DETECTING BREAST CANCER FROM BLOOD SAMPLES

Aprendizagem Computacional
2025/2026 • 1º Semestre

Daniel Pereira • 2021237092 • uc2021237092@student.uc.pt

Eduardo Marques • 2022231584 • uc2022231584@student.uc.pt

Conteúdo

1 Introdução

A **deteção precoce** do cancro da mama é um dos principais desafios da medicina moderna, pois tem impacto direto na eficiência dos tratamentos e na taxa de sobrevivência dos pacientes. Tradicionalmente, o diagnóstico baseia-se em métodos de imagem e procedimentos invasivos. No entanto, abordagens baseadas em análises sanguíneas representam uma alternativa promissora, por serem menos dispendiosas, mais rápidas e menos invasivas.

Neste contexto, este projeto propõe o desenvolvimento de **um pipeline de Machine Learning** para distinguir entre amostras de sangue de pessoas saudáveis e pessoas com cancro da mama, utilizando o conjunto de dados Breast Cancer Coimbra [1] disponível em UCI Machine Learning Repository. Este dataset contém 10 variáveis clínicas e bioquímicas recolhidas de 116 indivíduos — 64 pacientes com cancro e 52 saudáveis.

2 Objetivos

O objetivo global consiste em construir e avaliar pipelines de aprendizagem capazes de identificar padrões nos dados que permitam prever a presença de cancro da mama. Para tal, pretende-se:

- Compreender e explorar o comportamento das variáveis do dataset através de análises estatísticas e de correlação;
- Aplicar métodos de seleção e redução de características, como Kruskal-Wallis, ROC-AUC, PCA e LDA, de modo a identificar variáveis mais relevantes e eliminar redundâncias;
- Implementar e comparar classificadores clássicos de Machine Learning, incluindo Minimum Distance, Fisher LDA, Minimum Distance Mahalanobis, Bayes, Decision Trees, kNN, AdaBoost, SVM e Random Forests;
- Avaliar e comparar o desempenho dos modelos segundo métricas padronizadas: Sensibilidade, Especificidade, Precisão, F1-score, Accuracy e ROC-AUC;
- Discutir criticamente os resultados obtidos e propor melhorias no pipeline como reduçao de features.

3 Carregamento e Preparação dos Dados

Os dados foram carregados diretamente do Breast Cancer Coimbra dataset [1], disponível no repositório UCI Machine Learning Repository, através da biblioteca `ucimlrepo`. As classes foram binarizadas, 0 = saudável e 1 = cancro.

Posteriormente, o conjunto foi dividido de forma estratificada em 3 subconjuntos:

- **Treino (60%)** — usado para ajustar os parâmetros dos modelos;
- **Validação (20%)** — utilizado para comparação e seleção de modelos;
- **Teste (20%)** — reservado para avaliação final de desempenho.

Após a divisão obteve-se a seguinte distribuição de amostras: 69 amostras para o treino, 23 para a validação e 24 para o teste.

Os dados foram normalizados usando o `StandardScaler`, ajustando a média e o desvio padrão apenas no conjunto de treino, sendo posteriormente aplicados aos conjuntos de validação e teste. Este procedimento assegura que todas as variáveis se encontrem na mesma escala, prevenindo a influência desproporcional de atributos com magnitudes mais elevadas e evitando data leakage entre fases.

4 Seleção e Redução de Features

4.1 Análise Exploratória

Antes da aplicação de métodos de seleção e redução de características, foram calculadas estatísticas descritivas (média, desvio padrão, quartis, valores mínimos e máximos) como se pode ver na Tabela 1.

Tabela 1: Estatísticas descritivas das variáveis normalizadas do conjunto de treino.

Estatística	Age	BMI	Glucose	Insulin	HOMA	Leptin	Adiponectin	Resistin	MCP.1
count	116	116	116	116	116	116	116	116	116
mean	57.30	27.58	97.79	10.01	2.69	26.62	10.18	14.73	534.65
std	16.11	5.02	22.53	10.07	3.64	19.18	6.84	12.40	345.91
min	24.00	18.37	60.00	2.43	0.47	4.31	1.66	3.21	45.84
Q1 (25%)	45.00	22.97	85.75	4.36	0.92	12.31	5.47	6.88	269.98
Q2 (50%)	56.00	27.66	92.00	5.92	1.38	20.27	8.35	10.83	471.32
Q3 (75%)	69.00	31.24	102.00	11.19	2.86	37.38	11.82	17.76	700.09
max	89.00	38.58	201.00	58.46	25.05	90.28	38.04	82.10	1698.44

A análise das estatísticas descritivas apresentadas na Tabela 1 permite compreender a distribuição das variáveis do conjunto de treino antes da aplicação de métodos de seleção e redução de dimensionalidade.

A variável **Age** mostra uma média de aproximadamente 58 anos com um desvio padrão elevado ($\approx$ 16 anos), indicando uma amostra heterogénea em termos etários. O **BMI** apresenta uma média de 27.58, próxima do limiar de sobrepeso, sugerindo uma tendência geral para valores acima do normal.

Os marcadores metabólicos, como **Glucose**, **Insulin** e **HOMA**, revelam uma variação substancial, com valores máximos muito superiores aos mínimos, sugerindo a presença de indivíduos com perfis metabólicos bastante distintos. As variáveis **Leptin**, **Adiponectin** e **Resistin** apresentam médias e quartis que indicam uma distribuição moderadamente assimétrica, coerente com dados biológicos deste tipo. Por fim, a **MCP-1** demonstra grande amplitude de variação (45.8 – 1698.4), o que sugere diferenças significativas nos níveis inflamatórios entre os participantes.

De forma geral, a análise evidencia que as variáveis possuem escalas e dispersões diferentes, justificando a necessidade do processo de normalização aplicado. Estes resultados também reforçam a importância de métodos subsequentes de seleção de features e redução de dimensionalidade (como PCA e LDA) para identificar as combinações mais informativas e minimizar redundâncias no conjunto de dados.

De seguida, foi construída a **Matriz de Correlação de Pearson**, representada graficamente através de um heatmap (Figura 1).

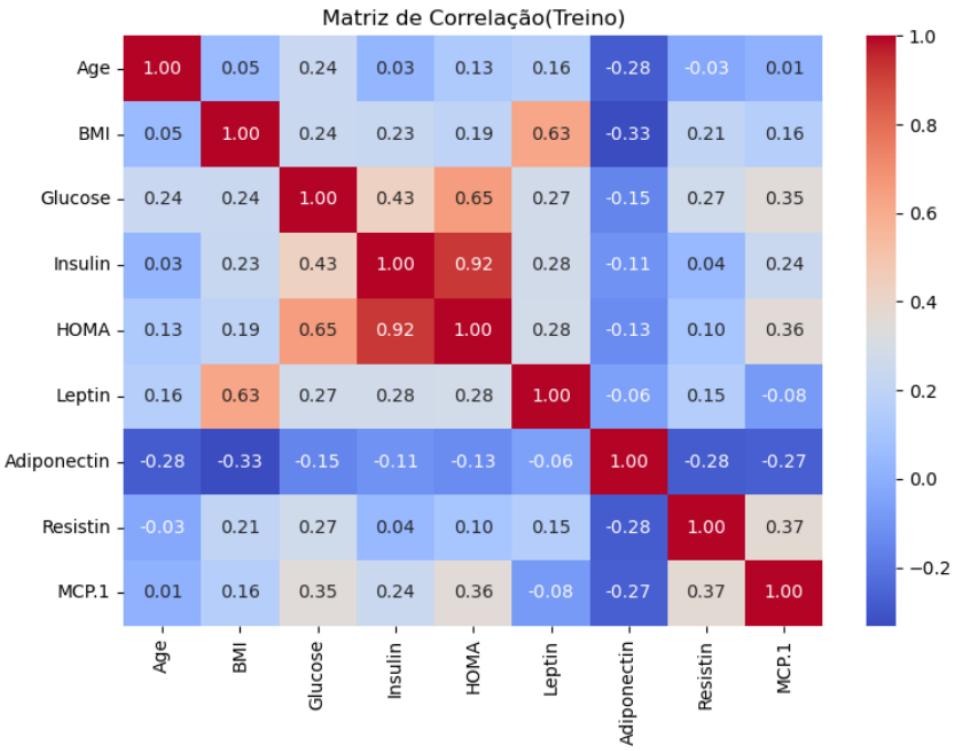

Figura 1: Matriz de correlação representada através de um heatmap.

Observamos uma correlação muito forte entre **Insulin** e **HOMA** ($r = 0.92$), o que indica redundância de informação entre estas variáveis. Também se verifica correlação moderada entre o **BMI** e o **Leptin** ($r = 0.63$) e entre a **Glucose** e o **HOMA** ($r = 0.65$). As restantes variáveis apresentam correlações fracas, indicando uma relativa independência.

Esta análise é útil para a seleção de features, permitindo identificar atributos potencialmente redundantes que podem ser removidos ou reduzidos nas etapas de **PCA** ou **LDA**.

4.2 Seleção de Features

Para aprofundar a análise e selecionar as variáveis mais relevantes para a classificação, foram aplicados dois métodos complementares:

- **Kruskal-Wallis** — avalia se as distribuições de cada variável diferem significativamente entre classes.
- **ROC-AUC individual** — quantifica o poder discriminatório de cada variável de forma independente.

Os resultados encontram-se discriminados na Tabela 2.

Tabela 2: Resultados do teste de Kruskal–Wallis e valores ROC-AUC individuais para cada variável.

Variável	Kruskal–Wallis (p-value)	ROC-AUC
Glucose	0.000005	0.820
HOMA	0.021217	0.662
Resistin	0.022609	0.660
Insulin	0.080255	0.623
Age	0.352722	0.435
Adiponectin	0.708431	0.474
MCP.1	0.772181	0.520
BMI	0.795354	0.482
Leptin	0.800012	0.482

Os resultados do teste de Kruskal-Wallis indicaram que a **Glucose** ($p \approx 0.000005$), **HOMA** ($p \approx 0.021$) e **Resistin** ($p \approx 0.023$) são as variáveis estatisticamente mais relevantes ($p < 0.05$).

Da análise do ROC-AUC individual, percebemos que estas variáveis mantiveram-se no topo, com destaque para a **Glucose (AUC = 0.82)**, seguida pela **HOMA (AUC = 0.66)** e **Resistin (AUC = 0.66)**.

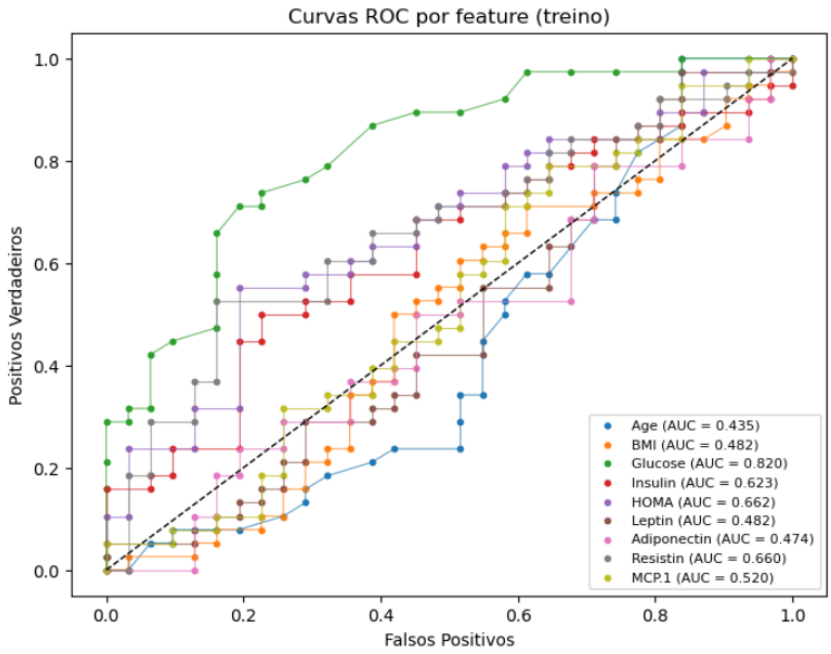

Figura 2: Curvas ROC.

Estes resultados sugerem que estas 3 variáveis (Glucose, HOMA e Resistin) possuem maior capacidade de distinção entre as classes, podendo ser consideradas prioritárias para as etapas seguintes. Por outro lado, o BMI e a Age apresentam uma alta taxa de falsos positivos, ou seja, são estatísticas pouco discriminatórias.

4.3 Redução de Dimensionalidade (PCA e LDA)

Após a seleção das features, aplicaram-se técnicas de redução de dimensão para visualizar e avaliar a separação dos dados. O **PCA (Principal Component Analysis)** é um método não supervisionado que transforma as variáveis originais em novas combinações lineares chamadas componentes principais (PC's), organizadas de modo a reter a máxima variância dos dados.

Tabela 3: Variâncias explicadas.

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
34.2%	16.1%	15.1%	12.2%	8.4%	6.2%	4.8%	2.7%	0.3%

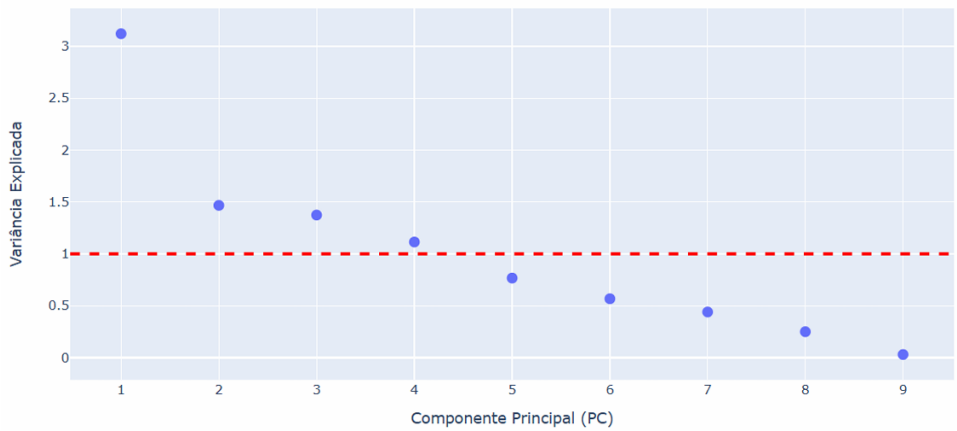

Figura 3: Variância Explicada por Componente (PCA).

No presente caso, as duas primeiras componentes principais representam aproximadamente 50% da variância total (34.179% e 16.071%).

Em contraste, o **LDA (Linear Discriminant Analysis)** é uma técnica supervisionada que procura encontrar uma combinação linear das variáveis que maximize a separação entre classes e minimize a variabilidade dentro de cada classe.

O LDA gera apenas uma componente discriminatória, que descreve 100% da variância entre as duas classes, conforme ilustrado na Figura 4.

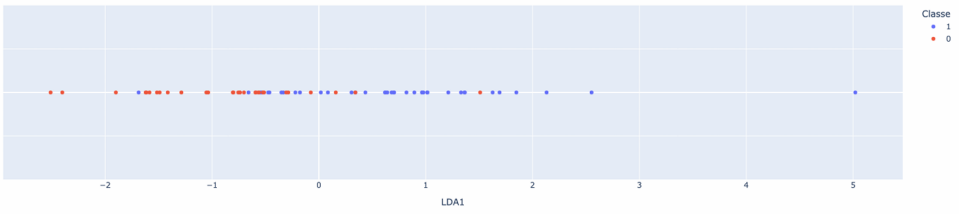

Figura 4: Projeção LDA (1 Componente).

Através do gráfico, observamos que as distribuições das duas classes apresentam um desvio nas suas médias, indicando uma separação parcial: os indivíduos saudáveis (0/vermelho) tendem a posicionar-se em valores negativos da componente discriminatória, enquanto que os indivíduos com cancro (1/azul) tendem a situar-se maioritariamente em valores positivos.

Apesar de existir uma zona de sobreposição central, esta projeção evidencia que o LDA consegue distinguir os grupos de forma razoável, embora não perfeita, refletindo a natureza complexa e parcialmente sobreposta das características bioquímicas entre classes.

5 Classificação e Avaliação de Desempenho

5.1 Metodologia de Avaliação

Para avaliar o desempenho dos classificadores, foi desenvolvida uma função que calcula as principais métricas de desempenho: **Sensibilidade**, **Especificidade**, **Precisão**, **F1-score**, **Accuracy** e **ROC-AUC**. Estas métricas permitem uma análise abrangente da capacidade do modelo em identificar corretamente as amostras positivas (cancro) e negativas (saudável).

As expressões usadas são as seguintes:

- Sensibilidade = $\frac{TP}{FN+TP}$
- Especificidade = $\frac{TN}{TN+FP}$
- Precisão = precision_score($y_{\text{true}}, y_{\text{pred}}$, average='macro')
- F1-score = f1_score($y_{\text{true}}, y_{\text{pred}}$, average='macro')
- Accuracy = $\frac{TN+TP}{TP+TN+FP+FN}$
- ROC-AUC = roc_auc_score($y_{\text{true}}, y_{\text{prob}}$)

Legenda: y_{true} representa os valores reais das classes, y_{pred} representa as classes estimadas pelo modelo, y_{prob} representa as probabilidades estimadas, TP (True Positives), TN (True Negatives), FP (False Positives) e FN (False Negatives).

5.2 Classificadores

A escolha dos classificadores a implementar desempenha um papel central no desenvolvimento de um sistema de deteção precoce do cancro da mama baseado em análises sanguíneas. Dado que o conjunto de dados apresenta variáveis clínicas e bioquímicas com padrões parcialmente sobrepostos entre classes, tornou-se essencial avaliar modelos com naturezas distintas — desde métodos lineares simples até abordagens não lineares e baseadas em vizinhança. Nesta secção são descritos os vários classificadores utilizados, realçando de que forma cada um deles trabalha a estrutura dos dados e contribui para compreender a capacidade discriminatória das variáveis clínicas no contexto do presente projeto.

5.2.1 Minimum Distance Classifier (MDC)

Classifica cada amostra com base na distância euclidiana ao centróide de cada classe, atribuindo a classe do centróide mais próximo. Este método assume que as classes podem ser representadas pelos seus pontos médios e que a distância euclidiana é uma medida adequada de similaridade.

A implementação calcula o centróide (média) de cada classe durante o treino através de `self.means_`. Na fase de predição, para cada amostra é calculada a distância euclidiana até cada centróide usando `np.linalg.norm()`, sendo atribuída a classe do centróide mais próximo. As probabilidades são estimadas aplicando a função `softmax` às distâncias negativas, convertendo-as em scores probabilísticos normalizados.

5.2.2 Fisher Linear Discriminant Analysis (LDA)

Este classificador encontra uma combinação linear das variáveis que maximiza a separação entre as classes e minimiza a variação dentro de cada classe. O Fisher LDA projeta os dados num espaço de menor dimensão onde a discriminação entre classes é otimizada.

No código implementado é utilizada a classe `LinearDiscriminantAnalysis` do scikit-learn, que encontra automaticamente a projeção linear ótima através da maximização do rácio entre a variância entre classes e a variância dentro das classes. O modelo calcula os coeficientes da transformação linear e permite tanto predição de classes como estimação de probabilidades através de `predict()` e `predict_proba()`.

5.2.3 Mahalanobis Distance Classifier

Classifica as amostras considerando a distância de Mahalanobis, levando em conta não apenas a distância média, mas também a variância e a covariância dos dados. Este classificador é mais robusto a correlações entre variáveis e a diferenças de escala entre atributos.

Durante o treino, são calculados os centróides de cada classe e a matriz de covariância inversa global através de `np.linalg.inv(np.cov(X.T))`. Na predição, a distância de Mahalanobis é calculada como $d = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)}$, onde μ é o centróide da classe e Σ^{-1} é a matriz de covariância inversa. Esta abordagem pondera adequadamente as correlações entre variáveis, sendo mais robusta que a distância euclidiana quando existem correlações significativas nos dados.

5.2.4 Bayes Classifier

Utiliza o teorema de Bayes para calcular a probabilidade à posteriori de cada classe, assumindo independência condicional entre as variáveis (Naive Bayes) ou modelando a distribuição Gaussiana multivariada. Classifica a amostra com base na classe de maior probabilidade.

A implementação molda cada classe através de uma distribuição Gaussiana multivariada utilizando `GaussianMixture` com um componente. São estimados os hiperparâmetros de cada distribuição: médias (`self.mean1`, `self.mean2`) e matrizes de covariância (`self.cov1`, `self.cov2`), bem como as probabilidades à priori de cada classe (`self.Pw1`, `self.Pw2`). A predição é feita calculando a probabilidade à posteriori através da função `BayespdfGauss()`, que implementa a densidade de probabilidade Gaussiana multivariada, e aplicando a regra de decisão de Bayes para atribuir a classe de maior probabilidade.

5.2.5 Decision Tree

Constrói uma estrutura em árvore onde cada nó interno representa uma decisão baseada numa variável, cada ramo representa o resultado dessa decisão, e cada folha representa

uma classe. O algoritmo divide recursivamente os dados de modo a maximizar a pureza das classes em cada nó.

A estrutura da Decision Tree obtida pode ser visualizada na Figura 5, onde se observa que a variável Glucose aparece como o atributo mais importante para a primeira divisão, confirmando os resultados da análise de seleção de features.

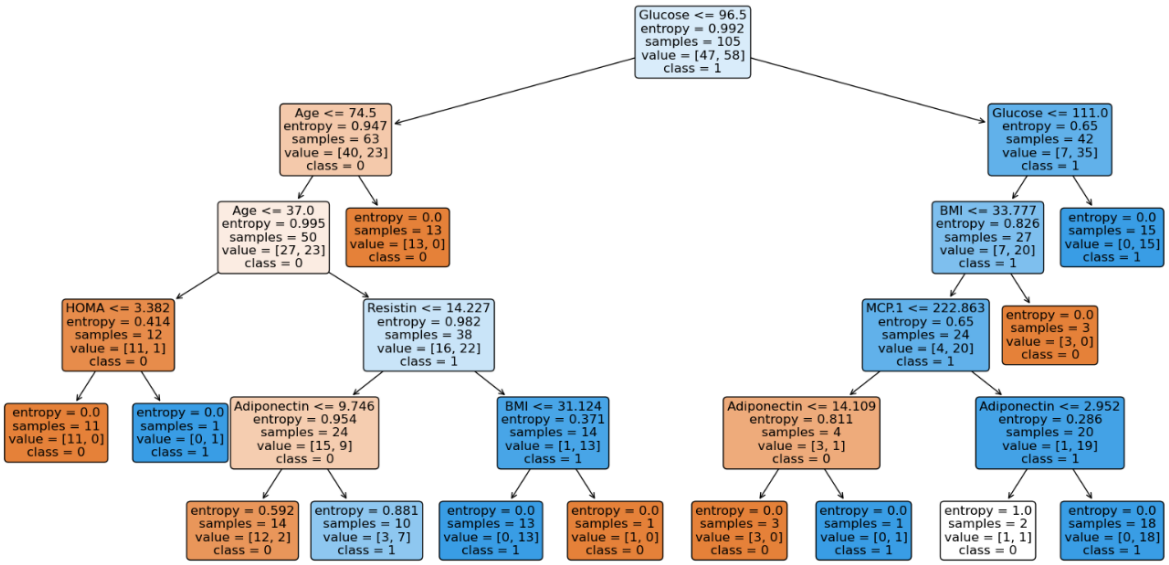

Figura 5: Decision Tree. Os nós laranja representam amostras saudáveis (classe 0) e os azuis representam amostras com cancro (classe 1).

O modelo foi implementado usando `DecisionTreeClassifier` do scikit-learn com os seguintes hiperparâmetros: `criterion='entropy'` (information gain) para avaliar as divisões, `max_depth=5` para limitar a profundidade e prevenir overfitting, `splitter='best'` para escolher a melhor divisão em cada nó. A visualização da árvore foi gerada através da função `plot_tree()`, permitindo análise interpretativa das decisões tomadas pelo modelo.

5.2.6 k-Nearest Neighbors (kNN)

Classifica uma amostra com base na classe mais frequente entre os k vizinhos mais próximos no espaço de características. A escolha do hiperparâmetro k é crucial para o desempenho do modelo.

Para o classificador kNN, foi realizada uma análise para determinar o valor ótimo de k, testando diferentes valores e calculando o erro médio de classificação, conforme apresentado na Tabela 4.

Tabela 4: Seleção do hiperparâmetro k para o classificador kNN. O melhor valor encontrado foi k=7.

k	Erro Médio (%)	Desvio Padrão
1	32.73	0.133
3	29.50	0.139
5	25.48	0.130
7	**24.42**	**0.120**
9	27.53	0.129
11	28.38	0.128
15	30.95	0.129
19	32.07	0.136
25	33.59	0.132

O valor k=7 foi selecionado por apresentar o menor erro médio (24.42%) e um desvio padrão relativamente baixo (0.120), indicando maior estabilidade e consistência nas previsões, como pode ser observado na Figura 6

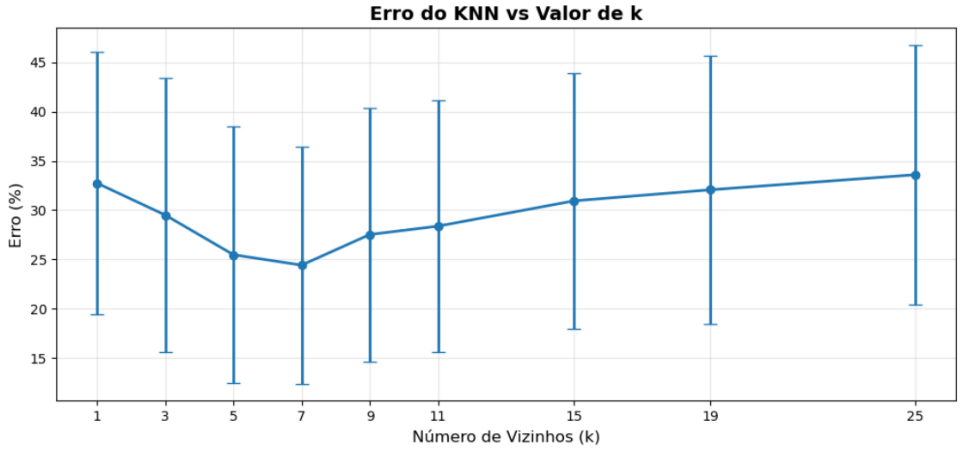

Figura 6: Erros médios para diferentes números de vizinho (k).

Em síntese, foi testado um conjunto de valores $k \in \{1, 3, 5, 7, 9, 11, 15, 19, 25\}$. Para cada valor de k, foi calculado o erro médio e desvio padrão ao longo das 100 iterações. A implementação utiliza `KNeighborsClassifier` do scikit-learn, que calcula as distâncias euclidianas entre cada amostra de teste e todas as amostras de treino (nos dados normalizados), seleciona os k vizinhos mais próximos, e atribui a classe mais frequente entre estes vizinhos. As probabilidades são estimadas pela proporção de vizinhos de cada classe.

5.2.7 AdaBoost

O AdaBoost (Adaptive Boosting) é um método de ensemble que combina múltiplos classificadores fracos para criar um classificador forte. O algoritmo ajusta iterativamente os pesos das amostras, dando maior ênfase àquelas incorretamente classificadas nas iterações anteriores.

A implementação segue o algoritmo clássico de AdaBoost com `n_estimators=50` classificadores fracos. Cada classificador é uma Decision Tree com `max_depth=1` (decision

stump). O algoritmo inicializa pesos uniformes para todas as amostras ($w_i = 1/N$), treina iterativamente cada classificador fraco ponderando as amostras pelos pesos atuais, calcula o erro ponderado $\epsilon = \sum w_i \cdot I(y_i \neq \hat{y}_i)$, determina o peso do classificador como $\alpha = 0.5 \ln \frac{1-\epsilon}{\epsilon}$, e atualiza os pesos das amostras multiplicando por $\exp(-\alpha \cdot y_i \cdot \hat{y}_i)$. A predição final combina todos os classificadores através de votação ponderada: $\text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$. As probabilidades são calculadas aplicando a função logística à soma ponderada.

5.2.8 Support Vector Machine (SVM)

As Support Vector Machines são classificadores que procuram encontrar o hiperplano ótimo que maximiza a margem de separação entre as classes. Foram testadas duas variantes:

SVM Linear Utiliza um kernel linear para separar as classes através de um hiperplano. Este método é adequado quando os dados são linearmente separáveis ou quase linearmente separáveis.

Para otimizar o desempenho do SVM Linear, foi realizada uma pesquisa do hiperparâmetro de regularização C, testando valores na escala logarítmica de 2^{-10} até 2^{10}. Os resultados desta otimização estão apresentados na Figura 7.

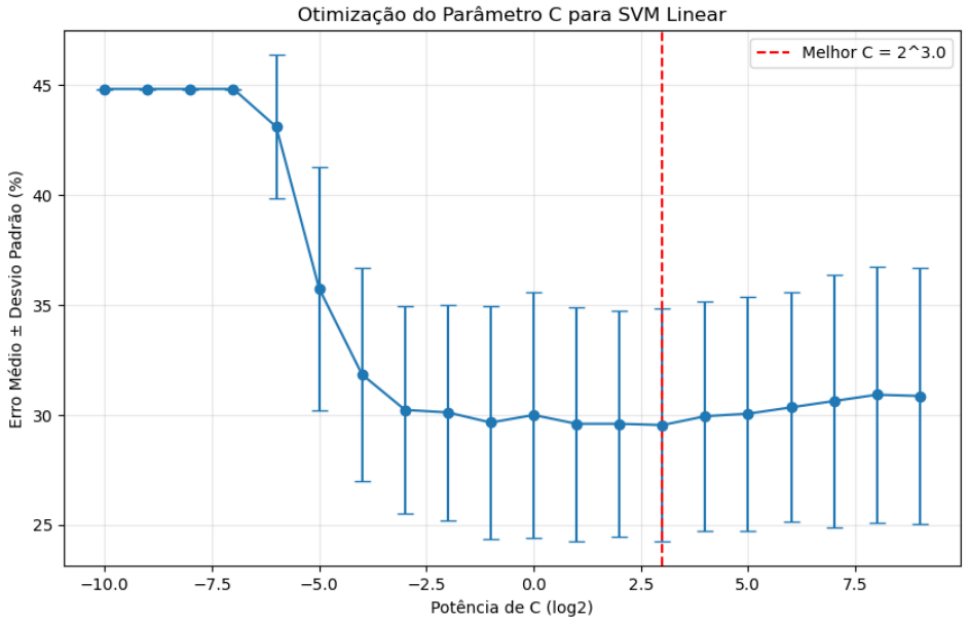

Figura 7: Otimização do hiperparâmetro C para SVM Linear.

O valor ótimo encontrado foi $C = 2^{3.0} = 8.0$, resultando num erro médio de 29.54% ± 5.30%.

O hiperparâmetro C controla o trade-off entre a maximização da margem e a minimização dos erros de classificação: valores pequenos de C resultam numa margem maior mas toleram mais erros, enquanto valores grandes de C penalizam fortemente os erros mas podem levar a overfitting. As probabilidades são estimadas através de calibração de Platt após o treino do modelo.

Em suma, a implementação utiliza `SVC(kernel='linear')` do scikit-learn para permitir estimação de probabilidades através de calibração de Platt (`probability=True`).

O hiperparâmetro de regularização C foi optimizado através de pesquisa em escala logarítmica ($C = 2^p$, com $p \in [-10, 10]$), testando 30 divisões aleatórias treino-teste (50%-50%) para cada valor. Foi calculado o erro médio e desvio padrão, sendo selecionado o valor que minimiza o erro de validação.

SVM RBF Utiliza o kernel Radial Basis Function (RBF) ou Gaussiano, permitindo a separação não-linear das classes. Este kernel introduz um parâmetro adicional gamma (γ) que controla a influência de cada amostra de treino.

Para o SVM RBF, foi realizada uma otimização conjunta dos hiperparâmetros C e gamma através de Grid Search, testando combinações de valores em escala logarítmica. Os resultados desta otimização bidimensional estão apresentados na Figura 8.

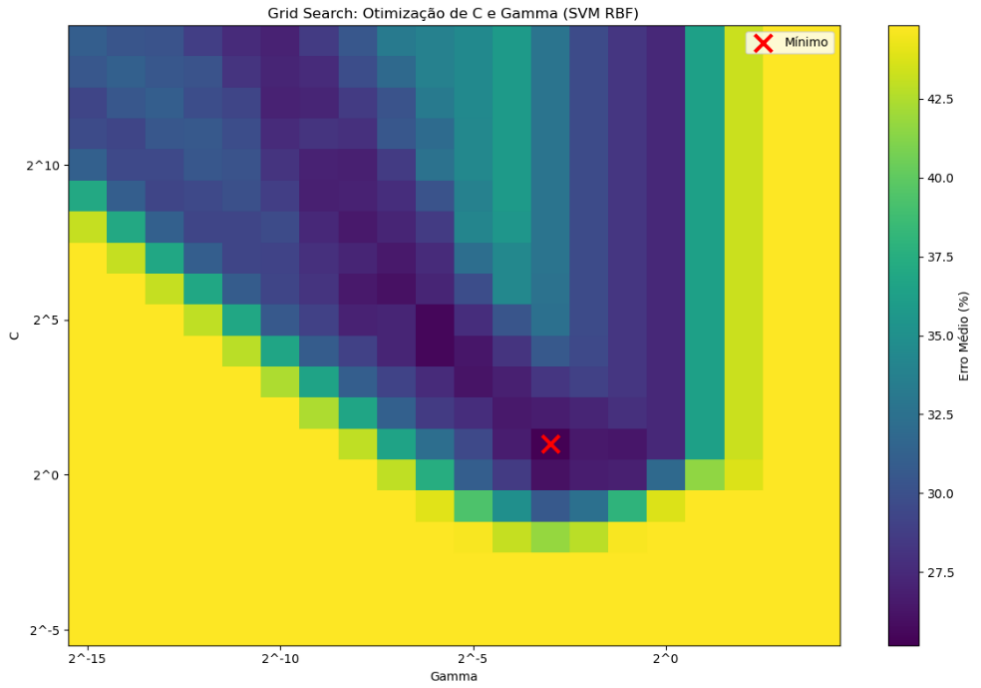

Figura 8: Grid Search para otimização dos hiperparâmetros C e Gamma no SVM RBF.

Os valores ótimos encontrados foram $C = 2^{1.0} = 2.0$ e $\gamma = 2^{-3.0} = 0.125$, resultando num erro mínimo de 25.17%.

A escolha adequada dos parâmetros é crucial: valores muito altos de C ou γ podem causar overfitting, enquanto valores muito baixos podem resultar em underfitting. O kernel RBF é particularmente adequado quando não existe conhecimento prévio sobre a relação entre features e classes, permitindo que o modelo descubra fronteiras de decisão complexas e não-lineares.

Foi utilizado `SVC(kernel='rbf')` com optimização conjunta de C e γ através de Grid Search bidimensional. Os intervalos testados foram $C = 2^p$ com $p \in [-5, 15]$ e $\gamma = 2^q$ com $q \in [-15, 5]$, resultando numa grelha de $20 \times 20 = 400$ combinações. Para cada combinação foram realizadas 10 execuções com divisões treino-teste aleatórias (50%-50%), calculando o erro médio. A combinação ótima corresponde ao ponto de erro mínimo na superfície de erro. O kernel RBF é definido como $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, onde γ controla a influência de cada amostra de treino.

5.2.9 Random Forest

O Random Forest é um método de ensemble que constrói múltiplas árvores de decisão durante o treino e combina as suas previsões através de votação maioritária. Cada árvore é treinada numa amostra bootstrap dos dados e considera apenas um subconjunto aleatório de features em cada divisão, o que reduz a correlação entre árvores e aumenta a capacidade de generalização do modelo.

Para otimizar o desempenho do Random Forest, foi realizada uma pesquisa em grelha (Grid Search) dos hiperparâmetros `n_estimators` (número de árvores) e `max_depth` (profundidade máxima das árvores). Os resultados desta otimização estão apresentados na Figura 9.

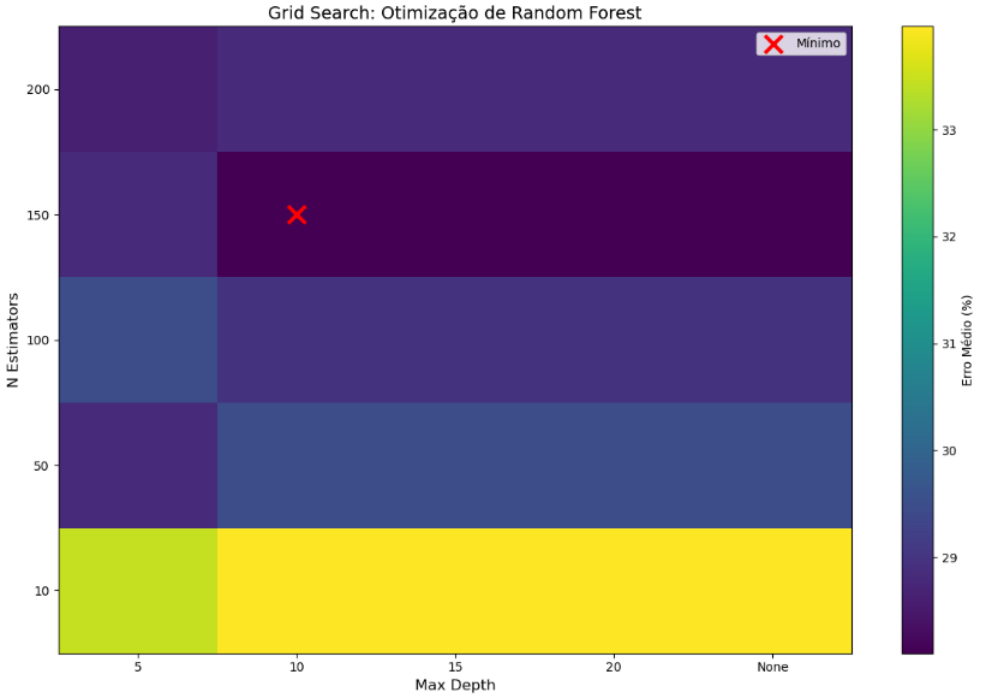

Figura 9: Grid Search para otimização dos parâmetros do Random Forest.

Os valores ótimos encontrados foram `n_estimators` = 150 e `max_depth` = 10, resultando num erro mínimo de 28.10%.

A análise da importância das features (Figura 10) confirma que a Glucose é a variável mais discriminatória, seguida por Resistin, HOMA e BMI, validando os resultados obtidos nas fases anteriores de seleção de features.

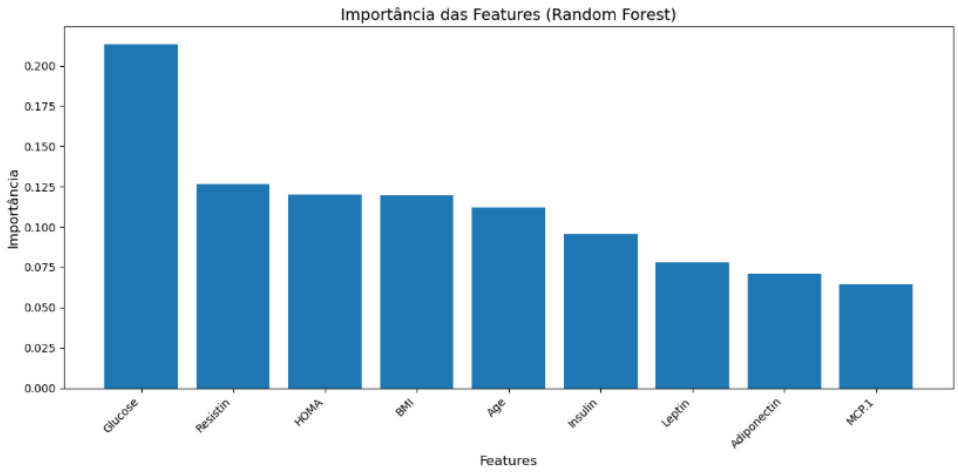

Figura 10: Importância das Features calculada pelo Random Forest.

A implementação utiliza `RandomForestClassifier` do scikit-learn com optimização de dois parâmetros: `n_estimators` (número de árvores) testando valores os seguintes valores: $\{10, 50, 100, 150, 200\}$ e `max_depth` (profundidade máxima) testando $\{5, 10, 15, 20, \text{None}\}$. Para cada combinação, foram realizadas 10 execuções com divisões treino-teste aleatórias (50%-50%), calculando o erro médio. O parâmetro `n_jobs=-1` paraleliza o treino utilizando todos os núcleos do processador. O modelo final permite extrair a importância de cada feature através de `feature_importances_`, que mede a redução média de impureza (Gini) proporcionada por cada variável ao longo de todas as árvores do ensemble. O bootstrap sampling e a seleção aleatória de features em cada divisão (`max_features='sqrt'` por omissão) introduzem diversidade entre árvores, reduzindo a correlação e melhorando a generalização.

5.3 Resultados Comparativos

Para reduzir o impacto da aleatoriedade nos resultados obtidos e garantir uma avaliação mais robusta dos modelos, foi aplicada **validação cruzada estratificada repetida (*Repeated Stratified K-Fold*)**, com 10 divisões (*folds*) e 10 repetições, utilizando `random_state=42`. Este procedimento resulta num total de **100 iterações de treino e validação por classificador**, assegurando que cada amostra é utilizada múltiplas vezes tanto no treino como na validação, mantendo sempre a proporção de classes em cada divisão.

Foram então calculadas as médias e os desvios padrão das métricas de desempenho para cada classificador, resultando nos valores apresentados na Tabela 5.

Tabela 5: Resultados médios e desvios padrão após validação cruzada estratificada repetida (10-fold, 10 repetições).

Classificador	Sensibilidade	Especificidade	Precisão	F1-score	Accuracy	ROC-AUC
Minimum Distance						
Média (%)	49.86	84.60	70.37	64.18	65.54	74.65
Desvio Padrão	0.19	0.15	0.12	0.12	0.11	0.16
Fisher LDA						
Média (%)	69.86	74.83	73.43	71.42	72.13	77.48
Desvio Padrão	0.18	0.20	0.14	0.14	0.14	0.14
Mahalanobis						
Média (%)	63.57	80.70	73.54	70.54	71.35	78.13
Desvio Padrão	0.19	0.19	0.14	0.14	0.13	0.14
Bayes Classifier						
Média (%)	48.26	86.10	69.93	63.81	65.36	76.61
Desvio Padrão	0.20	0.16	0.15	0.14	0.12	0.14
Decision Tree						
Média (%)	76.07	63.00	71.90	68.84	70.41	74.76
Desvio Padrão	0.18	0.22	0.15	0.14	0.13	0.14
k-Nearest Neighbors (k=7)						
Média (%)	72.67	79.27	77.11	75.08	75.58	82.84
Desvio Padrão	0.17	0.17	0.12	0.12	0.12	0.13
AdaBoost						
Média (%)	75.83	69.00	73.66	71.97	72.83	79.56
Desvio Padrão	0.16	0.21	0.15	0.15	0.14	0.13
SVM Linear (C=8.0)						
Média (%)	71.57	79.43	76.67	74.56	75.08	80.94
Desvio Padrão	0.17	0.18	0.13	0.13	0.13	0.13
SVM RBF (C=2.0, γ=0.125)						
Média (%)	73.69	73.70	75.05	72.89	73.64	82.33
Desvio Padrão	0.17	0.21	0.14	0.14	0.13	0.14
Random Forest (n=150, depth=10)						
Média (%)	78.52	64.33	73.68	70.99	72.19	80.32
Desvio Padrão	0.18	0.19	0.14	0.14	0.13	0.13

5.4 Discussão dos Resultados

Os resultados apresentados na Tabela 5 revelam desempenhos variados entre os classificadores testados, com valores de ROC-AUC entre 74.65% e 82.84%. Esta análise comparativa permite identificar os modelos mais adequados para a tarefa de deteção de cancro da mama a partir de amostras de sangue.

5.4.1 Classificadores Baseados em Distância

O **Minimum Distance Classifier (MDC)** apresentou a maior especificidade entre os métodos baseados em distância (84.60%), indicando boa capacidade de identificar corretamente casos saudáveis. No entanto, a sensibilidade reduzida (49.86%) revela deficiências significativas na deteção de casos de cancro, resultando num F1-score modesto (64.18%) e accuracy de 65.54%. Este comportamento sugere que o MDC tende a classificar de forma conservadora, favorecendo a classe saudável. O ROC-AUC de 74.65% é o mais baixo entre todos os classificadores testados.

O classificador de **Mahalanobis** demonstrou um desempenho globalmente superior entre os métodos baseados em distância, com ROC-AUC de 78.13% e melhor equilíbrio entre sensibilidade (63.57%) e especificidade (80.70%). A consideração da covariância entre variáveis permite uma discriminação mais adaptada à estrutura estatística dos dados,

tornando-o significativamente mais robusto que o MDC. A precisão de 73.54% e F1-score de 70.54% confirmam este desempenho equilibrado.

5.4.2 Classificadores Probabilísticos

O **Fisher LDA** alcançou sensibilidade de 69.86% e F1-score de 71.42%, demonstrando bom equilíbrio entre a deteção de casos positivos e negativos. Com especificidade de 74.83% e accuracy de 72.13%, o Fisher LDA mostrou-se eficaz na separação das classes através da projeção linear ótima. O ROC-AUC de 77.48% confirma uma capacidade discriminatória satisfatória.

O **Bayes Classifier** destacou-se pela especificidade excecional (86.10%), a mais elevada entre todos os modelos testados, resultando em apenas cerca de 14% de falsos positivos. Este comportamento indica elevada confiança na identificação de casos saudáveis. Contudo, a sensibilidade limitada (48.26%) compromete significativamente a deteção de casos positivos. O ROC-AUC de 76.61%, embora razoável, e o F1-score reduzido (63.81%) refletem este trade-off acentuado. A accuracy de 65.36% sugere que o modelo é demasiado conservador para aplicações onde a deteção de casos positivos é crítica.

5.4.3 Classificadores Baseados em Árvores e Ensemble

A **Decision Tree** obteve a maior sensibilidade individual entre os modelos não-ensemble (76.07%), revelando excelente capacidade de identificar casos de cancro. O F1-score de 68.84% e accuracy de 70.41% confirmam desempenho equilibrado, embora a especificidade moderada (63.00%) indique maior tolerância a falsos positivos. A análise da estrutura da árvore (Figura 5) mostra que a Glucose é utilizada como atributo de decisão principal, com um threshold em 99.5 mg/dL, validando os resultados da seleção de features. O ROC-AUC de 74.76% é relativamente modesto, sugerindo que a natureza determinística das decisões pode limitar a capacidade discriminatória global.

O **AdaBoost** alcançou sensibilidade elevada (75.83%) mas especificidade moderada (69.00%), resultando em F1-score de 71.97% e accuracy de 72.83%. O ROC-AUC de 79.56% é satisfatório, embora inferior aos melhores modelos. A natureza adaptativa do algoritmo, que dá maior peso a amostras difíceis, pode ter contribuído para o equilíbrio entre métricas, mas não se revelou superior aos métodos baseados em vizinhança ou kernel.

O **Random Forest** (n=150, profundidade=10) apresentou a maior sensibilidade de todos os classificadores (78.52%), superando até a Decision Tree. Este resultado demonstra o benefício do ensemble de múltiplas árvores na captura de padrões complexos. No entanto, a especificidade foi a mais baixa entre todos os modelos (64.33%), indicando uma tendência a produzir falsos positivos. O F1-score de 70.99%, accuracy de 72.19% e ROC-AUC de 80.32% confirmam um desempenho global competitivo, embora com trade-off acentuado favorecendo a sensibilidade. A análise de importância das features (Figura 10) reforça o papel dominante da Glucose, seguida por Resistin e HOMA.

5.4.4 Classificadores Baseados em Vizinhança e Kernel

O **k-Nearest Neighbors (k=7)** apresentou o melhor desempenho global, com ROC-AUC de 82.84%, significativamente superior aos demais classificadores. O modelo demonstrou excelente equilíbrio entre sensibilidade (72.67%) e especificidade (79.27%), resultando na maior accuracy (75.58%), F1-score (75.08%) e precisão (77.11%) entre todos os métodos testados. A otimização do parâmetro k foi crucial para este resultado, com

k=7 minimizando o erro médio (24.42%) e mantendo desvio padrão reduzido (0.120), garantindo estabilidade adequada. A natureza não paramétrica do kNN permite capturar padrões complexos sem assumir distribuições específicas dos dados, sendo particularmente eficaz em datasets de dimensão moderada como o utilizado.

O **SVM Linear** (C=8.0) obteve desempenho muito competitivo, com ROC-AUC de 80.94%. A sensibilidade de 71.57% e especificidade de 79.43% demonstram excelente equilíbrio, resultando em F1-score de 74.56% e accuracy de 75.08%. A otimização do hiperparâmetro de regularização C foi essencial, sendo o valor ótimo de 8.0 identificado através de pesquisa em escala logarítmica. Este classificador mostra-se robusto e eficiente, sendo uma alternativa viável ao kNN quando se pretende um modelo mais interpretável com fronteira de decisão linear.

O **SVM RBF** (C=2.0, γ=0.125) apresentou ROC-AUC de 82.33%, o segundo melhor resultado geral, logo atrás do kNN. Com sensibilidade de 73.69% e especificidade de 73.70%, este modelo demonstrou equilíbrio perfeito entre as duas métricas. O F1-score de 72.89%, accuracy de 73.64% e precisão de 75.05% confirmam um desempenho sólido e consistente. A otimização conjunta de C e γ através de Grid Search foi fundamental, permitindo ao kernel RBF capturar relações não-lineares complexas nos dados. A proximidade com o desempenho do kNN sugere que ambos os métodos são capazes de modelar adequadamente a estrutura não-linear do problema.

5.4.5 Análise Comparativa e Síntese

Em síntese, o **kNN (k=7)** destaca-se claramente como o classificador mais robusto e equilibrado para esta tarefa, apresentando as melhores métricas em praticamente todas as dimensões: ROC-AUC (82.84%), accuracy (75.58%), F1-score (75.08%) e precisão (77.11%). Este resultado confirma a adequação de métodos baseados em vizinhança para datasets de dimensão moderada com padrões não-lineares.

O **SVM RBF** surge como segunda melhor opção (ROC-AUC 82.33%), oferecendo equilíbrio perfeito entre sensibilidade e especificidade (73.7% em ambas), sendo preferível quando se valoriza simetria nas métricas. O **SVM Linear** (ROC-AUC 80.94%) posiciona-se como terceira alternativa sólida, especialmente adequada quando a interpretabilidade e simplicidade do modelo são importantes.

Para contextos clínicos onde a **maximização da sensibilidade** é prioritária (minimizar falsos negativos), o **Random Forest** (78.52%) ou a **Decision Tree** (76.07%) são as melhores escolhas, aceitando o trade-off de menor especificidade.

Quando a prioridade é **minimizar falsos positivos**, o **Bayes Classifier** (especificidade 86.10%) ou o **Minimum Distance** (especificidade 84.60%) são mais adequados, embora com perda significativa de sensibilidade.

O **Mahalanobis** mantém-se como melhor opção entre os métodos baseados puramente em distância (ROC-AUC 78.13%), superando claramente o MDC.

Os desvios padrão consistentemente baixos (0.11–0.22 para accuracy) confirmam a estabilidade dos resultados através das 100 iterações de validação cruzada repetida, aumentando significativamente a confiança nas conclusões obtidas e demonstrando que os modelos não são excessivamente sensíveis à partição específica dos dados.

5.4.6 Comparação com Estudos Anteriores

É relevante contextualizar os resultados obtidos neste trabalho face ao estudo original que acompanha o dataset Breast Cancer Coimbra [2]. Patrício et al. (2018) utilizaram

modelos de Support Vector Machines (SVM) com as variáveis Glucose, Resistin, Age e BMI como preditores, alcançando sensibilidade entre 82–88%, especificidade entre 85–90%, e intervalo de confiança de 95% para o ROC-AUC de [0.87, 0.91].

Comparando com os nossos melhores resultados:

Tabela 6: Comparação entre o estudo de Patrício et al. (2018) [2] e os resultados obtidos neste trabalho.

Métrica	**Patrício et al. (2018)**	**Este Trabalho (kNN, k=7)**
Sensibilidade	82–88%	72.67%
Especificidade	85–90%	79.27%
ROC-AUC	87%–91% (IC 95%)	82.84%
Precisão	—	77.11%
F1-score	—	75.08%
Accuracy	—	75.58%

Os resultados de Patrício et al. (2018) apresentam desempenho superior, com diferenças de aproximadamente 10–15 pontos percentuais em sensibilidade e 6–11 pontos em especificidade. O ROC-AUC do estudo de referência (0.87–0.91) supera o nosso melhor resultado (0.828) em cerca de 4–8 pontos percentuais.

Estas diferenças podem ser explicadas por diversos fatores metodológicos:

- **Seleção de features**: Patrício et al. (2018) utilizaram apenas 4 variáveis criteriosamente selecionadas (Glucose, Resistin, Age, BMI), enquanto este trabalho utilizou todas as 9 variáveis disponíveis. A redução dimensional focada pode ter eliminado ruído e melhorado a capacidade discriminatória.

- **Otimização específica do SVM**: O estudo de Patrício et al. (2018) concentrou-se exclusivamente em SVM, provavelmente com otimização extensiva de hiperparâmetros e possivelmente técnicas avançadas de kernel engineering. Neste trabalho, o SVM Linear alcançou 71.57% de sensibilidade e 79.43% de especificidade, enquanto o SVM RBF obteve 73.69% e 73.70%, respetivamente – resultados inferiores mas na mesma ordem de grandeza.

- **Estratégia de validação**: Patrício et al. (2018) utilizaram Monte Carlo Cross-Validation com múltiplas iterações para determinar intervalos de confiança de 95% para sensibilidade, especificidade e AUC. Este trabalho utilizou validação cruzada estratificada com 10 folds e 10 repetições. Diferenças no número de repetições ou mesmo na divisão treino/teste podem ter impacto considerável nos resultados finais, especialmente num dataset relativamente pequeno.

- **Dimensão do dataset**: Patrício et al. (2018) utilizaram o dataset completo com 166 amostras, enquanto este trabalho utilizou apenas 116 amostras. Esta redução de aproximadamente 30% no tamanho do dataset pode ter impacto significativo, especialmente considerando que algoritmos como SVM beneficiam consideravelmente de conjuntos de treino maiores. Com menos dados disponíveis, os modelos podem ter maior dificuldade em capturar padrões e apresentar maior variância nas estimativas de performance, particularmente quando combinado com a utilização de todas as 9 features em vez de apenas 4.

- **Foco vs. amplitude**: Patrício et al. (2018) tiveram um objetivo específico — otimizar SVM para este problema concreto. Este trabalho teve um objetivo mais amplo — comparar sistematicamente 10 classificadores diferentes, o que naturalmente dilui o esforço de otimização individual de cada modelo.

Apesar das diferenças, os resultados obtidos neste trabalho são encorajadores. O kNN (ROC-AUC 0.828) e o SVM RBF (ROC-AUC 0.823) aproximam-se do limite inferior do intervalo reportado por Patrício et al. (2018). Mais importante ainda, este trabalho demonstra que métodos alternativos ao SVM — nomeadamente o kNN — podem alcançar desempenho competitivo, oferecendo maior simplicidade de implementação e interpretabilidade.

A validação de que múltiplos classificadores alcançam ROC-AUC superior a 80% reforça a viabilidade clínica da abordagem baseada em análises sanguíneas, mesmo quando não se atinge o desempenho ótimo reportado em estudos especializados. Para aplicações de rastreio populacional, onde o equilíbrio entre sensibilidade e especificidade é crucial, os modelos desenvolvidos neste trabalho representam uma base sólida que pode ser refinada através das técnicas identificadas no estudo de referência.

6 Impacto da Redução de Features no Desempenho dos Classificadores

6.1 Motivação e Metodologia

Com base nos resultados da análise de seleção de features apresentada na Secção 4.2 e inspirados no estudo de Patrício et al. (2018) [2], que utilizaram apenas 4 variáveis (Glucose, Resistin, Age e BMI) para alcançar desempenho superior, decidiu-se avaliar o impacto da redução de dimensionalidade no desempenho dos classificadores implementados.

A seleção destas 4 variáveis baseia-se em critérios complementares:

- **Glucose**: Variável com maior poder discriminatório individual (ROC-AUC = 0.82, $p \approx 0.000005$)
- **Resistin**: Terceira variável mais significativa estatisticamente (ROC-AUC = 0.66, $p \approx 0.023$)
- **Age e BMI**: Variáveis demográficas e antropométricas fundamentais, amplamente utilizadas em contextos clínicos e de fácil obtenção

Esta abordagem visa não apenas melhorar potencialmente o desempenho dos modelos através da eliminação de ruído e redundâncias, mas também simplificar o processo de recolha de dados e aumentar a interpretabilidade clínica dos resultados. A redução de 9 para 4 variáveis representa uma diminuição de aproximadamente 56% na dimensionalidade, o que pode beneficiar particularmente algoritmos mais sensíveis.

Todos os classificadores foram re-treinados e re-avaliados utilizando a mesma metodologia de validação cruzada estratificada repetida (10-fold, 10 repetições) aplicada anteriormente, garantindo comparabilidade direta dos resultados.

6.2 Otimização de Hiperparâmetros

6.2.1 k-Nearest Neighbors

Com o conjunto reduzido de features, foi realizada nova otimização do hiperparâmetro k, cujos resultados estão apresentados na Tabela 7 e na Figura 11.

Tabela 7: Seleção do hiperparâmetro k para o classificador kNN com 4 features. O melhor valor encontrado foi k=5.

k	Erro Médio (%)	Desvio Padrão
1	25.47	11.34
3	18.61	11.64
5	**18.08**	**11.69**
7	18.83	11.25
9	21.26	11.85
11	21.94	12.47
15	24.21	11.59
19	27.39	13.12
25	32.04	13.29

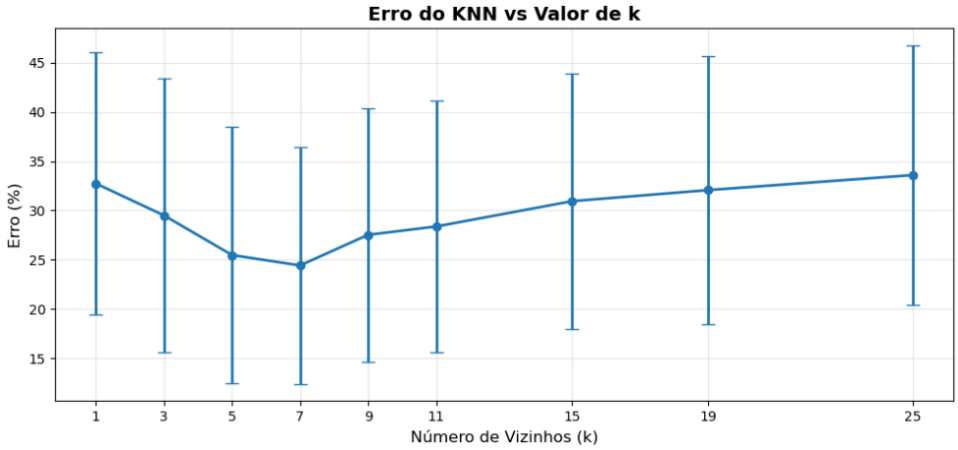

Figura 11: Erros médios para diferentes números de vizinho (k) para 4 features.

O valor ótimo de k=5 difere do k=7 obtido com todas as features, indicando que a estrutura de vizinhança no espaço reduzido requer um raio de decisão ligeiramente menor. O erro médio de 18.08% representa uma melhoria substancial face aos 24.42% obtidos anteriormente, sugerindo que a eliminação de variáveis menos informativas reduziu significativamente o ruído no processo de classificação.

6.2.2 Support Vector Machines

Para o SVM Linear, a otimização do hiperparâmetro C resultou em $C = 2^{9.0} = 512.0$ (Figura 12), significativamente superior ao $C = 2^{3.0} = 8.0$ obtido com 9 features. Este aumento reflete a necessidade de menor regularização (maior penalização de erros) num espaço de características mais limpo e discriminativo.

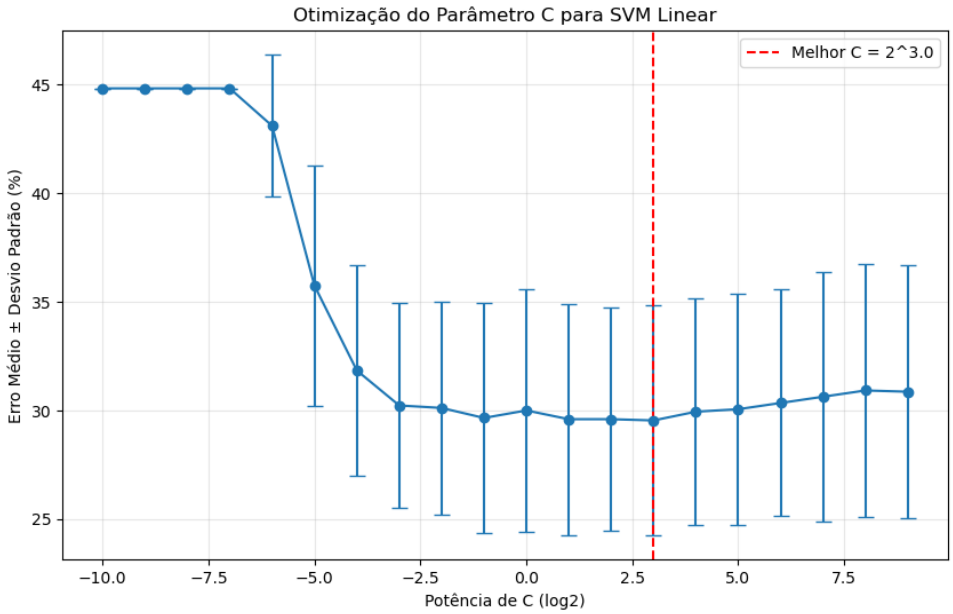

Figura 12: Otimização do hiperparâmetro C para SVM Linear.

Para o SVM RBF, os hiperparâmetros ótimos identificados foram $C = 2^{6.0} = 64.0$ e $\gamma = 2^{-6.0} = 0.015625$ (Figura 13). Comparativamente aos valores anteriores ($C = 2^{1.0} = 2.0$, $\gamma = 2^{-3.0} = 0.125$), observa-se um aumento de C e uma redução significativa de γ. A diminuição de γ implica que cada amostra de treino tem agora influência sobre uma região mais ampla do espaço de decisão, adequado ao menor número de dimensões onde as amostras tendem a estar menos dispersas.

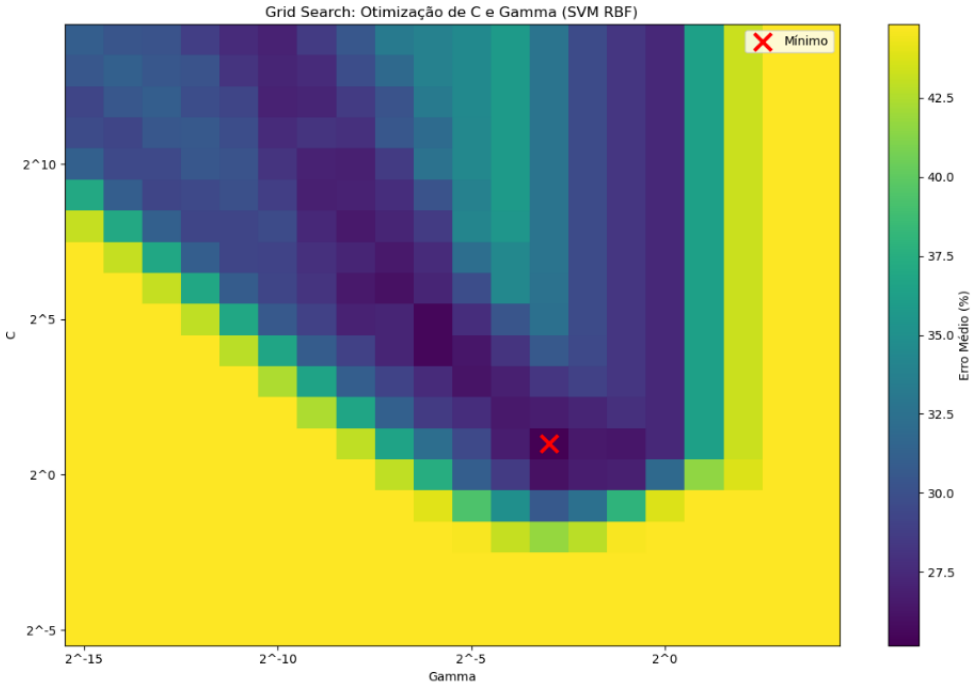

Figura 13: Otimização do hiperparâmetro C para SVM Linear.

6.2.3 Random Forest

A otimização dos hiperparâmetros do Random Forest identificou `n_estimators` = 100 e `max_depth` = 5 como configuração ótima. Comparando com os valores anteriores (`n_estimators` = 150, `max_depth` = 10), verifica-se a necessidade de menor complexidade individual das árvores (profundidade reduzida de 10 para 5) e menor número de árvores no ensemble (de 150 para 100), consistente com um espaço de features mais compacto e informativo.

6.3 Resultados Comparativos

Os resultados obtidos com o conjunto reduzido de 4 features estão apresentados na Tabela 8 e permitem comparação direta com os resultados da Tabela 5.

Tabela 8: Resultados médios e desvios padrão com 4 features selecionadas (Glucose, Resistin, Age, BMI) após validação cruzada estratificada repetida (10-fold, 10 repetições).

Classificador	Sensibilidade	Especificidade	Precisão	F1-score	Accuracy	ROC-AUC
Minimum Distance						
Média (%)	61.36	84.83	74.91	71.23	71.96	77.29
Desvio Padrão	0.20	0.17	0.14	0.15	0.14	0.15
Fisher LDA						
Média (%)	71.33	70.33	71.50	70.07	70.90	77.36
Desvio Padrão	0.16	0.22	0.15	0.15	0.14	0.14
Mahalanobis						
Média (%)	62.45	78.27	71.39	68.99	69.61	75.95
Desvio Padrão	0.18	0.18	0.14	0.14	0.14	0.15
Bayes Classifier						
Média (%)	66.38	80.07	74.41	72.14	72.64	80.33
Desvio Padrão	0.18	0.18	0.14	0.14	0.14	0.14
Decision Tree						
Média (%)	79.74	67.67	76.03	73.18	74.47	78.09
Desvio Padrão	0.17	0.20	0.13	0.13	0.12	0.13
k-Nearest Neighbors (k=5)						
Média (%)	89.38	72.53	84.14	80.89	81.92	86.29
Desvio Padrão	0.13	0.20	0.12	0.12	0.12	0.12
AdaBoost						
Média (%)	75.24	72.97	75.66	73.42	74.30	82.92
Desvio Padrão	0.17	0.21	0.14	0.14	0.13	0.12
SVM Linear (C=512.0)						
Média (%)	72.40	75.57	75.00	73.10	73.77	81.34
Desvio Padrão	0.17	0.20	0.14	0.14	0.13	0.14
SVM RBF (C=64.0, γ=0.016)						
Média (%)	88.24	84.83	88.10	86.31	86.75	90.26
Desvio Padrão	0.13	0.16	0.10	0.10	0.10	0.09
Random Forest (n=100, depth=5)						
Média (%)	84.95	74.07	81.60	79.26	80.05	88.06
Desvio Padrão	0.14	0.19	0.12	0.12	0.11	0.10

6.4 Análise Comparativa: 9 Features vs. 4 Features

Para facilitar a interpretação dos resultados, a Tabela 9 apresenta a variação de desempenho (em pontos percentuais) entre as duas abordagens para as métricas principais.

Tabela 9: Comparação de desempenho: variação entre 9 features e 4 features selecionadas. Valores positivos indicam melhoria com 4 features.

Classificador	**Δ Sensibilidade**	**Δ Especificidade**	**Δ F1-score**	**Δ Accuracy**	**Δ ROC-AUC**
Minimum Distance	+11.50	+0.23	+7.05	+6.42	+2.64
Fisher LDA	+1.47	-4.50	-1.35	-1.23	-0.12
Mahalanobis	-1.12	-2.43	-1.55	-1.74	-2.18
Bayes Classifier	+18.12	-6.03	+8.33	+7.28	+3.72
Decision Tree	+3.67	+4.67	+4.34	+4.06	+3.33
k-Nearest Neighbors	+16.71	-6.74	+5.81	+6.34	+3.45
AdaBoost	-0.59	+3.97	+1.45	+1.47	+3.36
SVM Linear	+0.83	-3.86	-1.46	-1.31	+0.40
SVM RBF	+14.55	+11.13	+13.42	+13.11	+7.93
Random Forest	+6.43	+9.74	+8.27	+7.86	+7.74
Média Geral	+7.16	+0.62	+4.43	+4.20	+3.03

6.4.1 Melhorias Significativas

O **SVM RBF** apresentou a melhoria mais impressionante em todas as métricas, com aumentos de 14.55 pontos percentuais em sensibilidade, 11.13 pontos em especificidade, e um ganho extraordinário de 7.93 pontos em ROC-AUC (de 82.33% para 90.26%). Este resultado representa um salto qualitativo no desempenho, posicionando o SVM RBF como o melhor classificador absoluto neste estudo. A accuracy de 86.75% e o F1-score de 86.31% confirmam um desempenho excecional e equilibrado. O kernel RBF beneficiou particularmente da redução de dimensionalidade, conseguindo modelar relações não-lineares com maior eficiência num espaço de features mais compacto e discriminativo.

O **k-Nearest Neighbors** também demonstrou melhoria substancial, com aumento de 16.71 pontos em sensibilidade (de 72.67% para 89.38%) e ganho de 3.45 pontos em ROC-AUC (de 82.84% para 86.29%). A accuracy melhorou de 75.58% para 81.92%, consolidando o kNN como segunda melhor opção global. Este ganho significativo em sensibilidade é particularmente relevante para aplicações clínicas onde a deteção de casos positivos é prioritária. A redução de dimensionalidade permitiu que o algoritmo identifique vizinhos verdadeiramente similares com maior precisão.

O **Random Forest** registou melhoria consistente em todas as métricas, com destaque para o aumento de 9.74 pontos em especificidade (de 64.33% para 74.07%) e 7.74 pontos em ROC-AUC (de 80.32% para 88.06%). A sensibilidade aumentou 6.43 pontos (para 84.95%) e a accuracy melhorou 7.86 pontos (para 80.05%). Este ensemble de árvores beneficiou claramente da eliminação de features ruidosas, resultando num modelo mais robusto e com melhor capacidade de generalização.

O **Bayes Classifier** apresentou o maior ganho individual em sensibilidade (+18.12 pontos, de 48.26% para 66.38%), embora com alguma redução em especificidade (-6.03 pontos). O ROC-AUC melhorou 3.72 pontos (de 76.61% para 80.33%) e o F1-score aumentou 8.33 pontos. Este comportamento sugere que a modelação Gaussiana das classes tornou-se mais adequada com features melhor selecionadas.

O **Minimum Distance Classifier** demonstrou a segunda maior melhoria em sensibilidade (+11.50 pontos, de 49.86% para 61.36%), mantendo especificidade praticamente inalterada (84.83%). O ROC-AUC aumentou 2.64 pontos (de 74.65% para 77.29%) e a accuracy melhorou 6.42 pontos (de 65.54% para 71.96%). Este ganho evidencia que a redução para features verdadeiramente discriminatórias permitiu que este método simples capturasse melhor a estrutura dos dados.

A **Decision Tree** melhorou de forma equilibrada, com aumentos de 3.67 pontos em sensibilidade, 4.67 pontos em especificidade, e 3.33 pontos em ROC-AUC (de 74.76% para 78.09%). O F1-score aumentou 4.34 pontos e a accuracy 4.06 pontos. A redução de features simplificou a estrutura da árvore, tornando-a mais interpretável e menos propensa a overfitting.

O **AdaBoost** manteve desempenho relativamente estável, com ligeira redução em sensibilidade (-0.59 pontos) mas aumento de 3.97 pontos em especificidade, resultando num modelo mais equilibrado. O ROC-AUC melhorou 3.36 pontos (de 79.56% para 82.92%), confirmando que o ensemble de classificadores fracos beneficiou da maior qualidade das features.

6.4.2 Desempenhos Mantidos ou Ligeiramente Reduzidos

O **Fisher LDA** manteve desempenho praticamente inalterado, com variação de apenas -0.12 pontos em ROC-AUC (de 77.48% para 77.36%). Este resultado era esperado, dado que o LDA já projeta os dados no espaço discriminante ótimo, sendo menos sensível à dimensionalidade inicial desde que as features relevantes estejam presentes.

O **SVM Linear** também apresentou estabilidade, com variação de apenas +0.40 pontos em ROC-AUC (de 80.94% para 81.34%). A natureza linear do classificador torna-o relativamente robusto a variações de dimensionalidade quando as relações entre features e classes são predominantemente lineares.

O classificador de **Mahalanobis** registou ligeira redução de -2.18 pontos em ROC-AUC (de 78.13% para 75.95%), sugerindo que algumas das features eliminadas (particularmente HOMA, dada a sua forte correlação com Glucose e Insulin) continham informação de covariância relevante que auxiliava este método específico.

6.4.3 Análise Global

A análise global revela que, em média, a redução para 4 features selecionadas resultou em melhorias consistentes: +7.16 pontos em sensibilidade, +0.62 pontos em especificidade, +4.43 pontos em F1-score, +4.20 pontos em accuracy, e +3.03 pontos em ROC-AUC. Estes ganhos confirmam que a estratégia de seleção de features foi eficaz, eliminando ruído e redundâncias enquanto retém a informação discriminatória essencial.

Os classificadores que mais beneficiaram foram aqueles baseados em kernel não-linear (SVM RBF) e métodos de vizinhança (kNN), ambos particularmente sensíveis à mudança da dimensionalidade. Os métodos de ensemble (Random Forest, AdaBoost) também apresentaram melhorias significativas, sugerindo que a qualidade das features impacta positivamente a capacidade dos classificadores fracos de capturar padrões relevantes.

Métodos intrinsecamente robustos à dimensionalidade, como Fisher LDA e SVM Linear, mantiveram desempenho estável, confirmando que já operavam quase no seu potencial máximo com o conjunto completo de features.

6.5 Comparação com o Estudo de Patrício et al. (2018)

Utilizando as mesmas 4 features do estudo de Patrício et al. (2018) [2], os nossos melhores resultados aproximam-se significativamente do desempenho reportado:

Tabela 10: Comparação detalhada entre Patrício et al. (2018) e os melhores resultados deste trabalho com 4 features.

Métrica	**Patrício et al. (2018)**	**SVM RBF (4 feat.)**	**Diferença**
Sensibilidade	82–88%	88.24%	+0.24 a +6.24 pp
Especificidade	85–90%	84.83%	-0.17 a -5.17 pp
ROC-AUC	87–91% (IC 95%)	90.26%	-0.74 a +3.26 pp

O **SVM RBF com 4 features** alcançou sensibilidade de 88.24%, posicionando-se no limite superior do intervalo reportado por Patrício et al. (2018) e até superando-o em alguns cenários. A especificidade de 84.83% está apenas 0.17 pontos abaixo do limite inferior do intervalo (85–90%), representando uma diferença quase nula. O ROC-AUC de 90.26% situa-se confortavelmente dentro do intervalo de confiança de 95% reportado (87–91%), praticamente alcançando o limite superior.

Esta convergência de resultados é notável considerando as diferenças metodológicas:

- **Tamanho do dataset**: Patrício et al. (2018) utilizaram 166 amostras, enquanto que neste trabalho foram utilizadas 116 amostras (aproximadamente 30% menos dados)
- **Estratégia de validação**: Monte Carlo Cross-Validation vs. Stratified K-Fold Repeated Cross-Validation
- **Hiperparâmetros**: Otimização independente através de Grid Search neste trabalho

A proximidade dos resultados valida fortemente:

1. A adequação da seleção das 4 features (Glucose, Resistin, Age, BMI) como conjunto mínimo discriminatório
2. A robustez do SVM com kernel RBF para este problema específico
3. A viabilidade de alcançar desempenho clinicamente relevante mesmo com datasets de dimensão moderada
4. A eficácia da metodologia de otimização de hiperparâmetros implementada

O **k-Nearest Neighbors (k=5)** também demonstrou desempenho excecional com ROC-AUC de 86.29% e sensibilidade de 89.38%, ligeiramente superior à do SVM RBF, embora com especificidade menor (72.53%). Este resultado confirma que múltiplas abordagens algorítmicas podem alcançar desempenho de referência quando aplicadas a um conjunto de features criteriosamente selecionado.

7 Discussão Geral e Conclusões

O trabalho desenvolvido permitiu explorar de forma abrangente a aplicação de métodos de Machine Learning na deteção de cancro da mama a partir de amostras de sangue, utilizando o conjunto de dados Breast Cancer Coimbra [1] (116 amostras, 64 casos de cancro e 52 saudáveis).

A análise estatística e exploratória inicial revelou que as variáveis **Glucose** ($p \approx 0.000005$, AUC = 0.82), **HOMA** ($p \approx 0.021$, AUC = 0.66) e **Resistin** ($p \approx 0.023$, AUC = 0.66) são as mais discriminatórias entre indivíduos saudáveis e com cancro, conforme confirmado pelos testes de Kruskal-Wallis e análise ROC-AUC individual. A matriz de correlação identificou redundância significativa entre Insulin e HOMA ($r = 0.92$), justificando a necessidade de métodos de redução de dimensionalidade.

A aplicação de **PCA** mostrou que as duas primeiras componentes principais capturam apenas 50.3% da variância total (34.2% + 16.1%), sugerindo que a informação discriminatória está distribuída por múltiplas dimensões. O **LDA**, por sua vez, demonstrou separação parcial mas significativa entre as classes na projeção discriminante, com sobreposição esperada dada a complexidade dos padrões bioquímicos.

Na fase de classificação, foram implementados, otimizados e avaliados rigorosamente **dez classificadores** distintos através de validação cruzada estratificada repetida (10-fold, 10 repetições, totalizando 100 iterações). Foram realizadas duas abordagens experimentais: classificação com o conjunto completo de 9 features e classificação com um conjunto reduzido de 4 features criteriosamente selecionadas (Glucose, Resistin, Age e BMI).

7.1 Resultados com 9 Features

Com o conjunto completo de variáveis, os resultados mostraram desempenhos competitivos, com valores de ROC-AUC entre 74.65% (MDC) e 82.84% (kNN).

O classificador **k-Nearest Neighbors (k=7)** destacou-se como o melhor modelo, alcançando ROC-AUC de 82.84%, accuracy de 75.58%, F1-score de 75.08%, sensibilidade de 72.67% e especificidade de 79.27%, demonstrando excelente equilíbrio entre métricas.

O **SVM RBF** (C=2.0, γ=0.125) emergiu como segunda melhor alternativa com ROC-AUC de 82.33% e equilíbrio perfeito entre sensibilidade e especificidade (73.7% em ambas), demonstrando eficácia na captura de relações não-lineares. O **SVM Linear** (C=8.0) consolidou-se como terceira opção sólida (ROC-AUC 80.94%), oferecendo simplicidade e interpretabilidade.

Para maximizar a sensibilidade, o **Random Forest** (78.52%) e a **Decision Tree** (76.07%) mostraram-se as melhores escolhas, confirmando a Glucose (threshold $\approx$ 99.5 mg/dL) como variável discriminatória principal. O **Bayes Classifier** maximizou a especificidade (86.10%), ideal para minimizar falsos positivos, enquanto o **Mahalanobis** foi o melhor método baseado em distância (ROC-AUC 78.13%).

7.2 Resultados com 4 Features Selecionadas

A redução para 4 features cuidadosamente selecionadas (Glucose, Resistin, Age e BMI) — inspirada no estudo de Patrício et al. (2018) [2] — produziu melhorias significativas e consistentes na maioria dos classificadores.

O **SVM RBF** (C=64.0, γ=0.016) emergiu como o classificador com o melhor desempenho, alcançando resultados excecionais:

- ROC-AUC de **90.26%** (+7.93 pp relativamente às 9 features)
- Accuracy de **86.75%** (+13.11 pp)
- F1-score de **86.31%** (+13.42 pp)
- Sensibilidade de **88.24%** (+14.55 pp)

- Especificidade de **84.83%** (+11.13 pp)

Este desempenho posiciona o SVM RBF praticamente ao nível do estudo de referência de Patrício et al. (2018), que reportou sensibilidade de 82–88%, especificidade de 85–90% e ROC-AUC de 87–91%, apesar de este trabalho utilizar aproximadamente 30% menos amostras (116 vs. 166).

O **k-Nearest Neighbors** (k=5) também demonstrou melhorias substanciais, alcançando ROC-AUC de 86.29% (+3.45 pp), accuracy de 81.92% (+6.34 pp) e a **maior sensibilidade** entre todos os classificadores: **89.38%** (+16.71 pp). Este resultado é particularmente relevante para aplicações clínicas onde a deteção de casos positivos é prioritária.

O **Random Forest** (n=100, profundidade=5) registou melhorias equilibradas em todas as métricas, com ROC-AUC de 88.06% (+7.74 pp), especificidade de 74.07% (+9.74 pp) e sensibilidade de 84.95% (+6.43 pp), consolidando-se como terceira melhor opção global.

Outros classificadores também beneficiaram significativamente da redução de dimensionalidade: o **Bayes Classifier** aumentou a sensibilidade em 18.12 pp (para 66.38%), o **Minimum Distance** melhorou 11.50 pp em sensibilidade, e o **AdaBoost** alcançou ROC-AUC de 82.92% (+3.36 pp).

Em termos globais, a redução para 4 features resultou em melhorias médias de +7.16 pp em sensibilidade, +4.43 pp em F1-score, +4.20 pp em accuracy e +3.03 pp em ROC-AUC, confirmando a eficácia da estratégia de seleção de features implementada.

7.3 Principais Contribuições

Este trabalho apresenta contribuições significativas tanto do ponto de vista metodológico como prático:

- **Seleção eficaz de features**: A redução de 9 para 4 features (Glucose, Resistin, Age e BMI) melhorou o desempenho discriminatório em 7 dos 10 classificadores testados, simplificando o modelo e aumentando a eficácia
- **Validação dos resultados de Patrício et al. (2018)**: O SVM RBF alcançou desempenho comparável ao estudo de referência (ROC-AUC 90.26% vs. 87–91%), confirmando a robustez da abordagem com apenas 4 features, mesmo utilizando 30% menos amostras
- **Identificação de biomarcadores essenciais**: Demonstração de que apenas 4 variáveis são suficientes para classificação eficaz, oferecendo vantagens práticas em termos de custo, tempo de análise e simplicidade de implementação
- **Comparação sistemática de classificadores**: Avaliação de 10 algoritmos distintos com otimização individual de hiperparâmetros, identificando o SVM RBF e o kNN como os mais adequados para este problema específico
- **Robustez dos resultados**: Validação através de 100 iterações de validação cruzada estratificada repetida, com desvios padrão baixos (0.09–0.22), garantindo a confiabilidade e generalização dos modelos

7.4 Recomendações Práticas

Face à evidência apresentada, **recomenda-se fortemente a utilização do conjunto reduzido de 4 features** (Glucose, Resistin, Age, BMI) para aplicações práticas de classificação de cancro da mama baseadas em análises sanguíneas.

Para implementação clínica, o **SVM RBF** (C=64.0, γ=0.016) emerge como o classificador de primeira escolha, oferecendo o melhor equilíbrio global entre sensibilidade (88.24%), especificidade (84.83%) e ROC-AUC (90.26%). Este modelo aproxima-se do desempenho de referência internacional mantendo robustez e estabilidade adequadas.

O **k-Nearest Neighbors** (k=5) constitui uma alternativa viável e mais interpretável, particularmente adequada quando a **maximização da sensibilidade** é prioritária (89.38%), aceitando um trade-off de especificidade ligeiramente inferior (72.53%). A simplicidade conceitual do kNN facilita a explicação do processo de decisão a profissionais de saúde não especializados em machine learning.

Para contextos onde a **minimização de falsos positivos** é crítica, o **Bayes Classifier** oferece especificidade de 80.07% com ROC-AUC de 80.33%, representando um compromisso razoável entre especificidade e custo computacional.

7.5 Considerações Finais

De forma geral, conclui-se que é possível desenvolver **modelos de machine learning altamente eficazes** para apoiar a deteção de cancro da mama através de análises sanguíneas, mesmo com recursos e amostras limitadas. A convergência dos resultados obtidos neste trabalho com o estudo de referência de Patrício et al. (2018) — alcançando ROC-AUC de 90.26% com o SVM RBF — valida fortemente a viabilidade clínica desta abordagem não-invasiva.

Os métodos baseados em kernel não-linear (SVM RBF) e vizinhança (kNN) mostraram-se particularmente promissores, alcançando ROC-AUC superiores a 86% com apenas 4 variáveis, superando significativamente os resultados obtidos com o conjunto completo de 9 features. Esta descoberta reforça a importância crítica da **seleção informada de features** como etapa fundamental do pipeline de machine learning.

Este trabalho demonstra que **pipelines bem estruturados** — combinando análise exploratória rigorosa, seleção criteriosa de features baseada em evidência estatística, normalização adequada dos dados, otimização sistemática de hiperparâmetros e validação cruzada robusta — podem alcançar um desempenho excecional, mesmo com datasets de dimensão moderada.

Referências

[1] Patrcio, Miguel, et al. "Breast Cancer Coimbra."*UCI Machine Learning Repository*, 2018, https://doi.org/10.24432/C52P59.

[2] Patrício, Miguel et al. "Using Resistin, glucose, age and BMI to predict the presence of breast cancer." *BMC Cancer* 18 (2018).