Aprendizagem 2024/25

Homework I – Group 99

(ist1106059, ist1106329)

I. Pen-and-paper

1)

$$H(y_{out}|y_1 \geq 0.3\,) = -\left(\frac{3}{7}log_2\left(\frac{3}{7}\right)+\frac{2}{7}\log_2\left(\frac{2}{7}\right)+\frac{2}{7}\log_2\left(\frac{2}{7}\right)\right) \approx 1.5567$$

$$H(y_{out}|y_1 \geq 0.3, y_2\,) = \frac{4}{7}\left(-\frac{1}{4}log_2\left(\frac{1}{4}\right)-\frac{1}{4}\log_2\left(\frac{1}{4}\right)-\frac{1}{2}\log_2\left(\frac{1}{2}\right)\right)+\frac{3}{7}(-\frac{2}{3}log_2\left(\frac{2}{3}\right)-\frac{1}{3}log_2\left(\frac{1}{3}\right)) \approx 1.2507$$

$$IG(y_2) = 1.5567 - 1.2507 = 0.3060$$

$$H(y_{out}|y_1 \geq 0.3, y_3\,) = \frac{2}{7}(-1log_2(1))+\frac{4}{7}(-\frac{1}{4}log_2\left(\frac{1}{4}\right)-\frac{1}{4}\log_2\left(\frac{1}{4}\right)-\frac{1}{2}\log_2\left(\frac{1}{2}\right))+\frac{1}{7}(-1log_2(1)) \approx 0.8571$$

$$IG(y_3) = 1.5567 - 0.8571 = 0.6996$$

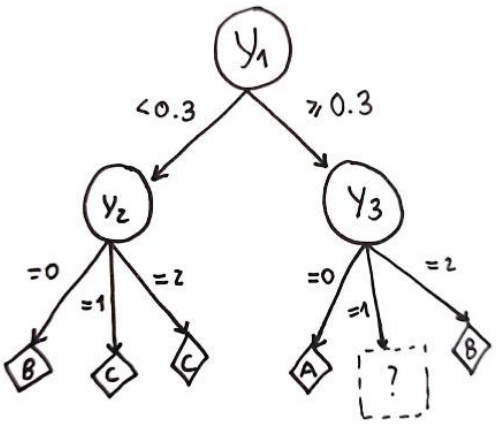
$$H(y_{out}|y_1 \geq 0.3, y_4) = \frac{4}{7}\left(-\frac{1}{2}log_2\left(\frac{1}{2}\right)-\frac{1}{2}log_2\left(\frac{1}{2}\right)\right)+\frac{3}{7}(-\frac{1}{3}log_2\left(\frac{1}{3}\right)-\frac{2}{3}log_2\left(\frac{2}{3}\right)) \approx 0.9650$$

$$IG(y_4) = 1.5567 - 0.9650 = 0.5917$$

Logo, y_3 é a variável que gera o maior ganho de informação e podemos fixá-la como raíz da sub-árvore.

Ficamos então com as partições {x8, x11}, {x6, x7, x9, x10}, {x12}.

Como {x8, x11} só tem instâncias da classe A e {x12} só tem uma instância da classe B, criamos as respetivas folhas. Como para a partição {x6, x7, x9, x10} não conseguimos tirar conclusões diretas e esta tem 4 observações, podemos continuar a dividir o nó.



$$H(y_{out}|y_1 \geq 0.3|y_3 = 1) = -\frac{1}{4}\left(\frac{1}{4}log_2\left(\frac{1}{4}\right)+\frac{1}{4}log_2\left(\frac{1}{4}\right)+\frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) = 1.5$$

$$H(y_{out}|y_1 \geq 0.3|y_3 = 1, y_2) = \frac{4}{4}\left(-\frac{1}{4}log_2\left(\frac{1}{4}\right) - \frac{1}{4}log_2\left(\frac{1}{4}\right) - \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right) = 1.5$$

$$IG(y_2) = 1.5000 - 1.5000 = 0$$

$$H(y_{out}|y_1 \geq 0.3|y_3 = 1, y_4) = \frac{1}{4}(-1\log_2 1) + \frac{3}{4}(-\frac{1}{3}\log_2\left(\frac{1}{3}\right) - \frac{2}{3}\log_2\left(\frac{2}{3}\right)) \approx 0.6887$$

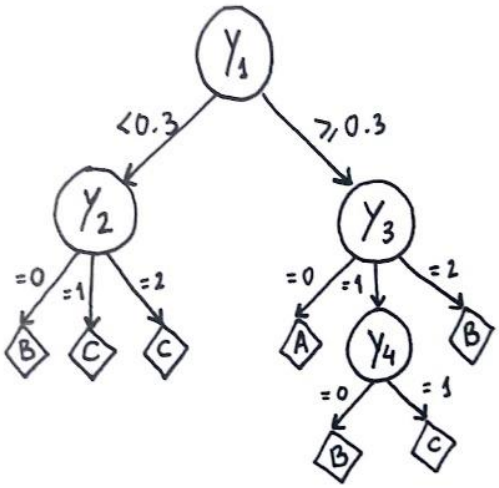
$$IG(y_4) = 1.5000 - 0.6887 = 0.8113$$

Logo, y_4 é a variável com mais ganho de informação e podemos fixá-la como raíz da sub-árvore.

Ficamos agora com as partições {x6} e {x7, x9, x10}.

A partição {x6} só tem uma instância da classe B, logo, criamos uma folha para B. No entanto, a partição {x7, x9, x10} tem instâncias de classes diferentes. Contrariamente ao que aconteceu antes, não podemos continuar a dividir a partição visto que esta só tem 3 instâncias.

Posto isto, criamos uma folha da classe C pois é a que tem maior frequência dentro da partição. Ficamos assim com a árvore de decisão apresentada abaixo:



2)

Predicted Values \ Real Values	**A**	**B**	**C**
A	2	0	0
B	0	4	0
C	1	0	5

3)

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN} \qquad F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

$$precision_A = \frac{2}{2+0} = 1$$
$$recall_A = \frac{2}{2+1} = \frac{2}{3}$$
$$F1_A = \frac{4}{5} = 80\%$$

$$precision_B = \frac{4}{4+0} = 1$$
$$recall_B = \frac{4}{4+0} = 1$$
$$F1_B = 1 = 100\%$$

$$precision_C = \frac{5}{5+1} = \frac{5}{6}$$
$$recall_C = \frac{5}{5+0} = 1$$
$$F1_C = \frac{10}{11} \approx 91\%$$

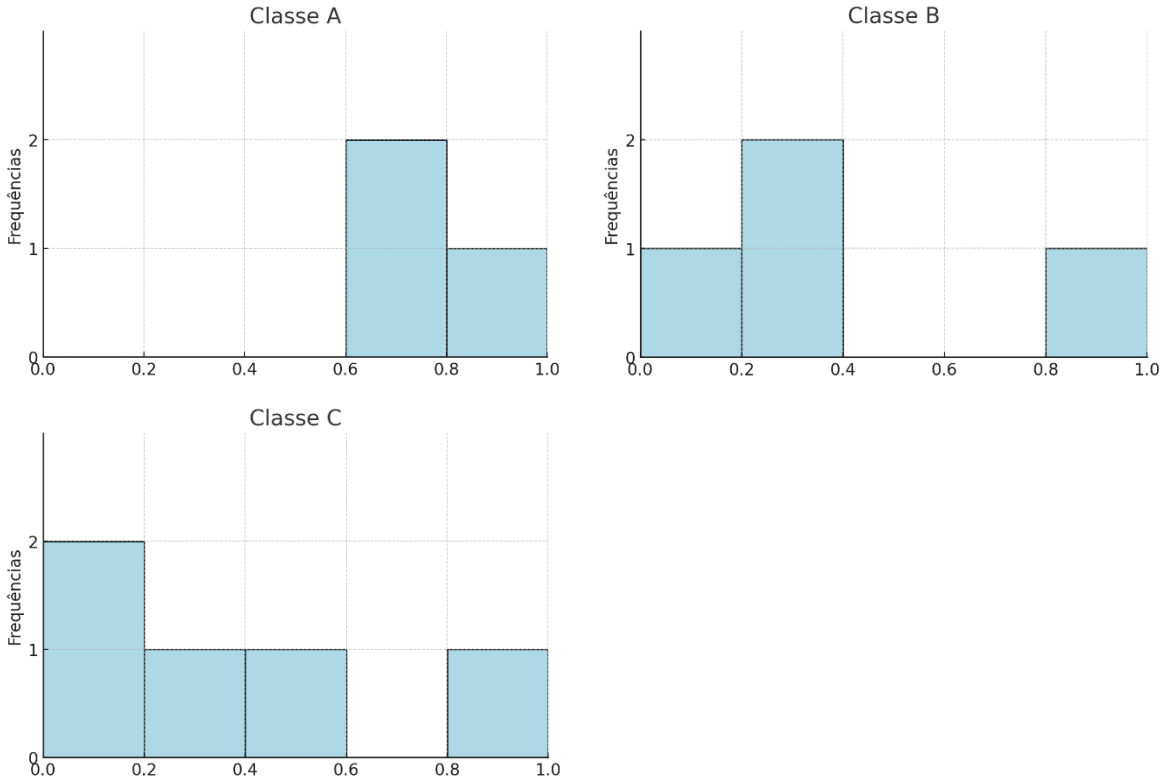
A classe A tem o F1 score mais baixo.

Homework I – Group 99

(ist1106059, ist1106329)

4)

Desenhando os histogramas condicionais por classe de y_1 usando 5 bins de igual tamanho no intervalo [0,1], ou seja, cada bin tem 0.2 de largura, obtemos o seguinte:



Ao analisar os histogramas, conseguimos tirar várias conclusões de como partir y_1 de forma a obter a melhor partição possível, ou seja, quantos e quais ramos devem sair de um root node com y_1 de forma a dar-nos mais informação sobre o target.

Olhando para o intervalo [0, 0.2[nos 3 histogramas, a classe com maior frequência é a C, portanto, a classe com maior número de observações quando y_1 pertence a esse intervalo. Desta forma, temos boas razões para criar um ramo neste intervalo e associá-lo à classe C.

Seguindo a mesma lógica para os outros bins temos que, para o intervalo [0.2 0.4[, a classe B tem maior frequência e, por isso, associamo-la ao ramo nesse intervalo e, para o intervalo [0.4, 0.6[, temos a classe C.

No entanto, para o intervalo [0.6, 0.8[, a classe A tem maior frequência, mas no último intervalo, [0.8, 1[, as 3 classes têm o mesmo número de observações. Isto significa que, se fossemos seguir esse ramo, não ganhávamos muita informação sobre o target. Logo, fazer um split de y_1 neste ramo não é ideal.

Em alternativa, podemos considerar o intervalo [0.6, 1[, ou seja, agrupar os dois últimos intervalos. Neste intervalo combinado, a frequência de A é 3, a de B é 1 e a de C é 1, logo, podemos associar esse intervalo à classe A.

Concluindo, com base nos histogramas condicionais por classe de y_1 apresentados acima, devemos partir os valores de y_1 em 4 ramos nos intervalos discutidos anteriormente.

Aprendizagem 2024/25

Homework I – Group 99

(ist1106059, ist1106329)

II. Programming and critical analysis

5)

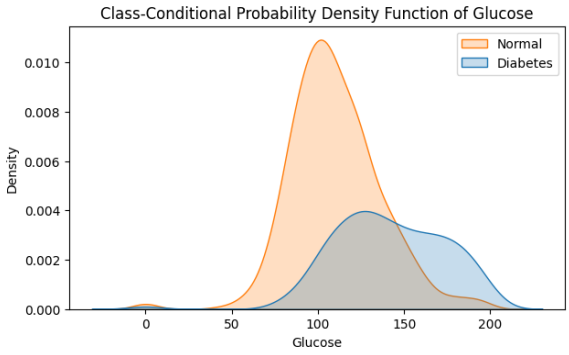
Utilizando a função f_classif da biblioteca sklearn, foi possível identificar a variável com maior e menor poder de discriminação. Para obtermos esta informação, começámos por fazer o carregamento dos dados do ficheiro fornecido, diabetes.arff, e converté-los para um DataFrame de forma a facilitar a manipulação e análise de dados.

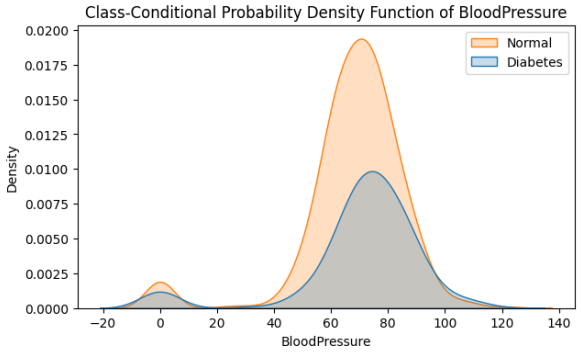
De seguida, separámos as variáveis: as Features foram armazenadas na variável x, exceto a coluna Outcome que foi guardada na variável y, representando o target. Com a função f_classif(x, y), foi possível calcular os valores F para cada Feature em relação ao Outcome. Quanto maior este valor, maior será a relevância da Feature para o modelo, logo, maior será o seu poder de discriminação.

Com estes valores calculados, e para facilmente visualizar os resultados obtidos como mostra a imagem abaixo, criámos um DataFrame com duas colunas – 'Feature' e 'F-Value', onde os elementos estão ordenados de forma decrescente do F-Value. Desta forma, podemos concluir que a variável com maior poder de discriminação é a Glucose (maior F-Value) e a variável com menor poder de discriminação é a BloodPressure (menor F-Value).

	Feature	F-Value
1	Glucose	213.161752
5	BMI	71.772072
7	Age	46.140611
0	Pregnancies	39.670227
6	DiabetesPedigreeFunction	23.871300
4	Insulin	13.281108
3	SkinThickness	4.304381
2	BloodPressure	3.256950

Com estes dados recolhidos, conseguimos criar gráficos onde podemos visualizar a função densidade de probabilidade de cada variável em relação às classes existentes. Desta forma, obtivemos os seguintes gráficos para as variáveis com maior e menor poder de discriminação.

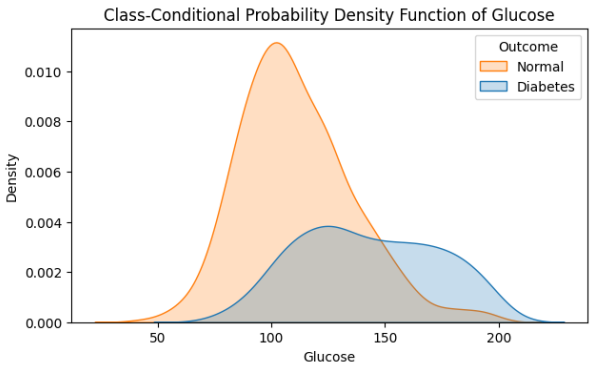


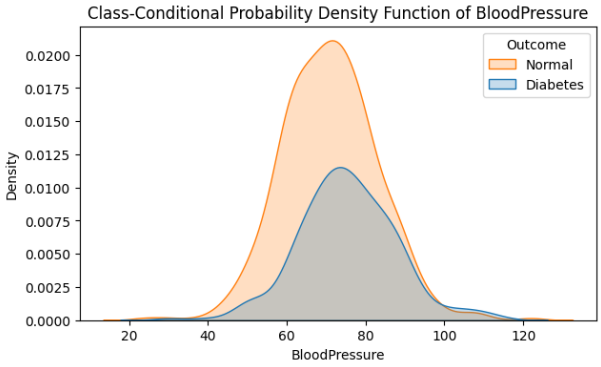


Ao estudarmos os gráficos obtidos, notámos que existia um aumento de densidade, tanto na glicose como na pressão do sangue, junto ao valor 0. Tendo em conta que é impossível estes valores existirem num ser humano vivo, adicionámos uma restrição para ignorar os valores abaixo do nível 0:

$$df = df[(df['BloodPressure'] > 0) \ \& \ (df['Glucose'] > 0)]$$

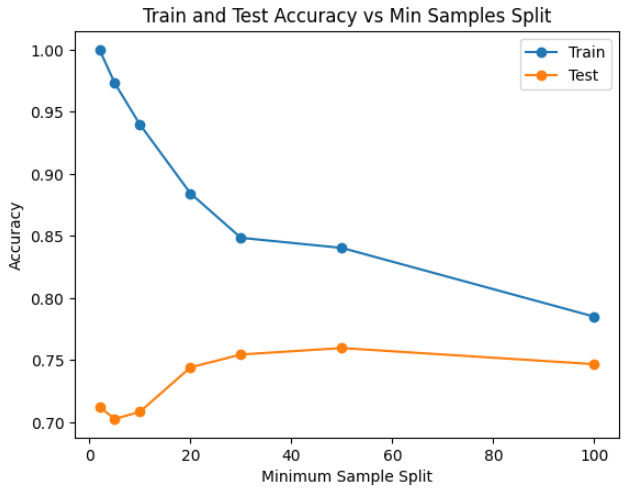
Desta forma, os gráficos tornam-se mais realistas uma vez que apenas consideram valores possíveis para estas variáveis. Os resultados obtidos após esta alteração encontram-se abaixo.





6)

O objetivo desta questão era realizar uma análise do desempenho e capacidade de generalização de um modelo de árvore de decisão, variando o parâmetro min_samples_split. Para isso, fizemos 10 execuções de avaliação da árvore, com diferentes valores desse parâmetro, calculando a accuracy média de teste e treino para cada valor. São feitas todas estas execuções de forma a garantir a estabilidade dos resultados obtidos. De seguida, fizemos o gráfico apresentado abaixo que nos permite visualizar a relação da accuracy com o aumento do parâmetro min_samples_split. Desta forma, conseguirmos concluir qual o melhor valor de min_samples_split de forma a existir um equilíbrio entre overfitting e underfitting.



7)

Ao observar o plot anterior, é possível ver que os gráficos da precisão do conjunto de treino e do conjunto de teste diferem muito um do outro.

Primeiramente, é importante definir o conceito de generalização, ou seja, a habilidade do modelo aprender padrões e fazer previsões corretas e precisas em dados não vistos (conjunto de teste), sem "memorizar" os dados de treino.

Quando o valor mínimo de samples para split é pequeno (entre 2 a 10), o modelo mostra sinais evidentes de overfitting, visto que a precisão do conjunto de treino começa perto de 1 (ou 100%). No entanto, estes valores altos não se refletem para a precisão do conjunto de teste, ou seja, a árvore de decisão (que é muito profunda nestes casos), ajusta-se demasiado aos dados de treino sem conseguir generalizar bem para o conjunto de treino.

Com o aumento do número de samples necessárias para fazer split (até aos 30), a precisão do conjunto de treino diminui bastante, cerca de 15%. Isto acontece porque, quanto maior este valor mínimo das samples, menos profunda tendem a ser as árvores de decisão, o que significa que o modelo faz menos splits, levando a um ajuste menos exato nos dados de treino (menos overfitting). Por outro lado, a precisão do conjunto de teste melhora e atinge um pico em torno dos 75%, o que sugere que o modelo está a generalizar melhor, evitando o underfitting.

O modelo apresenta a melhor capacidade de generalização quando o número mínimo de samples está entre os 30 e os 50, já que é neste intervalo que, tanto a precisão no conjunto de treino como na de teste estabilizam, indicando que o modelo encontrou um bom equilíbrio entre a complexidade e a capacidade de generalizar.

Um aumento adicional no número mínimo de samples para fazer split provoca uma diminuição em ambas as precisões, o que sugere que a árvore deixa de ter profundidade suficiente para conseguir capturar os padrões importantes dos dados, ou seja, começamos a ter underfitting (modelo fica demasiado simples para representar os dados de forma adequada).

Para terminar, com o plot realizado na pergunta anterior e depois de uma análise do comportamento do modelo, podemos concluir que os valores intermédios para o número mínimo de observações necessárias para dividir um nó (30 a 50) oferecem o melhor compromisso entre complexidade e capacidade de generalizar para novos dados.

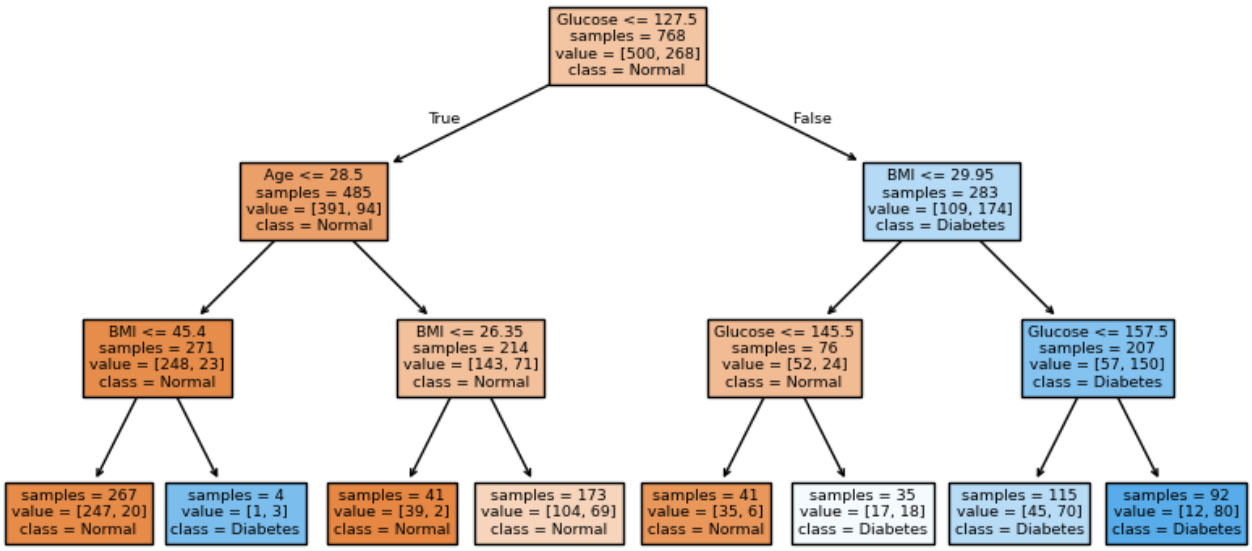
Aprendizagem 2024/25

Homework I – Group 99

(ist1106059, ist1106329)

8)

i) Plot da Árvore de decisão:



ii)

A árvore de decisão da alínea anterior foi treinada com todos os dados disponíveis e com uma profundidade máxima de 3 para evitar overfitting. O objetivo desta árvore é dividir as várias features do dataset e criar associações condicionais para, nas folhas, reunir as observações com características semelhantes e conseguir identificar o conjunto de características ou regras de associação que melhor identificam indivíduos com diabetes.

Em cada nó da árvore está presente a regra de associação que o descreve, assim como o número de samples/observações que se enquadram com esta regra e, na variável 'value' a distribuição dessas samples sobre a variável target. Ou seja, quantas samples, das totais desse nó, não têm diabetes e quantas é que têm. Para além disso, têm ainda o valor da classe que melhor representa esse nó, de acordo com a probabilidade posterior.

Esta probabilidade representa a probabilidade de uma sample que pertença a este nó ter realmente diabetes ou não. Calcula-se dividindo todas as observações desse nó que têm diabetes (valor da direita no array 'value') pelo número total de observações desse nó (variável samples).

Para identificar as associações condicionais que melhor descrevem indivíduos com diabetes, temos de calcular as probabilidades condicionais das folhas classificadas como diabetes:

Folha 1 (da esquerda):

- Regra de associação: (Glucose ≤ 127.5) $\cap$ (Age ≤ 28.5) $\cap$ (BMI > 45.4)

- Probabilidade posterior: $P = 3/4 = 0{,}75 = 75\%$

Aprendizagem 2024/25

Homework I – Group 99

(ist1106059, ist1106329)

Folha 2:

- Regra de associação: (Glucose $>$ 127.5) $\cap$ (BMI $\leq$ 29.95) $\cap$ (Glicose $>$ 145.5)
- Probabilidade posterior: P $= 18/35 = 0{,}514 = 51{,}4\%$

Folha 3:

- Regra de associação: (Glucose $>$ 127.5) $\cap$ (BMI $>$ 29.95) $\cap$ (Glucose $\leq$ 157.5)
- Probabilidade posterior: P $= 70/115 = 0{,}609 = 60{,}9\%$

Folha 4:

- Regra de associação: (Glucose $>$ 127.5) $\cap$ (BMI $>$ 29.95) $\cap$ (Glucose $>$ 157.5)
- Probabilidade posterior: P $= 80/92 = 0{,}870 = 87\%$

Com base nas regras de associação e probabilidades posterior calculadas, conseguimos concluir que níveis altos de glicose (>127.5) aumentam bastante o risco de diabetes, visto que 3 de 4 folhas com esses valores de glicose foram classificadas como tendo diabetes.

Para além disso, 3 das 4 folhas classificadas com diabetes apresentam níveis altos de BMI (>26.35) e a folha 1, que não apresenta glicose $>$ 127.5, mesmo assim foi classificada com diabetes por ter valores BMI ainda mais elevados (>45.4). Logo, podemos também concluir, com alguma certeza, que BMI alto aumenta o risco de diabetes.

Por fim, temos ainda a idade que, com valores inferiores a 28.5, e juntamente com valores baixos de glicose e altos de BMI, também pode ser um indicador de diabetes, ainda que não o possamos concluir com grande certeza.

END