão que são ilustrados na figura 7.

Figura 7 - Cross Validation com 10 folds

```
# Cross-validation na Árvore de Decisão c/ 10 folds
from sklearn.model_selection import cross_val_score, cross_val_predict
from sklearn import metrics
from sklearn.tree import DecisionTreeClassifier

modelodt = DecisionTreeClassifier()
#modelodt = modelodt.fit(xtreino,ytreino)
modelodt = modelodt.fit(X,y)
#print(np.mean(cross_val_score(modelodt, xtreino, ytreino, cv=10)))
print(np.mean(cross_val_score(modelodt, X, y, cv=10)))
previsoes_cv = cross_val_predict(modelodt, X, y, cv=10)
print(metrics.accuracy_score(y, previsoes_cv))
```

```
0.6799999999999999
0.65
```

Fonte: Autoria Própria.

6.2 Resultado obtido

6.2.1 Matriz de Confusão Global

A matriz de confusão pode ser gerada de duas formas, e ambas foram realizadas. Na primeira maneira, temos a matriz de confusão do SVM, utilizando o método confusion_matrix disponibilizado pelo sklearn, obtendo-se o resultado ilustrado na figura 8.

Figura 8 - Matriz de Confusão

```
# Matriz de Confusao do SVM
from sklearn.metrics import classification_report, confusion_matrix
print(confusion_matrix(yteste,yprevisao))
print(confusion_matrix(yprevisao, yteste))
```

```
[[113  10]
 [ 34  35]]
[[113  34]
 [ 10  35]]
```

Fonte: Autoria Própria.

Na segunda forma, usando novamente o sklearn, há o método de classification_report, e essa gera o resultado mostrado na figura 9.

Figura 9 - Resultado geral da matriz de confusão

```
# Resultado da Matriz de Confusão
from sklearn.metrics import classification_report, confusion_matrix
print(classification_report(yteste,yprevisao))
```

```
              precision    recall  f1-score   support

           0       0.77      0.92      0.84       123
           1       0.78      0.51      0.61        69

    accuracy                           0.77       192
   macro avg       0.77      0.71      0.73       192
weighted avg       0.77      0.77      0.76       192
```

Fonte: Autoria Própria.

6.2.2. Curva ROC

Para realizar a medição do valor da curva ROC, utilizando métodos do sklearn e logo depois, é calculada a probabilidade de predição de cada classe. O resultado da curva ROC varia com cada execução de código, mas seus valores variam de 30% a 40%.

Figura 10 - Valor da curva ROC

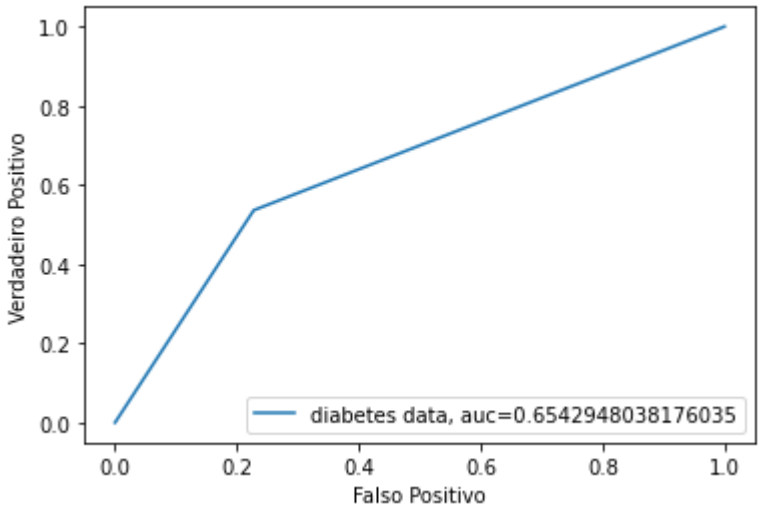
```
# Curva ROC
from sklearn.metrics import roc_auc_score
from sklearn.tree import DecisionTreeClassifier

modelodt = DecisionTreeClassifier()
modelodt = modelodt.fit(xtreino,ytreino)

# predic_proba() calcula a probabilidade de predição de cada classe
probs = modelodt.predict_proba(xteste)
probs = probs[:, 0]
roc_auc_score(yteste,probs)
```

```
0.3746907034287734
```

Fonte: Autoria Própria.

O gráfico ROC representa a performance do algoritmo e está ilustrado na figura 11. Além disso, nesse gráfico também é mostrado o AUC com o valor de 0,65, que representa um valor mediano.

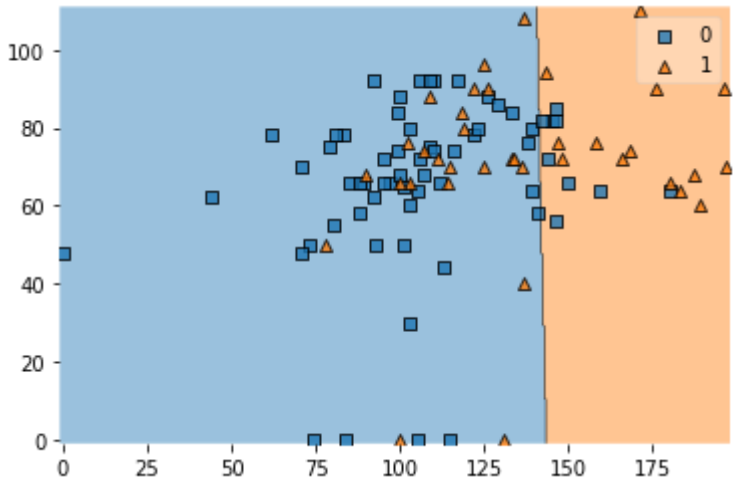
Figura 11 - Gráfico ROC



Fonte: Autoria Própria.

Na figura 12, está representado o gráfico de regiões de decisões para melhor visualizar o SVM, e ele em seguida irá separar quem tiver melhor acurácia, ou seja, que tenha mais classificações corretas no hiperplano.

Figura 12 - Gráfico de regiões de decisão



Fonte: Autoria Própria.

7 CONCLUSÕES FINAIS

O diagnóstico precoce da diabetes é muito importante para prevenir futuras complicações. Por isso, a utilização de algoritmos de classificação auxilia nos diagnósticos médicos. As máquinas de suporte de vetores (SVM) como uma técnica de aprendizado de máquina supervisionada conseguem auxiliar para a classificação de pacientes que podem vir a ter diabetes no futuro.

As SVMs conseguem classificar e as métricas de desempenho conseguem fazer a avaliação do algoritmo, para verificar principalmente com a acurácia, verdadeiros positivos e negativos, matriz de confusão, curva ROC e validação cruzada. Foi possível perceber que o algoritmo apresentou uma acurácia mediana. Logo, a utilização das SVMs foi necessária para classificar as amostras dos pacientes.

REFERÊNCIAS

Diabetes: tudo o que você precisa saber sobre a doença. Disponível em: <https://www.hermespardini.com.br/blog/?p=111>. Acesso: 10 jul 2021.

GUEDES, Pires E. **Máquinas de suporte de vetores (SVM)**. Disponível em: <https://www.kaggle.com/eriveltonguedes/6-m-quinas-de-suporte-de-vetores-svm-erivelton/code>. Acesso: 10 jul 2021.

KUMARI, Anuja V. CHITRA, R. **Classification Of Diabetes Disease Using Support Vector Machine**. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.435.3161&rep=rep1&type=pdf> Acesso: 11 jul 2021.

BRAGA, da Silva Cristina Ana. **Curvas ROC: Aspectos Funcionais e Aplicações.** Disponível em: <http://repositorium.sdum.uminho.pt/bitstream/1822/195/1/tese_doutACB.pdf>. Acesso: 11 jul 2021.

O que é acurácia? Entenda o conceito e sua importância. Disponível em: <https://blog.idwall.co/o-que-e-acuracia/>. Acesso 11 jul 2021.

O que é Matriz de Confusão? Disponível em: <https://ajuda.datarisk.io/knowledge/o-que-%C3%A9-matriz-de-confus%C3%A3o>. Acesso: 11 jul 2021.

Validação cruzada. Disponível em: <https://docs.aws.amazon.com/pt_br/machine-learning/latest/dg/cross-validation.html>. Acesso: 11 jul 2021.