```
# Manipulação de dados
import os
import pandas as pd
import numpy as np
from collections import Counter
import itertools

# Visualização
import matplotlib.pyplot as plt
import seaborn as sns
from tqdm import tqdm

# Processamento de imagens / Radiomics
import pylidc as pl
from pylidc.utils import consensus
from radiomics import featureextractor
import SimpleITK as sitk

# Pré-processamento e seleção de features
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import StandardScaler  # repetido, mantido
conforme original
from sklearn.impute import SimpleImputer
from sklearn.feature_selection import SelectKBest, f_classif

# Modelos e validação
from sklearn.model_selection import StratifiedGroupKFold
from sklearn.model_selection import train_test_split
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression, LassoCV
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.calibration import CalibratedClassifierCV

# Métricas
from sklearn.metrics import (
    roc_auc_score, average_precision_score, brier_score_loss,
    precision_score, recall_score, f1_score, accuracy_score,
    roc_curve, confusion_matrix
)

# Estatística
from scipy.stats import spearmanr, zscore
```

Índice

Introdução

O objetivo deste projeto é aplicar métodos de IA e Ciência de Dados para classificação de cancro do pulmão. Para tal, utilizam-se imagens de Tomografia Computorizada (CT) do tórax, com dados de nódulos pulmonares anotados por radiologistas e features extraídas em 3D e 2D. O projeto visa explorar estas informações para desenvolver sistemas capazes de avaliar automaticamente o risco de malignidade dos nódulos, apoiando decisões clínicas de forma não invasiva.

Ao descarregar o dataset objetivo deste projeto completo LIDC-IDRI, verifica-se que nem todas as séries de cada estudo estão incluídas. Isto ocorre porque apenas uma série por estudo foi anotada pelos radiologistas, ou seja, apenas essa série contém os ficheiros XML com as anotações dos nódulos pulmonares.

As restantes séries (por exemplo, reconstruções com diferentes espessuras de corte ou janelas de visualização) não contêm anotações e, consequentemente, não são incluídas no download oficial. Isto tem como objetivo reduzir o tamanho total do dataset e garantir que apenas as séries relevantes para a análise e deteção de nódulos sejam armazenadas e processadas.

Quando se utiliza o TCIA Downloader (ou o comando tcia_download através de um manifest), o sistema descarrega, por defeito, apenas a série associada às anotações. Contudo, nos casos em que um paciente não possui anotações, o sistema ainda descarrega uma série “representativa” sem ficheiros XML, de forma a manter a consistência estrutural do dataset.

É importante salientar que a ausência de ficheiros de anotações (.xml) nas pastas descarregadas não implica necessariamente que o paciente não apresente nódulos ≥ 3 mm. À primeira vista, poderia parecer que a inexistência desses ficheiros corresponde à falta de anotações radiológicas; no entanto, após análise da documentação oficial do PyLIDC e de discussões na respetiva comunidade de utilizadores, verificou-se que o PyLIDC não lê diretamente os ficheiros XML armazenados localmente.

Em vez disso, o PyLIDC utiliza uma base de dados interna (~/.pylidc/annotations.db), onde se encontram registadas as informações extraídas dos XML originais. Deste modo, um paciente

pode não ter os ficheiros XML visíveis no diretório local, mas continuar a ter as suas anotações acessíveis através da base de dados interna do PyLIDC.

Análise dos Datasets

Todos os datasets analisados neste projeto encontram-se disponíveis no site oficial do **The Cancer Imaging Archive (TCIA)** ou foram **gerados localmente** para efeitos de experimentação e validação.

> **Nota:**
> O dataset **LIDC-IDRI** contém originalmente **1012 identificadores de pacientes**, mas apenas **1010** estão incluídos na versão oficial fornecida pelo TCIA e acessível através da biblioteca **PyLIDC**.
> Os casos **LIDC-IDRI-0238** e **LIDC-IDRI-0585** estão ausentes nesta versão.

Construção do Dataset "anotacoes_pylidc"

O dataset anotacoes_pylidc foi construído a partir das anotações radiológicas disponibilizadas na base LIDC-IDRI, processadas através da biblioteca PyLIDC. Durante o pré-processamento, foram excluídos os pacientes sem nódulos anotados, que foram armazenados separadamente no ficheiro patients_no_nodules.csv.

Para cada anotação realizada pelos radiologistas, foram extraídos diversos atributos radiológicos, organizados em três categorias principais:

- Características visuais (ordinais): subtlety, spiculation, lobulation, margin, texture, sphericity, malignancy
- Características estruturais (categóricas): calcification, internal_structure
- Medidas quantitativas (contínuas): diâmetro médio e volume médio (em milímetros e milímetros cúbicos, respetivamente)

O resultado final é o dataset nodule_per_row_aggregated.csv, em que cada linha representa um nódulo individual, incluindo atributos quantitativos e qualitativos e identificadores do paciente e do estudo.

Este dataset constitui a principal base de dados utilizada na análise, uma vez que permite correspondência direta entre imagem e características radiológicas, contém informação detalhada sobre cada nódulo e consegue preservar a variabilidade interobservador através de um processo de agregação controlada.

Extração com o PyLIDC

```
# Corrige o uso de np.int no pylidc para versões novas do NumPy
if not hasattr(np, "int"):
```

```
    np.int = int

rows = []
rows_no_nodules = []

# Consulta todos os scans do LIDC
scans = pl.query(pl.Scan).all()

for scan in scans:
    # Agrupa as diferentes anotações de cada nódulo (até 4
radiologistas)
    nods = scan.cluster_annotations()

    if len(nods) == 0:
        # Paciente sem nódulos >= 3 mm
        rows_no_nodules.append({
            "patient_id": scan.patient_id,
            "study_uid": scan.study_instance_uid,
            "series_uid": scan.series_instance_uid,
            "num_nodules": 0
        })

    else:
    # Anotações dos pacientes que tem pelo menos 1 nódulo >= 3mm
        for i, nod in enumerate(nods):
            for anot in nod:
                rows.append({
                    "patient_id": scan.patient_id,
                    "study_uid": scan.study_instance_uid,
                    "series_uid": scan.series_instance_uid,
                    "nodule_cluster": i+1, # id que agrupa as
anotações de cada nódulo
                    "subtlety": anot.subtlety,
                    "spiculation": anot.spiculation,
                    "lobulation": anot.lobulation,
                    "margin": anot.margin,
                    "texture": anot.texture,
                    "diameter": anot.diameter,
                    "volume": anot.volume,
                    "malignancy": anot.malignancy,
                    "calcification": anot.calcification,
                    "sphericity": anot.sphericity,
                    "internal_structure": anot.internalStructure,
                })

df = pd.DataFrame(rows)
df_no_nodules = pd.DataFrame(rows_no_nodules)
df.to_csv("lidc_dataset_pylidc.csv", index=False)
df_no_nodules.to_csv("patients_no_nodules.csv", index=False)
print(df_no_nodules.head())
```

```
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
Failed to reduce all groups to <= 4 Annotations.
Some nodules may be close and must be grouped manually.
       patient_id
study_uid  \
0  LIDC-IDRI-0028  1.3.6.1.4.1.14519.5.2.1.6279.6001.313822485916...

1  LIDC-IDRI-0032  1.3.6.1.4.1.14519.5.2.1.6279.6001.188500211166...

2  LIDC-IDRI-0062  1.3.6.1.4.1.14519.5.2.1.6279.6001.121427641977...

3  LIDC-IDRI-0071  1.3.6.1.4.1.14519.5.2.1.6279.6001.194873495479...

4  LIDC-IDRI-0100  1.3.6.1.4.1.14519.5.2.1.6279.6001.318037728418...


                                              series_uid  num_nodules
0  1.3.6.1.4.1.14519.5.2.1.6279.6001.203741923654...             0
1  1.3.6.1.4.1.14519.5.2.1.6279.6001.803808126682...             0
2  1.3.6.1.4.1.14519.5.2.1.6279.6001.210426531621...             0
3  1.3.6.1.4.1.14519.5.2.1.6279.6001.290410217650...             0
4  1.3.6.1.4.1.14519.5.2.1.6279.6001.622243923620...             0
```

As informações das anotações radiológicas foram extraídas utilizando a biblioteca PyLIDC. A partir deste processo, foi obtido um dataset que contém os dados de todas as anotações associadas a nódulos com diâmetro igual ou superior a 3 mm.

Adicionalmente, foi criado um segundo dataset que inclui todos os pacientes cujos nódulos apresentam diâmetro inferior a 3 mm, permitindo assim distinguir entre casos anotados e não anotados segundo este critério de tamanho.

```
df_anno=pd.read_csv('anotacoes_pylidc.csv')

# Número de pacientes únicos
num_patients = df_anno["patient_id"].nunique()
print(f"Número de pacientes : {num_patients}")
```

```
Número de pacientes : 875
```

Após a extração dos dados, obtivemos um total de 875 pacientes a partir dos 1010 disponíveis no dataset LIDC-IDRI. Estes 875 pacientes correspondem aos casos que apresentam pelo menos um nódulo com diâmetro maior ou igual a 3 mm.

```
df_pacientseliminated=pd.read_csv('patients_no_nodules.csv')

# Número de pacientes
num_patients = df_pacientseliminated["patient_id"].nunique()
print(f"Número de pacientes : {num_patients}")
```

```
Número de pacientes : 135
```

Verificou-se que o PyLIDC conseguiu ler corretamente 1010 pacientes do dataset LIDC-IDRI. Desses, 135 pacientes apresentam apenas nódulos com diâmetro inferior a 3 mm, que não serão incluídos na análise.

Agrupamento de Anotações e Limpeza de Clusters

Durante a criação do dataset, ao processar as anotações dos nódulos com a biblioteca PyLIDC, foi gerado o aviso: "Failed to reduce all groups to ≤ 4 Annotations. Some nodules may be close and must be grouped manually."

Esta mensagem indica que o algoritmo de clustering do PyLIDC não conseguiu garantir que cada nódulo tivesse no máximo quatro anotações, correspondentes aos quatro radiologistas que participam na anotação do dataset LIDC-IDRI.

Este problema ocorre quando dois ou mais nódulos estão espacialmente muito próximos, levando o algoritmo a agrupá-los incorretamente como se fossem um único nódulo. Para evitar erros na análise, optou-se por descartar os clusters com mais de quatro anotações, uma vez que estes correspondem a agrupamentos incorretos.

```python
# Conta o número de anotações por cluster (nódulo)
cluster_counts = (
    df_anno.groupby(["patient_id", "nodule_cluster"])
    .size()
    .reset_index(name="num_annotations")
)

# Mantém apenas clusters com <= 4 anotações
valid_clusters = cluster_counts[cluster_counts["num_annotations"] <=
4]

# Faz merge para filtrar apenas os válidos
df_clean = df_anno.merge(valid_clusters[["patient_id",
"nodule_cluster"]], on=["patient_id", "nodule_cluster"], how="inner")

# Salva o dataset limpo
df_clean.to_csv("anotacoes_pylidc_clean.csv", index=False)

print(f"Dataset original: {len(df_anno)} linhas")
print(f"Dataset limpo: {len(df_clean)} linhas")
```

```
Dataset original: 6859 linhas
Dataset limpo: 6692 linhas
```

Agregação das Anotações e Definição da Malignancy

Cada nódulo no dataset LIDC-IDRI pode possuir até quatro anotações distintas, uma por cada radiologista. Para construir um dataset consolidado, foi necessário reduzir estas múltiplas observações a uma única entrada por nódulo.

As anotações foram agrupadas por paciente, estudo, série e identificador de cluster do nódulo, aplicando-se depois funções de agregação específicas a cada tipo de atributo, de forma a preservar a informação clínica e estatística mais representativa.

Atributos ordinais

- Incluem variáveis como subtlety, spiculation, lobulation, margin, texture e sphericity. Nestes casos, foi utilizada a moda (valor mais frequente entre as anotações). Em situações de empate (por exemplo, 3, 3, 4, 4), foi aplicada a média arredondada ao inteiro mais próximo como critério de desempate.

Atributos contínuos

- Para atributos quantitativos como diameter e volume, foi utilizada a média aritmética, garantindo uma representação numérica fiel das medições efetuadas.

Atributos categóricos

- Nos atributos internal_structure e calcification, foi utilizada a moda. Em caso de empate, manteve-se o primeiro valor anotado, dado que estas variáveis são nominais e não possuem relação ordinal que permita o uso da média.

Malignidade (malignancy)

- O atributo malignancy é o mais relevante para a análise e predição do risco de malignidade dos nódulos. A sua agregação seguiu o mesmo princípio dos atributos ordinais, moda e, em caso de empate, média arredondada.

Esta abordagem é consistente com o método descrito por J. L. Causey et al. (2018), no artigo *"Highly accurate model for prediction of lung nodule malignancy with CT scans"* (Scientific Reports), onde os autores utilizaram tanto a média como a moda para obter um valor representativo de malignidade a partir de múltiplas anotações.

Adicionalmente, implementámos um segundo critério baseado na concordância entre radiologistas, designado 3-vote. Neste caso, o valor de malignidade foi escolhido quando pelo menos três dos quatro radiologistas atribuíram a mesma classificação. Este procedimento segue a metodologia proposta em trabalhos baseados no dataset LIDC-IDRI, como o de S. G. Armato et al. (2011), *"The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A completed reference database of lung nodules on CT scans"* (Medical Physics), onde se enfatiza a importância da concordância interobservador na interpretação radiológica.

Esta combinação de critérios (moda, média e 3-vote) permitiu construir um dataset equilibrado, que reflete tanto a tendência central das avaliações como o grau de concordância entre observadores, assegurando maior consistência na análise posterior dos nódulos.

```
# Funções de agregação
def mode_or_first(series):
    # retorna moda; se empate, devolve o valor que aparece primeiro na
lista original
    vals = series.dropna().tolist()
    if not vals:
        return np.nan
    cnt = Counter(vals)
    modes = [v for v,c in cnt.items() if c == max(cnt.values())]
    if len(modes) == 1:
        return modes[0]
    # empate -> devolve o primeiro valor encontrado na ordem original
    for v in vals:
        if v in modes:
            return v
    return modes[0]

def tiebreak_mean(series):
    # tenta moda; em caso de empate usa média inteira arredondada
    vals = series.dropna().astype(float)
    if vals.empty:
        return np.nan
    cnt = Counter(vals)
    modes = [v for v,c in cnt.items() if c == max(cnt.values())]
```

```
    if len(modes) == 1:
        return modes[0]
    return int(round(vals.mean()))

# Agregar por cluster (um nódulo por linha)
aggr_rows = []
group_cols = ["patient_id", "study_uid", "series_uid",
"nodule_cluster"]
for key, group in df_clean.groupby(group_cols):
    patient_id, study_uid, series_uid, nodule_cluster = key
    n_ann = len(group)

    # malignancy: valores
    malign_modes = group["malignancy"].dropna().astype(float).tolist()

# 3-vote: existe uma categoria com >=3 votos?
    malign_vote_counts = Counter(malign_modes)
    malign_3vote = None
    for val, cnt in malign_vote_counts.items():
        if cnt >= 3:
            malign_3vote = val
            break

    # para atributos ordinais: usar mode, desempate com mean
arredondado
    ord_attrs = ["subtlety", "spiculation", "lobulation", "margin",
"texture", "sphericity","malignancy"]
    ord_aggr = {}
    for a in ord_attrs:
        if a in group.columns:
            ord_aggr[a] = tiebreak_mean(group[a])

    # medidas contínuas
    diameter_vals = group["diameter"].dropna().astype(float) if
"diameter" in group.columns else pd.Series(dtype=float)
    volume_vals   = group["volume"].dropna().astype(float)   if
"volume" in group.columns else pd.Series(dtype=float)

    diameter_mean = diameter_vals.mean() if not diameter_vals.empty
else np.nan
    volume_mean = volume_vals.mean() if not volume_vals.empty else
np.nan

    # internal_structure / calcification: mode (categorical)
    internal_struct = mode_or_first(group["internal_structure"]) if
"internal_structure" in group.columns else np.nan
    calcification = mode_or_first(group["calcification"]) if
"calcification" in group.columns else np.nan

    aggr_rows.append({
```

```
        "patient_id": patient_id,
        "study_uid": study_uid,
        "series_uid": series_uid,
        "nodule_cluster": nodule_cluster,
        "num_annotations": n_ann,
        "diameter": diameter_mean,
        "volume": volume_mean,
        "internal_structure": internal_struct,
        "calcification": calcification,
        **ord_aggr,
        "malignancy_3vote": malign_3vote,
    })

df_nodule = pd.DataFrame(aggr_rows)

# Converter colunas de atributos ordinais para Int64(permite) para
aliminar casas decimais artificais de float64
ord_attrs.append("malignancy_3vote")
for col in ord_attrs:
    df_nodule[col] = df_nodule[col].round().astype("Int64")

# Guarda dataset
df_nodule.to_csv("nodule_per_row_aggregated.csv", index=False)
print("Dataset agregado guardado em nodule_per_row_aggregated.csv")
```

```
Dataset agregado guardado em nodule_per_row_aggregated.csv
```

```
print(df_nodule.isna().sum())
```

```
patient_id               0
study_uid                0
series_uid               0
nodule_cluster           0
num_annotations          0
diameter                 0
volume                   0
internal_structure       0
calcification            0
subtlety                 0
spiculation              0
lobulation               0
margin                   0
texture                  0
sphericity               0
malignancy               0
malignancy_3vote      2154
dtype: int64
```

Escolha do Método de Agregação para Malignancy

Optou-se por utilizar o método de moda e média para a agregação da variável malignancy. A alternativa de considerar apenas os nódulos cuja avaliação fosse concordante por pelo menos três radiologistas teria resultado no descarte de uma grande porção dos dados, reduzindo significativamente a amostra disponível para análise.

Este procedimento permite preservar a maior parte da informação, mantendo ao mesmo tempo uma representação confiável da avaliação de malignidade dos nódulos.

```
df_nodule = df_nodule.drop(columns=["malignancy_3vote"],
errors="ignore")

# Salva novamente no mesmo CSV
df_nodule.to_csv("nodule_per_row_aggregated.csv", index=False)
```

Análise de Datasets

Dataset PatientDiagnoses

O dataset PatientDiagnoses contém informações clínicas e diagnósticos a nível de paciente e de nódulo, referentes a 157 pacientes. Para cada paciente (identificado pela coluna TCIA Patient ID), inclui-se o diagnóstico global e, opcionalmente, diagnósticos individuais para até cinco nódulos, com indicação do método de confirmação utilizado (por exemplo, biópsia, ressecção cirúrgica, estabilidade em imagens, entre outros).

Teoricamente, este dataset poderia ser utilizado para validar o diagnóstico clínico dos nódulos identificados via PyLIDC, comparando as anotações radiológicas com os diagnósticos confirmados no PatientDiagnoses.

No entanto, essa correspondência apresenta dificuldades significativas:

- O dataset não fornece identificadores únicos de nódulos (como Nodule_ID ou coordenadas 3D).
- Apenas indica o número do nódulo (Nodule 1, 2, ...) sem referência espacial, tornando impossível associar com segurança cada "Nodule X" do PatientDiagnoses ao respetivo nódulo extraído via PyLIDC.

Além disso, o dataset abrange apenas uma pequena amostra do total de 1010 pacientes do LIDC-IDRI. Por estas razões, o PatientDiagnoses não será integrado nem utilizado diretamente na análise principal nem no treino do modelo de classificação de nódulos.

```
df_patients = pd.read_excel('PatientDiagnoses.xlsx')

print(df_patients.head())
```

```
  TCIA Patient ID  \
0  LIDC-IDRI-0068
1  LIDC-IDRI-0071
2  LIDC-IDRI-0072
3  LIDC-IDRI-0088
4  LIDC-IDRI-0090

   Diagnosis at the Patient Level\n0=Unknown\n1=benign or non-
malignant disease\n2= malignant, primary lung cancer\n3 = malignant
metastatic\n  \
0                                                  3

1                                                  3

2                                                  2

3                                                  3

4                                                  2


   Diagnosis Method\n0 = unknown\n1 = review of radiological images to
show 2 years of stable nodule\n2 = biopsy\n3 = surgical resection\n4 =
progression or response  \
0                                                  4

1                                                  1

2                                                  4

3                                                  0

4                                                  3


  Primary tumor site for metastatic disease  \
0                       Head & Neck Cancer
1                             Head & Neck
2                              Lung Cancer
3                          Uterine Cancer
4                                   NSCLC

   Nodule 1\nDiagnosis at the Nodule Level \n0=Unknown\n1=benign or
non-malignant disease\n2= malignant, primary lung cancer\n3 =
malignant metastatic)\n  \
0                                                3.0

1                                                1.0

2                                                1.0
```

```
3                                                   0.0

4                                                   2.0


   Nodule 1\nDiagnosis Method at the Nodule Level\n0 = unknown\n1 =
review of radiological images to show 2 years of stable nodule\n2 =
biopsy\n3 = surgical resection\n4 = progression or response\n  \
0                                                   4.0

1                                                   1.0

2                                                   4.0

3                                                   0.0

4                                                   3.0


   Nodule 2\nDiagnosis at the Nodule Level \n0=Unknown\n1=benign or
non-malignant disease\n2= malignant, primary lung cancer\n3 =
malignant metastatic)\n  \
0                                                   NaN

1                                                   NaN

2                                                   NaN

3                                                   NaN

4                                                   NaN


   Nodule 2\nDiagnosis Method at the Nodule Level\n0 = unknown\n1 =
review of radiological images to show 2 years of stable nodule\n2 =
biopsy\n3 = surgical resection\n4 = progression or response\n  \
0                                                   NaN

1                                                   NaN

2                                                   NaN

3                                                   NaN

4                                                   NaN


  Nodule 3\nDiagnosis at the Nodule Level \n0=Unknown\n1=benign or
non-malignant disease\n2= malignant, primary lung cancer\n3 =
malignant metastatic)\n  \
0                                                   NaN
```

```
1                                                   NaN

2                                                   NaN

3                                                   NaN

4                                                   NaN


  Nodule 3\nDiagnosis Method at the Nodule Level\n0 = unknown\n1 =
review of radiological images to show 2 years of stable nodule\n2 =
biopsy\n3 = surgical resection\n4 = progression or response\n  \
0                                                   NaN

1                                                   NaN

2                                                   NaN

3                                                   NaN

4                                                   NaN


   Nodule 4\nDiagnosis at the Nodule Level \n0=Unknown\n1=benign or
non-malignant disease\n2= malignant, primary lung cancer\n3 =
malignant metastatic)\n  \
0                                                   NaN

1                                                   NaN

2                                                   NaN

3                                                   NaN

4                                                   NaN


   Nodule 4\nDiagnosis Method at the Nodule Level\n0 = unknown\n1 =
review of radiological images to show 2 years of stable nodule\n2 =
biopsy\n3 = surgical resection\n4 = progression or response\n  \
0                                                   NaN

1                                                   NaN

2                                                   NaN

3                                                   NaN

4                                                   NaN


   Nodule 5\nDiagnosis at the Nodule Level \n0=Unknown\n1=benign or
```

```
non-malignant disease\n2= malignant, primary lung cancer\n3 =
malignant metastatic)\n  \
0                                                    NaN

1                                                    NaN

2                                                    NaN

3                                                    NaN

4                                                    NaN


   Nodule 5\nDiagnosis Method at the Nodule Level\n0 = unknown\n1 =
review of radiological images to show 2 years of stable nodule\n2 =
biopsy\n3 = surgical resection\n4 = progression or response\n
0                                                    NaN

1                                                    NaN

2                                                    NaN

3                                                    NaN

4                                                    NaN
```

Dataset NoduleCounts.xlsx

O ficheiro NoduleCounts.xlsx fornece um resumo do número de nódulos detetados em cada paciente do conjunto de dados LIDC-IDRI.

As colunas principais incluem:

- TCIA Patient ID – identificador único do paciente;
- Total Number of Nodules – número total de nódulos identificados;
- Number of Nodules ≥3mm – número de nódulos considerados clinicamente significativos (≥3 mm);
- Number of Nodules <3mm – número de nódulos menores que 3 mm.

Este dataset serve principalmente como referência de conferência e validação, permitindo verificar se os pacientes descartados por não possuírem nódulos significativos (≥3 mm) coincidem com aqueles identificados via PyLIDC.

Apesar de fornecer informação quantitativa sobre os nódulos, o NoduleCounts.xlsx não inclui características radiológicas detalhadas nem dados de anotação por imagem, pelo que não será usado para extrair mais informações para o dataset principal.

```
df_counts = pd.read_excel('NoduleCounts.xlsx')

print(df_counts.head())
```

```
       Patient_ID  Total_Nodules  Nodules_>=3mm  Nodules_<3mm
0  LIDC-IDRI-0001              4              1             3
1  LIDC-IDRI-0002             12              1            11
2  LIDC-IDRI-0003              4              4             0
3  LIDC-IDRI-0004              4              1             3
4  LIDC-IDRI-0005              9              3             6
```

```
# Simplificamos os nomes das colunas
df_counts.columns = ['Patient_ID', 'Total_Nodules', 'Nodules_>=3mm',
'Nodules_<3mm']
```

Foi realizada uma análise para identificar os pacientes com 0 nódulos ≥ 3 mm. O objetivo é verificar se estes pacientes coincidem com aqueles que foram eliminados pelo PyLIDC por não possuírem nódulos clinicamente significativos.

Esta conferência permite validar a consistência entre os critérios de filtragem aplicados no dataset principal e os dados resumidos disponíveis no NoduleCounts.xlsx.

```
# Filtrar pacientes com 0 nódulos >= 3 mm
df_sem_grandes = df_counts[df_counts['Nodules_>=3mm'] == 0]

# Mostrar o resultado
#print("Número de pacientes sem nódulos >=3mm:", len(df_sem_grandes))
#print(df_sem_grandes[['Patient_ID', 'Total_Nodules',
'Nodules_>=3mm']])


df_no_nodules = pd.read_csv('patients_no_nodules.csv')
# Converter IDs para conjuntos
set_sem_grandes = set(df_sem_grandes['Patient_ID'])
set_no_nodules = set(df_no_nodules['patient_id'])

# Comparações
intersecao = set_sem_grandes.intersection(set_no_nodules)
apenas_pylidc = set_no_nodules - set_sem_grandes
apenas_excel = set_sem_grandes - set_no_nodules


# Resultados
print(f"Pacientes sem nódulos >=3mm segundo NoduleCounts:
{len(set_sem_grandes)}")
print(f"Pacientes sem nódulos >=3mm segundo PyLIDC:
{len(set_no_nodules)}")
print(f"Pacientes coincidentes: {len(intersecao)}")
```

```
Pacientes sem nódulos >=3mm segundo NoduleCounts: 135
Pacientes sem nódulos >=3mm segundo PyLIDC: 135
Pacientes coincidentes: 135
```

Após a conferência, concluímos que os pacientes identificados com 0 nódulos ≥ 3 mm coincidem exatamente com aqueles que foram eliminados pelo PyLIDC pelo mesmo motivo.

Esta confirmação reforça a consistência entre os critérios de filtragem aplicados e os dados resumidos disponíveis no NoduleCounts.xlsx.

```python
# Verificação de pacientes duplicados
duplicates = df_counts[df_counts['Patient_ID'].duplicated(keep=False)]
print(duplicates)
```

	Patient_ID	Total_Nodules	Nodules_>=3mm	Nodules_<3mm
131	LIDC-IDRI-0132	14	6	8
132	LIDC-IDRI-0132	12	8	4
151	LIDC-IDRI-0151	3	1	2
152	LIDC-IDRI-0151	7	1	6
315	LIDC-IDRI-0315	13	7	6
316	LIDC-IDRI-0315	8	5	3
333	LIDC-IDRI-0332	6	5	1
334	LIDC-IDRI-0332	3	2	1
357	LIDC-IDRI-0355	4	1	3
358	LIDC-IDRI-0355	3	2	1
368	LIDC-IDRI-0365	8	1	7
369	LIDC-IDRI-0365	5	1	4
446	LIDC-IDRI-0442	4	3	1
447	LIDC-IDRI-0442	3	3	0
489	LIDC-IDRI-0484	33	2	31
490	LIDC-IDRI-0484	4	2	2

Teoricamente, este dataset poderia ser utilizado para complementar o nosso dataset com a informação do número de nódulos ≥ 3 mm por paciente.

No entanto, verificou-se que alguns pacientes apresentam entradas duplicadas, o que pode gerar conflitos e dificuldades na identificação da entrada correta.

Por esta razão, o NoduleCounts.xlsx não será utilizado para complementar o dataset principal.

Dataset metadata.csv

O ficheiro metadata.csv é gerado automaticamente aquando do download do conjunto de dados LIDC-IDRI a partir do portal TCIA (The Cancer Imaging Archive). Este ficheiro contém informações de natureza administrativa e técnica sobre cada série de imagens DICOM incluída no dataset.

Embora seja útil para controlo e rastreabilidade das imagens, o metadata.csv não contém informação clínica, anatómica ou de anotação dos nódulos. Os seus atributos descrevem apenas

o contexto técnico das séries DICOM (como datas, descrições, tamanho e fabricante do scanner), e não contribuem diretamente para a caracterização ou classificação dos nódulos pulmonares.

Por esta razão, o ficheiro foi analisado, mas não foi integrado no processo de criação do dataset final, que se centra nas características radiológicas e diagnósticas extraídas via PyLIDC e nos ficheiros de anotação disponibilizados pelo LIDC-IDRI.

Dataset Final e Verificação de Duplicados

Por estes motivos, o ficheiro final gerado nesta etapa do pré-processamento é o nodule_per_row_aggregated.csv.

O próximo passo consiste em verificar se existem nódulos duplicados, garantindo a consistência e a integridade do dataset antes de avançar para a análise e modelagem.

```
# Lê o CSV com as features
anot = pd.read_csv("nodule_per_row_aggregated.csv")

# Conta o número de pacientes únicos
num_unique_patients = anot['patient_id'].nunique()

# Conta o número de pares (patient_id, nodule_cluster) únicos
num_unique_nodules = anot[['patient_id',
'nodule_cluster']].drop_duplicates().shape[0]

# Conta o número total de linhas do CSV
num_total_rows = len(anot)

print(f"Número de pacientes únicos: {num_unique_patients}")
print(f"Número total de linhas: {num_total_rows}")
print(f"Número de pares (patient_id, nodule_cluster) únicos:
{num_unique_nodules}")
```

```
Número de pacientes únicos: 867
Número total de linhas: 2612
Número de pares (patient_id, nodule_cluster) únicos: 2604
```

Observou-se que existem pares (patient_id, nodule_cluster) repetidos no dataset, ou seja, o ficheiro nodule_per_row_aggregated.csv contém mais linhas do que pares únicos.

O próximo passo consiste em identificar exatamente quais estes pares duplicados para avaliar a necessidade de limpeza ou correção do dataset.

```
# Encontra os pares duplicados (patient_id, nodule_cluster)
duplicados = (
    anot
    .groupby(['patient_id', 'nodule_cluster'])
    .size()
    .reset_index(name='count')
    .query('count > 1')
```

```
)

print(f"Número de pares duplicados: {len(duplicados)}")
print("\nPares duplicados e quantas vezes aparecem:")
print(duplicados)
```

```
Número de pares duplicados: 8

Pares duplicados e quantas vezes aparecem:
          patient_id  nodule_cluster  count
384   LIDC-IDRI-0132               1      2
385   LIDC-IDRI-0132               2      2
386   LIDC-IDRI-0132               3      2
387   LIDC-IDRI-0132               5      2
388   LIDC-IDRI-0132               6      2
803   LIDC-IDRI-0315               3      2
885   LIDC-IDRI-0355               1      2
1244  LIDC-IDRI-0484               2      2
```

Após identificar os pares (patient_id, nodule_cluster) duplicados, o passo seguinte é verificar se essas duplicações correspondem a linhas idênticas em todos os parâmetros do dataset ou se existem diferenças entre as entradas repetidas.

Esta análise permitirá decidir como tratar os duplicados de forma adequada, preservando a consistência dos dados sem perder informação relevante.

```
# Verifica duplicatas exatas (todas as colunas iguais)
duplicatas_exatas = anot[anot.duplicated(keep=False)]

print(f"Número de linhas totalmente duplicadas:
{duplicatas_exatas.shape[0]}")
print(f"Número de grupos de duplicatas exatas:
{duplicatas_exatas.drop_duplicates().shape[0]}")

# (Opcional) mostra algumas duplicadas
print("\nExemplos de duplicatas exatas:")
print(duplicatas_exatas.head(10))
```

```
Número de linhas totalmente duplicadas: 0
Número de grupos de duplicatas exatas: 0

Exemplos de duplicatas exatas:
Empty DataFrame
Columns: [patient_id, study_uid, series_uid, nodule_cluster,
num_annotations, diameter, volume, internal_structure, calcification,
subtlety, spiculation, lobulation, margin, texture, sphericity,
malignancy]
Index: []
```

Concluímos que, para o mesmo par (patient_id, nodule_cluster), a informação dos casos duplicados difere.

Para compreender a origem destas diferenças, decidimos explorar a hipótese de que alguns pacientes tenham realizado mais de um exame CT. Esta abordagem permitirá verificar se as duplicações estão relacionadas a múltiplas séries de imagens ou a anotações de diferentes exames do mesmo paciente.

```
# Carregar o dataset
df = pd.read_csv("anotacoes_pylidc.csv")

# Verificar quantos estudos (CT) cada paciente tem
study_counts = (
    df.groupby("patient_id")["study_uid"]
    .nunique()  # número de estudos distintos
    .reset_index(name="num_studies")
)

# Filtrar pacientes com mais de 1 TAC
patients_multi_ct = study_counts[study_counts["num_studies"] > 1]

# Mostrar resultados
print(f"Número de pacientes com mais de um TAC:
{len(patients_multi_ct)}")
print(patients_multi_ct.sort_values("num_studies", ascending=False))
```

```
Número de pacientes com mais de um TAC: 8
         patient_id  num_studies
126  LIDC-IDRI-0132            2
144  LIDC-IDRI-0151            2
290  LIDC-IDRI-0315            2
302  LIDC-IDRI-0332            2
321  LIDC-IDRI-0355            2
329  LIDC-IDRI-0365            2
394  LIDC-IDRI-0442            2
431  LIDC-IDRI-0484            2
```

> Nota: Os pacientes que realizaram mais do que um TAC também estão duplicados no ficheiro NoduleCounts.xlsx como é possível confirmar na análise que foi realizada anteriormente ao mesmo.

Os pacientes com pelo menos um nódulo duplicado no ficheiro nodule_per_row_aggregated.csv são:

- LIDC-IDRI-0132
- LIDC-IDRI-0315
- LIDC-IDRI-0355
- LIDC-IDRI-0484

Verificou-se que todos estes pacientes realizaram mais de uma CT, o que explica as duplicações. O PyLIDC não distingue múltiplos exames distintos do mesmo paciente e, por esse motivo, não agrupa corretamente os nódulos quando pertencem a séries de estudos diferentes.

O próximo passo é analisar os restantes pacientes que realizaram mais de uma CT, começando por verificar se estão entre os pacientes eliminados por apresentarem clusters com mais de quatro anotações.

```
# Lê os ficheiros CSV
df1 = pd.read_csv("anotacoes_pylidc.csv")
df2 = pd.read_csv("anotacoes_pylidc_clean.csv")

cols = ["patient_id", "nodule_cluster"]

# Filtra as linhas de df1 que não aparecem em df2
diff = df1.merge(df2[cols], on=cols, how="left", indicator=True)
missing_pairs = diff[diff["_merge"] == "left_only"][cols]

# Mantém apenas pares únicos
missing_pairs_unique = missing_pairs.drop_duplicates()

# Guarda num CSV
missing_pairs_unique.to_csv("missing_pairs.csv", index=False)

print(f"Total de pares únicos exclusivos:
{len(missing_pairs_unique)}")
print("CSV 'missing_pairs.csv' criado com sucesso!")
```

```
Total de pares únicos exclusivos: 26
CSV 'missing_pairs.csv' criado com sucesso!
```

```
# Lê o CSV dos pares únicos
missing_pairs = pd.read_csv("missing_pairs.csv")

# Lista dos pacientes a verificar
patients_to_check = [
    "LIDC-IDRI-0132",
    "LIDC-IDRI-0151",
    "LIDC-IDRI-0315",
    "LIDC-IDRI-0332",
    "LIDC-IDRI-0355",
    "LIDC-IDRI-0365",
    "LIDC-IDRI-0442",
    "LIDC-IDRI-0484"
]

# Filtra os pares que têm esses pacientes
filtered =
missing_pairs[missing_pairs["patient_id"].isin(patients_to_check)]

print(filtered)
```

```
# Mostra quais pacientes estão ausentes no CSV
missing_patients = set(patients_to_check) -
set(filtered["patient_id"])
print("\nPacientes que não aparecem em missing_pairs.csv:",
missing_patients)
```

```
        patient_id  nodule_cluster
0   LIDC-IDRI-0132               4
1   LIDC-IDRI-0151               1
7   LIDC-IDRI-0315               1
8   LIDC-IDRI-0315               2
9   LIDC-IDRI-0315               4
10  LIDC-IDRI-0315               5
11  LIDC-IDRI-0332               1
12  LIDC-IDRI-0332               2
14  LIDC-IDRI-0365               1
17  LIDC-IDRI-0442               1
18  LIDC-IDRI-0442               2
19  LIDC-IDRI-0442               3
20  LIDC-IDRI-0484               1

Pacientes que não aparecem em missing_pairs.csv: {'LIDC-IDRI-0355'}
```

Observamos então que os nódulos dos pacientes que realizaram mais que uma CT que não se encontram duplicados em nodule_per_row_aggregated.csv, já tinham sido previamente eliminados por terem um número de anotações superior a 4.

No entanto, não é possível atribuir um novo valor a nodule_cluster devido à forma como foram extraídas as features 2D e 3D posteriormente neste trabalho. Para manter a consistência do dataset, optou-se por eliminar as 16 linhas duplicadas, correspondentes a 8 nódulos repetidos, garantindo que cada nódulo restante fique representado por uma única linha no ficheiro final.

```
# Lê o CSV
nodule_per_row = pd.read_csv("nodule_per_row_aggregated.csv")

# Lista dos pares a remover
pairs_to_remove = [
    ("LIDC-IDRI-0132", 1),
    ("LIDC-IDRI-0132", 2),
    ("LIDC-IDRI-0132", 3),
    ("LIDC-IDRI-0132", 5),
    ("LIDC-IDRI-0132", 6),
    ("LIDC-IDRI-0315", 3),
    ("LIDC-IDRI-0355", 1),
    ("LIDC-IDRI-0484", 2)
]

# Filtra para manter apenas as linhas que **não estão** em
pairs_to_remove
```

```
nodule_per_row =
nodule_per_row[~nodule_per_row.set_index(["patient_id",
"nodule_cluster"]).index.isin(pairs_to_remove)]


# Sobrescreve o CSV
nodule_per_row.to_csv("nodule_per_row_aggregated.csv", index=False)

num_total_rows = len (nodule_per_row)
num_unique_patients = nodule_per_row['patient_id'].nunique()

print("Linhas removidas e CSV atualizado com sucesso!")
print(f"Número total de linhas: {num_total_rows}")
print(f"Número de pacientes únicos: {num_unique_patients}")
```

```
Linhas removidas e CSV atualizado com sucesso!
Número total de linhas: 2596
Número de pacientes únicos: 866
```

Após a remoção das linhas duplicadas, o processo foi concluído com sucesso. A operação resultou na eliminação de apenas 1 paciente, mas permitiu excluir 16 nódulos duplicados, garantindo a consistência e a integridade do dataset final.

Extração de Features 2D

O script seguinte tem como objetivo realizar a extração de features radiómicas 2D de nódulos pulmonares, utilizando as anotações de nódulos previamente agregadas e o pacote PyRadiomics.

O script começa por ler os ficheiros de anotações e identificar os pacientes disponíveis. Para cada paciente, verifica-se a existência de exames de CT associados e os clusters de nódulos anotados.

Para cada nódulo dentro de um scan, é criada uma máscara de consenso com base nas anotações disponíveis, considerando um limiar de 50%. Clusters com mais de quatro anotações ou cuja área máxima em qualquer fatia seja inferior a três pixels são ignorados, garantindo a qualidade das regiões analisadas.

O volume do scan é normalizado aplicando uma windowing entre -1000 e 400 unidades Hounsfield, correspondendo à janela de visualização pulmonar típica, e convertido para int16 para compatibilidade com o PyRadiomics. (Larue et al., 2017; Aerts et al., 2014).

Para cada nódulo válido, seleciona-se a fatia axial com maior área de lesão, isto é, a secção onde o nódulo apresenta a maior extensão visível. Esta abordagem é apoiada por estudos recentes, como, por exemplo, CZM+18 em *"Highly accurate model for prediction of lung nodule malignancy with CT scans."* e reduz a redundância entre fatias adjacentes e o custo computacional da análise completa em 3D, ao mesmo tempo que mantém a porção mais representativa da morfologia e heterogeneidade do nódulo.

A extração de features é então realizada pelo PyRadiomics, configurado para gerar todas as features disponíveis em modo 2D, com interpolação B-Spline e resampling isotrópico de 1 mm × 1 mm para consistência espacial entre a imagem e a máscara (Zwanenburg et al., 2020). Após a extração, as features são armazenadas num DataFrame e salvas progressivamente num ficheiro CSV, garantindo que o progresso não é perdido em caso de interrupções.

O pipeline inclui ainda tratamento de erros e remoção segura de ficheiros temporários criados para a extração, assegurando a robustez do processo.

```
# Corrige np.int para versões novas do NumPy
if not hasattr(np, "int"):
    np.int = int


# Lê anotações prévias
df_anotacoes = pd.read_csv("nodule_per_row_aggregated.csv")
patients_ids = df_anotacoes["patient_id"].unique()

# Arquivos de saída
output_file = "radiomics_2d.csv"

# Configura PyRadiomics
extractor = featureextractor.RadiomicsFeatureExtractor()
extractor.enableAllFeatures()
extractor.settings['resampledPixelSpacing'] = [1, 1]  # 1mm x 1mm para
2D
extractor.settings['interpolator'] = sitk.sitkBSpline
extractor.settings['force2D'] = True
extractor.settings['force2Ddimension'] = 0

# DataFrame de saída
if os.path.exists(output_file):
    df_features = pd.read_csv(output_file)
else:
    df_features = pd.DataFrame()

# Loop principal
for patient_id in tqdm(patients_ids, desc="Pacientes"):
    try:
        scans = pl.query(pl.Scan).filter(pl.Scan.patient_id ==
patient_id).all()
        if not scans:
            print(f"[WARNING] Nenhum scan encontrado para
{patient_id}")
            continue

        clusters_patient = df_anotacoes.loc[
            df_anotacoes.patient_id == patient_id, "nodule_cluster"
        ].tolist()
```

```
        for scan in scans:
            volume = scan.to_volume()
            volume = np.clip(volume, -1000, 400).astype(np.int16)
            nods = scan.cluster_annotations()

            for i, nod in enumerate(nods):
                nodule_cluster_id = i + 1
                if nodule_cluster_id not in clusters_patient:
                    continue

                # Ignora clusters com mais de 4 anotações, double
check
                if len(nod) > 4:
                    print(f"[IGNORADO] {patient_id} cluster
{nodule_cluster_id} tem {len(nod)} anotações (>4)")
                    continue

                # Cria máscara de consenso 50%
                mask, bbox, _ = consensus(nod, clevel=0.5)
                mask_full = np.zeros_like(volume, dtype=np.uint8)
                z, y, x = bbox
                mask_full[z.start:z.stop, y.start:y.stop,
x.start:x.stop] = mask

                # Calcula área em cada slice
                slice_areas = [np.sum(mask_full[z, :, :]) for z in
range(mask_full.shape[0])]
                if max(slice_areas) < 3:
                    print(f"[IGNORADO] {patient_id} - cluster
{nodule_cluster_id} tem área < 3 pixels")
                    continue

                # Escolhe slice com maior área
                best_slice_idx = int(np.argmax(slice_areas))
                slice_vol = volume[best_slice_idx, :, :]
                slice_mask = mask_full[best_slice_idx, :, :]

                # Salva temporariamente
                vol_fname =
f"temp_vol_{patient_id}_{nodule_cluster_id}.nii.gz"
                mask_fname =
f"temp_mask_{patient_id}_{nodule_cluster_id}.nii.gz"
                sitk.WriteImage(sitk.GetImageFromArray(slice_vol),
vol_fname)
                sitk.WriteImage(sitk.GetImageFromArray(slice_mask),
mask_fname)

                try:
                    # Extrai features 2D
                    features = extractor.execute(vol_fname,
```

```
mask_fname)
                    features = {k: v for k, v in features.items() if
not k.startswith("diagnostics")}
                    features.update({
                        "patient_id": patient_id,
                        "nodule_cluster": nodule_cluster_id,
                    })

                    # Adiciona ao DataFrame
                    df_features = pd.concat([df_features,
pd.DataFrame([features])], ignore_index=True)

                except Exception as e:
                    print(f"[ERRO] Falha ao extrair features de
{patient_id} cluster {nodule_cluster_id}: {e}")

                finally:
                    # Garante remoção dos ficheiros temporários mesmo
em caso de erro
                    for fname in [vol_fname, mask_fname]:
                        if os.path.exists(fname):
                            try:
                                os.remove(fname)
                            except Exception as cleanup_error:
                                print(f"[AVISO] Falha ao apagar
{fname}: {cleanup_error}")


        # Salva progresso após cada paciente
        df_features.to_csv(output_file, index=False)

    except Exception as e:
        print(f"[ERRO] paciente {patient_id}: {e}")

print("Features 2D extraídas com base na fatia de maior área por
nódulo!")
```

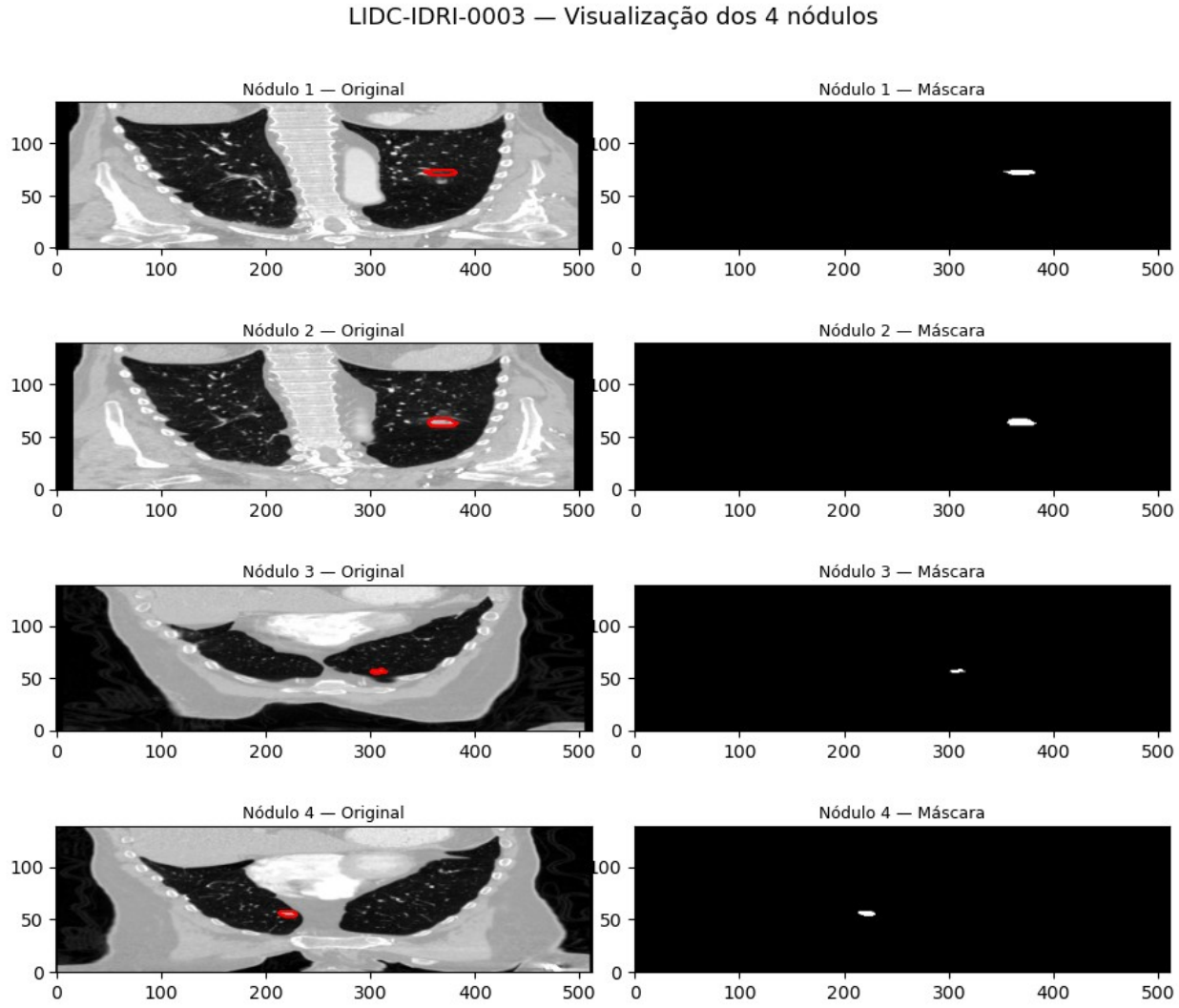
Visualização dos nódulos

Decidimos visualizar os nódulos diretamente nas imagens de tomografia, contornando as suas máscaras, de forma a confirmar que o PyRadiomics estava a identificar corretamente as regiões de interesse (ROIs) e a extrair as features apenas dos nódulos pretendidos. Esta verificação visual permitiu garantir que não havia erros de localização ou segmentação antes do cálculo das features radiómicas.

```
if not hasattr(np, "int"):
    np.int = int
```

```
# Configuração
patient_id = "LIDC-IDRI-0003"
hu_min, hu_max = -1000, 400

# --- Carrega o scan ---
scan = pl.query(pl.Scan).filter(pl.Scan.patient_id ==
patient_id).first()
if scan is None:
    raise ValueError(f"Nenhum scan encontrado para {patient_id}")

vol = np.clip(scan.to_volume(), hu_min, hu_max)
clusters = scan.cluster_annotations()
n_nodulos = len(clusters)
print(f"{patient_id} — {n_nodulos} nódulos encontrados.")

# --- Figura limpa e compacta ---
fig, axes = plt.subplots(
    nrows=n_nodulos, ncols=2,
    figsize=(9, 2 * n_nodulos),

)

if n_nodulos == 1:
    axes = np.array([axes])

for i, nod in enumerate(clusters, start=1):
    # Máscara de consenso
    mask, bbox, _ = consensus(nod, clevel=0.5)
    mask_full = np.zeros_like(vol, dtype=np.uint8)
    z, y, x = bbox
    mask_full[z.start:z.stop, y.start:y.stop, x.start:x.stop] = mask

    # Slice com maior área
    slice_areas = [np.sum(mask_full[z]) for z in
range(mask_full.shape[0])]
    best_slice_idx = int(np.argmax(slice_areas))
    slice_vol = vol[best_slice_idx]
    slice_mask = mask_full[best_slice_idx]

    # Orientação agradável
    if slice_vol.shape[0] > slice_vol.shape[1]:
        slice_vol = np.rot90(slice_vol)
        slice_mask = np.rot90(slice_mask)

    ax_img, ax_mask = axes[i-1]

    # Original + contorno
    print('slice_vol_:', slice_vol.shape)
    ax_img.imshow(slice_vol, cmap="gray", vmin=hu_min, vmax=hu_max,
```

```
origin='lower', aspect='equal')
    ax_img.contour(slice_mask, colors='r', linewidths=0.9)
    ax_img.set_title(f"Nódulo {i} — Original", fontsize=9, pad=4)
    #ax_img.axis('off')

    # Máscara isolada
    ax_mask.imshow(slice_mask, cmap="gray", origin='lower',
aspect='equal')
    ax_mask.set_title(f"Nódulo {i} — Máscara", fontsize=9, pad=4)
    #ax_mask.axis('off')

# --- Layout final ---
fig.suptitle(f"{patient_id} — Visualização dos {n_nodulos} nódulos",
fontsize=13, y=0.995)
plt.subplots_adjust(left=0.02, right=0.98, top=0.93, bottom=0.03,
hspace=0.18, wspace=0.08)
plt.show()


Loading dicom files ... This may take a moment.
LIDC-IDRI-0003 — 4 nódulos encontrados.
slice_vol_: (140, 512)
slice_vol_: (140, 512)
slice_vol_: (140, 512)
slice_vol_: (140, 512)
```




```
rad = pd.read_csv("radiomics_2d.csv")
print(len(rad))

agg = pd.read_csv("nodule_per_row_aggregated.csv")
print(len(agg))
```

```
2515
2596
```

O ficheiro CSV gerado contém menos linhas do que o CSV original das anotações. O próximo passo é identificar quais nódulos estão em falta e analisar também a média dos diâmetros anotados pelos radiologistas, de forma a compreender melhor os critérios que levaram à exclusão desses nódulos durante o pré-processamento.

```
# Seleciona colunas relevantes
agg_cols = ["patient_id", "nodule_cluster", "diameter"]
rad_cols = ["patient_id", "nodule_cluster"]
```

```python
agg_pairs = agg[agg_cols].drop_duplicates()
rad_pairs = rad[rad_cols].drop_duplicates()

# Faz a diferença: pares em agg mas não em rad
diff = pd.merge(agg_pairs, rad_pairs, on=["patient_id",
"nodule_cluster"], how="left", indicator=True)
missing = diff[diff["_merge"] == "left_only"].drop(columns=["_merge"])

# Mostra resultados
print(f"Total de pares em aggregated: {len(agg_pairs)}")
print(f"Total de pares em radiomics_2d: {len(rad_pairs)}")
print(f"Pares em aggregated e não em radiomics_2d: {len(missing)}")

print("\nPares ausentes (com diâmetro):")
pd.set_option("display.max_rows", None)
print(missing)

# Mostra o range de diâmetros
if not missing.empty:
    print(f"\nRange de diâmetros dos nódulos ausentes:
[{missing['diameter'].min()} ; {missing['diameter'].max()}]")
else:
    print("\nNão há nódulos ausentes.")
```

```
Total de pares em aggregated: 2596
Total de pares em radiomics_2d: 2515
Pares em aggregated e não em radiomics_2d: 81

Pares ausentes (com diâmetro):
          patient_id  nodule_cluster   diameter
11    LIDC-IDRI-0006               2   6.628324
87    LIDC-IDRI-0031               2   8.463862
108   LIDC-IDRI-0040               4   5.475996
119   LIDC-IDRI-0042               8   6.385717
155   LIDC-IDRI-0049               4   5.314939
198   LIDC-IDRI-0061               3   8.234042
223   LIDC-IDRI-0069               2   4.686495
227   LIDC-IDRI-0070               3   4.978748
292   LIDC-IDRI-0102               2   6.176320
294   LIDC-IDRI-0103               2   5.101897
322   LIDC-IDRI-0116               1   9.618020
344   LIDC-IDRI-0124               4   7.202769
351   LIDC-IDRI-0124              11   4.752158
397   LIDC-IDRI-0136               1   6.296668
430   LIDC-IDRI-0145               3   4.941059
439   LIDC-IDRI-0148               1   4.752158
442   LIDC-IDRI-0149               1   8.450511
447   LIDC-IDRI-0149               6   4.207160
448   LIDC-IDRI-0149               7   5.524272
```

504	LIDC-IDRI-0171	5	6.197041
505	LIDC-IDRI-0171	6	7.923046
519	LIDC-IDRI-0179	6	6.482492
557	LIDC-IDRI-0188	3	5.354840
638	LIDC-IDRI-0229	1	7.030684
642	LIDC-IDRI-0229	5	7.952580
643	LIDC-IDRI-0229	6	7.519226
646	LIDC-IDRI-0230	2	4.917767
665	LIDC-IDRI-0240	4	6.670429
680	LIDC-IDRI-0247	1	5.038911
689	LIDC-IDRI-0254	2	6.314524
716	LIDC-IDRI-0273	1	4.585097
740	LIDC-IDRI-0289	1	8.202296
742	LIDC-IDRI-0289	3	4.838237
744	LIDC-IDRI-0290	2	6.007503
825	LIDC-IDRI-0334	2	9.604686
829	LIDC-IDRI-0334	6	6.708204
846	LIDC-IDRI-0341	3	5.705707
1002	LIDC-IDRI-0399	2	5.939551
1006	LIDC-IDRI-0400	4	6.842631
1013	LIDC-IDRI-0402	7	7.485674
1014	LIDC-IDRI-0402	8	7.421880
1015	LIDC-IDRI-0402	9	8.027991
1097	LIDC-IDRI-0435	3	7.162672
1101	LIDC-IDRI-0435	7	6.114526
1118	LIDC-IDRI-0445	1	5.709543
1157	LIDC-IDRI-0458	1	4.351001
1228	LIDC-IDRI-0480	2	8.658309
1295	LIDC-IDRI-0499	6	7.026544
1296	LIDC-IDRI-0499	7	4.313621
1314	LIDC-IDRI-0509	3	8.282262
1328	LIDC-IDRI-0516	1	4.872748
1330	LIDC-IDRI-0517	1	7.152167
1357	LIDC-IDRI-0526	3	6.514304
1369	LIDC-IDRI-0530	4	6.680721
1396	LIDC-IDRI-0547	3	9.768125
1496	LIDC-IDRI-0586	3	4.645247
1588	LIDC-IDRI-0626	1	5.580808
1641	LIDC-IDRI-0643	1	6.113280
1686	LIDC-IDRI-0659	5	4.454667
1734	LIDC-IDRI-0675	1	9.059559
1828	LIDC-IDRI-0713	1	6.430490
1831	LIDC-IDRI-0713	4	14.831313
1882	LIDC-IDRI-0740	3	5.373401
1971	LIDC-IDRI-0767	1	4.076374
2038	LIDC-IDRI-0787	2	4.454663
2128	LIDC-IDRI-0824	1	7.516459
2142	LIDC-IDRI-0832	1	4.316380
2179	LIDC-IDRI-0846	2	5.596813

```
2206  LIDC-IDRI-0855               3   6.395777
2214  LIDC-IDRI-0855              11   7.210332
2337  LIDC-IDRI-0899               1   2.878157
2377  LIDC-IDRI-0916               5   5.078921
2412  LIDC-IDRI-0935               3   5.796489
2414  LIDC-IDRI-0935               5   5.197244
2442  LIDC-IDRI-0949               3  11.004303
2456  LIDC-IDRI-0957               2   6.424009
2460  LIDC-IDRI-0961               2   6.356739
2501  LIDC-IDRI-0980               2  10.531589
2537  LIDC-IDRI-0998               3   7.001783
2547  LIDC-IDRI-0999               3   5.628476
2554  LIDC-IDRI-1000               5   7.380000

Range de diâmetros dos nódulos ausentes: [2.878156861505987 ;
14.831313149261105]
```

Para compreender a ausência de alguns nódulos no CSV final, vamos explorar a hipótese de que a exclusão esteja relacionada com o diâmetro dos nódulos. Esta análise permitirá verificar se os nódulos em falta são, por exemplo, menores que 3 mm, ou se existe outro critério que justifique a sua omissão durante o pré-processamento.

```
# Define o intervalo de diametro
min_d, max_d = 2.878156861505987, 14.831313149261105

# Filtra nódulos dentro do range
in_range = agg[(agg["diameter"] >= min_d) & (agg["diameter"] <=
max_d)]

# Conta quantos existem
count = len(in_range)

print(f"Nódulos com diâmetro entre {min_d} e {max_d}: {count}")

Nódulos com diâmetro entre 2.878156861505987 e 14.831313149261105:
2203
```

Verificou-se que o problema da não detecção não está relacionado com o diâmetro, uma vez que existem 2203 nódulos presentes no CSV final cujo diâmetro se encaixa no intervalo esperado.

A questão centra-se nos 81 nódulos restantes, para os quais não é possível extrair features 2D.

O motivo é que a extração 2D utiliza apenas a fatia em que o nódulo apresenta a maior área. Se um nódulo for muito pequeno ou estiver orientado de forma oblíqua, a fatia selecionada pode ter área insuficiente ou até estar praticamente vazia, tornando impossível calcular as features radiómicas.

Por essa razão, esses nódulos são ignorados durante o processamento 2D, garantindo que apenas nódulos com áreas adequadas sejam analisados.

O próximo passo consiste em identificar quantos pacientes ficam sem features 2D, ou seja, aqueles cujos nódulos não foram processados durante a extração 2D. Esta análise permitirá avaliar o impacto da exclusão de nódulos pequenos ou mal orientados no conjunto de dados final e garantir que a amostra de pacientes analisada se mantém representativa.

```
# Lê o CSV com as features
radiomics_3d = pd.read_csv("radiomics_2d.csv")


# Conta o número de pacientes únicos
num_unique_patients = radiomics_3d['patient_id'].nunique()

print(f"Número de pacientes: {num_unique_patients}")
```

```
Número de pacientes: 862
```

A exclusão dos 81 nódulos que não permitem extração de features 2D resulta na perda de dados de 4 pacientes. No total, após essa remoção, permanecem 862 pacientes com dados 2D válidos, garantindo que a maioria do conjunto de dados é preservada para análise.

Extração de Features 3D

Seguindo a mesma lógica do pipeline 2D, este código realiza a extração de features radiómicas 3D diretamente no volume completo do scan, sem reduzir para uma fatia única. O extrator é configurado com resampling isotrópico de 1 mm^3 e interpolação B-Spline, assegurando consistência espacial entre a imagem e a máscara.

O volume é normalizado com windowing entre -1000 e 400 HU, como na abordagem 2D. Para cada nódulo, é criada uma máscara de consenso 3D, e as features de forma, textura e intensidade são extraídas diretamente do volume inteiro.

Os resultados são acumulados num DataFrame e guardados incrementalmente, garantindo a preservação do progresso. Ao final, obtém-se um conjunto completo de features 3D por nódulo, pronto para análises ou modelagem preditiva, mantendo o rigor espacial característico da extração tridimensional.

```
# --- Corrige compatibilidade com NumPy novo ---
if not hasattr(np, "int"):
    np.int = int

# --- CSVs de entrada e saída ---
df_anotacoes = pd.read_csv("nodule_per_row_aggregated.csv")
patients_ids = df_anotacoes["patient_id"].unique()
output_file = "radiomics_3d.csv"

# --- Inicializa extractor PyRadiomics ---
extractor = featureextractor.RadiomicsFeatureExtractor()
extractor.enableAllFeatures()
```

```
extractor.settings['resampledPixelSpacing'] = [1, 1, 1] # 1mm³
isotrópico
extractor.settings['interpolator'] = sitk.sitkBSpline

# --- Reaproveita progresso se já existir ---
if os.path.exists(output_file):
    df_features = pd.read_csv(output_file)
else:
    df_features = pd.DataFrame()

# --- Loop por paciente ---
for patient_id in tqdm(patients_ids, desc="Pacientes"):
    try:
        scans_patient = pl.query(pl.Scan).filter(pl.Scan.patient_id ==
patient_id).all()
        if not scans_patient:
            print(f"[AVISO] Nenhum scan encontrado para {patient_id}")
            continue

        # Clusters válidos deste paciente conforme CSV de anotações
        clusters_patient = df_anotacoes.loc[
            df_anotacoes.patient_id == patient_id, "nodule_cluster"
        ].tolist()

        for scan in scans_patient:
            nods = scan.cluster_annotations()
            volume = scan.to_volume()
            volume = np.clip(volume, -1000, 400).astype(np.int16)

            for i, nod in enumerate(nods):
                nodule_cluster_id = i + 1

                # Ignora nódulos que não estão no CSV de anotações
                if nodule_cluster_id not in clusters_patient:
                    continue

                #Ignora clusters com mais de 4 anotações
                if len(nod) > 4:
                    print(f"[IGNORADO] {patient_id} cluster
{nodule_cluster_id} tem {len(nod)} anotações (>4)")
                    continue

                # Máscara de consenso 50%
                mask, bbox, _ = consensus(nod, clevel=0.5)
                mask_full = np.zeros_like(volume, dtype=np.uint8)
                z, y, x = bbox
                mask_full[z.start:z.stop, y.start:y.stop,
x.start:x.stop] = mask

                # Arquivos temporários para PyRadiomics
```

```
                vol_fname =
f"temp_vol_{patient_id}_{nodule_cluster_id}.nii.gz"
                mask_fname =
f"temp_mask_{patient_id}_{nodule_cluster_id}.nii.gz"
                sitk.WriteImage(sitk.GetImageFromArray(volume),
vol_fname)
                sitk.WriteImage(sitk.GetImageFromArray(mask_full),
mask_fname)

                # Extrai features radiômicas
                features = extractor.execute(vol_fname, mask_fname)
                features = {k: v for k, v in features.items() if not
k.startswith("diagnostics")}

                # Meta-info para identificação posterior
                features["patient_id"] = patient_id
                features["nodule_cluster"] = nodule_cluster_id

                # Adiciona ao DataFrame acumulado
                df_features = pd.concat([df_features,
pd.DataFrame([features])], ignore_index=True)

                # Remove arquivos temporários
                os.remove(vol_fname)
                os.remove(mask_fname)

        # Salva incrementalmente
        df_features.to_csv(output_file, index=False)

    except Exception as e:
        print(f"[ERRO] no paciente {patient_id}: {e}")

print("Features 3D extraídas (clusters com <= 4 anotações) e salvas em
'radiomics_3d.csv'.")

# Lê o CSV com as features
radiomics_3d = pd.read_csv("radiomics_3d.csv")


# Conta o número de pacientes únicos
num_unique_patients = radiomics_3d['patient_id'].nunique()

print(f"Número de pacientes: {num_unique_patients}")
```

```
Número de pacientes: 866
```

Verificamos que o número de pacientes é consistente com o observado em nodule_per_row_aggregated.csv.

O próximo passo consiste em verificar se existem pares (patient_id, nodule_cluster) presentes no nodule_per_row_aggregated.csv que não aparecem no dataset de features 3D

(radiomics_3d). Esta análise permite identificar nódulos que não tiveram extração 3D e avaliar se algum problema de processamento ou filtragem afetou o conjunto de dados tridimensional.

```
# Lê os dois ficheiros
agg = pd.read_csv("nodule_per_row_aggregated.csv")
rad = pd.read_csv("radiomics_3d.csv")

# Seleciona colunas relevantes
agg_cols = ["patient_id", "nodule_cluster", "diameter"]
rad_cols = ["patient_id", "nodule_cluster"]

agg_pairs = agg[agg_cols].drop_duplicates()
rad_pairs = rad[rad_cols].drop_duplicates()

# Faz a diferença: pares em agg mas não em rad
diff = pd.merge(agg_pairs, rad_pairs, on=["patient_id",
"nodule_cluster"], how="left", indicator=True)
missing = diff[diff["_merge"] == "left_only"].drop(columns=["_merge"])

print(f"Pares em aggregated e não em radiomics_3d: {len(missing)}")

print("\nPares ausentes (com diâmetro):")
pd.set_option("display.max_rows", None)
print(missing)

# Mostra o range de diâmetros
if not missing.empty:
    print(f"\nRange de diâmetros dos nódulos ausentes:
[{missing['diameter'].min()} ; {missing['diameter'].max()}]")
else:
    print("\nNão há nódulos ausentes.")
```

```
Pares em aggregated e não em radiomics_3d: 0

Pares ausentes (com diâmetro):
Empty DataFrame
Columns: [patient_id, nodule_cluster, diameter]
Index: []

Não há nódulos ausentes.
```

Contrariamente ao que aconteceu na extração de features 2D, na extração de features 3D não existem nódulos que tenham ficado sem extração.

Isto acontece porque, em 3D, é criada uma máscara volumétrica completa do nódulo, o que garante que mesmo nódulos pequenos ou orientados obliquamente são representados de forma adequada no volume. Desta forma, a máscara 3D é sempre válida, ao contrário do caso 2D, em que a fatia selecionada podia ter área insuficiente para a extração de features.

As features extraídas no ficheiro final resultante da extração 3D estão organizadas em diferentes grupos, consoante o tipo de informação que representam:

- **Forma (Shape Features)** — descrevem a geometria tridimensional do nódulo: Sphericity, Elongation, Flatness, MeshVolume, SurfaceVolumeRatio.
- **Estatísticas de Primeira Ordem** — caracterizam a distribuição das intensidades de voxel dentro do nódulo: Entropy, Skewness, Kurtosis, Mean, Variance, Energy.

As restantes correspondem a features de textura, que capturam padrões de heterogeneidade e estrutura interna do nódulo:

- **GLCM (Gray Level Co-occurrence Matrix):** Contrast, Correlation, Idm, Imc1, Entropy.
- **GLRLM (Gray Level Run Length Matrix):** RunEntropy, ShortRunEmphasis, LongRunEmphasis, GrayLevelVariance.
- **GLSZM (Gray Level Size Zone Matrix):** ZoneEntropy, LargeAreaHighGrayLevelEmphasis, SmallAreaEmphasis.
- **GLDM (Gray Level Dependence Matrix):** DependenceEntropy, LargeDependenceHighGrayLevelEmphasis.
- **NGTDM (Neighborhood Gray Tone Difference Matrix):** Contrast, Coarseness, Busyness.

Nesta fase, optou-se por manter todas as features extraídas pelo PyRadiomics, sem aplicar qualquer forma de redução dimensional ou eliminação de colunas correlacionadas. Embora seja reconhecido que várias destas variáveis apresentam forte correlação entre si, especialmente entre as shape features e as métricas de textura, esta decisão visa garantir que nenhuma informação potencialmente relevante seja perdida numa fase inicial do estudo.

A literatura demonstra que diferentes classes de features radiómicas, nomeadamente as características de forma (shape features) e de textura (texture features), estão fortemente associadas à avaliação da malignidade de nódulos pulmonares. Em particular, Aerts et al. (2014) e Hawkins et al. (2016) demonstram que medidas como Sphericity, SurfaceArea, Elongation, Entropy e Uniformity se relacionam com o grau de irregularidade e heterogeneidade tumoral, fatores amplamente reconhecidos como indicadores de malignidade.

Assim, a eliminação de redundâncias e a seleção das features mais informativas serão realizadas apenas após o merge, recorrendo a técnicas de feature selection e análise de correlação.

Merge de Datasets

Devido à ausência de algumas features 2D (81 nódulos não extraídos), foram consideradas duas abordagens distintas para a integração dos dados.

1. Merge entre nodule_per_row_aggregated.csv e radiomics_3d.csv

Esta abordagem constitui o modelo principal, uma vez que as features 3D captam de forma mais fiel a forma e a textura tridimensional dos nódulos no volume completo da TAC. As anotações

radiológicas provenientes do PyLIDC fornecem o rótulo de malignidade associado a cada nódulo.

Utilidade: Treinar um modelo robusto e realista, baseado em informação volumétrica completa, sem restrições impostas pela seleção de fatias únicas (como ocorre na abordagem 2D).

```
# Lê os ficheiros
df_ann = pd.read_csv("nodule_per_row_aggregated.csv")
df_3d = pd.read_csv("radiomics_3d.csv")

# Faz o merge direto
merged = pd.merge(df_ann, df_3d, on=["patient_id", "nodule_cluster"],
how="inner")

# Guarda o resultado
merged.to_csv("merged_anot_3d.csv", index=False)

print(f"Merge concluído com sucesso! Total de linhas: {len(merged)}")
```

```
Merge concluído com sucesso! Total de linhas: 2596
```

Para garantir que o merge entre os ficheiros nodule_per_row_aggregated.csv e radiomics_3d.csv foi realizado corretamente, foi carregado o ficheiro resultante (merged_anot_3d.csv) e inspecionada a sua estrutura.

```
merged = pd.read_csv("merged_anot_3d.csv")

pd.set_option('display.max_columns', None)

# Mostra todas as linhas (opcional, cuidado se forem muitas)
pd.set_option('display.max_rows', None)

# Impede truncamento do texto nas células
pd.set_option('display.max_colwidth', None)

merged.info()
merged.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1611 entries, 0 to 1610
Columns: 123 entries, patient_id to original_ngtdm_Strength
dtypes: float64(109), int64(11), object(3)
memory usage: 1.5+ MB
```

```
       patient_id  \
0  LIDC-IDRI-0001
1  LIDC-IDRI-0002
2  LIDC-IDRI-0003
3  LIDC-IDRI-0003
4  LIDC-IDRI-0003
```

```
                                                                 study_uid  \
0  1.3.6.1.4.1.14519.5.2.1.6279.6001.298806137288633453246975630178
1  1.3.6.1.4.1.14519.5.2.1.6279.6001.490157381160200744295382098329
2  1.3.6.1.4.1.14519.5.2.1.6279.6001.101370605276577556143013894866
3  1.3.6.1.4.1.14519.5.2.1.6279.6001.101370605276577556143013894866
4  1.3.6.1.4.1.14519.5.2.1.6279.6001.101370605276577556143013894866

                                                                series_uid  \
0  1.3.6.1.4.1.14519.5.2.1.6279.6001.179049373636438705059720603192
1  1.3.6.1.4.1.14519.5.2.1.6279.6001.619372068417051974713149104919
2  1.3.6.1.4.1.14519.5.2.1.6279.6001.170706757615202213033480003264
3  1.3.6.1.4.1.14519.5.2.1.6279.6001.170706757615202213033480003264
4  1.3.6.1.4.1.14519.5.2.1.6279.6001.170706757615202213033480003264

   nodule_cluster  num_annotations   diameter       volume  \
0               1                4  32.755812  6989.673615
1               1                2  30.781671  7244.667508
2               1                1  31.664468  4731.410934
3               2                4  31.001964  6519.463698
4               3                4  13.309155   472.089669

   internal_structure  calcification  subtlety  spiculation
lobulation  \
0                   1              6         5            5
3
1                   1              6         2            1
1
2                   1              6         1            1
1
3                   1              6         5            2
2
4                   1              6         4            2
1

   margin  texture  sphericity  malignancy  original_shape_Elongation
\
0       4        5           3           1                   0.972560

1       2        2           4           1                   0.918731

2       2        1           5           0                   0.778668

3       3        4           4           1                   0.815524

4       4        5           4           1                   0.686744


   original_shape_Flatness  original_shape_LeastAxisLength  \
0                 0.242197                        7.854413
1                 0.643298                       22.362177
```

```
2                  0.187225                            5.949320
3                  0.238987                            6.960832
4                  0.248045                            3.243108

   original_shape_MajorAxisLength
original_shape_Maximum2DDiameterColumn  \
0                       32.429873
44.011362
1                       34.761745
42.190046
2                       31.776383
32.015621
3                       29.126416
30.149627
4                       13.074666
10.049876

   original_shape_Maximum2DDiameterRow
original_shape_Maximum2DDiameterSlice  \
0                            45.617979
35.227830
1                            45.398238
48.166378
2                            38.832976
31.016125
3                            37.696154
29.000000
4                            15.297059
15.000000

   original_shape_Maximum3DDiameter  original_shape_MeshVolume  \
0                          48.456166                5382.208333
1                          51.176166               14160.500000
2                          39.673669                2509.250000
3                          37.696154                3208.666667
4                          15.297059                 251.083333

   original_shape_MinorAxisLength  original_shape_Sphericity  \
0                       31.539994                   0.539498
1                       31.936697                   0.571845
2                       24.743257                   0.486288
3                       23.753283                   0.608560
4                        8.978947                   0.673826

   original_shape_SurfaceArea  original_shape_SurfaceVolumeRatio  \
0                 2752.969818                           0.511494
1                 4949.844790                           0.349553
2                 1836.340800                           0.731829
3                 1728.742762                           0.538773
4                  285.637275                           1.137619
```

```
   original_shape_VoxelVolume  original_firstorder_10Percentile  \
0                       5428.0                            -443.3
1                      14252.0                            -857.0
2                       2542.0                            -798.9
3                       3241.0                            -521.0
4                        261.0                            -644.0

   original_firstorder_90Percentile  original_firstorder_Energy  \
0                             129.0                3.050890e+08
1                            -524.0                7.314935e+09
2                            -465.1                1.095544e+09
3                              38.0                2.596406e+08
4                             -24.0                4.784480e+07

   original_firstorder_Entropy  original_firstorder_InterquartileRange
\
0                     4.657107                                  322.25

1                     4.326124                                  184.00

2                     4.382796                                  165.75

3                     4.717848                                  366.00

4                     4.871635                                  428.00


   original_firstorder_Kurtosis  original_firstorder_Maximum  \
0                      2.860359                        253.0
1                      4.041780                         84.0
2                      5.932879                        176.0
3                      2.162446                        276.0
4                      1.645562                         34.0

   original_firstorder_MeanAbsoluteDeviation  original_firstorder_Mean
\
0                                 188.336555                -77.425571

1                                 106.458877               -703.924993

2                                 106.743973               -640.765932

3                                 189.781453               -179.850663

4                                 207.210258               -359.831418


   original_firstorder_Median  original_firstorder_Minimum  \
0                        24.0                       -808.0
1                      -725.0                       -951.0
```

```
2                        -660.0                          -891.0
3                        -104.0                          -763.0
4                        -405.0                          -775.0

   original_firstorder_Range
original_firstorder_RobustMeanAbsoluteDeviation  \
0                      1061.0
139.954647
1                      1035.0
76.182678
2                      1067.0
70.275407
3                      1039.0
151.450228
4                       809.0
172.721595

   original_firstorder_RootMeanSquared
original_firstorder_Skewness  \
0                           237.079152                       -0.991518

1                           716.419407                        0.903494

2                           656.488463                        1.301749

3                           283.039354                       -0.650805

4                           428.151135                        0.153644


   original_firstorder_TotalEnergy  original_firstorder_Uniformity  \
0                      3.050890e+08                        0.057200
1                      7.314935e+09                        0.057694
2                      1.095544e+09                        0.057768
3                      2.596406e+08                        0.051301
4                      4.784480e+07                        0.037624

   original_firstorder_Variance  original_glcm_Autocorrelation  \
0                   50211.805256                    1075.201487
1                   17746.371374                     148.949831
2                   20396.121845                     141.043637
3                   47765.015033                     715.469574
4                   53834.745526                     385.776901

   original_glcm_ClusterProminence  original_glcm_ClusterShade  \
0                     133332.544098                -3109.453922
1                      26925.040468                  569.477634
2                      34519.090642                  673.075386
3                     110759.361550                -2411.349194
4                      78316.029630                  -66.597193
```

```
   original_glcm_ClusterTendency  original_glcm_Contrast  \
0                      187.720526               62.307974
1                       91.782319                9.978881
2                       85.068236               26.918112
3                      198.261432               53.854624
4                      184.380782              173.242096

   original_glcm_Correlation  original_glcm_DifferenceAverage  \
0                   0.472382                         5.253912
1                   0.802635                         2.247481
2                   0.500230                         3.512968
3                   0.566204                         5.143833
4                   0.052964                        10.438563

   original_glcm_DifferenceEntropy
original_glcm_DifferenceVariance  \
0                         3.748224                         30.643902

1                         2.770701                          4.776610

2                         3.312290                         13.868489

3                         3.811617                         25.529798

4                         4.220778                         46.025299


   original_glcm_Id  original_glcm_Idm  original_glcm_Idmn
original_glcm_Idn  \
0          0.357261           0.288097            0.971680
0.904572
1          0.453400           0.382826            0.994764
0.952437
2          0.376161           0.295529            0.987262
0.931219
3          0.340451           0.267210            0.973792
0.902670
4          0.195261           0.120869            0.881915
0.785138

   original_glcm_Imc1  original_glcm_Imc2
original_glcm_InverseVariance  \
0           -0.139251            0.758946
0.265217
1           -0.206279            0.899166
0.365215
2           -0.138308            0.799431
0.297136
3           -0.133980            0.799749
```

```
0.247107
4          -0.386747            0.987298
0.125380

   original_glcm_JointAverage  original_glcm_JointEnergy  \
0                   32.300579                   0.010244
1                   11.335576                   0.007444
2                   11.247295                   0.005794
3                   26.059871                   0.008435
4                   19.560313                   0.005687

   original_glcm_JointEntropy  original_glcm_MCC  \
0                    8.114369           0.533478
1                    7.662240           0.823689
2                    8.008119           0.627493
3                    8.406129           0.612321
4                    7.708290           0.707442

   original_glcm_MaximumProbability  original_glcm_SumAverage  \
0                          0.039398                 64.601158
1                          0.019715                 22.671152
2                          0.015005                 22.494590
3                          0.038846                 52.119742
4                          0.020410                 39.120626

   original_glcm_SumEntropy  original_glcm_SumSquares  \
0                  5.242091                 62.507125
1                  5.203230                 25.440300
2                  5.134887                 27.996587
3                  5.458571                 63.029014
4                  5.342054                 89.405719

   original_gldm_DependenceEntropy
original_gldm_DependenceNonUniformity  \
0                          7.254482
775.294399
1                          7.606468
1496.455515
2                          7.089424
391.596381
3                          7.093503
547.277075
4                          6.039805
92.984674

   original_gldm_DependenceNonUniformityNormalized  \
0                                         0.142832
1                                         0.105000
2                                         0.154051
3                                         0.168861
```

```
4                                              0.356263

   original_gldm_DependenceVariance
original_gldm_GrayLevelNonUniformity  \
0                           11.968504
310.478998
1                            9.283568
822.258350
2                            4.464253
146.847364
3                           11.417617
166.266276
4                            1.322500
9.819923

   original_gldm_GrayLevelVariance
original_gldm_HighGrayLevelEmphasis  \
0                           80.346951
1005.793294
1                           28.446482
157.609739
2                           32.764765
151.387884
3                           76.390491
668.121259
4                           85.998356
379.444444

   original_gldm_LargeDependenceEmphasis  \
0                                28.704864
1                                35.164047
2                                16.877262
3                                25.645480
4                                 4.747126

   original_gldm_LargeDependenceHighGrayLevelEmphasis  \
0                                          38769.442152
1                                           3632.963374
2                                           1967.623918
3                                          25502.179883
4                                           2903.850575

   original_gldm_LargeDependenceLowGrayLevelEmphasis  \
0                                            0.025580
1                                            0.768121
2                                            0.336243
3                                            0.037212
4                                            0.041258

   original_gldm_LowGrayLevelEmphasis
```

```
original_gldm_SmallDependenceEmphasis  \
0                              0.002347
0.322797
1                              0.016070
0.147483
2                              0.024884
0.255768
3                              0.005159
0.354250
4                              0.024161
0.588421

   original_gldm_SmallDependenceHighGrayLevelEmphasis  \
0                                          220.883256
1                                           37.614915
2                                           59.795997
3                                          156.321515
4                                          164.730716

   original_gldm_SmallDependenceLowGrayLevelEmphasis  \
0                                           0.001341
1                                           0.001921
2                                           0.006047
3                                           0.002929
4                                           0.021511

   original_glrlm_GrayLevelNonUniformity  \
0                              234.003206
1                              676.235096
2                              129.760647
3                              127.807039
4                                9.381428

   original_glrlm_GrayLevelNonUniformityNormalized  \
0                                         0.048749
1                                         0.056283
2                                         0.056521
3                                         0.044037
4                                         0.037141

   original_glrlm_GrayLevelVariance
original_glrlm_HighGrayLevelRunEmphasis  \
0                           83.008492
963.715924
1                           29.456242
165.300296
2                           34.405029
155.219733
3                           77.103928
634.416786
```

```
4                           84.664549
371.579120

   original_glrlm_LongRunEmphasis  \
0                        1.538529
1                        1.676540
2                        1.371470
3                        1.453784
4                        1.114913

   original_glrlm_LongRunHighGrayLevelEmphasis  \
0                                   1672.175853
1                                    241.613341
2                                    198.351179
3                                   1069.266228
4                                    441.422137

   original_glrlm_LongRunLowGrayLevelEmphasis  \
0                                    0.003065
1                                    0.029135
2                                    0.033175
3                                    0.006347
4                                    0.025434

   original_glrlm_LowGrayLevelRunEmphasis
original_glrlm_RunEntropy  \
0                                 0.002532                    5.269948

1                                 0.015398                    5.078861

2                                 0.025200                    4.884973

3                                 0.005560                    5.258696

4                                 0.024814                    5.002294


   original_glrlm_RunLengthNonUniformity  \
0                           3926.870579
1                           8903.806214
2                           1913.821268
3                           2410.740687
4                            238.232254

   original_glrlm_RunLengthNonUniformityNormalized  \
0                                         0.814938
1                                         0.738580
2                                         0.830722
3                                         0.829927
4                                         0.942074
```

```
   original_glrlm_RunPercentage  original_glrlm_RunVariance  \
0                      0.881115                    0.227563
1                      0.842797                    0.256578
2                      0.902953                    0.135549
3                      0.893385                    0.190614
4                      0.967286                    0.042802

   original_glrlm_ShortRunEmphasis  \
0                         0.918743
1                         0.882628
2                         0.927897
3                         0.927213
4                         0.976638

   original_glrlm_ShortRunHighGrayLevelEmphasis  \
0                                    859.017695
1                                    151.011036
2                                    146.771940
3                                    567.041409
4                                    357.572623

   original_glrlm_ShortRunLowGrayLevelEmphasis  \
0                                     0.002444
1                                     0.013177
2                                     0.023558
3                                     0.005409
4                                     0.024685

   original_glszm_GrayLevelNonUniformity  \
0                              64.165314
1                              81.637537
2                              29.566248
3                              45.824335
4                               6.398844

   original_glszm_GrayLevelNonUniformityNormalized  \
0                                          0.032538
1                                          0.040535
2                                          0.041236
3                                          0.034847
4                                          0.036988

   original_glszm_GrayLevelVariance
original_glszm_HighGrayLevelZoneEmphasis  \
0                         74.857104
669.850913
1                         50.281057
278.495035
2                         60.660808
```

```
230.096234
3                          64.103223
433.006084
4                          68.637576
291.734104

   original_glszm_LargeAreaEmphasis  \
0                         488.826572
1                        2223.009930
2                          90.443515
3                         214.930038
4                           4.583815

   original_glszm_LargeAreaHighGrayLevelEmphasis  \
0                                   682173.205882
1                                   179302.644489
2                                     9541.788006
3                                   223006.799240
4                                     2917.878613

   original_glszm_LargeAreaLowGrayLevelEmphasis  \
0                                      0.357907
1                                     45.153453
2                                      1.399587
3                                      0.224404
4                                      0.047061

   original_glszm_LowGrayLevelZoneEmphasis  \
0                                 0.004092
1                                 0.012942
2                                 0.028148
3                                 0.008343
4                                 0.033373

   original_glszm_SizeZoneNonUniformity  \
0                            953.870183
1                            698.389275
2                            265.097629
3                            619.606844
4                            100.768786

   original_glszm_SizeZoneNonUniformityNormalized  \
0                                        0.483707
1                                        0.346767
2                                        0.369732
3                                        0.471184
4                                        0.582479

   original_glszm_SmallAreaEmphasis  \
0                          0.723007
```

```
1                           0.610801
2                           0.632115
3                           0.712886
4                           0.791535

   original_glszm_SmallAreaHighGrayLevelEmphasis  \
0                                      450.048684
1                                      191.254121
2                                      175.560472
3                                      285.645229
4                                      202.401011

   original_glszm_SmallAreaLowGrayLevelEmphasis
original_glszm_ZoneEntropy  \
0                                       0.003344
6.525033
1                                       0.007831
6.967980
2                                       0.013967
6.693464
3                                       0.006369
6.398453
4                                       0.031783
5.650718

   original_glszm_ZonePercentage  original_glszm_ZoneVariance  \
0                       0.363301                   481.250120
1                       0.141313                  2172.933577
2                       0.282061                    77.874166
3                       0.405739                   208.855594
4                       0.662835                     2.307728

   original_ngtdm_Busyness  original_ngtdm_Coarseness  \
0                 0.466158                   0.001206
1                 2.040966                   0.001017
2                 0.540909                   0.003731
3                 0.375968                   0.002146
4                 0.176433                   0.012777

   original_ngtdm_Complexity  original_ngtdm_Contrast
original_ngtdm_Strength
0                3162.569676                 0.328881
1.274217
1                1057.086620                 0.053434
0.926980
2                1918.619966                 0.090857
4.007826
3                2521.046243                 0.342239
1.578325
```

```
4                 3284.429889                    1.238819
5.874138

merged[['patient_id', 'nodule_cluster']].drop_duplicates().shape[0]

2596
```

A inspeção confirmou que o merge foi realizado corretamente. O número de nódulos e colunas é o esperado, e não foram encontradas inconsistências ou valores em falta nos identificadores principais.

```
cols_com_nan = merged.columns[merged.isna().any()].tolist()
print(f"Número de colunas com NaN: {len(cols_com_nan)}")
print(cols_com_nan)

Número de colunas com NaN: 0
[]

merged.dtypes

patient_id                                  object
study_uid                                   object
series_uid                                  object
nodule_cluster                               int64
num_annotations                              int64
diameter                                   float64
volume                                     float64
internal_structure                           int64
calcification                                int64
subtlety                                     int64
spiculation                                  int64
lobulation                                   int64
margin                                       int64
texture                                      int64
sphericity                                   int64
malignancy                                   int64
original_shape_Elongation                  float64
original_shape_Flatness                    float64
original_shape_LeastAxisLength             float64
original_shape_MajorAxisLength             float64
original_shape_Maximum2DDiameterColumn     float64
original_shape_Maximum2DDiameterRow        float64
original_shape_Maximum2DDiameterSlice      float64
original_shape_Maximum3DDiameter           float64
original_shape_MeshVolume                  float64
original_shape_MinorAxisLength             float64
original_shape_Sphericity                  float64
original_shape_SurfaceArea                 float64
original_shape_SurfaceVolumeRatio          float64
original_shape_VoxelVolume                 float64
```

```
original_firstorder_10Percentile                        float64
original_firstorder_90Percentile                        float64
original_firstorder_Energy                              float64
original_firstorder_Entropy                             float64
original_firstorder_InterquartileRange                  float64
original_firstorder_Kurtosis                            float64
original_firstorder_Maximum                             float64
original_firstorder_MeanAbsoluteDeviation               float64
original_firstorder_Mean                                float64
original_firstorder_Median                              float64
original_firstorder_Minimum                             float64
original_firstorder_Range                               float64
original_firstorder_RobustMeanAbsoluteDeviation         float64
original_firstorder_RootMeanSquared                     float64
original_firstorder_Skewness                            float64
original_firstorder_TotalEnergy                         float64
original_firstorder_Uniformity                          float64
original_firstorder_Variance                            float64
original_glcm_Autocorrelation                           float64
original_glcm_ClusterProminence                         float64
original_glcm_ClusterShade                              float64
original_glcm_ClusterTendency                           float64
original_glcm_Contrast                                  float64
original_glcm_Correlation                               float64
original_glcm_DifferenceAverage                         float64
original_glcm_DifferenceEntropy                         float64
original_glcm_DifferenceVariance                        float64
original_glcm_Id                                        float64
original_glcm_Idm                                       float64
original_glcm_Idmn                                      float64
original_glcm_Idn                                       float64
original_glcm_Imc1                                      float64
original_glcm_Imc2                                      float64
original_glcm_InverseVariance                           float64
original_glcm_JointAverage                              float64
original_glcm_JointEnergy                               float64
original_glcm_JointEntropy                              float64
original_glcm_MCC                                       float64
original_glcm_MaximumProbability                        float64
original_glcm_SumAverage                                float64
original_glcm_SumEntropy                                float64
original_glcm_SumSquares                                float64
original_gldm_DependenceEntropy                         float64
original_gldm_DependenceNonUniformity                   float64
original_gldm_DependenceNonUniformityNormalized         float64
original_gldm_DependenceVariance                        float64
original_gldm_GrayLevelNonUniformity                    float64
original_gldm_GrayLevelVariance                         float64
original_gldm_HighGrayLevelEmphasis                     float64
```

```
original_gldm_LargeDependenceEmphasis                   float64
original_gldm_LargeDependenceHighGrayLevelEmphasis      float64
original_gldm_LargeDependenceLowGrayLevelEmphasis       float64
original_gldm_LowGrayLevelEmphasis                      float64
original_gldm_SmallDependenceEmphasis                   float64
original_gldm_SmallDependenceHighGrayLevelEmphasis      float64
original_gldm_SmallDependenceLowGrayLevelEmphasis       float64
original_glrlm_GrayLevelNonUniformity                   float64
original_glrlm_GrayLevelNonUniformityNormalized         float64
original_glrlm_GrayLevelVariance                        float64
original_glrlm_HighGrayLevelRunEmphasis                 float64
original_glrlm_LongRunEmphasis                          float64
original_glrlm_LongRunHighGrayLevelEmphasis             float64
original_glrlm_LongRunLowGrayLevelEmphasis              float64
original_glrlm_LowGrayLevelRunEmphasis                  float64
original_glrlm_RunEntropy                               float64
original_glrlm_RunLengthNonUniformity                   float64
original_glrlm_RunLengthNonUniformityNormalized         float64
original_glrlm_RunPercentage                            float64
original_glrlm_RunVariance                              float64
original_glrlm_ShortRunEmphasis                         float64
original_glrlm_ShortRunHighGrayLevelEmphasis            float64
original_glrlm_ShortRunLowGrayLevelEmphasis             float64
original_glszm_GrayLevelNonUniformity                   float64
original_glszm_GrayLevelNonUniformityNormalized         float64
original_glszm_GrayLevelVariance                        float64
original_glszm_HighGrayLevelZoneEmphasis                float64
original_glszm_LargeAreaEmphasis                        float64
original_glszm_LargeAreaHighGrayLevelEmphasis           float64
original_glszm_LargeAreaLowGrayLevelEmphasis            float64
original_glszm_LowGrayLevelZoneEmphasis                 float64
original_glszm_SizeZoneNonUniformity                    float64
original_glszm_SizeZoneNonUniformityNormalized          float64
original_glszm_SmallAreaEmphasis                        float64
original_glszm_SmallAreaHighGrayLevelEmphasis           float64
original_glszm_SmallAreaLowGrayLevelEmphasis            float64
original_glszm_ZoneEntropy                              float64
original_glszm_ZonePercentage                           float64
original_glszm_ZoneVariance                             float64
original_ngtdm_Busyness                                 float64
original_ngtdm_Coarseness                               float64
original_ngtdm_Complexity                               float64
original_ngtdm_Contrast                                 float64
original_ngtdm_Strength                                 float64
dtype: object
```

De seguida, analisou-se a distribuição da variável target (malignancy), de forma a compreender o equilíbrio entre classes e avaliar a representatividade dos diferentes níveis de malignidade no conjunto de dados.

```
df = pd.read_csv("merged_anot_3d.csv")

# Verificar a distribuição (contagem de cada valor)
print(df['malignancy'].value_counts())
```

```
malignancy
3    985
2    784
1    332
4    320
5    175
Name: count, dtype: int64
```

Com base no artigo de J. L. Causey et al. (2018), *"Highly accurate model for prediction of lung nodule malignancy with CT scans"* (Scientific Reports), optou-se por binarizar a variável malignancy para simplificar a classificação entre nódulos benignos e malignos. Os valores 4 e 5 foram agrupados como malignancy = 1 (maligno) e os valores 1 e 2 como malignancy = 0 (benigno). Os nódulos com valor 3 foram excluídos, por representarem casos de incerteza diagnóstica.

Esta decisão segue a metodologia proposta por Xie et al. (2018) em *"Deep learning in detecting lung nodules and assessing their malignancy on CT scans"*, onde foram comparadas diferentes formas de agrupar as classes (1,2,3 vs 4,5 e 1,2 vs 3,4,5). Os autores concluíram que a divisão 1,2 vs 4,5 proporciona uma melhor separação entre classes, resultando em modelos mais estáveis e com maior capacidade discriminativa, ao reduzir a ambiguidade dos casos intermédios.

```
df = df[df['malignancy'] != 3]

# Guardar de volta no mesmo ficheiro (sobrescreve o anterior)
df.to_csv("merged_anot_3d.csv", index=False)

df = pd.read_csv("merged_anot_3d.csv")

# Substituir valores:
df['malignancy'] = df['malignancy'].replace({1: 0, 2: 0, 4: 1, 5: 1})

# Guardar de volta no mesmo ficheiro
df.to_csv("merged_anot_3d.csv", index=False)

print(df['malignancy'].value_counts())
```

```
malignancy
0    1116
1     495
Name: count, dtype: int64
```

A distribuição da variável malignancy é aproximadamente 70/30, indicando algum desbalanceamento, mas não extremo. Durante o treino do modelo, este fator será considerado, embora se trate de um cenário totalmente tratável para a maioria dos algoritmos de machine learning.

Por fim, foi realizada uma verificação para identificar se existem colunas com apenas um único valor no dataset. A remoção dessas colunas é importante, pois não fornecem informação discriminativa para o treino do modelo.

```
int((merged.nunique() == 1).sum())
```

```
0
```

Não foram encontradas colunas com valores únicos. Como o CSV não apresenta problemas relativos a valores únicos, nulos ou tipos de dados, e a variável malignancy já foi devidamente tratada, o dataset encontra-se pronto para a fase de feature selection.

2. Merge de merge_anot_3d.csv com radiomics_2d.csv (somente nódulos comuns)

Esta abordagem é utilizada para avaliar o contributo das features 2D quando combinadas com as 3D, considerando apenas os nódulos presentes em ambos os datasets.

Utilidade: Permitir testar se a informação da fatia de maior área (2D) complementa as features volumétricas (3D), podendo trazer ganhos adicionais de desempenho para os modelos preditivos

```
# Lê os dois CSVs
merged_anot_3d = pd.read_csv("merged_anot_3d.csv")
radiomics_2d = pd.read_csv("radiomics_2d.csv")

# Faz o merge apenas dos nódulos comuns (inner join)
merged_common = pd.merge(
    merged_anot_3d,
    radiomics_2d,
    on=["patient_id", "nodule_cluster"],
    how="inner",
    suffixes=("_3d", "_2d")
)

# Mostra informações de verificação
print(f"Linhas no merged_anot_3d: {len(merged_anot_3d)}")
print(f"Linhas no radiomics_2d: {len(radiomics_2d)}")
print(f"Nódulos em comum: {len(merged_common)}")

# Guarda o resultado
merged_common.to_csv("merged_anot_3d_2d.csv", index=False)
print("Ficheiro guardado: merged_anot_3d_2d.csv")
```

```
Linhas no merged_anot_3d: 1611
Linhas no radiomics_2d: 2515
Nódulos em comum: 1574
Ficheiro guardado: merged_anot_3d_2d.csv
```

```python
merged_common = pd.read_csv("merged_anot_3d_2d.csv")

pd.set_option('display.max_columns', None)

# Mostra todas as linhas (opcional, cuidado se forem muitas)
pd.set_option('display.max_rows', None)

# Impede truncamento do texto nas células
pd.set_option('display.max_colwidth', None)

merged_common.info()
merged_common.head()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1574 entries, 0 to 1573
Columns: 225 entries, patient_id to original_ngtdm_Strength_2d
dtypes: float64(211), int64(11), object(3)
memory usage: 2.7+ MB
```

```
       patient_id  \
0  LIDC-IDRI-0001
1  LIDC-IDRI-0002
2  LIDC-IDRI-0003
3  LIDC-IDRI-0003
4  LIDC-IDRI-0003

                                                                 study_uid  \
0  1.3.6.1.4.1.14519.5.2.1.6279.6001.298806137288633453246975630178
1  1.3.6.1.4.1.14519.5.2.1.6279.6001.490157381160200744295382098329
2  1.3.6.1.4.1.14519.5.2.1.6279.6001.101370605276577556143013894866
3  1.3.6.1.4.1.14519.5.2.1.6279.6001.101370605276577556143013894866
4  1.3.6.1.4.1.14519.5.2.1.6279.6001.101370605276577556143013894866

                                                                series_uid  \
0  1.3.6.1.4.1.14519.5.2.1.6279.6001.179049373636438705059720603192
1  1.3.6.1.4.1.14519.5.2.1.6279.6001.619372068417051974713149104919
2  1.3.6.1.4.1.14519.5.2.1.6279.6001.170706757615202213033480003264
3  1.3.6.1.4.1.14519.5.2.1.6279.6001.170706757615202213033480003264
4  1.3.6.1.4.1.14519.5.2.1.6279.6001.170706757615202213033480003264

   nodule_cluster  num_annotations   diameter       volume  \
0               1                4  32.755812  6989.673615
1               1                2  30.781671  7244.667508
2               1                1  31.664468  4731.410934
3               2                4  31.001964  6519.463698
4               3                4  13.309155   472.089669

   internal_structure  calcification  subtlety  spiculation
lobulation  \
0                   1              6         5            5
```

```
3
1                   1              6          2              1
1
2                   1              6          1              1
1
3                   1              6          5              2
2
4                   1              6          4              2
1

   margin  texture  sphericity  malignancy  original_shape_Elongation
\
0       4        5           3           1                   0.972560

1       2        2           4           1                   0.918731

2       2        1           5           0                   0.778668

3       3        4           4           1                   0.815524

4       4        5           4           1                   0.686744


   original_shape_Flatness  original_shape_LeastAxisLength  \
0                 0.242197                        7.854413
1                 0.643298                       22.362177
2                 0.187225                        5.949320
3                 0.238987                        6.960832
4                 0.248045                        3.243108

   original_shape_MajorAxisLength
original_shape_Maximum2DDiameterColumn  \
0                       32.429873
44.011362
1                       34.761745
42.190046
2                       31.776383
32.015621
3                       29.126416
30.149627
4                       13.074666
10.049876

   original_shape_Maximum2DDiameterRow
original_shape_Maximum2DDiameterSlice  \
0                           45.617979
35.227830
1                           45.398238
48.166378
2                           38.832976
```

```
31.016125
3                              37.696154
29.000000
4                              15.297059
15.000000

   original_shape_Maximum3DDiameter  original_shape_MeshVolume  \
0                         48.456166                5382.208333
1                         51.176166               14160.500000
2                         39.673669                2509.250000
3                         37.696154                3208.666667
4                         15.297059                 251.083333

   original_shape_MinorAxisLength  original_shape_Sphericity  \
0                       31.539994                   0.539498
1                       31.936697                   0.571845
2                       24.743257                   0.486288
3                       23.753283                   0.608560
4                        8.978947                   0.673826

   original_shape_SurfaceArea  original_shape_SurfaceVolumeRatio  \
0                 2752.969818                           0.511494
1                 4949.844790                           0.349553
2                 1836.340800                           0.731829
3                 1728.742762                           0.538773
4                  285.637275                           1.137619

   original_shape_VoxelVolume  original_firstorder_10Percentile_3d  \
0                      5428.0                                -443.3
1                     14252.0                                -857.0
2                      2542.0                                -798.9
3                      3241.0                                -521.0
4                       261.0                                -644.0

   original_firstorder_90Percentile_3d  original_firstorder_Energy_3d
\
0                                129.0                   3.050890e+08

1                               -524.0                   7.314935e+09

2                               -465.1                   1.095544e+09

3                                 38.0                   2.596406e+08

4                                -24.0                   4.784480e+07


   original_firstorder_Entropy_3d
original_firstorder_InterquartileRange_3d  \
0                        4.657107
```

```
322.25
1                          4.326124
184.00
2                          4.382796
165.75
3                          4.717848
366.00
4                          4.871635
428.00

   original_firstorder_Kurtosis_3d  original_firstorder_Maximum_3d  \
0                         2.860359                           253.0
1                         4.041780                            84.0
2                         5.932879                           176.0
3                         2.162446                           276.0
4                         1.645562                            34.0

   original_firstorder_MeanAbsoluteDeviation_3d
original_firstorder_Mean_3d  \
0                                    188.336555                    -
77.425571
1                                    106.458877                   -
703.924993
2                                    106.743973                   -
640.765932
3                                    189.781453                   -
179.850663
4                                    207.210258                   -
359.831418

   original_firstorder_Median_3d  original_firstorder_Minimum_3d  \
0                           24.0                          -808.0
1                         -725.0                          -951.0
2                         -660.0                          -891.0
3                         -104.0                          -763.0
4                         -405.0                          -775.0

   original_firstorder_Range_3d  \
0                        1061.0
1                        1035.0
2                        1067.0
3                        1039.0
4                         809.0

   original_firstorder_RobustMeanAbsoluteDeviation_3d  \
0                                          139.954647
1                                           76.182678
2                                           70.275407
3                                          151.450228
4                                          172.721595
```

```
   original_firstorder_RootMeanSquared_3d
original_firstorder_Skewness_3d  \
0                               237.079152                           -
0.991518
1                               716.419407
0.903494
2                               656.488463
1.301749
3                               283.039354                           -
0.650805
4                               428.151135
0.153644

   original_firstorder_TotalEnergy_3d
original_firstorder_Uniformity_3d  \
0                         3.050890e+08
0.057200
1                         7.314935e+09
0.057694
2                         1.095544e+09
0.057768
3                         2.596406e+08
0.051301
4                         4.784480e+07
0.037624

   original_firstorder_Variance_3d
original_glcm_Autocorrelation_3d  \
0                      50211.805256                          1075.201487

1                      17746.371374                           148.949831

2                      20396.121845                           141.043637

3                      47765.015033                           715.469574

4                      53834.745526                           385.776901


   original_glcm_ClusterProminence_3d
original_glcm_ClusterShade_3d  \
0                        133332.544098                     -3109.453922

1                         26925.040468                      569.477634

2                         34519.090642                      673.075386

3                        110759.361550                     -2411.349194
```

```
4                         78316.029630                      -66.597193


   original_glcm_ClusterTendency_3d  original_glcm_Contrast_3d  \
0                         187.720526                  62.307974
1                          91.782319                   9.978881
2                          85.068236                  26.918112
3                         198.261432                  53.854624
4                         184.380782                 173.242096

   original_glcm_Correlation_3d  original_glcm_DifferenceAverage_3d  \
0                      0.472382                            5.253912
1                      0.802635                            2.247481
2                      0.500230                            3.512968
3                      0.566204                            5.143833
4                      0.052964                           10.438563

   original_glcm_DifferenceEntropy_3d
original_glcm_DifferenceVariance_3d  \
0                            3.748224
30.643902
1                            2.770701
4.776610
2                            3.312290
13.868489
3                            3.811617
25.529798
4                            4.220778
46.025299

   original_glcm_Id_3d  original_glcm_Idm_3d  original_glcm_Idmn_3d  \
0             0.357261              0.288097               0.971680
1             0.453400              0.382826               0.994764
2             0.376161              0.295529               0.987262
3             0.340451              0.267210               0.973792
4             0.195261              0.120869               0.881915

   original_glcm_Idn_3d  original_glcm_Imc1_3d  original_glcm_Imc2_3d
\
0              0.904572              -0.139251               0.758946

1              0.952437              -0.206279               0.899166

2              0.931219              -0.138308               0.799431

3              0.902670              -0.133980               0.799749

4              0.785138              -0.386747               0.987298
```

```
   original_glcm_InverseVariance_3d  original_glcm_JointAverage_3d  \
0                          0.265217                      32.300579
1                          0.365215                      11.335576
2                          0.297136                      11.247295
3                          0.247107                      26.059871
4                          0.125380                      19.560313

   original_glcm_JointEnergy_3d  original_glcm_JointEntropy_3d  \
0                      0.010244                       8.114369
1                      0.007444                       7.662240
2                      0.005794                       8.008119
3                      0.008435                       8.406129
4                      0.005687                       7.708290

   original_glcm_MCC_3d  original_glcm_MaximumProbability_3d  \
0              0.533478                             0.039398
1              0.823689                             0.019715
2              0.627493                             0.015005
3              0.612321                             0.038846
4              0.707442                             0.020410

   original_glcm_SumAverage_3d  original_glcm_SumEntropy_3d  \
0                    64.601158                     5.242091
1                    22.671152                     5.203230
2                    22.494590                     5.134887
3                    52.119742                     5.458571
4                    39.120626                     5.342054

   original_glcm_SumSquares_3d  original_gldm_DependenceEntropy_3d  \
0                    62.507125                            7.254482
1                    25.440300                            7.606468
2                    27.996587                            7.089424
3                    63.029014                            7.093503
4                    89.405719                            6.039805

   original_gldm_DependenceNonUniformity_3d  \
0                                775.294399
1                               1496.455515
2                                391.596381
3                                547.277075
4                                 92.984674

   original_gldm_DependenceNonUniformityNormalized_3d  \
0                                           0.142832
1                                           0.105000
2                                           0.154051
3                                           0.168861
4                                           0.356263

   original_gldm_DependenceVariance_3d  \
```

```
0                                11.968504
1                                 9.283568
2                                 4.464253
3                                11.417617
4                                 1.322500

   original_gldm_GrayLevelNonUniformity_3d  \
0                                310.478998
1                                822.258350
2                                146.847364
3                                166.266276
4                                  9.819923

   original_gldm_GrayLevelVariance_3d
original_gldm_HighGrayLevelEmphasis_3d  \
0                            80.346951
1005.793294
1                            28.446482
157.609739
2                            32.764765
151.387884
3                            76.390491
668.121259
4                            85.998356
379.444444

   original_gldm_LargeDependenceEmphasis_3d  \
0                                 28.704864
1                                 35.164047
2                                 16.877262
3                                 25.645480
4                                  4.747126

   original_gldm_LargeDependenceHighGrayLevelEmphasis_3d  \
0                                          38769.442152
1                                           3632.963374
2                                           1967.623918
3                                          25502.179883
4                                           2903.850575

   original_gldm_LargeDependenceLowGrayLevelEmphasis_3d  \
0                                             0.025580
1                                             0.768121
2                                             0.336243
3                                             0.037212
4                                             0.041258

   original_gldm_LowGrayLevelEmphasis_3d  \
0                                0.002347
1                                0.016070
```

```
2                                0.024884
3                                0.005159
4                                0.024161

   original_gldm_SmallDependenceEmphasis_3d  \
0                                  0.322797
1                                  0.147483
2                                  0.255768
3                                  0.354250
4                                  0.588421

   original_gldm_SmallDependenceHighGrayLevelEmphasis_3d  \
0                                          220.883256
1                                           37.614915
2                                           59.795997
3                                          156.321515
4                                          164.730716

   original_gldm_SmallDependenceLowGrayLevelEmphasis_3d  \
0                                           0.001341
1                                           0.001921
2                                           0.006047
3                                           0.002929
4                                           0.021511

   original_glrlm_GrayLevelNonUniformity_3d  \
0                                234.003206
1                                676.235096
2                                129.760647
3                                127.807039
4                                  9.381428

   original_glrlm_GrayLevelNonUniformityNormalized_3d  \
0                                           0.048749
1                                           0.056283
2                                           0.056521
3                                           0.044037
4                                           0.037141

   original_glrlm_GrayLevelVariance_3d  \
0                            83.008492
1                            29.456242
2                            34.405029
3                            77.103928
4                            84.664549

   original_glrlm_HighGrayLevelRunEmphasis_3d  \
0                                  963.715924
1                                  165.300296
2                                  155.219733
```

```
3                                    634.416786
4                                    371.579120

   original_glrlm_LongRunEmphasis_3d  \
0                           1.538529
1                           1.676540
2                           1.371470
3                           1.453784
4                           1.114913

   original_glrlm_LongRunHighGrayLevelEmphasis_3d  \
0                                     1672.175853
1                                      241.613341
2                                      198.351179
3                                     1069.266228
4                                      441.422137

   original_glrlm_LongRunLowGrayLevelEmphasis_3d  \
0                                        0.003065
1                                        0.029135
2                                        0.033175
3                                        0.006347
4                                        0.025434

   original_glrlm_LowGrayLevelRunEmphasis_3d
original_glrlm_RunEntropy_3d  \
0                                  0.002532
5.269948
1                                  0.015398
5.078861
2                                  0.025200
4.884973
3                                  0.005560
5.258696
4                                  0.024814
5.002294

   original_glrlm_RunLengthNonUniformity_3d  \
0                              3926.870579
1                              8903.806214
2                              1913.821268
3                              2410.740687
4                               238.232254

   original_glrlm_RunLengthNonUniformityNormalized_3d  \
0                                           0.814938
1                                           0.738580
2                                           0.830722
3                                           0.829927
4                                           0.942074
```

```
   original_glrlm_RunPercentage_3d  original_glrlm_RunVariance_3d  \
0                         0.881115                       0.227563
1                         0.842797                       0.256578
2                         0.902953                       0.135549
3                         0.893385                       0.190614
4                         0.967286                       0.042802

   original_glrlm_ShortRunEmphasis_3d  \
0                            0.918743
1                            0.882628
2                            0.927897
3                            0.927213
4                            0.976638

   original_glrlm_ShortRunHighGrayLevelEmphasis_3d  \
0                                       859.017695
1                                       151.011036
2                                       146.771940
3                                       567.041409
4                                       357.572623

   original_glrlm_ShortRunLowGrayLevelEmphasis_3d  \
0                                        0.002444
1                                        0.013177
2                                        0.023558
3                                        0.005409
4                                        0.024685

   original_glszm_GrayLevelNonUniformity_3d  \
0                                 64.165314
1                                 81.637537
2                                 29.566248
3                                 45.824335
4                                  6.398844

   original_glszm_GrayLevelNonUniformityNormalized_3d  \
0                                            0.032538
1                                            0.040535
2                                            0.041236
3                                            0.034847
4                                            0.036988

   original_glszm_GrayLevelVariance_3d  \
0                            74.857104
1                            50.281057
2                            60.660808
3                            64.103223
4                            68.637576
```

```
   original_glszm_HighGrayLevelZoneEmphasis_3d  \
0                                    669.850913
1                                    278.495035
2                                    230.096234
3                                    433.006084
4                                    291.734104

   original_glszm_LargeAreaEmphasis_3d  \
0                           488.826572
1                          2223.009930
2                            90.443515
3                           214.930038
4                             4.583815

   original_glszm_LargeAreaHighGrayLevelEmphasis_3d  \
0                                     682173.205882
1                                     179302.644489
2                                       9541.788006
3                                     223006.799240
4                                       2917.878613

   original_glszm_LargeAreaLowGrayLevelEmphasis_3d  \
0                                          0.357907
1                                         45.153453
2                                          1.399587
3                                          0.224404
4                                          0.047061

   original_glszm_LowGrayLevelZoneEmphasis_3d  \
0                                    0.004092
1                                    0.012942
2                                    0.028148
3                                    0.008343
4                                    0.033373

   original_glszm_SizeZoneNonUniformity_3d  \
0                                953.870183
1                                698.389275
2                                265.097629
3                                619.606844
4                                100.768786

   original_glszm_SizeZoneNonUniformityNormalized_3d  \
0                                           0.483707
1                                           0.346767
2                                           0.369732
3                                           0.471184
4                                           0.582479

   original_glszm_SmallAreaEmphasis_3d  \
```

```
0                                0.723007
1                                0.610801
2                                0.632115
3                                0.712886
4                                0.791535

   original_glszm_SmallAreaHighGrayLevelEmphasis_3d  \
0                                         450.048684
1                                         191.254121
2                                         175.560472
3                                         285.645229
4                                         202.401011

   original_glszm_SmallAreaLowGrayLevelEmphasis_3d  \
0                                          0.003344
1                                          0.007831
2                                          0.013967
3                                          0.006369
4                                          0.031783

   original_glszm_ZoneEntropy_3d  original_glszm_ZonePercentage_3d  \
0                       6.525033                          0.363301
1                       6.967980                          0.141313
2                       6.693464                          0.282061
3                       6.398453                          0.405739
4                       5.650718                          0.662835

   original_glszm_ZoneVariance_3d  original_ngtdm_Busyness_3d  \
0                      481.250120                    0.466158
1                     2172.933577                    2.040966
2                       77.874166                    0.540909
3                      208.855594                    0.375968
4                        2.307728                    0.176433

   original_ngtdm_Coarseness_3d  original_ngtdm_Complexity_3d  \
0                      0.001206                   3162.569676
1                      0.001017                   1057.086620
2                      0.003731                   1918.619966
3                      0.002146                   2521.046243
4                      0.012777                   3284.429889

   original_ngtdm_Contrast_3d  original_ngtdm_Strength_3d  \
0                    0.328881                    1.274217
1                    0.053434                    0.926980
2                    0.090857                    4.007826
3                    0.342239                    1.578325
4                    1.238819                    5.874138

   original_shape2D_Elongation  original_shape2D_MajorAxisLength  \
0                     0.235850                         35.587289
```

```
1                      0.593511                           40.192150
2                      0.175496                           29.701047
3                      0.273460                           27.371071
4                      0.236064                           14.674310

   original_shape2D_MaximumDiameter  original_shape2D_MeshSurface  \
0                         35.057096                         233.5
1                         48.166378                         709.5
2                         31.016125                         120.5
3                         28.017851                         157.5
4                         15.000000                          37.5

   original_shape2D_MinorAxisLength  original_shape2D_Perimeter  \
0                          8.393263                   77.798990
1                         23.854468                  163.053824
2                          5.212416                   66.142136
3                          7.484894                   63.556349
4                          3.464072                   33.313708

   original_shape2D_PerimeterSurfaceRatio
original_shape2D_PixelSurface  \
0                                0.333186
234.0
1                                0.229815
710.0
2                                0.548897
121.0
3                                0.403532
158.0
4                                0.888366
38.0

   original_shape2D_Sphericity  original_firstorder_10Percentile_2d  \
0                     0.696265                                -388.4
1                     0.579095                                -835.0
2                     0.588329                                -754.0
3                     0.699981                                -485.6
4                     0.651625                                -602.5

   original_firstorder_90Percentile_2d  original_firstorder_Energy_2d
\
0                                123.7                     9965185.0

1                               -493.7                   317943747.0

2                               -496.0                    51182327.0

3                                 42.6                     8966793.0

4                                 -4.8                     5855304.0
```

```
   original_firstorder_Entropy_2d
original_firstorder_InterquartileRange_2d  \
0                          4.038865
198.25
1                          4.358968
190.25
2                          3.862082
133.00
3                          4.252739
242.75
4                          4.396229
480.50

   original_firstorder_Kurtosis_2d  original_firstorder_Maximum_2d  \
0                         4.241415                           167.0
1                         3.991544                           -51.0
2                         7.610824                           -85.0
3                         3.244939                            94.0
4                         1.470253                            33.0

   original_firstorder_MeanAbsoluteDeviation_2d
original_firstorder_Mean_2d  \
0                                    159.708744                   -
30.431624
1                                    109.772997                  -
654.545070
2                                     96.297384                  -
636.520661
3                                    164.215590                  -
128.069620
4                                    223.537396                  -
307.210526

   original_firstorder_Median_2d  original_firstorder_Minimum_2d  \
0                           72.5                          -645.0
1                         -657.0                          -895.0
2                         -679.0                          -830.0
3                          -41.0                          -760.0
4                         -290.0                          -731.0

   original_firstorder_Range_2d  \
0                         812.0
1                         844.0
2                         745.0
3                         854.0
4                         764.0

   original_firstorder_RobustMeanAbsoluteDeviation_2d  \
```

```
0                                            99.311365
1                                            78.512409
2                                            64.910315
3                                           115.651801
4                                           193.013333

   original_firstorder_RootMeanSquared_2d
original_firstorder_Skewness_2d  \
0                              206.364388                          -
1.521721
1                              669.184649
0.688269
2                              650.380226
1.906900
3                              238.226477                          -
1.185180
4                              392.539103                          -
0.074304

   original_firstorder_TotalEnergy_2d
original_firstorder_Uniformity_2d  \
0                            9965185.0
0.099642
1                          317943747.0
0.055596
2                           51182327.0
0.085445
3                            8966793.0
0.076350
4                            5855304.0
0.055402

   original_firstorder_Variance_2d
original_glcm_Autocorrelation_2d  \
0                     41660.176949                          743.638191

1                     19378.845152                          131.606864

2                     17835.885937                           84.655556

3                     40350.026799                          803.323077

4                     59708.639889                          411.304348


   original_glcm_ClusterProminence_2d
original_glcm_ClusterShade_2d  \
0                        67087.766774                    -2085.106780

1                        34742.968319                      501.978865
```

```
2                          24903.382441                        851.282074

3                         112168.749077                      -2956.235776

4                          27350.028681                         81.662037


   original_glcm_ClusterTendency_2d  original_glcm_Contrast_2d  \
0                         114.834120                  67.145729
1                          99.427927                  11.840874
2                          67.912222                  26.277778
3                         156.213018                  44.553846
4                         105.323251                 286.739130

   original_glcm_Correlation_2d  original_glcm_DifferenceAverage_2d  \
0                      0.262053                            5.366834
1                      0.787166                            2.577223
2                      0.442026                            3.744444
3                      0.556163                            4.784615
4                     -0.462722                           14.826087

   original_glcm_DifferenceEntropy_2d
original_glcm_DifferenceVariance_2d  \
0                            3.713958
38.342820
1                            2.897660
5.198795
2                            3.216514
12.256914
3                            3.673332
21.661302
4                            3.567040
66.926276

   original_glcm_Id_2d  original_glcm_Idm_2d  original_glcm_Idmn_2d  \
0             0.349985              0.281548               0.950943
1             0.406512              0.323694               0.990219
2             0.340390              0.260942               0.975579
3             0.360549              0.285583               0.967465
4             0.140699              0.079096               0.802351

   original_glcm_Idn_2d  original_glcm_Imc1_2d  original_glcm_Imc2_2d
\
0              0.879138              -0.245879               0.914357

1              0.932847              -0.231903               0.929911

2              0.900177              -0.317908               0.952255
```

```
3               0.890827               -0.374287                0.974183

4               0.707771               -0.650981                0.997473


   original_glcm_InverseVariance_2d  original_glcm_JointAverage_2d  \
0                          0.294964                     27.050251
1                          0.324094                     10.474259
2                          0.328718                      8.616667
3                          0.212081                     27.846154
4                          0.058234                     21.369565

   original_glcm_JointEnergy_2d  original_glcm_JointEntropy_2d  \
0                      0.021893                      6.449894
1                      0.006439                      7.626272
2                      0.016358                      6.277623
3                      0.019793                      6.464375
4                      0.022684                      5.480084

   original_glcm_MCC_2d  original_glcm_MaximumProbability_2d  \
0              0.554516                             0.065327
1              0.826688                             0.015601
2              0.796088                             0.038889
3              0.834842                             0.092308
4              1.000000                             0.043478

   original_glcm_SumAverage_2d  original_glcm_SumEntropy_2d  \
0                    54.100503                     4.417887
1                    20.948518                     5.233230
2                    17.233333                     4.466753
3                    55.692308                     4.680286
4                    42.739130                     4.229871

   original_glcm_SumSquares_2d  original_gldm_DependenceEntropy_2d  \
0                    45.494962                            4.590247
1                    27.817200                            5.122827
2                    23.547500                            4.216838
3                    50.191716                            4.725287
4                    98.015595                            4.523986

   original_gldm_DependenceNonUniformity_2d  \
0                                164.829060
1                                457.867606
2                                 96.239669
3                                 98.556962
4                                 34.210526

   original_gldm_DependenceNonUniformityNormalized_2d  \
0                                           0.704398
1                                           0.644884
```

```
2                                                 0.795369
3                                                 0.623778
4                                                 0.900277

   original_gldm_DependenceVariance_2d  \
0                             0.200672
1                             0.220186
2                             0.102315
3                             0.245794
4                             0.049861

   original_gldm_GrayLevelNonUniformity_2d  \
0                                23.316239
1                                39.473239
2                                10.338843
3                                12.063291
4                                 2.105263

   original_gldm_GrayLevelVariance_2d
original_gldm_HighGrayLevelEmphasis_2d  \
0                            66.617941
707.529915
1                            31.219091
137.977465
2                            28.966327
110.413223
3                            64.472841
759.696203
4                            96.299169
429.842105

   original_gldm_LargeDependenceEmphasis_2d  \
0                                  1.632479
1                                  1.763380
2                                  1.347107
3                                  1.848101
4                                  1.157895

   original_gldm_LargeDependenceHighGrayLevelEmphasis_2d  \
0                                          1291.145299
1                                           216.214085
2                                           126.429752
3                                          1654.113924
4                                           571.947368

   original_gldm_LargeDependenceLowGrayLevelEmphasis_2d  \
0                                           0.027115
1                                           0.087141
2                                           0.047011
3                                           0.009693
```

```
4                                                 0.035306

   original_gldm_LowGrayLevelEmphasis_2d  \
0                                0.026428
1                                0.046871
2                                0.033797
3                                0.008886
4                                0.035131

   original_gldm_SmallDependenceEmphasis_2d  \
0                                  0.865622
1                                  0.829499
2                                  0.913223
3                                  0.816104
4                                  0.960526

   original_gldm_SmallDependenceHighGrayLevelEmphasis_2d  \
0                                          582.993590
1                                          119.958216
2                                          106.409091
3                                          566.724684
4                                          394.315789

   original_gldm_SmallDependenceLowGrayLevelEmphasis_2d  \
0                                           0.026282
1                                           0.038465
2                                           0.030494
3                                           0.008710
4                                           0.035087

   original_glrlm_GrayLevelNonUniformity_2d  \
0                                 17.758294
1                                 34.278846
2                                  9.403509
3                                  8.328467
4                                  1.918919

   original_glrlm_GrayLevelNonUniformityNormalized_2d  \
0                                           0.084163
1                                           0.054934
2                                           0.082487
3                                           0.060792
4                                           0.051863

   original_glrlm_GrayLevelVariance_2d  \
0                            70.683183
1                            31.691065
2                            29.687365
3                            67.735202
4                            95.078159
```

```
   original_glrlm_HighGrayLevelRunEmphasis_2d  \
0                                   683.890995
1                                   142.682692
2                                   114.359649
3                                   714.824818
4                                   417.135135

   original_glrlm_LongRunEmphasis_2d  \
0                           1.402844
1                           1.461538
2                           1.184211
3                           1.547445
4                           1.081081

   original_glrlm_LongRunHighGrayLevelEmphasis_2d  \
0                                     1054.426540
1                                      188.810897
2                                      122.859649
3                                     1294.175182
4                                      490.108108

   original_glrlm_LongRunLowGrayLevelEmphasis_2d  \
0                                       0.029629
1                                       0.072031
2                                       0.040547
3                                       0.010621
4                                       0.036140

   original_glrlm_LowGrayLevelRunEmphasis_2d
original_glrlm_RunEntropy_2d  \
0                                   0.029190
4.555044
1                                   0.046261
4.915693
2                                   0.033535
4.140390
3                                   0.010102
4.772870
4                                   0.036050
4.520000

   original_glrlm_RunLengthNonUniformity_2d  \
0                                 177.625592
1                                 492.836538
2                                 100.859649
3                                 106.781022
4                                  35.054054

   original_glrlm_RunLengthNonUniformityNormalized_2d  \
```

```
0                                                0.841827
1                                                0.789802
2                                                0.884734
3                                                0.779424
4                                                0.947407

   original_glrlm_RunPercentage_2d  original_glrlm_RunVariance_2d  \
0                         0.901709                       0.172952
1                         0.878873                       0.166903
2                         0.942149                       0.057633
3                         0.867089                       0.217380
4                         0.973684                       0.026297

   original_glrlm_ShortRunEmphasis_2d  \
0                            0.933707
1                            0.909655
2                            0.953947
3                            0.904197
4                            0.979730

   original_glrlm_ShortRunHighGrayLevelEmphasis_2d  \
0                                       622.233412
1                                       132.915075
2                                       112.234649
3                                       614.699818
4                                       398.891892

   original_glrlm_ShortRunLowGrayLevelEmphasis_2d  \
0                                        0.029119
1                                        0.042052
2                                        0.031782
3                                        0.010010
4                                        0.036028

   original_glszm_GrayLevelNonUniformity_2d  \
0                                 17.758294
1                                 34.278846
2                                  9.403509
3                                  8.328467
4                                  1.918919

   original_glszm_GrayLevelNonUniformityNormalized_2d  \
0                                            0.084163
1                                            0.054934
2                                            0.082487
3                                            0.060792
4                                            0.051863

   original_glszm_GrayLevelVariance_2d  \
0                            70.683183
```

```
1                              31.691065
2                              29.687365
3                              67.735202
4                              95.078159

   original_glszm_HighGrayLevelZoneEmphasis_2d  \
0                                   683.890995
1                                   142.682692
2                                   114.359649
3                                   714.824818
4                                   417.135135

   original_glszm_LargeAreaEmphasis_2d  \
0                             1.402844
1                             1.461538
2                             1.184211
3                             1.547445
4                             1.081081

   original_glszm_LargeAreaHighGrayLevelEmphasis_2d  \
0                                       1054.426540
1                                        188.810897
2                                        122.859649
3                                       1294.175182
4                                        490.108108

   original_glszm_LargeAreaLowGrayLevelEmphasis_2d  \
0                                          0.029629
1                                          0.072031
2                                          0.040547
3                                          0.010621
4                                          0.036140

   original_glszm_LowGrayLevelZoneEmphasis_2d  \
0                                    0.029190
1                                    0.046261
2                                    0.033535
3                                    0.010102
4                                    0.036050

   original_glszm_SizeZoneNonUniformity_2d  \
0                              177.625592
1                              492.836538
2                              100.859649
3                              106.781022
4                               35.054054

   original_glszm_SizeZoneNonUniformityNormalized_2d  \
0                                           0.841827
1                                           0.789802
```

```
2                                                0.884734
3                                                0.779424
4                                                0.947407

   original_glszm_SmallAreaEmphasis_2d  \
0                             0.933707
1                             0.909655
2                             0.953947
3                             0.904197
4                             0.979730

   original_glszm_SmallAreaHighGrayLevelEmphasis_2d  \
0                                         622.233412
1                                         132.915075
2                                         112.234649
3                                         614.699818
4                                         398.891892

   original_glszm_SmallAreaLowGrayLevelEmphasis_2d  \
0                                          0.029119
1                                          0.042052
2                                          0.031782
3                                          0.010010
4                                          0.036028

   original_glszm_ZoneEntropy_2d  original_glszm_ZonePercentage_2d  \
0                       4.555044                          0.901709
1                       4.915693                          0.878873
2                       4.140390                          0.942149
3                       4.772870                          0.867089
4                       4.520000                          0.973684

   original_glszm_ZoneVariance_2d  original_ngtdm_Busyness_2d  \
0                        0.172952                    0.062753
1                        0.166903                    0.232359
2                        0.057633                    0.154527
3                        0.217380                    0.035224
4                        0.026297                    0.071751

   original_ngtdm_Coarseness_2d  original_ngtdm_Complexity_2d  \
0                      0.015036                   2140.930148
1                      0.013947                    874.197164
2                      0.043706                    764.793900
3                      0.028410                   1647.541694
4                      0.034142                   4188.531046

   original_ngtdm_Contrast_2d  original_ngtdm_Strength_2d
0                    0.847859                    9.657467
1                    0.132337                    6.770865
2                    0.382922                   15.715631
```

```
3                    0.671182                    15.214641
4                    4.797513                    17.234550
```

```
merged_common[['patient_id',
'nodule_cluster']].drop_duplicates().shape[0]
```

```
1574
```

```
cols_com_nan =
merged_common.columns[merged_common.isna().any()].tolist()
print(f"Número de colunas com NaN: {len(cols_com_nan)}")
print(cols_com_nan)
```

```
Número de colunas com NaN: 0
[]
```

```
merged_common.dtypes
```

```
patient_id                                    object
study_uid                                     object
series_uid                                    object
nodule_cluster                                 int64
num_annotations                                int64
diameter                                     float64
volume                                       float64
internal_structure                             int64
calcification                                  int64
subtlety                                       int64
spiculation                                    int64
lobulation                                     int64
margin                                         int64
texture                                        int64
sphericity                                     int64
malignancy                                     int64
original_shape_Elongation                    float64
original_shape_Flatness                      float64
original_shape_LeastAxisLength               float64
original_shape_MajorAxisLength               float64
original_shape_Maximum2DDiameterColumn       float64
original_shape_Maximum2DDiameterRow          float64
original_shape_Maximum2DDiameterSlice        float64
original_shape_Maximum3DDiameter             float64
original_shape_MeshVolume                    float64
original_shape_MinorAxisLength               float64
original_shape_Sphericity                    float64
original_shape_SurfaceArea                   float64
original_shape_SurfaceVolumeRatio            float64
original_shape_VoxelVolume                   float64
original_firstorder_10Percentile_3d          float64
original_firstorder_90Percentile_3d          float64
original_firstorder_Energy_3d                float64
```

original_firstorder_Entropy_3d	float64
original_firstorder_InterquartileRange_3d	float64
original_firstorder_Kurtosis_3d	float64
original_firstorder_Maximum_3d	float64
original_firstorder_MeanAbsoluteDeviation_3d	float64
original_firstorder_Mean_3d	float64
original_firstorder_Median_3d	float64
original_firstorder_Minimum_3d	float64
original_firstorder_Range_3d	float64
original_firstorder_RobustMeanAbsoluteDeviation_3d	float64
original_firstorder_RootMeanSquared_3d	float64
original_firstorder_Skewness_3d	float64
original_firstorder_TotalEnergy_3d	float64
original_firstorder_Uniformity_3d	float64
original_firstorder_Variance_3d	float64
original_glcm_Autocorrelation_3d	float64
original_glcm_ClusterProminence_3d	float64
original_glcm_ClusterShade_3d	float64
original_glcm_ClusterTendency_3d	float64
original_glcm_Contrast_3d	float64
original_glcm_Correlation_3d	float64
original_glcm_DifferenceAverage_3d	float64
original_glcm_DifferenceEntropy_3d	float64
original_glcm_DifferenceVariance_3d	float64
original_glcm_Id_3d	float64
original_glcm_Idm_3d	float64
original_glcm_Idmn_3d	float64
original_glcm_Idn_3d	float64
original_glcm_Imc1_3d	float64
original_glcm_Imc2_3d	float64
original_glcm_InverseVariance_3d	float64
original_glcm_JointAverage_3d	float64
original_glcm_JointEnergy_3d	float64
original_glcm_JointEntropy_3d	float64
original_glcm_MCC_3d	float64
original_glcm_MaximumProbability_3d	float64
original_glcm_SumAverage_3d	float64
original_glcm_SumEntropy_3d	float64
original_glcm_SumSquares_3d	float64
original_gldm_DependenceEntropy_3d	float64
original_gldm_DependenceNonUniformity_3d	float64
original_gldm_DependenceNonUniformityNormalized_3d	float64
original_gldm_DependenceVariance_3d	float64
original_gldm_GrayLevelNonUniformity_3d	float64
original_gldm_GrayLevelVariance_3d	float64
original_gldm_HighGrayLevelEmphasis_3d	float64
original_gldm_LargeDependenceEmphasis_3d	float64
original_gldm_LargeDependenceHighGrayLevelEmphasis_3d	float64
original_gldm_LargeDependenceLowGrayLevelEmphasis_3d	float64

```
original_gldm_LowGrayLevelEmphasis_3d                    float64
original_gldm_SmallDependenceEmphasis_3d                 float64
original_gldm_SmallDependenceHighGrayLevelEmphasis_3d    float64
original_gldm_SmallDependenceLowGrayLevelEmphasis_3d     float64
original_glrlm_GrayLevelNonUniformity_3d                 float64
original_glrlm_GrayLevelNonUniformityNormalized_3d       float64
original_glrlm_GrayLevelVariance_3d                      float64
original_glrlm_HighGrayLevelRunEmphasis_3d               float64
original_glrlm_LongRunEmphasis_3d                        float64
original_glrlm_LongRunHighGrayLevelEmphasis_3d           float64
original_glrlm_LongRunLowGrayLevelEmphasis_3d            float64
original_glrlm_LowGrayLevelRunEmphasis_3d                float64
original_glrlm_RunEntropy_3d                             float64
original_glrlm_RunLengthNonUniformity_3d                 float64
original_glrlm_RunLengthNonUniformityNormalized_3d       float64
original_glrlm_RunPercentage_3d                          float64
original_glrlm_RunVariance_3d                            float64
original_glrlm_ShortRunEmphasis_3d                       float64
original_glrlm_ShortRunHighGrayLevelEmphasis_3d          float64
original_glrlm_ShortRunLowGrayLevelEmphasis_3d           float64
original_glszm_GrayLevelNonUniformity_3d                 float64
original_glszm_GrayLevelNonUniformityNormalized_3d       float64
original_glszm_GrayLevelVariance_3d                      float64
original_glszm_HighGrayLevelZoneEmphasis_3d              float64
original_glszm_LargeAreaEmphasis_3d                      float64
original_glszm_LargeAreaHighGrayLevelEmphasis_3d         float64
original_glszm_LargeAreaLowGrayLevelEmphasis_3d          float64
original_glszm_LowGrayLevelZoneEmphasis_3d               float64
original_glszm_SizeZoneNonUniformity_3d                  float64
original_glszm_SizeZoneNonUniformityNormalized_3d        float64
original_glszm_SmallAreaEmphasis_3d                      float64
original_glszm_SmallAreaHighGrayLevelEmphasis_3d         float64
original_glszm_SmallAreaLowGrayLevelEmphasis_3d          float64
original_glszm_ZoneEntropy_3d                            float64
original_glszm_ZonePercentage_3d                         float64
original_glszm_ZoneVariance_3d                           float64
original_ngtdm_Busyness_3d                               float64
original_ngtdm_Coarseness_3d                             float64
original_ngtdm_Complexity_3d                             float64
original_ngtdm_Contrast_3d                               float64
original_ngtdm_Strength_3d                               float64
original_shape2D_Elongation                              float64
original_shape2D_MajorAxisLength                         float64
original_shape2D_MaximumDiameter                         float64
original_shape2D_MeshSurface                             float64
original_shape2D_MinorAxisLength                         float64
original_shape2D_Perimeter                               float64
original_shape2D_PerimeterSurfaceRatio                   float64
original_shape2D_PixelSurface                            float64
```

original_shape2D_Sphericity	float64
original_firstorder_10Percentile_2d	float64
original_firstorder_90Percentile_2d	float64
original_firstorder_Energy_2d	float64
original_firstorder_Entropy_2d	float64
original_firstorder_InterquartileRange_2d	float64
original_firstorder_Kurtosis_2d	float64
original_firstorder_Maximum_2d	float64
original_firstorder_MeanAbsoluteDeviation_2d	float64
original_firstorder_Mean_2d	float64
original_firstorder_Median_2d	float64
original_firstorder_Minimum_2d	float64
original_firstorder_Range_2d	float64
original_firstorder_RobustMeanAbsoluteDeviation_2d	float64
original_firstorder_RootMeanSquared_2d	float64
original_firstorder_Skewness_2d	float64
original_firstorder_TotalEnergy_2d	float64
original_firstorder_Uniformity_2d	float64
original_firstorder_Variance_2d	float64
original_glcm_Autocorrelation_2d	float64
original_glcm_ClusterProminence_2d	float64
original_glcm_ClusterShade_2d	float64
original_glcm_ClusterTendency_2d	float64
original_glcm_Contrast_2d	float64
original_glcm_Correlation_2d	float64
original_glcm_DifferenceAverage_2d	float64
original_glcm_DifferenceEntropy_2d	float64
original_glcm_DifferenceVariance_2d	float64
original_glcm_Id_2d	float64
original_glcm_Idm_2d	float64
original_glcm_Idmn_2d	float64
original_glcm_Idn_2d	float64
original_glcm_Imc1_2d	float64
original_glcm_Imc2_2d	float64
original_glcm_InverseVariance_2d	float64
original_glcm_JointAverage_2d	float64
original_glcm_JointEnergy_2d	float64
original_glcm_JointEntropy_2d	float64
original_glcm_MCC_2d	float64
original_glcm_MaximumProbability_2d	float64
original_glcm_SumAverage_2d	float64
original_glcm_SumEntropy_2d	float64
original_glcm_SumSquares_2d	float64
original_gldm_DependenceEntropy_2d	float64
original_gldm_DependenceNonUniformity_2d	float64
original_gldm_DependenceNonUniformityNormalized_2d	float64
original_gldm_DependenceVariance_2d	float64
original_gldm_GrayLevelNonUniformity_2d	float64
original_gldm_GrayLevelVariance_2d	float64

```
original_gldm_HighGrayLevelEmphasis_2d                    float64
original_gldm_LargeDependenceEmphasis_2d                  float64
original_gldm_LargeDependenceHighGrayLevelEmphasis_2d     float64
original_gldm_LargeDependenceLowGrayLevelEmphasis_2d      float64
original_gldm_LowGrayLevelEmphasis_2d                     float64
original_gldm_SmallDependenceEmphasis_2d                  float64
original_gldm_SmallDependenceHighGrayLevelEmphasis_2d     float64
original_gldm_SmallDependenceLowGrayLevelEmphasis_2d      float64
original_glrlm_GrayLevelNonUniformity_2d                  float64
original_glrlm_GrayLevelNonUniformityNormalized_2d        float64
original_glrlm_GrayLevelVariance_2d                       float64
original_glrlm_HighGrayLevelRunEmphasis_2d                float64
original_glrlm_LongRunEmphasis_2d                         float64
original_glrlm_LongRunHighGrayLevelEmphasis_2d            float64
original_glrlm_LongRunLowGrayLevelEmphasis_2d             float64
original_glrlm_LowGrayLevelRunEmphasis_2d                 float64
original_glrlm_RunEntropy_2d                              float64
original_glrlm_RunLengthNonUniformity_2d                  float64
original_glrlm_RunLengthNonUniformityNormalized_2d        float64
original_glrlm_RunPercentage_2d                           float64
original_glrlm_RunVariance_2d                             float64
original_glrlm_ShortRunEmphasis_2d                        float64
original_glrlm_ShortRunHighGrayLevelEmphasis_2d           float64
original_glrlm_ShortRunLowGrayLevelEmphasis_2d            float64
original_glszm_GrayLevelNonUniformity_2d                  float64
original_glszm_GrayLevelNonUniformityNormalized_2d        float64
original_glszm_GrayLevelVariance_2d                       float64
original_glszm_HighGrayLevelZoneEmphasis_2d               float64
original_glszm_LargeAreaEmphasis_2d                       float64
original_glszm_LargeAreaHighGrayLevelEmphasis_2d          float64
original_glszm_LargeAreaLowGrayLevelEmphasis_2d           float64
original_glszm_LowGrayLevelZoneEmphasis_2d                float64
original_glszm_SizeZoneNonUniformity_2d                   float64
original_glszm_SizeZoneNonUniformityNormalized_2d         float64
original_glszm_SmallAreaEmphasis_2d                       float64
original_glszm_SmallAreaHighGrayLevelEmphasis_2d          float64
original_glszm_SmallAreaLowGrayLevelEmphasis_2d           float64
original_glszm_ZoneEntropy_2d                             float64
original_glszm_ZonePercentage_2d                          float64
original_glszm_ZoneVariance_2d                            float64
original_ngtdm_Busyness_2d                                float64
original_ngtdm_Coarseness_2d                              float64
original_ngtdm_Complexity_2d                              float64
original_ngtdm_Contrast_2d                                float64
original_ngtdm_Strength_2d                                float64
dtype: object
```

```
print(merged_common['malignancy'].value_counts(dropna=False))
```

```
malignancy
0    1079
1     495
Name: count, dtype: int64
```

A distribuição da variável malignancy é idêntica à observada no outro CSV.

De forma análoga ao dataset anterior, não foram detetados problemas relativos a valores únicos, valores nulos ou tipos de dados, e a variável malignancy já foi devidamente tratada. O dataset encontra-se, assim, pronto para a fase de feature selection.

Análise Exploratória de Dados

Realizamos uma análise exploratória comparativa dos três datasets disponíveis: nodule_per_row_aggregated (Anot_PyLIDC), merged_anot_3d (Anot_3D) e merged_anot_3d_2d (Anot_3D_2D).

```
datasets = {
    "Anot_PyLIDC": "nodule_per_row_aggregated.csv",
    "Anot_3D": "merged_anot_3d.csv",
    "Anot_3D_2D": "merged_anot_3d_2d.csv"
}

target_col = "malignancy"  # coluna alvo

# Função de análise

def analyze_dataset(name, path):
    df = pd.read_csv(path)

    # Manter apenas colunas numéricas
    df_num = df.select_dtypes(include=[np.number])

    # Estatísticas básicas
    n_features = df_num.shape[1] - 1 if target_col in df_num.columns
else df_num.shape[1]
    var_mean = df_num.var().mean()

    # Correlação com a variável alvo
    if target_col in df_num.columns:
        corr_target = df_num.corr()[target_col].dropna()
        corr_mean = corr_target.abs().mean()
        high_corr = (corr_target.abs() > 0.4).sum()
    else:
        corr_mean = np.nan
        high_corr = 0

    # Percentagem média de outliers
```

```
    outlier_ratio =
(np.abs(zscore(df_num.select_dtypes(include=[np.number]))) >
3).mean().mean() * 100

    # Proporção de malignos
    if target_col in df.columns:
        class_ratio =
df[target_col].value_counts(normalize=True).to_dict()
        malignant_ratio = round(class_ratio.get(1, np.nan) * 100, 2)
# se 1 for maligno
    else:
        malignant_ratio = np.nan

    # Correlação média entre features
    feature_corr_mean =
df_num.corr().abs().where(~np.eye(df_num.corr().shape[0],
dtype=bool)).mean().mean()

    # Retornar resumo
    summary = {
        "Dataset": name,
        "Nº Features": n_features,
        "Variância Média": round(var_mean, 4),
        "Correlação Média (|r|)": round(corr_mean, 3),
        "Features com |r|>0.4": high_corr,
        "Outliers (%)": round(outlier_ratio, 2),
        "% Malignos": malignant_ratio,
        "Correlação média entre features": round(feature_corr_mean, 3)
    }

    return summary, df_num

# Executar análise

results = []
for name, path in datasets.items():
    summary, df_num = analyze_dataset(name, path)
    results.append(summary)

# Converter resultados em DataFrame comparativo
eda_summary = pd.DataFrame(results)
eda_summary = eda_summary.sort_values(by="Correlação Média (|r|)",
ascending=False)

print("\n Comparação entre datasets:")
print(eda_summary)
```

```
 Comparação entre datasets:
      Dataset  Nº Features  Variância Média  Correlação Média (|
```

```
r|)  \
0  Anot_PyLIDC           12     2.447299e+05                    0.312

1      Anot_3D          119     2.584895e+15                    0.271

2   Anot_3D_2D          221     1.430854e+15                    0.259


   Features com |r|>0.4  Outliers (%)  % Malignos  \
0                     3          0.92       12.79
1                    34          1.32       30.73
2                    58          1.35       31.45

   Correlação média entre features
0                            0.173
1                            0.291
2                            0.295
```

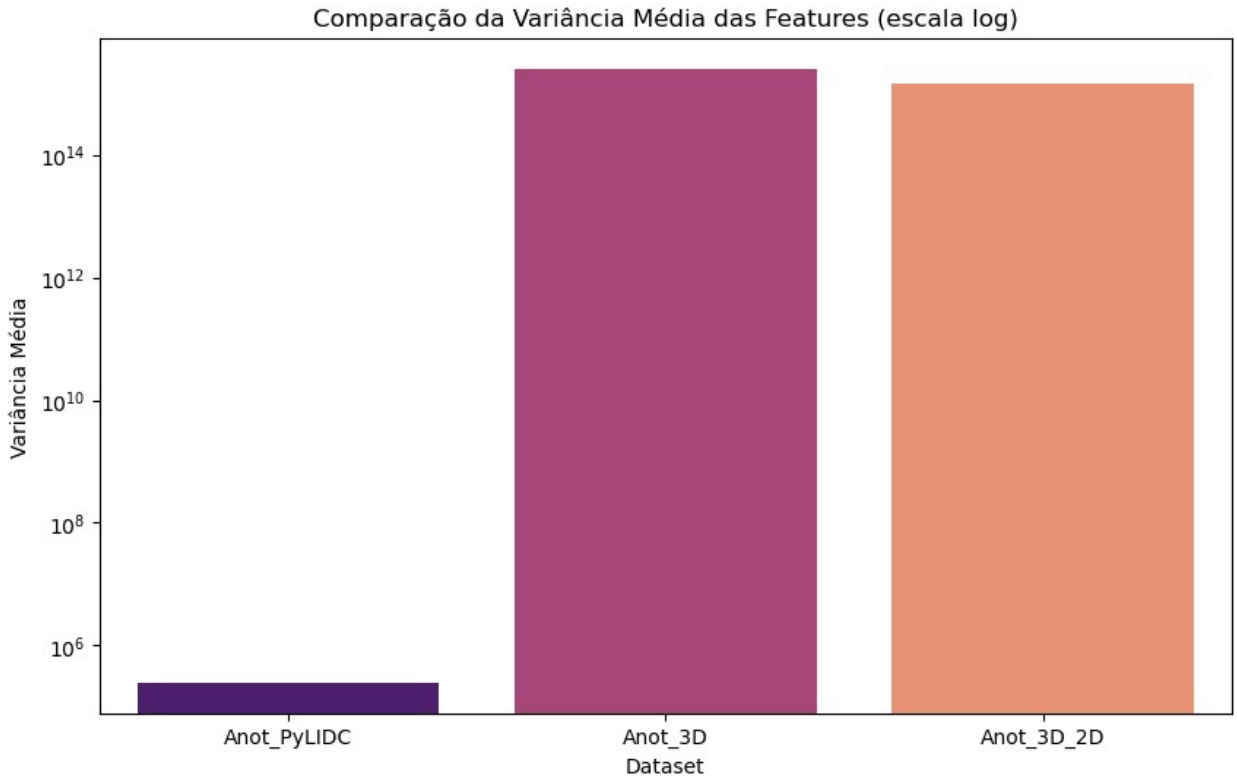
Número de Features

- O Anot_3D tem um bom equilíbrio: não é tão pequeno como o Anot_PyLIDC (pouca informação), nem tão grande como o 3D_2D (potencialmente redundante e mais ruidoso).

Variância Média

```
plt.figure(figsize=(10, 6))
sns.barplot(data=eda_summary, x="Dataset", y="Variância Média",
palette="magma")
plt.yscale("log")
plt.title("Comparação da Variância Média das Features (escala log)")
plt.show()
```

```
/tmp/ipykernel_5175/3994887946.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(data=eda_summary, x="Dataset", y="Variância Média",
palette="magma")
```

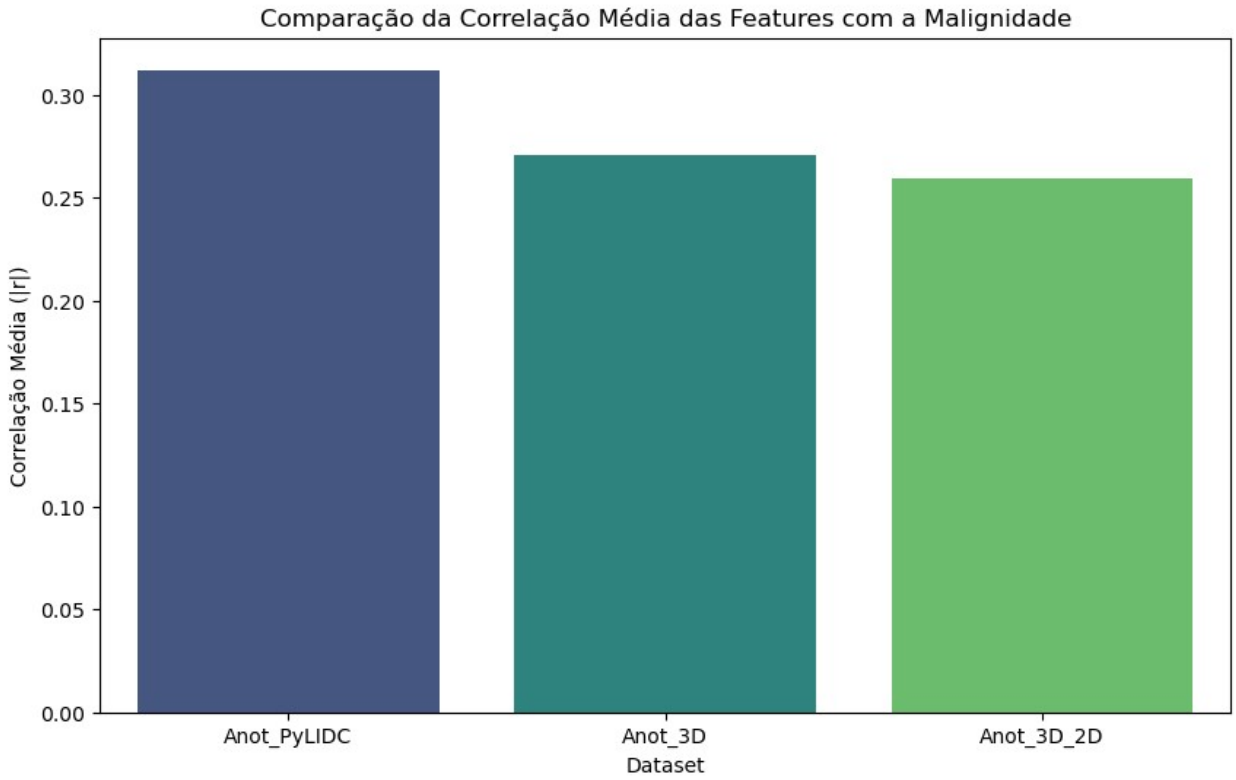


- É mais elevada no Anot_3D indicando maior dispersão dos dados, o que significa que há maior capacidade de distinguir padrões, não existindo diferença elevada para o Anot_3D_2D.
- O Anot_PyLIDC tem variância muito baixa, logo as variáveis têm pouca variação, o que pode ser negativo para modelos preditivos.

Correlação Média com Malignidade:

```
plt.figure(figsize=(10, 6))
sns.barplot(data=eda_summary, x="Dataset", y="Correlação Média (|r|)",
palette="viridis")
plt.title("Comparação da Correlação Média das Features com a
Malignidade")
plt.show()
```

```
/tmp/ipykernel_5175/1356941375.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(data=eda_summary, x="Dataset", y="Correlação Média (|
r|)", palette="viridis")
```

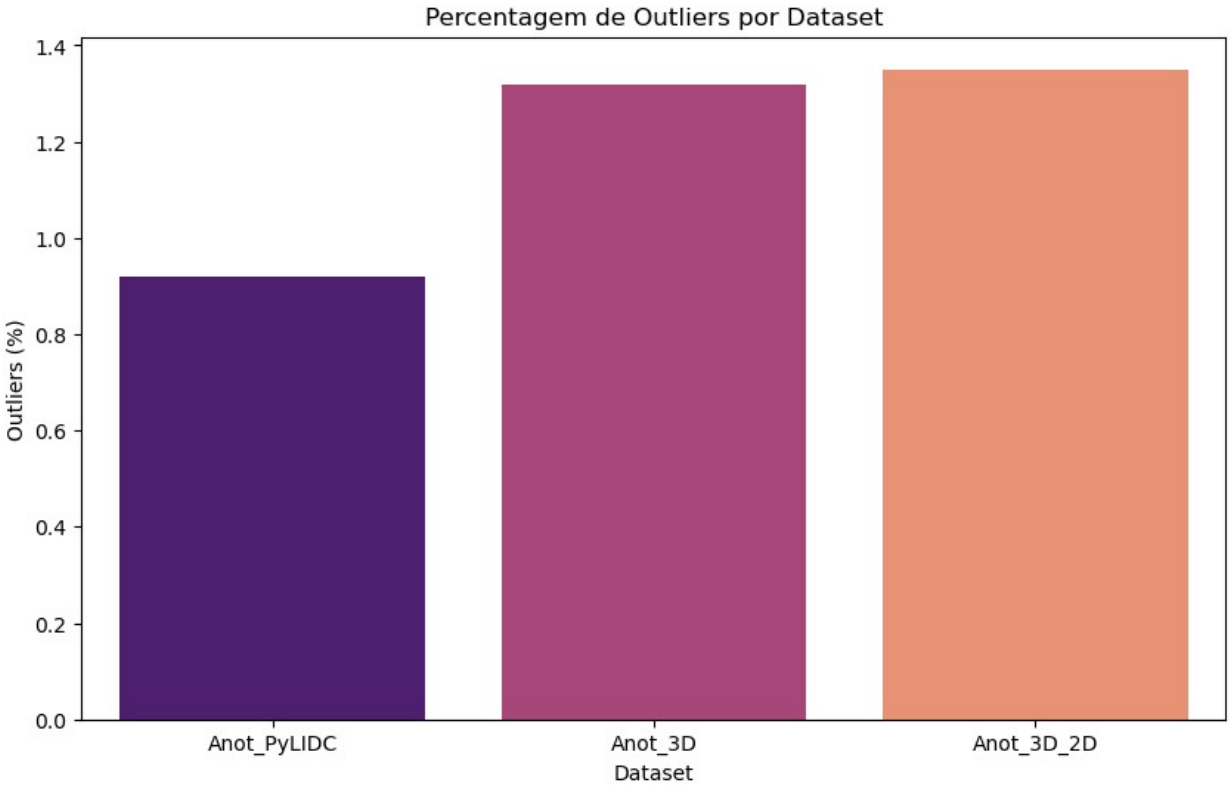


- O Anot_PyLIDC até tem ligeiramente mais correlação com a malignidade, mas só tem 12 features portanto, essa correlação pode ser instável.

Percentagem de Outliers

```
plt.figure(figsize=(10, 6))
sns.barplot(data=eda_summary, x="Dataset", y="Outliers (%)",
palette="magma")
plt.title("Percentagem de Outliers por Dataset")
plt.ylabel("Outliers (%)")
plt.show()
```

```
/tmp/ipykernel_5175/336252848.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(data=eda_summary, x="Dataset", y="Outliers (%)",
palette="magma")
```

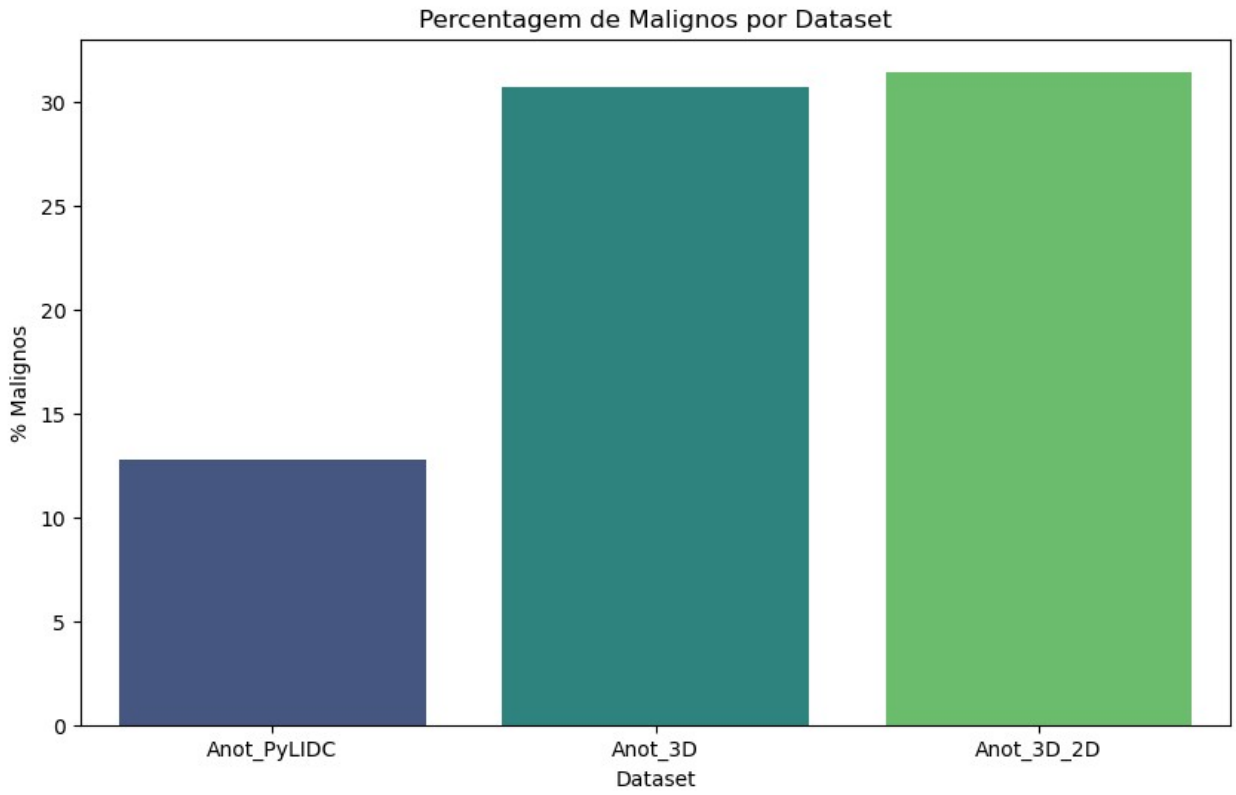


- Valores baixos em todos os datasets.
- Ligeiro aumento no Anot_3D e Anot_3D_2D. Este aumento é esperado, porque estes datasets incluem mais variáveis.

Percentagem de Malignidade

```
plt.figure(figsize=(10, 6))
sns.barplot(data=eda_summary, x="Dataset", y="% Malignos",
palette="viridis")
plt.title("Percentagem de Malignos por Dataset")
plt.show()
```

```
/tmp/ipykernel_5175/4021217398.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(data=eda_summary, x="Dataset", y="% Malignos",
palette="viridis")
```

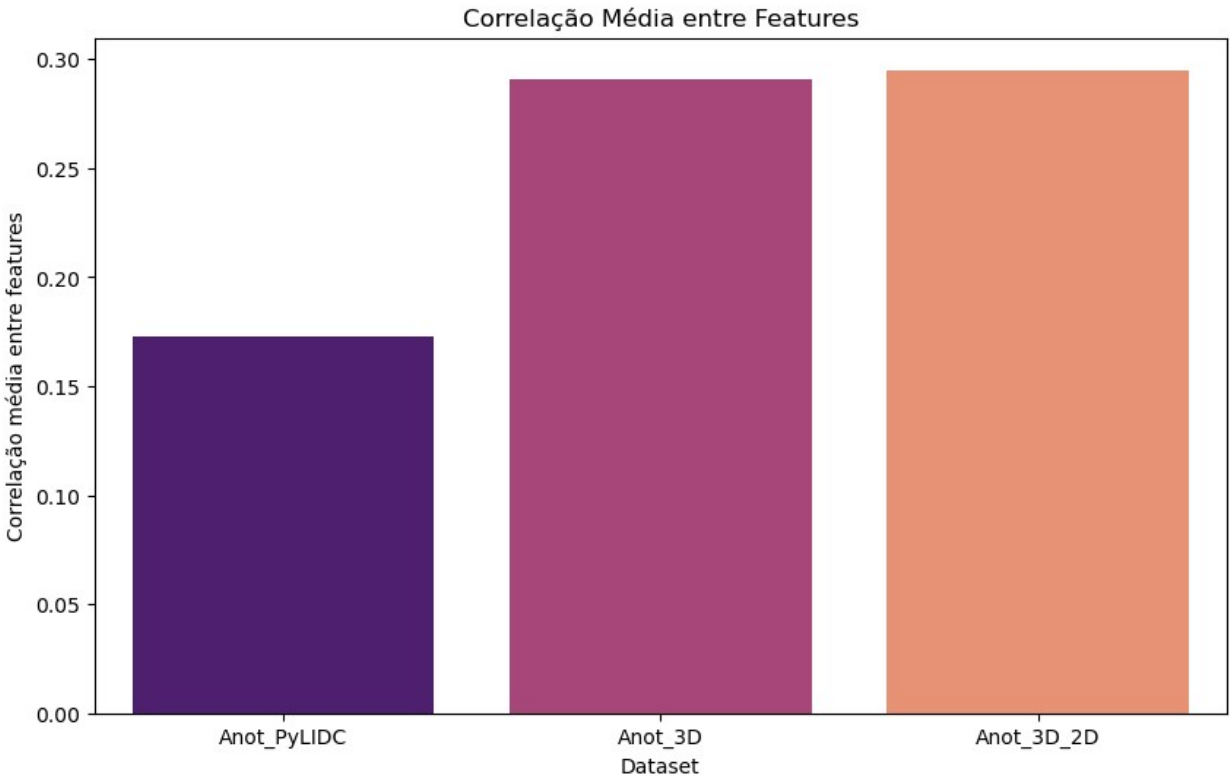


- O Anot_PyLIDC apresenta um forte desbalanceamento (12% malignos).
- Confirma-se o anteriormente visto que Anot_3D e o Anot_3D_2D têm cerca de 30% de casos malignos, o que é mais equilibrado.

Correlação média entre features

```
plt.figure(figsize=(10, 6))
sns.barplot(data=eda_summary, x="Dataset", y="Correlação média entre features", palette="magma")
plt.title("Correlação Média entre Features")
plt.show()
```

```
/tmp/ipykernel_5175/825497634.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.barplot(data=eda_summary, x="Dataset", y="Correlação média entre features", palette="magma")
```



- O Anot_3D_2D tem a correlação entre features mais alta (0.295), ou seja, maior redundância.
- O Anot_3D está logo a seguir, mas mantém-se dentro de um intervalo aceitável (0.29).
- O Anot_PyLIDC tem menor correlação entre features, mas à custa de informação limitada.

Após a análise exploratória comparativa entre os três datasets, o **Anot_3D** foi considerado o mais adequado para usar na feature selection e avaliação dos modelos.

Este dataset combina as anotações dos radiologistas com features 3D, oferecendo um equilíbrio favorável entre número de variáveis, variância média e proporção de casos malignos, refletindo maior diversidade e qualidade informativa das features.

O Anot_PyLIDC, contendo apenas anotações, apresenta uma correlação ligeiramente superior com a "malignancy", mas o reduzido número de atributos e o desbalanceamento entre classes limitam o seu valor preditivo.

O Anot_3D_2D, que adiciona features 2D às features 3D e às anotações, aumenta significativamente o número de variáveis e a redundância entre elas, sem acrescentar informação relevante adicional. Com as features 3D já disponíveis, a informação 2D torna-se essencialmente redundante.

Portanto, o Anot_3D oferece o melhor compromisso entre informação radiológica, equilíbrio das classes e estabilidade estatística das variáveis, sendo o dataset selecionado para a análise exploratória subsequente.

Seleção de variáveis

Nesta etapa foram aplicadas diferentes técnicas com o objetivo de identificar as variáveis mais relevantes para a predição da malignidade dos nódulos pulmonares.

Foram removidas as colunas não numéricas (com exceção de patient_id pois iremos avaliar a performance dos modelos por paciente e não por nódulo) uma vez que estas representam apenas identificadores administrativos e não contêm informação radiológica relevante. A sua presença poderia introduzir ruído, sem contribuir para a capacidade preditiva da malignidade dos nódulos. As colunas do tipo object, como observado acima na análise deste dataset são "series_uid" e "study_uid".

A escolha dos métodos foi baseada no artigo *"Feature selection methods and predictive models in CT lung cancer radiomics"* de Ge & Zhang (2023) que destaca o LASSO como o método de seleção de features mais utilizado tanto em estudos de classificação como de prognóstico, pela sua robustez na eliminação de atributos redundantes e preservação dos mais preditivos. O mesmo artigo também menciona o uso frequente de abordagens baseadas em Random Forest e em correlação de Spearman, o que sustentou a escolha destes métodos neste trabalho.

Adicionalmente, usamos o teste ANOVA para identificar variáveis com diferenças estatisticamente significativas entre classes de malignidade.

```
# Carregar dataset
df = pd.read_csv("merged_anot_3d.csv")

# Definir variável alvo
y = df["malignancy"]

# Definir features, excluindo as colunas indesejadas
X = df.drop(columns=["malignancy", "study_uid", "series_uid"],
errors="ignore")
```

As colunas removidas como não têm relação com o diagnóstico, a sua remoção não afetará o desempenho do modelo nem comprometerá a integridade da análise.

Correlação (Spearman)

O método de correlação de Spearman foi utilizado para identificar as features com maior relação com a variável alvo (malignancy). Ao contrário da correlação de Pearson, a correlação de Spearman é não paramétrica e adequada para relações não lineares, o que é particularmente útil em dados médicos, onde a relação entre as variáveis pode não ser linear. Após calcular a correlação entre cada feature e o alvo, foram removidas variáveis altamente correlacionadas entre si ($\rho > 0.9$), mantendo apenas as mais informativas e independentes para o modelo.

```python
# Calcular a correlação de Spearman entre cada feature e o target
spearman_corr = X.apply(lambda col: spearmanr(col, y).correlation)

# Selecionar features com correlação absoluta maior que um threshold
(ex.: 0.1)
threshold = 0.1
selected_features = spearman_corr[abs(spearman_corr) >=
threshold].index.tolist()

#print(f"Número de features selecionadas: {len(selected_features)}")
#print(selected_features)

# Remover features altamente correlacionadas entre si
corr_matrix = X[selected_features].corr(method='spearman').abs()
upper = corr_matrix.where(np.triu(np.ones(corr_matrix.shape),
k=1).astype(bool))
to_drop = [col for col in upper.columns if any(upper[col] > 0.9)]
print(f"Features a remover por alta correlação: {to_drop}")

# Mantém apenas as features selecionadas e não altamente
correlacionadas
spearman_features = [f for f in selected_features if f not in to_drop]
print(f"Features finais após correlação: {len(spearman_features)}")
print(spearman_features)
```

```
Features a remover por alta correlação: ['volume',
'original_shape_MajorAxisLength',
'original_shape_Maximum2DDiameterColumn',
'original_shape_Maximum2DDiameterRow',
'original_shape_Maximum2DDiameterSlice',
'original_shape_Maximum3DDiameter', 'original_shape_MeshVolume',
'original_shape_MinorAxisLength', 'original_shape_SurfaceArea',
'original_shape_SurfaceVolumeRatio', 'original_shape_VoxelVolume',
'original_firstorder_TotalEnergy', 'original_firstorder_Uniformity',
'original_glcm_DifferenceAverage', 'original_glcm_Idm',
'original_glcm_Idn', 'original_glcm_Imc1', 'original_glcm_Imc2',
'original_glcm_JointAverage', 'original_glcm_JointEntropy',
'original_glcm_MCC', 'original_glcm_MaximumProbability',
'original_glcm_SumAverage', 'original_glcm_SumEntropy',
'original_gldm_DependenceEntropy',
'original_gldm_DependenceNonUniformity',
'original_gldm_GrayLevelNonUniformity',
'original_gldm_HighGrayLevelEmphasis',
'original_gldm_LargeDependenceEmphasis',
'original_gldm_SmallDependenceEmphasis',
'original_gldm_SmallDependenceLowGrayLevelEmphasis',
'original_glrlm_GrayLevelNonUniformity',
'original_glrlm_GrayLevelNonUniformityNormalized',
'original_glrlm_HighGrayLevelRunEmphasis',
```

```
'original_glrlm_LongRunEmphasis',
'original_glrlm_LongRunHighGrayLevelEmphasis',
'original_glrlm_LongRunLowGrayLevelEmphasis',
'original_glrlm_LowGrayLevelRunEmphasis',
'original_glrlm_RunLengthNonUniformity',
'original_glrlm_RunLengthNonUniformityNormalized',
'original_glrlm_RunPercentage', 'original_glrlm_RunVariance',
'original_glrlm_ShortRunEmphasis',
'original_glrlm_ShortRunHighGrayLevelEmphasis',
'original_glrlm_ShortRunLowGrayLevelEmphasis',
'original_glszm_GrayLevelNonUniformity',
'original_glszm_GrayLevelNonUniformityNormalized',
'original_glszm_HighGrayLevelZoneEmphasis',
'original_glszm_LargeAreaEmphasis',
'original_glszm_LargeAreaHighGrayLevelEmphasis',
'original_glszm_LowGrayLevelZoneEmphasis',
'original_glszm_SizeZoneNonUniformity',
'original_glszm_SmallAreaEmphasis',
'original_glszm_SmallAreaHighGrayLevelEmphasis',
'original_glszm_SmallAreaLowGrayLevelEmphasis',
'original_glszm_ZoneEntropy', 'original_glszm_ZonePercentage',
'original_glszm_ZoneVariance', 'original_ngtdm_Coarseness',
'original_ngtdm_Complexity']
Features finais após correlação: 41
['num_annotations', 'diameter', 'calcification', 'subtlety',
'spiculation', 'lobulation', 'margin', 'sphericity',
'original_shape_Flatness', 'original_shape_LeastAxisLength',
'original_shape_Sphericity', 'original_firstorder_Energy',
'original_firstorder_Entropy', 'original_firstorder_Kurtosis',
'original_firstorder_Maximum', 'original_firstorder_Median',
'original_firstorder_Minimum', 'original_firstorder_Range',
'original_firstorder_RootMeanSquared', 'original_firstorder_Skewness',
'original_glcm_Autocorrelation', 'original_glcm_ClusterProminence',
'original_glcm_ClusterShade', 'original_glcm_Contrast',
'original_glcm_Correlation', 'original_glcm_DifferenceEntropy',
'original_glcm_Id', 'original_glcm_Idmn',
'original_glcm_InverseVariance', 'original_glcm_JointEnergy',
'original_gldm_DependenceNonUniformityNormalized',
'original_gldm_DependenceVariance',
'original_gldm_LargeDependenceHighGrayLevelEmphasis',
'original_gldm_LargeDependenceLowGrayLevelEmphasis',
'original_gldm_LowGrayLevelEmphasis',
'original_gldm_SmallDependenceHighGrayLevelEmphasis',
'original_glrlm_RunEntropy',
'original_glszm_SizeZoneNonUniformityNormalized',
'original_ngtdm_Busyness', 'original_ngtdm_Contrast',
'original_ngtdm_Strength']
```

ANOVA

O teste ANOVA avalia a relação entre cada feature e a variável alvo, medindo se as médias entre grupos (ex.: benigno vs maligno) diferem significativamente. No contexto deste projeto, foi utilizado para identificar as features que apresentam maior poder discriminativo entre classes, ajudando a reduzir o número de variáveis e a melhorar a interpretabilidade do modelo.

```
# Normalizar os dados (opcional, mas ajuda a estabilizar os valores)
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Selecionar as K melhores features com base no teste ANOVA
selector = SelectKBest(score_func=f_classif, k=20)  # podes ajustar o valor de k
X_new = selector.fit_transform(X_scaled, y)

# Features selecionadas
anova_features = X.columns[selector.get_support()]

print(f"Número de features selecionadas: {len(anova_features)}")
print(anova_features.tolist())
```

```
Número de features selecionadas: 20
['diameter', 'original_shape_LeastAxisLength',
'original_shape_MajorAxisLength',
'original_shape_Maximum2DDiameterColumn',
'original_shape_Maximum2DDiameterRow',
'original_shape_Maximum2DDiameterSlice',
'original_shape_Maximum3DDiameter', 'original_shape_MinorAxisLength',
'original_shape_SurfaceArea', 'original_shape_SurfaceVolumeRatio',
'original_glcm_Idmn', 'original_glcm_Idn', 'original_glcm_Imc1',
'original_glcm_Imc2', 'original_glcm_JointEntropy',
'original_glcm_MCC', 'original_gldm_DependenceEntropy',
'original_glszm_GrayLevelNonUniformity', 'original_glszm_ZoneEntropy',
'original_ngtdm_Strength']
```

LASSO (L1 Regularization)

O método LASSO utiliza regularização L1 para penalizar coeficientes menos relevantes durante o treino de um modelo linear, forçando alguns deles a zero. Desta forma, o LASSO atua simultaneamente como técnica de regressão e de seleção de features. No contexto deste projeto, foi usado para identificar as variáveis com maior impacto na predição de malignidade, reduzindo a dimensionalidade do conjunto de dados e prevenindo overfitting.

```
# Normalizar os dados — essencial para o LASSO
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

```
# LASSO com validação cruzada para escolher o melhor alpha
(penalização)
lasso = LassoCV(cv=5, random_state=42, max_iter=10000)
lasso.fit(X_scaled, y)

# Selecionar apenas as features com coeficiente diferente de 0
lasso_features = X.columns[lasso.coef_ != 0]

print(f"Número de features selecionadas: {len(lasso_features)}")
print(lasso_features.tolist())
```

```
Número de features selecionadas: 32
['num_annotations', 'diameter', 'volume', 'calcification', 'subtlety',
'spiculation', 'lobulation', 'margin', 'texture', 'sphericity',
'original_shape_Elongation', 'original_shape_Flatness',
'original_shape_MinorAxisLength', 'original_firstorder_Kurtosis',
'original_firstorder_Minimum', 'original_firstorder_Range',
'original_glcm_ClusterShade', 'original_glcm_Idn',
'original_glcm_Imc1', 'original_glcm_Imc2',
'original_glcm_JointAverage', 'original_glcm_MCC',
'original_glcm_SumAverage', 'original_glcm_SumEntropy',
'original_gldm_DependenceEntropy',
'original_gldm_LargeDependenceHighGrayLevelEmphasis',
'original_gldm_SmallDependenceLowGrayLevelEmphasis',
'original_glszm_LargeAreaHighGrayLevelEmphasis',
'original_glszm_SmallAreaEmphasis', 'original_glszm_ZoneEntropy',
'original_ngtdm_Complexity', 'original_ngtdm_Contrast']
```

Random Forest

O Random Forest avalia a importância das features durante a construção das árvores, permitindo selecionar as variáveis mais relevantes para a classificação dos nódulos.

```
# Supondo que já tens X (features) e y (labels)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42, stratify=y)

# Treina apenas no conjunto de treino
rf = RandomForestClassifier(n_estimators=200, random_state=42)
rf.fit(X_train, y_train)

# Obtém importâncias
importances = pd.Series(rf.feature_importances_,
index=X.columns).sort_values(ascending=False)

print("Top 10 features mais importantes:")
print(importances.head(10))

# Seleção de features acima de um limiar
```

```python
threshold = 0.01
rf_features = importances[importances > threshold].index.tolist()

print(f"\nNúmero de features selecionadas: {len(rf_features)}")
print(rf_features)
```

```
Top 10 features mais importantes:
diameter                                0.066793
original_shape_MinorAxisLength          0.056581
original_glrlm_RunLengthNonUniformity   0.053380
original_shape_SurfaceArea              0.041958
volume                                  0.038518
original_shape_MeshVolume               0.032723
original_shape_Maximum3DDiameter        0.031296
original_glszm_GrayLevelNonUniformity   0.029346
original_gldm_DependenceEntropy         0.028438
original_glszm_SizeZoneNonUniformity    0.024872
dtype: float64

Número de features selecionadas: 22
['diameter', 'original_shape_MinorAxisLength',
'original_glrlm_RunLengthNonUniformity', 'original_shape_SurfaceArea',
'volume', 'original_shape_MeshVolume',
'original_shape_Maximum3DDiameter',
'original_glszm_GrayLevelNonUniformity',
'original_gldm_DependenceEntropy',
'original_glszm_SizeZoneNonUniformity',
'original_shape_Maximum2DDiameterRow',
'original_shape_Maximum2DDiameterSlice',
'original_gldm_DependenceNonUniformity', 'original_shape_VoxelVolume',
'original_glcm_Imc1', 'original_glrlm_GrayLevelNonUniformity',
'original_gldm_GrayLevelNonUniformity',
'original_shape_Maximum2DDiameterColumn',
'original_glszm_ZoneEntropy', 'original_ngtdm_Strength',
'original_firstorder_90Percentile', 'original_ngtdm_Coarseness']
```

```python
# Criar dicionário com todos os conjuntos de features selecionadas
feature_sets = {
    'SPEARMAN': spearman_features,
    'ANOVA': anova_features,
    'LASSO': lasso_features,
    'RANDOM_FOREST': rf_features
}
```

Modelação e Avaliação

Justificação: Avaliação ao Nível do Paciente

Embora a extração e análise das features radiómicas seja realizada ao nível do nódulo, a decisão clínica sobre malignidade ocorre ao nível do paciente. Um paciente é considerado positivo (maligno) se apresentar pelo menos um nódulo maligno.

Avaliar o desempenho ao nível do nódulo poderia inflacionar artificialmente as métricas, uma vez que vários nódulos pertencem ao mesmo paciente, violando o pressuposto de independência das amostras. Assim, optou-se por avaliar ao nível do paciente, obtendo resultados mais conservadores, mas clinicamente mais realistas e robustos.

Preparação dos Dados

Antes da modelação, os dados são preparados de forma a garantir consistência e evitar fugas de informação:

- A variável alvo (malignancy) é binária: 0 = benigno, 1 = maligno.
- As variáveis de identificação (patient_id, nodule_cluster, etc.) são removidas do conjunto de treino.
- A validação cruzada é realizada de modo agrupado por paciente, garantindo que nódulos do mesmo paciente nunca aparecem simultaneamente em treino e teste. Isto assegura uma avaliação justa e evita overfitting devido à correlação intra-paciente.

```
# Carregar o dataset final (ajustar o nome do ficheiro se necessário)
df = pd.read_csv("merged_anot_3d.csv")

# Selecionar coluna alvo
TARGET_CANDS = ("malignancy", "label", "risk")
target = next((c for c in TARGET_CANDS if c in df.columns), None)
assert target is not None, "Coluna alvo não encontrada."

# Converter alvo para binário: 1-2 benigno (0), 4-5 maligno (1), 3
excluído
df = df.loc[df[target].notna()].copy()
vals = pd.to_numeric(df[target], errors="coerce")
if not set(vals.dropna().astype(int).unique()) <= {0,1}:
    df = df.loc[~vals.isin([3])].copy()
    df[target] = vals.map({1:0, 2:0, 4:1, 5:1}).astype(int)

# Definir grupos por paciente
groups = df["patient_id"].astype(str).values

# Construir X (apenas variáveis numéricas, sem identificadores ou
alvo)
LEAK = {target, "malignancy","label", "risk"}
IDS  =
{"patient_id","study_uid","series_uid","nodule_cluster","lesion_id","n
odule_id",
        "cluster_id","InstanceNumber","SOPInstanceUID"}
```

```
X = df.drop(columns=[c for c in (LEAK|IDS) if c in df.columns],
errors="ignore")
X = X.select_dtypes(include=[np.number]).copy()
y = df[target].astype(int).values

print("Dimensões:", X.shape, "Distribuição alvo:", np.bincount(y))
```

```
Dimensões: (1611, 118) Distribuição alvo: [1116  495]
```

Definição dos Modelos e Validação

Nesta fase, serão comparados três modelos clássicos de classificação binária amplamente utilizados em problemas de diagnóstico médico:

- Regressão Logística: modelo linear de referência, útil pela sua interpretabilidade.
- Support Vector Machine (SVM) com kernel RBF: capaz de modelar relações não lineares entre as features e a malignidade.
- Random Forest: modelo baseado em múltiplas árvores de decisão, robusto a ruído e capaz de capturar interações complexas entre variáveis.

Os seguintes artigos demonstram a eficácia dos modelos escolhidos e foram a base para a escolha dos mesmos:

- Regressão Logística: "A Machine-Learning Approach Using PET-Based Radiomics to Predict the Histological Subtypes of Lung Cancer" - Seung Hyup Hyun, Mi Sun Ahn, Young Wha Koh, Su Jin Lee (2019).
- Support Vector Machine (SVM): “Automated system for lung nodules classification based on wavelet feature descriptor and support vector machine” - Barrientos A. E. (2015)
- Random Forest: "Highly accurate model for prediction of lung nodule malignancy with CT scans" - Causey, J. L., Chae, M., Hsieh, S., & Choi, J. (2018).

A validação será realizada através de StratifiedGroupKFold, garantindo:

- Estratificação da variável alvo (mantém a proporção de malignos/benignos em cada fold).
- Separação por paciente (nódulos do mesmo paciente não aparecem simultaneamente em treino e teste).

Esta abordagem assegura uma avaliação justa e realista, evitando fuga de informação entre folds.

```

```

```
# Pré-processamentos
prep_scaled = Pipeline([
    ("imp", SimpleImputer(strategy="median")),
    ("scaler", StandardScaler()),
])
prep_impute = Pipeline([
    ("imp", SimpleImputer(strategy="median")),
])

# Modelos
models = {
    "LogisticRegression": Pipeline([
        ("prep", prep_scaled),
        ("clf", LogisticRegression(max_iter=2000,
class_weight="balanced", solver="lbfgs", random_state=42))
    ]),
    "SVM_RBF": Pipeline([
        ("prep", prep_scaled),
        ("clf", SVC(kernel="rbf", probability=True,
class_weight="balanced", random_state=42))
    ]),
    "RandomForest": Pipeline([
        ("prep", prep_impute),
        ("clf", RandomForestClassifier(n_estimators=400,
random_state=42, n_jobs=-1, class_weight="balanced"))
    ]),
}

# Validação cruzada por paciente
cv = StratifiedGroupKFold(n_splits=5, shuffle=True, random_state=42)
```

Métricas e Calibração

A avaliação dos modelos será realizada com base nas seguintes métricas principais:

- AUC ROC: mede a capacidade global de separação entre classes.
- AUC PR: mais informativa em contextos de classes desbalanceadas.
- Sensibilidade (Recall): proporção de nódulos malignos corretamente identificados.
- Precisão (Precision): proporção de predições malignas que são efetivamente corretas.
- F1-score: média harmónica entre precisão e sensibilidade, útil em cenários desbalanceados.

- Matriz de confusão: permite observar diretamente os verdadeiros e falsos positivos/negativos.

As probabilidades de saída dos modelos serão calibradas e o limiar de decisão será ajustado de forma a privilegiar a sensibilidade, assegurando simultaneamente uma especificidade mínima aceitável. Esta abordagem reflete o contexto clínico, onde detetar malignidades é prioritário face ao risco de falsos positivos.

```
def avaliar_modelos_com_feature_sets(models, feature_sets, X, y,
groups, espec_min=0.85):
    resultados = []

    for fs_nome, fs_cols in feature_sets.items():
        X_fs = X[fs_cols].copy()  # selecionar só as features deste
set

        for nome, pipe in models.items():
            y_all, p_all = [], []
            aucs, pr_aucs, briers = [], [], []

            # Validação cruzada
            for tr, te in cv.split(X_fs, y, groups):
                Xtr, Xte = X_fs.iloc[tr], X_fs.iloc[te]
                ytr, yte = y[tr], y[te]

                cal = CalibratedClassifierCV(pipe, cv=3,
method="isotonic")
                cal.fit(Xtr, ytr)
                p = cal.predict_proba(Xte)[:, 1]

                y_all.extend(yte)
                p_all.extend(p)

                aucs.append(roc_auc_score(yte, p))
                pr_aucs.append(average_precision_score(yte, p))
                briers.append(brier_score_loss(yte, p))

            # Optimização do limiar (max sensibilidade com espec >=
espec_min)
            y_all = np.array(y_all)
            p_all = np.array(p_all)
            fpr, tpr, thr = roc_curve(y_all, p_all)
            spec = 1 - fpr
            mask = spec >= espec_min
            thr_opt = thr[mask][np.argmax(tpr[mask])] if np.any(mask)
else 0.5

            yhat = (p_all >= thr_opt).astype(int)
```

```
            resultados.append({
                "Feature Set": fs_nome,
                "Modelo": nome,
                "AUC ROC": np.mean(aucs),
                "AUC PR": np.mean(pr_aucs),
                "Brier": np.mean(briers),
                "Limiar Ótimo": float(thr_opt),
                "Accuracy": accuracy_score(y_all, yhat),
                "Precision": precision_score(y_all, yhat,
zero_division=0),
                "Recall": recall_score(y_all, yhat, zero_division=0),
                "F1": f1_score(y_all, yhat, zero_division=0),
                "Matriz Confusão": confusion_matrix(y_all, yhat)
            })

    return pd.DataFrame(resultados).sort_values(["Feature Set", "AUC
ROC"], ascending=[True, False])


# Executar a avaliação em todos os feature sets
res = avaliar_modelos_com_feature_sets(models, feature_sets, X, y,
groups, espec_min=0.85)
res
```

	Feature Set	Modelo	AUC ROC	AUC PR	Brier	\
5	ANOVA	RandomForest	0.922776	0.859006	0.096125	
3	ANOVA	LogisticRegression	0.920302	0.840408	0.095267	
4	ANOVA	SVM_RBF	0.912989	0.822573	0.097714	
7	LASSO	SVM_RBF	0.944772	0.909333	0.073952	
6	LASSO	LogisticRegression	0.943808	0.903248	0.077615	
8	LASSO	RandomForest	0.943594	0.903412	0.077919	
11	RANDOM_FOREST	RandomForest	0.931139	0.877147	0.089475	
10	RANDOM_FOREST	SVM_RBF	0.923299	0.836028	0.094723	
9	RANDOM_FOREST	LogisticRegression	0.921561	0.838137	0.097475	
1	SPEARMAN	SVM_RBF	0.944112	0.902101	0.077323	
0	SPEARMAN	LogisticRegression	0.943908	0.894792	0.080320	
2	SPEARMAN	RandomForest	0.943566	0.895528	0.080241	

	Limiar Ótimo	Accuracy	Precision	Recall	F1	\
5	0.224444	0.852266	0.718166	0.854545	0.780443	
3	0.244507	0.855369	0.722789	0.858586	0.784857	
4	0.273623	0.862818	0.735395	0.864646	0.794800	
7	0.193563	0.865922	0.727569	0.901010	0.805054	
6	0.203922	0.866543	0.731023	0.894949	0.804723	
8	0.188034	0.863439	0.725780	0.892929	0.800725	
11	0.222823	0.859714	0.729131	0.864646	0.791128	
10	0.232181	0.857852	0.722408	0.872727	0.790485	
9	0.250836	0.855990	0.721754	0.864646	0.786765	
1	0.219577	0.869646	0.733224	0.905051	0.810127	

```
0       0.251337  0.871508   0.741611  0.892929  0.810266
2       0.227986  0.864680   0.728171  0.892929  0.802178

             Matriz Confusão
5   [[950, 166], [72, 423]]
3   [[953, 163], [70, 425]]
4   [[962, 154], [67, 428]]
7   [[949, 167], [49, 446]]
6   [[953, 163], [52, 443]]
8   [[949, 167], [53, 442]]
11  [[957, 159], [67, 428]]
10  [[950, 166], [63, 432]]
9   [[951, 165], [67, 428]]
1   [[953, 163], [47, 448]]
0   [[962, 154], [53, 442]]
2   [[951, 165], [53, 442]]
```

Apesar do desbalanceamento do target (70/30), os resultados obtidos mostraram-se bastante satisfatórios. Por esse motivo, optámos por não aplicar técnicas de data augmentation para os nódulos malignos (classe 30%).

De forma a simplificar a avaliação e focar nos aspetos clínicos mais críticos, decidimos selecionar, para cada modelo, apenas o subset de features que apresenta maior recall. No contexto deste estudo, o recall é a métrica mais importante, uma vez que o pior cenário seria classificar um paciente como benigno quando ele possui um nódulo maligno. Abaixo estão apresentados os subsets de features escolhidos com base neste critério.

```
melhores_modelos = {
    "RandomForest": (models["RandomForest"], feature_sets["LASSO"]),
    "LogisticRegression": (models["LogisticRegression"],
feature_sets["LASSO"]),
    "SVM_RBF": (models["SVM_RBF"], feature_sets["SPEARMAN"])
}
```

Como referido acima, os resultados obtidos representam a performance dos modelos ao nível do paciente, e não apenas ao nível do nódulo individual.

A avaliação ao nível do paciente permite compreender melhor a capacidade do modelo em apoiar diagnósticos clínicos, refletindo o verdadeiro impacto de um modelo preditivo num contexto médico real.

Visualização de Resultados

Esta secção apresenta as principais representações gráficas para a análise comparativa dos modelos desenvolvidos:

- Curvas ROC: comparativas entre os três modelos com os melhores subsets, obtidas a partir das previsões out-of-fold e avaliadas ao nível do paciente.

- Matrizes de confusão: geradas para cada modelo, considerando o limiar de decisão otimizado para maximizar a sensibilidade, mantendo uma especificidade mínima pré-definida.
- Distribuição das probabilidades preditas: histogramas das probabilidades atribuídas à classe positiva (maligna), permitindo comparar a separação entre os casos benignos e malignos em cada modelo.

Estas visualizações permitem avaliar tanto a discriminação global dos modelos como o comportamento prático na tomada de decisão clínica.

```
# Parâmetro: especificidade mínima para optimização do limiar
ESPEC_MIN = 0.85


# 1) Obter previsões out-of-fold (probabilidades) para cada modelo com
calibração interna
oof = {}
for nome, (pipe, fs_cols) in melhores_modelos.items():
    y_all, p_all = [], []
    for tr, te in cv.split(X, y, groups):
        Xtr, Xte = X.iloc[tr][fs_cols], X.iloc[te][fs_cols]   # só
usar features do set correspondente
        ytr, yte = y[tr], y[te]
        cal = CalibratedClassifierCV(pipe, cv=3, method="isotonic")
        cal.fit(Xtr, ytr)
        p = cal.predict_proba(Xte)[:, 1]
        y_all.extend(yte); p_all.extend(p)

    y_all = np.array(y_all); p_all = np.array(p_all)

    # Optimização do limiar: máxima sensibilidade com especificidade
>= ESPEC_MIN
    fpr, tpr, thr = roc_curve(y_all, p_all)
    spec = 1 - fpr
    mask = spec >= ESPEC_MIN
    thr_opt = thr[mask][np.argmax(tpr[mask])] if np.any(mask) else 0.5

    yhat = (p_all >= thr_opt).astype(int)
    cm = confusion_matrix(y_all, yhat)

    oof[nome] = {
        "y_true": y_all,
        "y_prob": p_all,
        "thr_opt": float(thr_opt),
        "cm": cm,
        "fpr": fpr,
        "tpr": tpr,
        "roc_auc": auc(fpr, tpr)
    }
```

```
# Ajustes gráficos globais
plt.rcParams.update({"figure.figsize": (7,5), "font.size": 11})
```

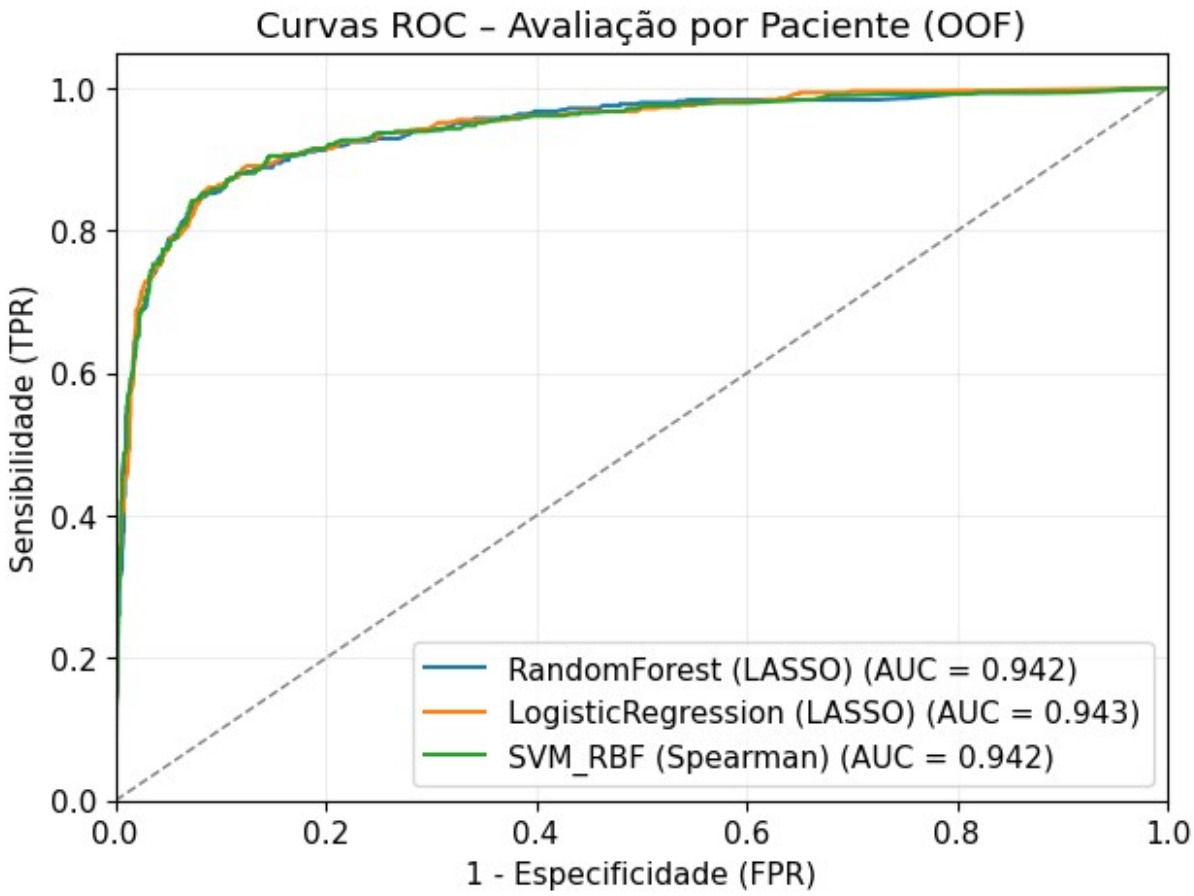
Curvas ROC

As curvas ROC (Receiver Operating Characteristic) seguintes foram construídas com as previsões out-of-fold, garantindo uma avaliação independente e livre de data leakage. Estas curvas ilustram o trade-off entre a sensibilidade (recall) e 1 - especificidade (false positive rate) para cada modelo testado, permitindo comparar a sua capacidade discriminativa.

A Área sob a Curva (AUC ROC) é apresentada para cada modelo e resume o desempenho global:

- Valores mais próximos de 1 indicam melhor separação entre classes (maligno vs benigno).
- Valores próximos de 0.5 indicam desempenho equivalente ao acaso.

```
plt.figure()
for nome, d in oof.items():
    plt.plot(d["fpr"], d["tpr"], label=f"{nome} (AUC =
{d['roc_auc']:.3f})")
plt.plot([0,1],[0,1], linestyle="--", linewidth=1, color="grey")
plt.xlim([0.0, 1.0]); plt.ylim([0.0, 1.05])
plt.xlabel("1 - Especificidade (FPR)")
plt.ylabel("Sensibilidade (TPR)")
plt.title("Curvas ROC — Avaliação por Paciente (OOF)")
plt.legend(loc="lower right")
plt.grid(alpha=0.2)
plt.show()
```



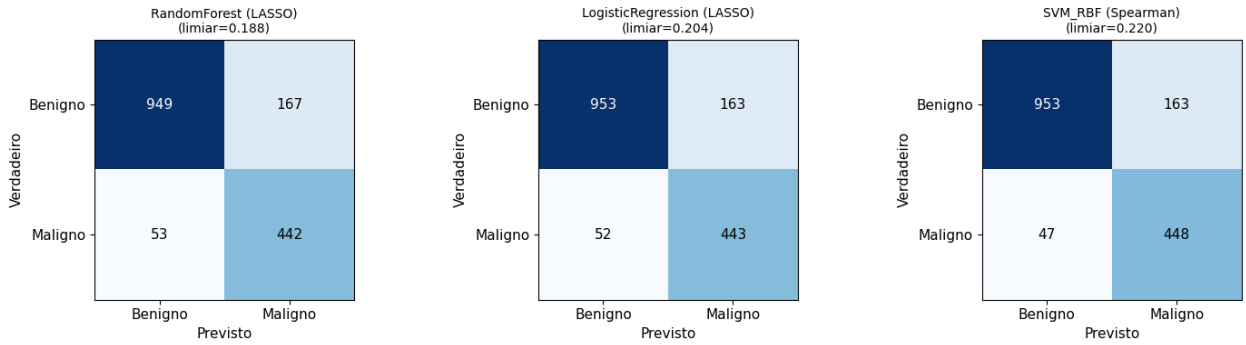
A análise das curvas ROC não revelou diferenças significativas entre os modelos, indicando que todos apresentam capacidade discriminativa semelhante ao nível do paciente.

Matrizes de Confusão por Modelo

As matrizes de confusão abaixo foram geradas utilizando o limiar de decisão otimizado para cada modelo, definido pela maximização da sensibilidade sob a restrição de uma especificidade mínima predefinida. Esta abordagem permite avaliar o equilíbrio entre deteção de casos malignos e controlo de falsos positivos, refletindo melhor o comportamento clínico esperado dos modelos.

```
def plot_confusion_matrix_subplot(ax, cm,
classes=("Benigno","Maligno"), title="Matriz de Confusão"):
    im = ax.imshow(cm, interpolation='nearest', cmap=plt.cm.Blues)
    ax.set_title(title, fontsize=10)
    tick_marks = np.arange(len(classes))
    ax.set_xticks(tick_marks); ax.set_xticklabels(classes)
    ax.set_yticks(tick_marks); ax.set_yticklabels(classes)

    thresh = cm.max() / 2.0
```

```
    for i, j in itertools.product(range(cm.shape[0]),
range(cm.shape[1])):
        ax.text(j, i, format(cm[i, j], "d"),
                ha="center", va="center",
                color="white" if cm[i, j] > thresh else "black")

    ax.set_ylabel("Verdadeiro")
    ax.set_xlabel("Previsto")

# Criar subplots horizontais
fig, axes = plt.subplots(1, len(oof), figsize=(15,4))

for ax, (nome, d) in zip(axes, oof.items()):
    cm = d["cm"]
    thr = d["thr_opt"]
    title = f"{nome}\n(limiar={thr:.3f})"
    plot_confusion_matrix_subplot(ax, cm, title=title)

plt.tight_layout()
plt.show()
```



No SVM, o número de falsos negativos foi o mais baixo, refletindo o maior recall entre os modelos. Em termos de falsos positivos, o Logistic Regression ficou empatado com o SVM.

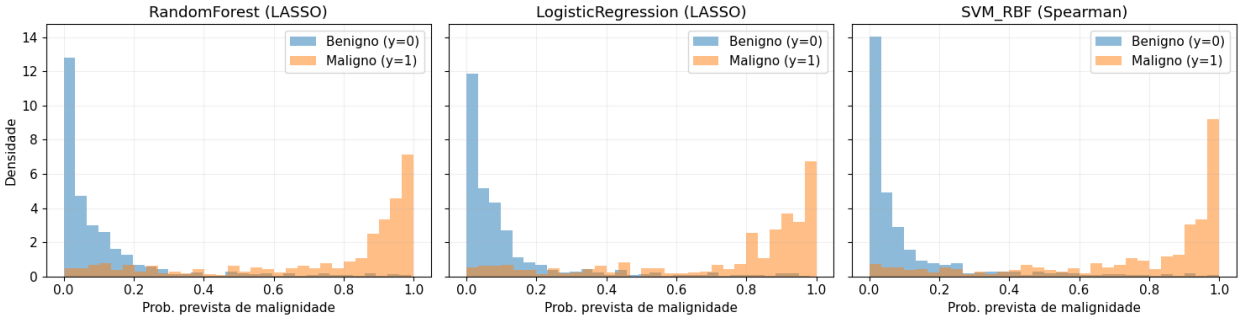
No geral, as diferenças entre os modelos foram mínimas em pontos percentuais, indicando desempenho comparável no equilíbrio entre sensibilidade e especificidade.

Histogramas de Probabilidades Previstas

Os histogramas seguintes mostram, para cada modelo, a distribuição das probabilidades previstas para a classe positiva, comparando casos benignos e malignos. Esta visualização permite avaliar o grau de separação entre as classes e identificar eventuais sobreposições nas previsões, fornecendo uma perspetiva intuitiva sobre a discriminabilidade após calibração.

```
bins = 30
fig, axes = plt.subplots(1, len(oof), figsize=(15,4), sharey=True)

for ax, (nome, d) in zip(axes, oof.items()):
    y_true = d["y_true"]
```

```
    p = d["y_prob"]
    p_neg = p[y_true == 0]
    p_pos = p[y_true == 1]

    ax.hist(p_neg, bins=bins, density=True, alpha=0.5, label="Benigno
(y=0)")
    ax.hist(p_pos, bins=bins, density=True, alpha=0.5, label="Maligno
(y=1)")
    ax.set_title(f"{nome}")
    ax.set_xlabel("Prob. prevista de malignidade")
    ax.grid(alpha=0.2)
    if ax == axes[0]:
        ax.set_ylabel("Densidade")
    ax.legend()

plt.tight_layout()
plt.show()
```



A análise dos histogramas confirma a mesma tendência observada nas métricas anteriores: os modelos apresentam desempenhos muito semelhantes, com uma ligeira vantagem para o SVM na separação entre casos benignos e malignos.

Nota sobre interpretação:

1. Curvas ROC: Quanto mais próxima do canto superior esquerdo a curva estiver e quanto maior for o valor da AUC, melhor é a capacidade do modelo em discriminar entre pacientes benignos e malignos.
2. Matrizes de confusão: Com o limiar optimizado, a prioridade foi maximizar a sensibilidade, mantendo a especificidade mínima definida. É importante observar os falsos negativos e falsos positivos resultantes.
3. Histogramas: Uma boa separação das distribuições indica clara distinção entre casos benignos e malignos. Sobreposição sugere casos limite, nos quais a escolha do limiar é crítica para o desempenho do modelo.

Conclusão Geral da Modelação

AUC ROC

Todos os modelos apresentam valores elevados de AUC ROC, entre ~0.91 e ~0.94, indicando uma excelente capacidade discriminativa entre pacientes benignos e malignos. Os modelos baseados em SVM RBF e Logistic Regression com subsets LASSO ou SPEARMAN atingem os valores mais altos (~0.944), sugerindo ligeira vantagem sobre Random Forest.

AUC PR

A análise da AUC PR confirma a tendência da AUC ROC, com valores também elevados (~0.82 a ~0.91). Os modelos SVM RBF com LASSO e SPEARMAN obtêm as melhores AUC PR (~0.909–0.902), indicando boa performance mesmo considerando o desbalanceamento da classe positiva (~30%).

Recall (Sensibilidade)

O recall é a métrica mais crítica neste contexto clínico. Os valores mais altos de recall (~0.90–0.91) são alcançados pelo SVM RBF e Logistic Regression com subsets LASSO e SPEARMAN, evidenciando que estes modelos são os mais eficazes em detetar pacientes malignos, minimizando falsos negativos. Random Forest apresenta um recall ligeiramente inferior (~0.85–0.89), ainda assim aceitável.

Conclusão Integrada

De forma geral, todos os modelos têm desempenho muito semelhante, com pequenas diferenças percentuais entre métricas. O SVM RBF com subsets LASSO ou SPEARMAN apresenta ligeira vantagem global, principalmente em recall, o que o torna o mais adequado para o contexto clínico, onde não detetar malignidade representa o maior risco. As outras métricas (precision, F1, accuracy) confirmam que os modelos mantêm equilíbrio entre sensibilidade e especificidade, sem comprometer a robustez geral.

> Relembrando que no pré-processamento foram eliminados os nódulos cuja malignidade estava avaliada como 3. Por isso, esta separação e bons resultados observados é particularmente evidente, conforme esperado segundo o estudo mencionado nessa fase de processamento.

Avaliação pelo diâmetro

Apenas 16% dos casos de cancro do pulmão são diagnosticados precocemente, pelo que a deteção de nódulos em estádios iniciais é de extrema importância, especialmente porque a taxa de sobrevivência aos cinco anos diminui drasticamente em fases avançadas da doença. Isto reforça o valor do diagnóstico precoce.

No contexto da avaliação dos modelos, é fundamental não considerar apenas a accuracy global, mas também o seu impacto clínico. Métricas como a sensibilidade na deteção de tumores pequenos ou a capacidade de identificar nódulos em fases iniciais são particularmente críticas. (American Cancer Society, 2019)

Por estes motivos, analisámos o comportamento da malignidade em função do tamanho do nódulo, seguindo estudos e guidelines publicados, nomeadamente a Fleischner Society (2017), a British Thoracic Society (BTS, 2015) e referências clínicas complementares como Radiopaedia.org.

O tamanho do nódulo é um fator conhecido de risco de malignidade, e categorizar os nódulos ajuda a compreender a aplicabilidade clínica do modelo.

Justificação dos Níveis de Malignidade

Nódulos Pequenos ([3, 6[mm)

- Risco de malignidade ainda muito baixo, consistente com a literatura.
- Para a maioria dos pacientes de baixo risco, não é necessário seguimento ativo.
- Estes nódulos foram incluídos nos modelos, mas são esperados valores de malignidade próximos de 0.

Nódulos Intermédios ([6, 8[mm)

- O risco de malignidade aumenta ligeiramente.
- A recomendação clínica é seguimento com TC em 6–12 meses, particularmente para pacientes de alto risco ou nódulos com morfologia suspeita.
- Para avaliação do modelo, estes nódulos representam casos críticos, sendo importantes para medir a capacidade discriminativa do classificador.

Nódulos Grandes (>= 8 mm)

- Alto risco de malignidade.
- A investigação adicional é necessária, incluindo PET-CT ou biópsia, dependendo das características do paciente.
- Nestas classes, o modelo deve apresentar alta sensibilidade, evitando falsos negativos.

```
# Função para categorizar diâmetro
def categorizar_diametro(d):
    if d < 6:
        return "3-6 mm"
    elif d < 8:
        return "6-8 mm"
    else:
        return ">= 8 mm"

diam_cat = X["diameter"].apply(categorizar_diametro)

def avaliar_por_diametro(melhores_modelos, X, y, groups, diam_cat,
espec_min=0.85):
    resultados = []

    for nome, (pipe, fs_cols) in melhores_modelos.items():
        X_fs = X[fs_cols].copy()
```

```
        # Out-of-fold predictions
        y_all, p_all, diam_all = [], [], []
        for tr, te in cv.split(X_fs, y, groups):
            Xtr, Xte = X_fs.iloc[tr], X_fs.iloc[te]
            ytr, yte = y[tr], y[te]
            dte = diam_cat.iloc[te]

            cal = CalibratedClassifierCV(pipe, cv=3,
method="isotonic")
            cal.fit(Xtr, ytr)
            p = cal.predict_proba(Xte)[:, 1]

            y_all.extend(yte)
            p_all.extend(p)
            diam_all.extend(dte)

        y_all = np.array(y_all)
        p_all = np.array(p_all)
        diam_all = np.array(diam_all)

        # Limiar ótimo global
        fpr, tpr, thr = roc_curve(y_all, p_all)
        spec = 1 - fpr
        mask_thr = spec >= espec_min
        thr_opt = thr[mask_thr][np.argmax(tpr[mask_thr])] if
np.any(mask_thr) else 0.5

        yhat_all = (p_all >= thr_opt).astype(int)

        # Avaliar por cada intervalo de diâmetro
        for cat in ["3-6 mm", "6-8 mm", ">= 8 mm"]:
            mask = diam_all == cat
            if mask.sum() == 0:
                continue
            yt = y_all[mask]
            yp = p_all[mask]
            yh = (yp >= thr_opt).astype(int)

            resultados.append({
                "Modelo": nome,
                "Faixa Diâmetro": cat,
                "N": len(yt),
                "N_malignancy": int(np.sum(yt == 1)),  # <-- nova
coluna
                "AUC ROC": roc_auc_score(yt, yp) if len(np.unique(yt))
> 1 else np.nan,
                "AUC PR": average_precision_score(yt, yp) if
len(np.unique(yt)) > 1 else np.nan,
                "Brier": brier_score_loss(yt, yp),
                "Accuracy": accuracy_score(yt, yh),
```

```
                "Precision": precision_score(yt, yh, zero_division=0),
                "Recall": recall_score(yt, yh, zero_division=0),
                "F1": f1_score(yt, yh, zero_division=0)
            })

    return pd.DataFrame(resultados)

# Executar
res_diametros = avaliar_por_diametro(melhores_modelos, X, y, groups,
diam_cat, espec_min=0.85)
res_diametros
```

	Modelo	Faixa Diâmetro	N	N_malignancy	AUC ROC
0	RandomForest (LASSO)	3-6 mm	370	8	0.787120
1	RandomForest (LASSO)	6-8 mm	439	33	0.720481
2	RandomForest (LASSO)	>= 8 mm	802	454	0.917154
3	LogisticRegression (LASSO)	3-6 mm	370	8	0.772790
4	LogisticRegression (LASSO)	6-8 mm	439	33	0.754142
5	LogisticRegression (LASSO)	>= 8 mm	802	454	0.919278
6	SVM_RBF (Spearman)	3-6 mm	370	8	0.688536
7	SVM_RBF (Spearman)	6-8 mm	439	33	0.746604
8	SVM_RBF (Spearman)	>= 8 mm	802	454	0.920439

	AUC PR	Brier	Accuracy	Precision	Recall	F1
0	0.272532	0.019867	0.975676	0.333333	0.125000	0.181818
1	0.230933	0.064932	0.902050	0.307692	0.242424	0.271186
2	0.932755	0.111042	0.790524	0.746552	0.953744	0.837524
3	0.118122	0.022042	0.959459	0.111111	0.125000	0.117647
4	0.304533	0.061710	0.899772	0.358974	0.424242	0.388889
5	0.936120	0.110865	0.805486	0.767025	0.942731	0.845850
6	0.054874	0.022746	0.964865	0.000000	0.000000	0.000000
7	0.254922	0.063952	0.911162	0.392857	0.333333	0.360656
8	0.934461	0.109264	0.802993	0.756055	0.962555	0.846899

Os resultados obtidos demonstram claramente que o desempenho do modelo varia conforme o tamanho do nódulo, em concordância com a literatura e guidelines clínicas (Fleischner Society, 2017; BTS, 2015; Radiopaedia.org):

Nódulos Pequenos (3–6 mm)

- O número de casos malignos é muito reduzido (N_malignancy = 8).
- O recall é extremamente baixo (0–0.125), refletindo a dificuldade de detectar malignidade em nódulos pequenos, que são clinicamente esperados como benignos.
- A precision também é baixa, mas tem menor relevância dado o pequeno número de positivos.

Conclusão: os modelos corretamente refletem o baixo risco clínico nesta faixa; falsos negativos têm impacto limitado devido à raridade de malignidade.

Nódulos Intermédios (6–8 mm)

- Número de casos malignos aumenta (N_malignancy = 33).
- O recall melhora ligeiramente (0.242–0.424), mostrando alguma capacidade de identificar malignos, mas ainda com dificuldades, o que é esperado dado que esta faixa tem risco de malignidade baixo a moderado.
- A precision permanece baixa (~0.31–0.39), refletindo a sobreposição de características entre nódulos benignos e malignos nesta faixa.

Conclusão: os modelos captam parcialmente a malignidade, mas o risco de falsos negativos ainda é significativo; reforça a necessidade de seguimento clínico adicional para nódulos intermediários.

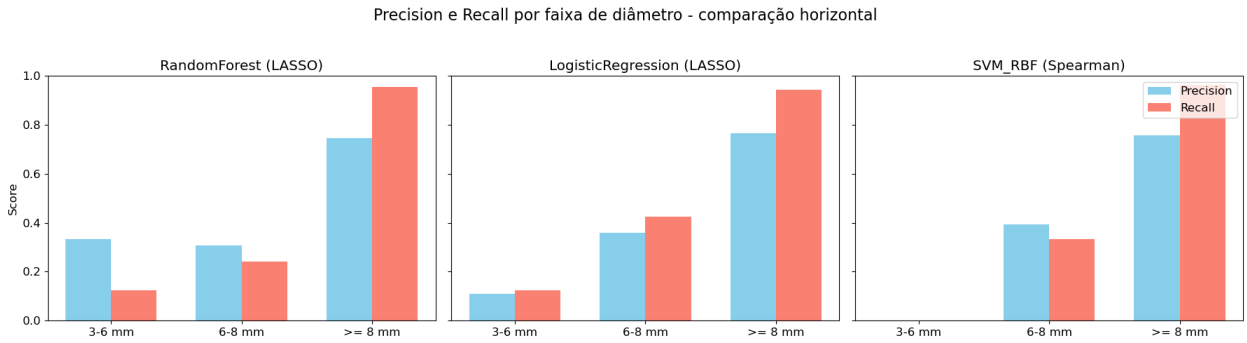
Nódulos Grandes (≥ 8 mm)

- Número de casos malignos elevado (N_malignancy = 454).
- O recall é muito alto (0.942–0.963), demonstrando que os modelos conseguem identificar a grande maioria dos nódulos malignos, que são prioritários do ponto de vista clínico.
- A precision também é elevada (0.746–0.767), mostrando que a maioria das previsões positivas é correta.

Conclusão: nesta faixa, o modelo cumpre plenamente o objetivo clínico, minimizando falsos negativos em nódulos com alto risco de malignidade.

```
# Configurações básicas
plt.rcParams.update({"figure.figsize": (18,5), "font.size": 12})
modelos = res_diametros["Modelo"].unique()
faixas = ["3-6 mm", "6-8 mm", ">= 8 mm"]
width = 0.35  # largura das barras

fig, axes = plt.subplots(1, 3, figsize=(18,5), sharey=True)

for i, modelo in enumerate(modelos):
    df_modelo = res_diametros[res_diametros["Modelo"] == modelo]
    precision = [df_modelo[df_modelo["Faixa Diâmetro"] == f]
```

```
["Precision"].values[0] for f in faixas]
    recall = [df_modelo[df_modelo["Faixa Diâmetro"] == f]
["Recall"].values[0] for f in faixas]

    x = np.arange(len(faixas))
    ax = axes[i]
    ax.bar(x - width/2, precision, width, label="Precision",
color="skyblue")
    ax.bar(x + width/2, recall, width, label="Recall", color="salmon")

    ax.set_xticks(x)
    ax.set_xticklabels(faixas)
    ax.set_ylim(0, 1.0)
    ax.set_ylabel("Score" if i==0 else "")
    ax.set_title(modelo)
    if i == 2:
        ax.legend(loc="upper right")

plt.suptitle("Precision e Recall por faixa de diâmetro - comparação
horizontal", fontsize=16)
plt.tight_layout(rect=[0, 0, 1, 0.95])
plt.show()
```



Como pode ser observado, os modelos refletem bem o padrão clínico esperado, com baixa detecção em nódulos pequenos (baixo risco) e excelente performance em nódulos grandes (alto risco).

Recall é elevado para casos clinicamente relevantes, garantindo que pacientes com alto risco de malignidade são corretamente identificados, enquanto precision confirma a confiabilidade das previsões positivas.

Nódulos intermédios permanecem um desafio, mas o comportamento do modelo está alinhado com o risco moderado esperado, reforçando a necessidade de seguimento clínico direcionado.

Conclusão

A avaliação dos modelos de classificação binária revelou que Random Forest, Logistic Regression e SVM RBF apresentam desempenho global muito semelhante, com pequenas diferenças observadas entre eles. A priorização do recall foi fundamental, dado que o principal objetivo clínico é reduzir o risco de falsos negativos, ou seja, não deixar de identificar pacientes

com nódulos malignos. Os modelos demonstraram alta capacidade discriminativa, refletida em métricas como AUC ROC, AUC PR e distribuição das probabilidades previstas.

Quando analisados por faixa de diâmetro dos nódulos, os modelos apresentaram comportamento coerente com o risco clínico esperado: nódulos grandes (≥ 8 mm) foram identificados com elevado recall e precision, nódulos intermédios (6–8 mm) apresentaram performance moderada, e nódulos pequenos (3–6 mm) mostraram baixo recall, condizente com o baixo risco de malignidade nesta faixa. De forma geral, todos os modelos se mostraram robustos e confiáveis, com o SVM RBF apresentando ligeira vantagem em recall, mas sem diferenças significativas que comprometam a aplicabilidade clínica. Esta avaliação evidencia que os modelos fornecem previsões consistentes e alinhadas com a prática clínica, permitindo suporte à decisão em radiologia com foco na detecção de nódulos malignos.

Pipeline do trabalho

TCs (LIDC-IDRI)
↓
Extração de anotações (PyLIDC)
↓
Features 2D (Pyradiomics)
↓
Features 3D (PyRadiomics)
↓
Feature Selection
↓
Modelos ML (SVM, RF, LR)
↓
Predição ao nível do paciente
↓
Predição pelo diâmetro

Bibliografia

Armato, S. G., McLennan, G., Bidaut, L., McNitt-Gray, M. F., Meyer, C. R., Reeves, A. P., ... & Clarke, L. P. (2011). The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A completed reference database of lung nodules on CT scans. Medical Physics, 38(2), 915–931.

Causey, J. L., Chae, M., Hsieh, S., & Choi, J. (2018). Highly accurate model for prediction of lung nodule malignancy with CT scans. Scientific Reports, 8, 9285.

Larue, R. T. H. M., van Timmeren, J. E., de Jong, E. E. C., van Elmpt, W., Reymen, B., & Lambin, P. (2017). Quantitative radiomics studies for tissue characterization: A review of feature extraction, reproducibility, and applications. Medical Physics, 44(6), e1–e20.

Aerts, H. J. W. L., Velazquez, E. R., Leijenaar, R. T. H., Parmar, C., Grossmann, P., Cavalho, S., ... & Lambin, P. (2014). Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach. Nature Communications, 5, 4006.

Hawkins, S., Wang, H., Liu, Y., Garcia, A., Stringfield, O., Rubin, D., & Ding, K. (2016). Predicting malignant nodules from CT scans using radiomic features. Radiology, 280(2), 883–892.

Zwanenburg, A., Vallières, M., Abdalah, M. A., Aerts, H., Andrearczyk, V., Apte, A., ... & Löck, S. (2020). The Image Biomarker Standardisation Initiative: Standardized quantitative radiomics for high-throughput image-based phenotyping. Radiology, 295(2), 328–338.

Xie, Y., Sun, Y., Jiang, T., Liu, Y., & Zhao, J. (2018). Comparison of different lung nodule malignancy rating schemes for improved prediction performance in CT imaging. [Journal/Conference].

Fleischner Society. (2017). Guidelines for Management of Pulmonary Nodules Detected on CT Images. Radiology, 284(1), 228–243.

British Thoracic Society (BTS). (2015). Guidelines for the Investigation and Management of Pulmonary Nodules. Thorax, 70(Suppl 2), ii1–ii54.

American Cancer Society. Facts & Figures 2019. Technical report, 2019.

Hyun, S. H., Ahn, M. S., Koh, Y. W., & Lee, S. J. (2019). A Machine-Learning Approach Using PET-Based Radiomics to Predict the Histological Subtypes of Lung Cancer. European Journal of Nuclear Medicine and Molecular Imaging, 46(2), 288–298.

Barrientos, A. E. (2015). Automated system for lung nodules classification based on wavelet feature descriptor and support vector machine. BioMedical Engineering OnLine, 14(1), 9.

Radiopaedia.org. Pulmonary Nodules. Disponível em: https://radiopaedia.org