UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS

Classification of major galaxy mergers

using machine learning algorithms trained with N-body simulations

Clasificación de fusiones de galaxias

utilizando algoritmos de aprendizaje automático entrenados con simulaciones de N-cuerpos

Profesor Guía: Dominik Schleicher
Departamento de Astronomía
Facultad de Ciencias Físicas y Matemáticas
Universidad de Concepción

Tesis presentada a la Dirección de Postgrado de la Universidad de Concepción para optar al grado académico de Magíster en Astronomía

Jorge Eduardo Saavedra Bastidas
Octubre de 2024
Concepción, Chile

© 2024, J.Saavedra-Bastidas
Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento

To my mother, whose silver strands I am one of the main contributors to.

ACKNOWLEDGEMENTS

The first person who comes to mind when I think of those I want to thank for my achievements and studies is undoubtedly my mother, *Roxana Esmeralda Bastidas Gatica*. She has been a pillar of support throughout my life, striving to provide for our small family even in the face of adversity. To my close family, who, although they may not fully understand what I do, have been a source of unconditional help and affection, for which I am deeply grateful. To my advisor, *Dr. Dominik Schleicher*, for being a mentor both in and out of the classroom, for presenting me with opportunities that have challenged and nurtured my career, for supporting me more times than I can count, and for always striving to help his students reach their highest potential. To my partner, *Vanessa Nayem Molina Espinoza*, who had witnessed everything behind the scenes of a research project and cared for me when the weight of responsibilities became too heavy to bear alone. To my friends, with whom I face each semester of this journey together, who are a constant source of inspiration for me. To *Dr. Ezequiel Treister*, for his contributions to the development of this project, his guidance in the use of the implemented software in this study, and his efforts in establishing connections with researchers in the same field. To *Dr. Guillermo Cabrera-Vives*, for introducing me to a new area in my field, in which I still have much to learn. To *Dr. George Privon*, for answering my countless questions, always with great patience, no matter how basic they were. To *M.Sc. Daniel Gaete*, for his help in creating the observational dataset. Part of the results presented here were computed at Instituto de Astronomía, Pontificia Universidad Católica. I gratefully acknowledge support from the ANID BASAL project FB21003, as well as from Fondecyt Regular (project code 1201280) and ANID QUIMAL220002.

Resumen

Las fusiones de galaxias son eventos significativos en astronomía, impulsando la transformación morfológica de galaxias espirales a elípticas y alterando la mecánica interna del gas, lo que incrementa la formación estelar, potencia la actividad nuclear y contribuye a la formación y evolución de los agujeros negros supermasivos. Los métodos tradicionales de detección de fusiones de galaxias carecen de la efectividad y eficiencia necesarias para manejar grandes conjuntos de datos. En este estudio, realizamos una comparación sistemática de diferentes modelos de aprendizaje automático como clasificadores de fusiones mayores de galaxias y sus etapas de fusión, basándonos únicamente en información morfológica. Probamos clasificadores basados en ensambles como Random Forest (RF) y Extreme Gradient Boosting (XGBoost) y arquitecturas de deep learning como Convolutional Neural Networks (CNNs). Proponemos el uso de imágenes extraídas de simulaciones de N-cuerpos diseñadas para replicar las características morfológicas de las interacciones entre galaxias como datos de entrenamiento para los algoritmos de clasificación. Evaluamos estos modelos en tres niveles de realismo observacional: galaxias idealizadas extraídas de nuestras simulaciones, galaxias convolucionadas con una función de dispersión puntual (PSF) Gaussiana, y galaxias convolucionadas con la PSF Gaussiana y complementadas con ruido de fondo real. Encontramos que los modelos con mejor rendimiento en el conjunto de pruebas sintético con mayor realismo observacional son aquellos entrenados en datos de la misma distribución. Las CNNs logran un área bajo la curva ROC de 95.2%, mientras que XGBoost y RF obtuvieron 93.5% y 93.0%, respectivamente. A pesar de quedar en segundo lugar, XGBoost muestra mayor estabilidad que las CNNs al predecir fusiones de galaxias proporcionadas por diferentes distribuciones de datos. Probamos XGBoost en una muestra de galaxias masivas y de bajo desplazamiento al rojo ($z \leq 0.15$) del Dark Energy Camera Legacy Survey - Galaxy Zoo Data Release 5, demostrando su capacidad para diferenciar pares de galaxias de manera efectiva. Concluimos que las características morfológicas son una base sólida para entrenar un clasificador de aprendizaje automático para fusiones de galaxias; sin embargo, las diferencias entre galaxias aisladas y post-fusiones recientes requieren de una física más detallada para caracterizar completamente ambas etapas.

Keywords – Métodos: numéricos, Métodos: estadistica, Galaxias: interacciones

Abstract

Galaxy mergers are significant events in astronomy, driving the morphological transformation from spiral to elliptical galaxies and disrupting internal gas mechanics, increasing star formation, enhancing nuclear activity, and contributing to the formation and evolution of supermassive black holes. Traditional detection methods for galaxy mergers lack the effectiveness and efficiency required to handle large datasets. In this study, we perform a systematic comparison of different machine learning models as classifiers for major galaxy mergers and their merger stages, relying solely on morphological information. We test ensemble-based classifiers like Random Forest (RF) and Extreme Gradient Boosting (XGboost) and deep learning architectures like Convolutional Neural Networks (CNNs). We propose the implementation of images extracted from N-body simulations designed to replicate the morphological features of galaxy-galaxy interactions as training data for the classification algorithms. We evaluate the performance of these models across three levels of observational realism: highly idealized galaxies extracted from our simulations, galaxies convoluted with a Gaussian point spread function (PSF), and galaxies convoluted with the Gaussian PSF and complemented with real background noise. We found that models with the best performance on the highest observational realism synthetic test set are those trained on data from the same distribution. CNNs achieved an average area under the receiver operating characteristic curve of 95.2%, while XGBoost and RF obtained 93.5% and 93.0%, respectively. Despite being in second place, XGBoost shows greater stability than CNNs when predicting mergers from galaxies provided by different data distributions. We test XGBoost on a sample of massive, low-redshift ($z \leq 0.15$) galaxies from the Dark Energy Camera Legacy Survey - Galaxy Zoo Data Release 5, showing the ability to differentiate galaxy pairs effectively. We conclude that morphological features are a solid base for training a machine learning classifier for galaxy mergers, however, the differences between isolated galaxies and recent post-mergers require more detailed physics to completely characterize both stages.

Keywords – Methods: numerical, Methods: statistical, Galaxies: interactions

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Context

Nowadays, the initial conditions that shaped the evolution of the universe from its earliest moments have been determined. Measurements of anisotropies in the cosmic microwave background (CMB; e.g., Spergel et al., 2003, 2007; Planck Collaboration et al., 2016) and observations of high-redshift supernovae (e.g., Riess et al., 1998; Perlmutter et al., 1999) have established a standard picture of how the universe evolved, with an unknown form of energy dominating the energy density that drives the accelerated cosmic expansion. In this scenario, most of the mass consists of non-baryonic matter, with a 71.4% of dark energy, a 24% of dark matter, and only a 4.6% of baryonic matter. On small scales, structure formation occurs through gravitational instability, where objects grow hierarchically in the currently favored Λ cold dark matter (ΛCDM) paradigm. Essentially, baryonic matter falls into dark matter potential wells, undergoes shock heating, and then cools radiatively, leading to the formation of stars and galaxies in a bottom-up progression (White & Rees, 1978). This hierarchical ensemble of the universe leads directly to close interactions between galaxies, where strong tides cause the inevitable merging of galaxy pairs through the transfer of energy and momentum from their motion to their internal degrees of freedom (Toomre, 1977; White, 1978; Barnes, 1988). Thus, the rate of occurrence of galaxy mergers is a direct consequence of the evolution of the universe, providing observable insights into the history of galaxy mass assembly (for a review, see Conselice, 2014).

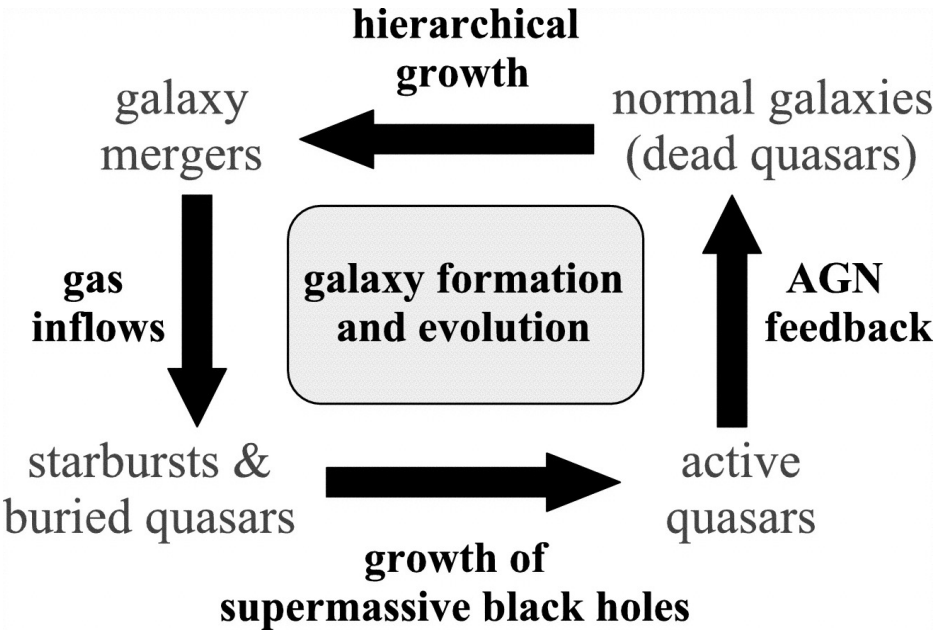

Figure 1.1.1: Schematic representation of the "cosmic cycle" for galaxy formation and evolution regulated by black hole growth in mergers. Figure from Hopkins et al. (2006a).

Major galaxy mergers (i.e., mergers where the mass ratio of the galaxies $\mu \geq 1/4$) drive the morphological transformation of galaxies, turning spiral disc galaxies into ellipticals and creating structures such as bridges, tails, and rings in the process (Toomre & Toomre, 1972; Lynds & Toomre, 1976; Theys & Spiegel, 1977). Furthermore, these mergers disrupt the internal gas dynamics of galaxies by compressing the gas, leading to nuclear and global starbursts (Mihos & Hernquist, 1994, 1996; Hopkins et al., 2006b). Their capacity to strip gas of sufficient angular momentum and induce high accretion rates toward the central object within colliding galaxies has made them a promising mechanism for fueling Active Galactic Nuclei (AGN) and a potential pathway for the formation and evolution of supermassive black holes (SMBHs; e.g., Kauffmann & Haehnelt, 2000; Cox et al., 2008; Hopkins et al., 2008; Treister et al., 2012; Kormendy & Ho, 2013). The connection between these processes has led to the conceptualization of a "cosmic cycle" in galaxy formation and evolution (Fig. 1.1.1). As displayed by Hopkins et al. (2006a), starbursts, quasars, and the concurrent formation of spheroids and supermassive black holes represent interlinked phases in the life cycles of galaxies. Mergers between gas-rich galaxies drive nuclear inflows of gas, triggering starbursts and promoting the growth of supermassive black holes. During most of this phase, quasar activity is obscured, but once a black hole dominates the

energetics of the central region, feedback mechanisms expel gas and dust, making the black hole briefly visible as a bright quasar. Eventually, as the gas is further heated and expelled, quasar activity ceases, and the merger remnant stabilizes into a normal galaxy with a spheroid and a supermassive black hole.

1.2 The AGN-merger connection

The connection between nuclear activity and galaxy mergers can be investigated observationally by comparing the morphologies of AGN hosts and comparable inactive galaxies. If AGN are more likely to live in merging systems compared to matched control galaxies this could imply a causal link between the two. It was originally proposed that AGN were initially obscured during the merger process, only to be revealed later as optical quasars (Heckman et al., 1986; Sanders et al., 1988). Treister et al. (2012) found a strong correlation between AGN luminosity and galaxy mergers, increasing from 4% of low-luminosity AGN to $\sim$ 90% of high-luminosity AGN in mergers. Ellison et al. (2013) found four times as many AGN in post-merger galaxies in the Sloan Digital Sky Service (SDSS; York et al., 2000) at low redshifts, with an increasing incidence of AGN activity and luminosity as the separation between galaxies decreased. These results were subsequently reinforced by Satyapal et al. (2014), who found a $\sim$ 10 times enhancement of AGN in mergers in the same sample by selecting SDSS AGN using infrared (IR) colors. Both Kocevski et al. (2015) and Weston et al. (2017) identified a correlation between the fractions of AGN in mergers and the degree of obscuration, suggesting that mergers trigger black hole growth while also concealing the AGN behind thick columns of gas and dust. In contrast, several studies have compared AGN hosts with matched quiescent galaxies and found no significant differences between them in either disc fractions, asymmetry distributions, or visual classification (e.g., Grogin et al., 2005; Cisternas et al., 2011; Villforth et al., 2014, 2017; Hewlett et al., 2017).

AGN properties, such as luminosity, vary significantly on timescales ranging from days to thousands of years (e.g., Sartori et al., 2018), whereas galaxy morphologies evolve over hundreds of millions of years. Since luminosities are measured in a single snapshot, relating instantaneous accretion rates to galaxy morphologies may be misleading. If there is a high degree of stochasticity, very large samples would

be required to discern underlying relationships, which is challenging due to the rarity of major mergers (around 2-4% of galaxies undergo major mergers per Gyr, Rodriguez-Gomez et al., 2015). Additionally, AGN can be detected using various indicators across the electromagnetic spectrum, each with its own biases: soft X-ray or optical selection methods tend to under-identify Compton-thick AGN, while mergers likely increase line-of-sight hydrogen column densities, making such AGN more commonly Compton-thick (Ricci et al., 2017). Relying solely on soft X-ray or optical selection may therefore omit some AGN in mergers, potentially biasing results against finding an AGN-merger connection. In contrast, mid-infrared (MIR) colors can robustly identify AGN despite extinction, but the probability of AGN detection increases with higher dust fractions. Dust is produced during star formation, which is often enhanced by mergers, potentially leading to a bias in favor of identifying a merger-AGN connection through MIR detection.

1.3 Uncertainties in the rate of galaxy mergers

The significance of galaxy mergers remains a subject of debate within the astronomical community. Specifically, there is contention regarding how the merger fraction and merger rate[1] scale with cosmic time, as observations and simulations often produce conflicting results. We now know that the rate at which mergers occur evolves strongly out to $z \sim 1.5$, as seen by many observational studies and cosmological simulations (e.g., Kartaltepe et al., 2007; Lotz et al., 2011; Rodriguez-Gomez et al., 2015; Mantha et al., 2018). However, many uncertainties in the observations and tensions with simulation results remain. Duncan et al. (2019) conducted a comparative analysis of merger rates up to $z \sim 6$ observed in the Cosmic Assembly Near-infrared Dark Energy Legacy Survey (CANDELS; Grogin et al. 2011; Koekemoer et al. 2011) against prior research (see Fig 1.3.1). They discovered that for galaxies with masses $\log_{10}(M_*/M_\odot) > 10.3$, their merger rate was in agreement with findings from the Illustris simulation (Genel et al., 2014; Vogelsberger et al., 2014) by Rodriguez-Gomez et al. (2015) up to $z \sim 6$, and with

[1]Merger fraction and merger rate are terms often used interchangeably in the literature because they share similar characteristics, yet they represent different quantities. The merger fraction, defined as the number of galaxy mergers relative to the rest of the population, represents an observable quantity that can be calculated from observations of a significant sample of galaxy mergers at a given redshift. In contrast, the merger rate represents the frequency at which galaxy mergers occur as a function of various parameters over a comoving timescale, which cannot be measured directly from observations.

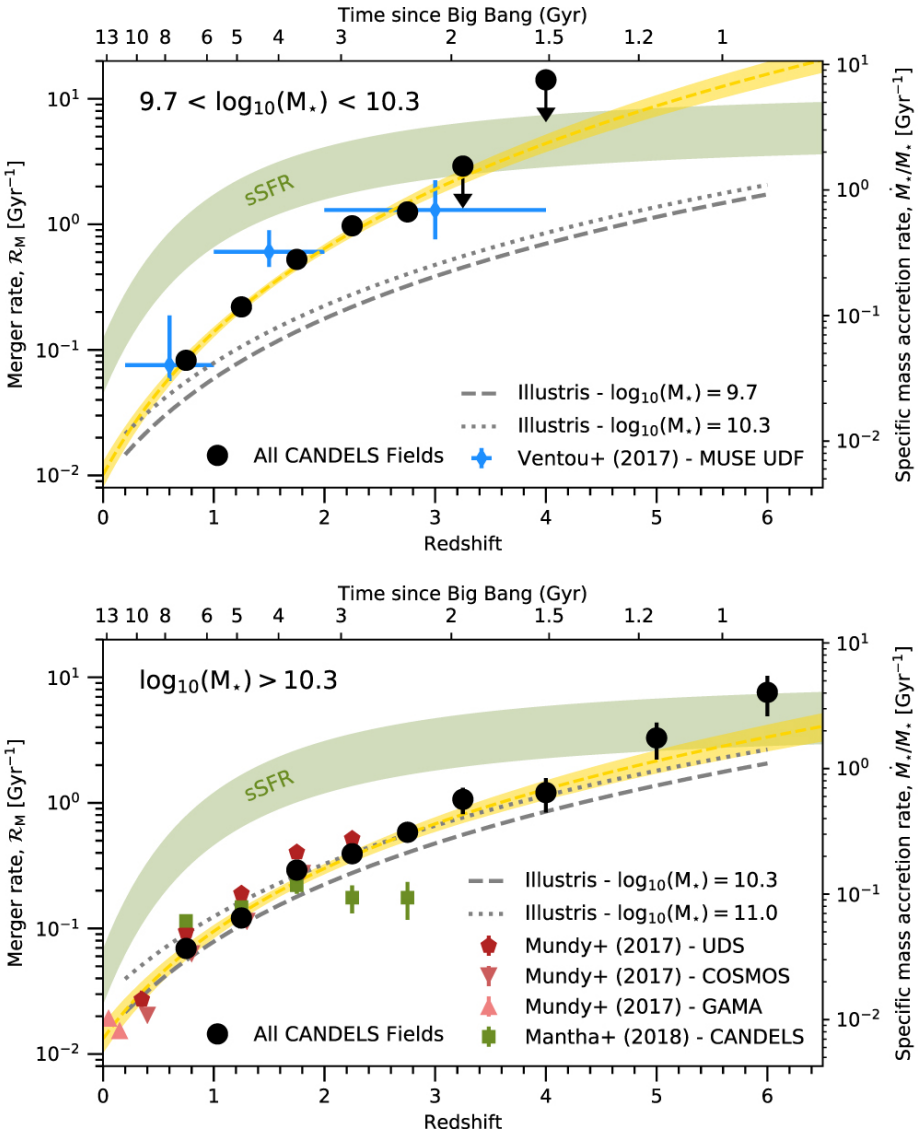

Figure 1.3.1: Estimated major merger rate per galaxy as a function of redshift for galaxies with stellar mass $9.7 < \log_{10}(M_*/M_\odot) < 10.3$ (top) and $\log_{10}(M_*/M_\odot) > 10.3$ (bottom). Also shown are the merger rates based on the close-pair statistics of Mundy et al. (2017), Ventou et al. (2017), and Mantha et al. (2018). The golden line and shaded region in each figure show the best-fitting power-law model found by Duncan et al. (2019). The right-hand scale illustrates the inferred specific mass accretion rate through major mergers based on the observed merger rate. For reference the observed specific star formation rates for similar mass galaxies as a function of redshift is shown (green shaded region). Figure from Duncan et al. (2019).

the results of Mundy et al. (2017) up to $z \sim 2$. However, beyond $z \sim 2$, there is a discrepancy with the merger rate reported by Mantha et al. (2018), which exhibits a declining trend. Regarding galaxies with masses $9.7 < \log_{10}(M_*/M_\odot) < 10.3$, Duncan et al. (2019) observed that their increasing merger rate aligns with Ventou et al. (2017) up to $z \sim 3$ with a short margin of error, yet contradicts the shallower rate predicted by the Illustris simulation, indicating a discrepancy between simulated and observed merger rates. This tension between observations and simulations also occurs when comparing the major merger fraction for galaxies with $\log_{10}(M_*/M_\odot) > 10.3$, as the values obtained from the EAGLE simulation (Schaye et al., 2015) calculated by Qu et al. (2017) show a higher estimate than those observed in the galaxy catalogs. Duncan et al. (2019) further highlighted substantial uncertainties in their comoving merger rates at $z > 4$.

1.4 Previous works on merger classification

Both in Sec. 1.2 and Sec. 1.3, an important aspect has not been fully considered: the impact of false positive and false negative merger detections on the interpretation of the results. Simulations indicate that approximately 2% of intermediate to massive local galaxies experience a major merger per Gyr (Rodriguez-Gomez et al., 2015). Therefore, even if only a small fraction of non-mergers are misclassified as mergers, the resulting 'merger' sample could be predominantly composed of non-mergers. Moreover, many studies report significantly higher fractions of mergers than those suggested by the simulations (e.g., Cisternas et al., 2011; Villforth et al., 2014, 2017; Ellison et al., 2019), indicating that this is indeed a substantial concern.

1.4.1 Traditional detection methods

The main traditional methods for detecting galaxy-galaxy interactions in astronomy can be summarized in two categories: close-pair detection and morphological perturbation detection.

Close-pair detection usually involve the identification of pairs of galaxies that fulfill a maximum separation criterion (both in redshift and angular separation) using either spectroscopic redshifts (e.g., Lin et al., 2004; Ellison et al., 2013; Satyapal et al., 2014; Ventou et al., 2017; Shah et al., 2020), or a sophisticated analysis on

photometric redshifts (e.g., Lin et al., 2004; Mundy et al., 2017; Mantha et al., 2018; Duncan et al., 2019), such that their orbits will decay with time resulting in a merger event. Spectroscopic pair identification is effective even at high-redshift, but is often incomplete due to the *fibre-collision*[2] and space sampling (Lin et al., 2007; Robotham et al., 2014).

Another approach consists of identifying advanced mergers through morphological disturbances, such as double nuclei, tidal tails, or other asymmetries. This procedure has been performed by visual inspection for subsets of different galaxy surveys such as the SDSS (Darg et al., 2010a,b), CANDELS (Kartaltepe et al., 2015), and Dark Energy Camera Legacy Survey (DECaLS; Walmsley et al., 2022). Due to the volume of data present in these surveys, projects such as Galaxy Zoo[3] have opted to use multiple volunteers to manually perform the classifications. However, this method can be both subjective and time-consuming, especially for large surveys (e.g., Lambrides et al., 2021). Structural parameters provide a quantitative indication of morphology without the subjectivity of human visual inspection. These parameters can take the form of parametric indicators, such as the change in light intensity with radial distance from the galaxy center described by the Sersic profile (Peng et al., 2002), or non-parametric indicators such as CAS (Concentration, Asymmetry, Smoothness), G, M_{20}, and the MID (Multimode, Intensity, Deviation) statistic (Conselice, 2003; Lotz et al., 2004; Freeman et al., 2013). The main obstacle with structural parameters is defining empirical thresholds that separate merges from non-mergers, which are highly sensitive to resolution and surface bright limits (e.g., Ji et al., 2014; Bottrell et al., 2019b; Rose et al., 2023).

1.4.2 Machine learning methods

Traditional detection methods, while effective, can be unfeasible for gathering large data samples or may introduce bias, compromising the validity of statistical conclusions drawn from extensive datasets. In recent years, there has been a proliferation of machine learning studies focused on galaxy image analysis and

[2]Fibre collision is an instrumental problem that can be summarized as the inability of the spectroscopic camera to resolve the space between two galactic nuclei, due to the positions of the fibres used to collect the light. This results in the inability to obtain spectra for all close pairs, leading to potential biases in the data.

[3]Galaxy Zoo: https://zoo4.galaxyzoo.org/

classification problems (e.g., Dieleman et al., 2015; Huertas-Company et al., 2015; Domínguez Sánchez et al., 2018; Cabrera-Vives et al., 2018; Pérez-Carrasco et al., 2019; Cheng et al., 2020; Tarsitano et al., 2022; Walmsley et al., 2022; Medina-Rosales et al., 2024). Machine learning (ML) is a branch of artificial intelligence that develops algorithms to identify patterns and make predictions based on data without an underlying physical model. However, training these algorithms requires labeled datasets where each example includes input data and the corresponding output label (i.e., supervised learning).

Many studies use human visual classification to construct these training sets. For instance, Huertas-Company et al. (2015) cataloged the morphologies of ∼50000 galaxies across five CANDELS fields (GOODS-N, GOODS-S, UDS, EGS, and COSMOS) using a convolutional neural network (CNN) trained on GOODS-S, which had detailed visual classifications for ∼8000 objects (Kartaltepe et al., 2015). Their model assigned probabilities to each galaxy for categories like spheroid, disc, irregularity, compactness, point source, and unclassifiable. Similarly, Pérez-Carrasco et al. (2019) also applied a CNN architecture trained on the GOODS-S dataset and using the same taxonomy, incorporating inception blocks (Szegedy et al., 2015), and fine-tuned the network to predict classifications for 8412 galaxies from the Cluster Lensing And Supernova survey with Hubble (CLASH; Postman et al., 2012). Domínguez Sánchez et al. (2018) compiled a catalog using CNNs to classify morphologies under the T-type Hubble sequence scheme for ∼670000 galaxies from the SDSS-DR7 Main Sample Galaxy (Abazajian et al., 2009), leveraging classifications from Nair & Abraham (2010) and Galaxy Zoo Data Release 2 (GZ2; Willett et al., 2013) for the training. Cheng et al. (2020) carried out a systematic comparison between several traditional ML methods for galaxy binary classification in the Dark Energy Survey (DES; Drlica-Wagner et al., 2018) combined with visual classification from Galaxy Zoo Data Release 1 (GZ1; Lintott et al., 2008).

These studies employ different methodologies to address uncertainties in the visual classifications of galaxies. The use of custom-defined metrics to quantify uncertainties in classifications (Cabrera-Vives et al., 2018) or the analysis of images through deep learning models (Medina-Rosales et al., 2024) aims to correct large surveys of observations to achieve cleaner training sets. These variations in methodology make direct comparisons of results difficult and do not eliminate

human bias but rather mitigate its impact on these classifiers. It is impossible to access the complete ground truth from observations.

Simulations, on the other hand, provide a full history from the start to the end of the phenomenon under study, allowing us to build datasets with prior knowledge of its parameters. For example, Bottrell et al. (2019a) created a synthetic imaging dataset from hydrodynamic simulations using FIRE-2 models (Hopkins et al., 2018) for galaxy merger classification using a CNN architecture. These models include a treatment of radiative cooling and heating from free-free, photo-ionization and recombination, Compton, photoelectric, dust-collisional, cosmic rays, molecular, metal-line, and fine structure processes. They employed GIZMO (Hopkins, 2015, 2017) to simulate the systems, with orbital parameters permuted across four galactic models (see Moreno et al., 2019), resulting in 27 unique interactions, and a dataset of ∼9000 unique images. Their study also analyzed the impact of observational realism in the classifier by introducing varying levels of synthetic instrumental noise into the images before training the model. They concluded that it is possible to avoid high computational costs associated with the physics of simulations, such as radiative transfer, without compromising classifier performance during inference. Ferreira et al. (2020) built a dataset from images extracted from the cosmological simulation IllustrisTNG (Nelson et al., 2019) to train a CNN capable of identifying major galaxy mergers and their merger stages across all CANDELS fields using near-infrared imaging. They restricted their exploration to the merger trees in the TNG300-1 simulation of massive galaxies (i.e., those with stellar masses $M_* > 10^{10}$ $M_\odot$) up to a redshift of $z \leq 3$, finding ∼30000 suitable candidates. Instrumental features present in the CANDELS survey were added using tools available in the IllustrisTNG API, such as photoionization and stellar population synthesis codes (for details, refer to Nelson et al., 2018). Using the predictions of the classifier of the CANDELS survey, they computed the galaxy merger rates in the catalog up to $z \leq 3$, finding a close agreement with results computed using close pair statistics. Rose et al. (2023) investigated the use of Random Forest (RF) to classify merging galaxies obtained from IllustrisTNG at high redshift ($z \leq 4$). Similar to Ferreira et al. (2020), the images were modified using the Illustris TNG API to resemble observations, in this case galaxies expected from the James Webb Space Telescope (JWST) Cosmic Evolution Early Release Science Survey (CEERS), obtaining ∼70000

suitable candidates. They employed the *galapagos-2*[4] and *statmorph*[5] programs to compute 12 structural parameters and residual images for each galaxy, which were then used to train the RF. They found that asymmetry features tend to be most important for merger classifications at low redshift, while bulge and clump features tend to be more important at higher redshifts.

Galaxies are not perfectly reconstructect in simulations, but these previous studies have shown that by properly imcorporating observational realism, simulation-trained algorithms can be applyied to real observations with high fidelity.

1.5 This project

In this investigation, we explore the performance of different ML algorithms optimized through an automated pipeline for classifying major galaxy mergers, relying solely on morphological information derived from imaging data.

In a pioneering study on the impact of mergers, Toomre (1977) proposed a merging sequence of 11 peculiar galaxies at various stages, now known as the *Toomre sequence*. Thirty years later, Rossa et al. (2007) revisited the properties of the nuclei of these galaxies using high spatial resolution observations in the near-infrared obtained with NICMOS on the Hubble Space Telescope (HST). They concluded that the merger stage significantly alters the properties of colliding galaxies, showing that the nuclei become more luminous as a function of the merger stage—something that could be explained by an increase in stellar density, a decrease in the average stellar age, or both. This finding, along with the theoretical expectation that dust obscuration of AGN occurs during the early stages of a merger and is only revealed later, motivates us to train a classifier capable of distinguishing the specific stage of the collision event, rather than limiting our study to a binary classification between mergers and non-mergers.

To avoid the inclusion of human bias during the training of ML models, we extract images from snapshots of self-consistent N-body simulations of binary interactions between massive galaxies, differentiating their stages based on the kinematics of particles within the two colliding bodies. To construct a comprehensive library of encounters, we use the IDENTIKIT (Barnes & Hibbard, 2009; Barnes, 2011a)

[4]Galapagos-2: https://github.com/MegaMorph/galapagos
[5]Statmorph: https://statmorph.readthedocs.io/en/latest/

methodology and the ZENO (Barnes, 2011b) programming framework. We focus on this type of simulation because we aim to study a classifier that relies solely on the morphological features produced during the collision event, excluding effects such as radiative transfer, photoionization, or stellar population synthesis. Additionally, the computational cost of exploring the parameter space in the initial conditions of each interaction would be significantly higher if a detailed hydrodynamic treatment were included.

Selecting the most suitable ML model is a computationally expensive task, often influenced by the nature and dimensionality of the data. To ensure we identify the best algorithm from a range of possibilities, we execute an automated machine learning (AutoML) pipeline on the datasets retrieved from the simulations. We also evaluate the classifiers obtained using different noise levels in the simulated images and compare the results of our best classifier on a sample of galaxy mergers obtained from the DECaLS Galaxy Zoo Data Release 5 (GZ5; Walmsley et al., 2022).

The manuscript is organized as follows. In Chapter 2, we describe the numerical methods and the codes used to perform the simulations in this study. Chapter 3 presents the simulated dataset and the observational sample retrieved from GZ5. A description of the AutoML pipeline and the experiments performed are provided in Chapter 4. The results are presented in Chapter 5. Finally, we summarize our work and provide the main conclusions and discussion in Chapter 6.

Chapter 2

Simulations

The diverse morphological and kinematic characteristics of interacting disc galaxies can be explained as the result of the strong tidal forces they experience due to their close encounters, leading to the inevitable merger of the two galaxies. During this process, galaxies become scrambled as they merge, but the stars and dark matter, which constitute most of their mass, evolve without colliding. The evolution of a collisionless, self-gravitating system is described by two pairs of equations: the Vlasov equation,

$$\frac{\partial f}{\partial t} + \boldsymbol{v} \cdot \frac{\partial f}{\partial \boldsymbol{r}} - \nabla\Phi \cdot \frac{\partial f}{\partial \boldsymbol{v}} = 0, \tag{2.0.1}$$

where $f(\boldsymbol{r}, \boldsymbol{v}, t)$ is the one-particle distribution function and $\Phi(\boldsymbol{r}, t)$ is the gravitational potential, and the Poisson equation,

$$\nabla^2\Phi = 4\pi G\rho = 4\pi G \int f(\boldsymbol{r}, \boldsymbol{v}, t) d\boldsymbol{v}. \tag{2.0.2}$$

N-body simulations often use a Monte Carlo method to solve these equations. The distribution function is represented by a collection of N particles:

$$f(\boldsymbol{r}, \boldsymbol{v}, t) \approx \sum_{i=1}^{N} m_i \delta^3(\boldsymbol{r} - \boldsymbol{r}_i(t)) \delta^3(\boldsymbol{v} - \boldsymbol{v}_i(t)) \tag{2.0.3}$$

where m_i, $\boldsymbol{r}_i$, and $\boldsymbol{v}_i$ are the mass, position, and velocity of particle i, which follows Newton's equations of motion. Over time, particles move along characteristics of Eq. [2.0.1]; at each instant, their positions provide the density needed for Eq. [2.0.2] (Vlasov, 1961; Klimontovich, 1967).

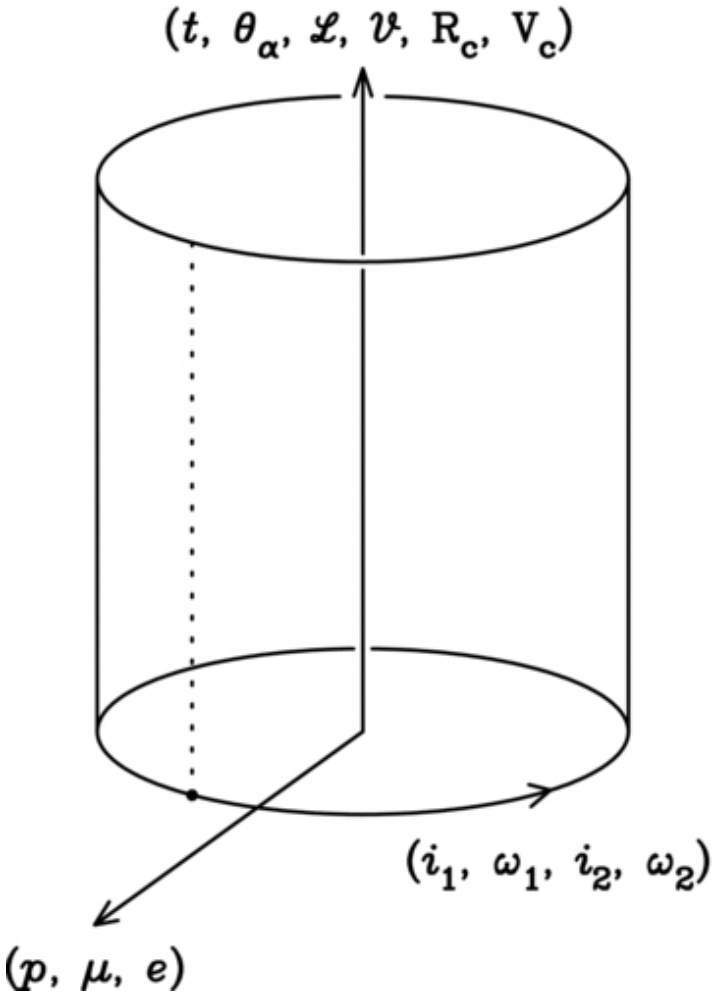

Figure 2.0.1: An abstract representation of the sixteen-dimensional parameter space of galaxy interactions. These parameters can be grouped into three classes: The radial coordinate represents the initial orbit of the galaxies; the azimuthal coordinate represents the disc orientations; and the vertical coordinate represents the viewing parameters chosen after a simulation is run. A conventional N-body simulation explores the parameter subspace represented by the dotted line, while a single IDENTIKIT simulation explores the entire cylindrical surface. Figure from Barnes (2011a).

Nevertheless, the numerical integration of the above equations in traditional N-body simulations requires the definition of several initial conditions to model a single system (see Fig. 2.0.1), such as the mass ratio of the galaxies (μ), the disc orientations[1] ((i_1, ω_1) and (i_2, ω_2)), the eccentricity of the orbit (e), and the initial pericentric separation (p). In addition to this, nine parameters are needed to compare simulations to observations: the length scale ($\mathcal{L}$), the velocity scale ($\mathcal{V}$), the center of mass position on the plane of the sky ($\boldsymbol{R}_c$), the radial velocity (V_m), the viewing angles ($\Theta_x, \Theta_y, \Theta_z$), and the time of viewing (t). Without varying the internal structure of the galaxies involved, 16 free parameters are present (Barnes & Hibbard, 2009). This plethora of parameters has long posed a challenge for systematic surveys of galactic collisions (e.g., Toomre & Toomre, 1972; Farouki & Shapiro, 1982; Wallin & Stuart, 1992; Howard et al., 1993; Naab & Burkert, 2003).

Exploring the parameter space in the initial conditions of these simulations is computationally expensive, particularly given the large amounts of data required by ML algorithms to avoid overfitting. In this work, we utilize the ZENO[2] programming framework (Barnes, 2011b) and the IDENTIKIT[3] methodology (Barnes & Hibbard, 2009; Barnes, 2011a) to create a comprehensive library of major merger encounters. IDENTIKIT, a type of N-body simulation, enables rapid exploration of the parameter space of galactic encounters, facilitating the matching of simulated encounters with real observations (e.g., Privon et al., 2013; Mortazavi et al., 2016, 2018; Pearson et al., 2018). This approach significantly reduces the computational time required to generate a sufficiently large and diverse dataset.

The procedure can be summarized as follows: Four galactic models are constructed, varying the total mass, using hybrid test-particle discs embedded in live N-body dark matter halos. These configurations are positioned at different distances with a fixed orbital eccentricity and are allowed to evolve until the system is fully merged. The simulations are then visualized using the IDENTIKIT graphical interface.

[1] These angles follow the definitions presented in Toomre & Toomre (1972). Inclination (i) is the angle between the disc's angular momentum vector and the z-axis, while azimuth (ω) is the angle between the disc's angular momentum vector, projected onto the (x, y) plane, and the y-axis.

[2] ZENO: https://home.ifa.hawaii.edu/users/barnes/zeno/

[3] IDENTIKIT: https://home.ifa.hawaii.edu/users/barnes/research/identikit/

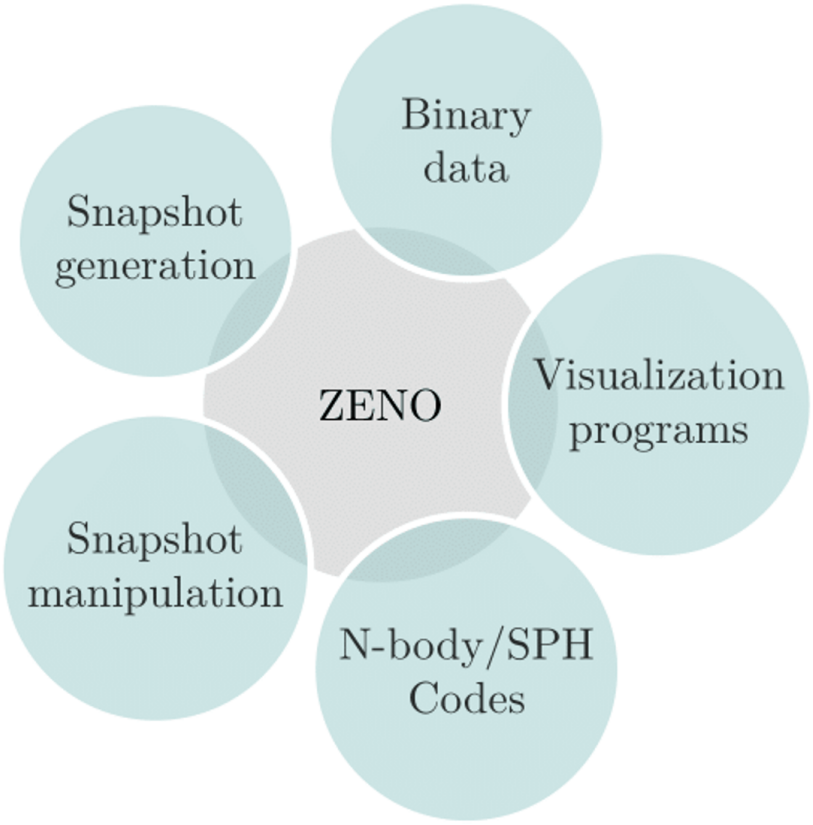

Figure 2.1.1: Different tasks and programs integrated in the ZENO framework for N-body and SPH particle simulations.

In the subsequent sections, we describe the ZENO programming framework, the details of the IDENTIKIT simulations employed in this project, the construction of the galactic models, and the development of the full library of galactic encounters.

2.1 The ZENO framework

The generation of models and integration of systems in this project is carried out through the ZENO framework. The ZENO software package integrates N-body and Smoothed-Particle Hydrodynamics (SPH) simulation codes with various programs for generating initial conditions and analyzing numerical simulations. Written in C, the ZENO system is portable across Mac, Linux, and Unix platforms. The framework consists of programs invoked and controlled directly from the UNIX command line. The system utilizes a variety of techniques to provide a flexible and powerful working environment (for a schematic representation, see

Fig. 2.1.1):

- A structured language for binary data files identifies data elements by name and type. Input routines can access binary files, retrieving only the data required for the computation at hand. Explicit typing facilitates data exchange between machines with different data formats.
- On-the-fly code generation. Many ZENO programs permit the user to specify C-language expressions as input parameters. These expressions are used to automatically generate executables to perform specific tasks.
- Dynamically extensible particle representation. The data associated with a particle depends on the specific application. Instead of trying to anticipate all possibilities, the software determines the particle structure at run-time.
- Snapshot generation routines create particle distributions with various properties. Systems with user-specified density profiles can be realized in collisionless or gaseous form; multiple spherical and disc components may be set up in mutual equilibrium.
- Simulation codes include both pure N-body and combined N-body/SPH programs. Pure N-body codes are available in both uniprocessor and parallel versions. SPH codes offer a wide range of options for gas physics, including isothermal, adiabatic, and radiating models.

Thanks to their modular design and their use of a common and general data format, these programs combine in a tremendous variety of ways. Generation and analysis pipelines can be prototyped in minutes, formalized using the UNIX make utility, and tuned for efficient processing of large data volumes.

2.2 The IDENTIKIT methodology

Created by Barnes & Hibbard (2009), IDENTIKIT-1 combines *test particles*[4] and *self-consistent*[5] techniques for modeling the initial conditions of major galaxy mergers. Further improvements in IDENTIKIT-2 (Barnes, 2011a) included a new method for computing the similarity between model and observational data, incorporating a score based on the number of disc test particles residing in small boxes defined by the user.

For the creation of a merged system, each galaxy is modeled as an initially spherical configuration of massive particles in equilibrium with a cumulative mass profile,

$$m(r) = m_b(r) + m_d(r) + m_h(r), \tag{2.2.1}$$

where $m_b(r)$, $m_d(r)$ and $m_h(r)$ corresponds to the mass profile of the bulge, disc and halo, respectively. The effect of Plummer softening (i.e., correcting for the finite resolution of the N-body force calculation) is taken into account by introducing a quasi-empirical transformation of the total mass profile (for further details on the derivation of this transformation, see Barnes, 2012; Appendix A),

$$\bar{m}(r) = [1 + (2/3)^{(\kappa/\alpha)}(\bar{\epsilon}/r)^{\kappa}]^{(\alpha/\kappa)} m(r), \tag{2.2.2}$$

where α is the logarithmic derivative of the density profile as $r \to 0$, the parameter $\bar{\epsilon}$ its comparable to the softening length in a regular softening transformation and κ adjust the shape of the transition near $r \sim \bar{\epsilon}$. This smoothed mass profile is then used to compute the gravitational potential,

$$\frac{d\Phi}{dr} G = \frac{\bar{m}(r)}{r^2}, \tag{2.2.3}$$

where $\Phi \to 0$ as $r \to \infty$. After expressing the density profile $\rho(r)$ in terms of the

[4] In the context of an N-body simulation, a test particle is a simulated object that interacts with the system's gravitational field but does not influence it due to its negligible mass. Test particles are used to study trajectories without affecting the overall dynamics of the system.

[5] Self-consistent refers to an approach where the gravitational field is continuously updated based on the current mass distribution, ensuring that all particles mutually influence and respond to the evolving field, providing a realistic simulation of the dynamics of the system.

corrected mass profile, the isotropic distribution

$$f(\varepsilon) = \frac{1}{\sqrt{8}\pi^2}\frac{d}{d\varepsilon}\int_{\varepsilon}^{0} d\Phi(\Phi - \varepsilon)^{-1/2}\frac{d\rho}{d\Phi} \tag{2.2.4}$$

is computed for each component using the Eddington (1916) formula.

Once $f(\varepsilon)$ has been calculated, an N-body realization with $N_{\rm sphr}$ equal-mass particles is generated. This realization is then populated with $N_{\rm test}$ massless particles moving on circular orbits. These test particles represent multiple discs whose trajectories can be tracked after the simulation has ended. The orientation of each particle i is chosen by randomly drawing its normalized angular momentum $\hat{\boldsymbol{s}}_i$ from a uniform distribution on the unit sphere $\boldsymbol{S}^2$. The radial distribution of the test particles follows the cumulative profile of the disc $m_d(r)$, biased by a factor of r^2, aimed at improving the sampling of particles in the outer parts of the disc. At the moment of visualization, the test particles representing the disc with a normalized spin vector $\hat{\boldsymbol{s}}_d$ are those with

$$1 - \hat{\boldsymbol{s}}_d \cdot \hat{\boldsymbol{s}}_i \leq \sigma/\max(q_i, r_{\rm min})^2. \tag{2.2.5}$$

Here σ is a tolerance parameter proportional to the number of particles selected, q_i is the initial orbital radius of particle i and $r_{\rm min}$ is a parameter which keeps Eq. [2.2.5] from diverging for small q_i. Finally, two such configurations are placed on a relative orbit with a given pericentric separation p and eccentricity e and are evolved using a standard treecode with a leapfrog integrator (Barnes & Hut, 1986; Barnes, 1990). We save positions and velocities every few time steps, creating a database of several hundred frames that trace the system's history from start to finish.

2.3 Galaxy models

We construct four galaxy models following parameters similar to those presented in Barnes & Hibbard (2009) and Pearson et al. (2018). All models were set up in approximate initial dynamical equilibrium. As previously stated, each galaxy model contains three collisionless components, initialized with explicit density profiles: a halo, a bulge, and a disc. The halo is composed of a Navarro, Frenk & White (NFW) dark matter halo (Navarro et al., 1996) without adiabatic

Table 2.3.1: Conversion factors from natural units used by IDENTIKIT to physical units.

Sim.Unit	Conversion Factor	Phys.Unit
1 (length)	62.50	kpc
1 (velocity)	127.83	kpc/Gyr
1 (time)	488.91	Myr
1 (mass)	226.90	$GM_\odot$

compression, containing 80% of the total mass, which tapers at large radii (b_h):

$$\rho_h(r) = \begin{cases} \frac{M_h(a_h)}{4\pi(\ln(2)-1/2)} \frac{1}{r(r+a_h)^2} & , r \leq b_h, \\ \rho_h^* \left(\frac{b_h}{r}\right)^\beta e^{-r/a_h} & , r > b_h, \end{cases} \tag{2.3.1}$$

where $M_h(a_h)$ is the halo mass within the scale radius of the halo (a_h), and ρ_h^* and β are fixed by requiring that both $\rho_h(r)$ and its derivative ($\partial_r \rho_h$) are continuous at $r = b_h$. The density profile tapers off exponentially, using the functional form devised by Springel & White (1999). The bulge follows a Hernquist (1990) model out to a radius $b_b = 200a_b$, where a_b is the bulge scale radius. At larger radii, the model tapers to avoid placing a small number of particles at greater distances,

$$\rho_b(r) = \begin{cases} \frac{a_b M_b}{2\pi} \frac{1}{r(a_b+r)^3} & , r \leq b_b \\ \rho_b^* \left(\frac{b_b}{r}\right)^2 e^{-2r/b_b} & , r > b_b \end{cases} \tag{2.3.2}$$

where M_b is the bulge mass, containing 5% of the total mass, and ρ_b^* is fixed by requiring that $\rho_b(r)$ is continuous at $r = b_b$. Such a compact bulge has little direct effect on the dynamics of tidal interactions but helps to stabilize the disc against bar instabilities. The disc has an exponential radial profile (Freeman, 1970) and a sech^2 vertical profile (van der Kruit & Searle, 1981),

$$\rho_d(q, \phi, z) = \frac{M_d}{4\pi a_d^2 z_d} e^{-q/a_d} \mathrm{sech}^2\left(\frac{z}{z_d}\right), \tag{2.3.3}$$

where $(q = \sqrt{x^2 + y^2}, \phi, z)$ are cylindrical coordinates, a_d is the disc scale radius, z_d is the disc scale height, and M_d is the mass of the disc, which contains the remaining 15% of the total mass. These density profiles are used to calculate the mass profile described in Eq. [2.2.1]. Following the procedure outlined in Sec. 2.2, a spherical N-body realization is constructed, embedding N_{test} massless particles that track multiple disc orbits. The simulations use natural units with

Table 2.3.2: List of parameters for all galactic models used in this study: full-mass (FM), half-mass (HM), quarter-mass (QM), and eighth-mass (EM). Values are provided in physical units for the three components that comprise each galactic model, including the hole system values such as the number of particles and the resolution achieved through the softening length.

Parameter	FM	HM	QM	EM	Phys.Unit
Halo					
$M_{\rm h}$	226.9	113.5	56.73	28.36	$GM_\odot$
$a_{\rm h}$	15.63	11.05	7.81	5.52	kpc
$b_{\rm h}$	61.26	43.32	30.63	21.55	kpc
Bulge					
$M_{\rm b}$	14.18	7.09	3.55	1.77	$GM_\odot$
$a_{\rm b}$	1.25	0.88	0.63	0.44	kpc
Disc					
$M_{\rm d}$	42.54	21.27	10.64	5.32	$GM_\odot$
$a_{\rm d}$	5.21	3.68	3.02	1.84	kpc
$z_{\rm d}$	0.47	0.47	0.47	0.28	kpc
System					
$N_{\rm sphr}$	$80k$	$80k$	$80k$	$40k$	-
$N_{\rm test}$	$256k$	$128k$	$64k$	$64k$	-
$\bar{\epsilon}$	$7.5 \cdot 10^{-3}$	$7.5 \cdot 10^{-3}$	$7.5 \cdot 10^{-3}$	$7.5 \cdot 10^{-3}$	-

Newton's constant $G = 1$; however, the conversion factors to physical units are automatically calculated through the IDENTIKIT interface. These factors are shown in Table 2.3.1. We aim to explore major merger interactions (i.e., mass ratio $\mu \geq 1/4$), so we define the parameters to create a stable galactic model based on the values presented in Barnes & Hibbard (2009), which we refer to as the *full-mass* (FM) model. From this, we generate two galactic models with reduced mass: the *half-mass* (HM) and *quarter-mass* (QM) models. Additionally, because galaxies resulting from minor interactions are expected in the local universe, we include an *eighth-mass* (EM) model, which was created by Pearson et al. (2018) to model the kinematics of NGC 4485. The complete set of parameters required to compute the galactic models used in this study is shown in Table 2.3.2.

We conducted isolated realizations of the four galaxy mass models to assess their long-term stability. In Fig. 2.3.1, we show the evolution of the galaxies in isolation after approximately 5 Gyr (i.e., 10 simulation units) of self-interaction. We observe that all four models remain in equilibrium without significant changes. The most massive model (FM) begins to show signs of spiral arms in its disc, while the other, more compact models do not exhibit any changes in their morphology.

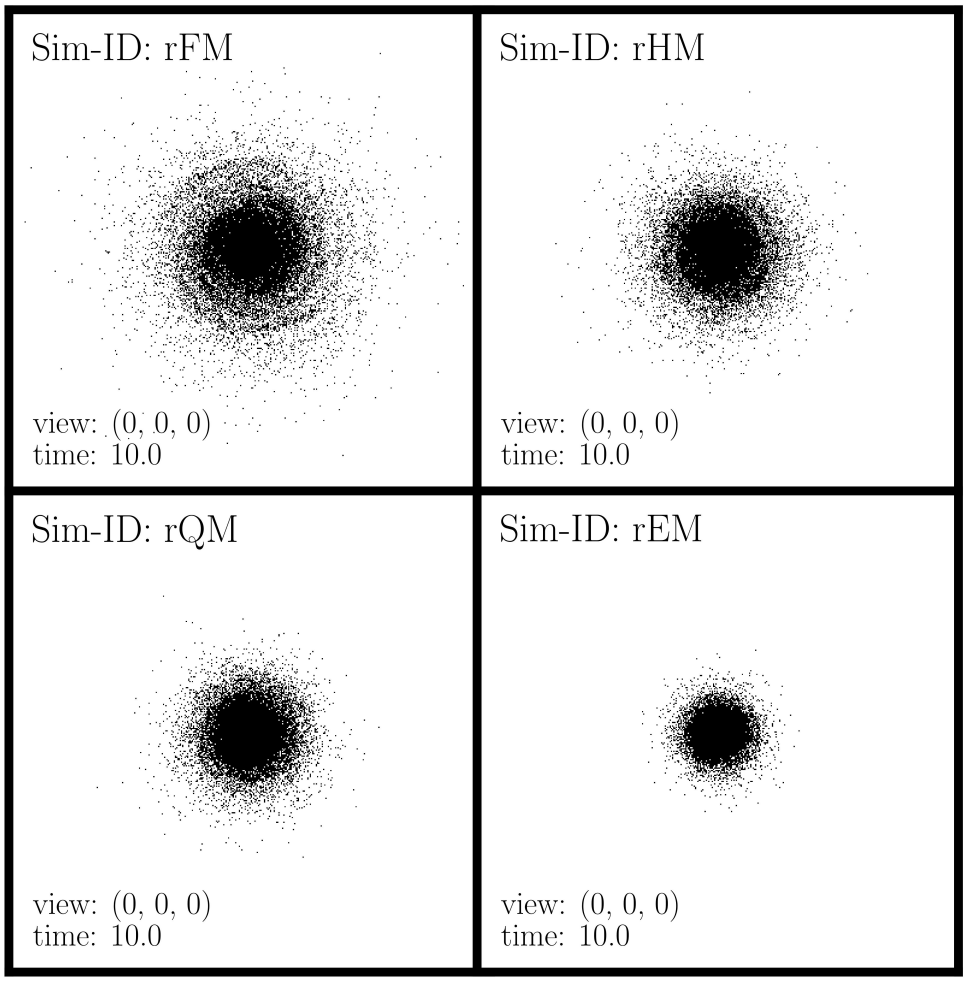

Figure 2.3.1: Snapshots of the simulations for the different galactic models evolved in isolation. (*top-left*) Full-mass model. (*top-right*) Half-mass model. (*bottom-left*) Quarter-mass model. (*bottom-right*) Eighth-mass model. All galaxies were run for approximately 5 Gyr (i.e., 10 simulation units) and are visualized face-on to check for any morphological instabilities.

2.4 Library of encounters

Of the sixteen parameters needed to simulate the encounter of two disc galaxies, only a few have a priori constraints. Toomre & Toomre (1972) argued that the orbital eccentricity should be $e \simeq 1$, a point generally supported by cosmological simulations (e.g., Khochfar & Burkert, 2006). Therefore, we set the orbital eccentricity to $e = 1$ for all our simulations, as $e < 1$ could imply prior encounters. Nevertheless, the eccentricity will evolve over the course of the encounter due to dynamical friction. Regarding the other two main parameters that define the initial orbit, we use different combinations of our galactic models to generate

Table 2.4.1: Summary of IDENTIKIT simulations for galaxy mergers. We highlight the combination of galactic models, the galactic mass ratio, the initial pericentric separation of the system, the total simulation time, and the number of particles displayed per galactic disc in the graphical interface once the simulation is complete.

ID	Gal-1	Gal-2	μ	p [kpc]	t [Myr]	N_{test} [(Gal-1; Gal-2)]
r121	FM	FM	1/1	3.91	4,889	(11883; 11908)
r124	FM	FM	1/1	15.63	4,889	(11883; 11908)
r128	FM	FM	1/1	31.25	7,823	(11883; 11908)
r1212	FM	FM	1/1	46.88	9,778	(11883; 11908)
r131	FM	HM	1/2	3.91	5,867	(16646; 8424)
r134	FM	HM	1/2	15.63	4,889	(16646; 8424)
r138	FM	HM	1/2	31.25	7,823	(16646; 8424)
r1312	FM	HM	1/2	46.88	9,778	(16646; 8424)
r151	FM	QM	1/4	3.91	5,867	(19828; 4926)
r154	FM	QM	1/4	15.63	5,867	(19828; 4926)
r158	FM	QM	1/4	31.25	7,823	(19828; 4926)
r1512	FM	QM	1/4	46.88	9,778	(19828; 4926)
r191	FM	EM	1/8	3.91	11,733	(20028; 5027)

systems with varying mass ratios (specifically, $\mu = \{(1/1), (1/2), (1/4), (1/8)\}$). For major mergers, we considered four different initial pericentric separations in the range $p \in$[3.91, 46.88] kpc, which resulted in a longer simulation time to reach the final equilibrium between the two galaxies as we increase the value of p. The combination of different values of μ and p resulted in 13 unique interacting systems, each of which contains all possible interactions between galactic discs with different inclination angles. The summary of this simulations can be seen on Table 2.4.1.

The IDENTIKIT software includes interactive routines allowing the user to select the disc orientations, viewing direction, scale factors, and centroid positions. The resulting test-particle coordinates are instantly projected onto the (X, Y), (X, V), (V, Y), and (X, Z) planes. The user can also step forward or backward in time, switch between databases, and vary the tolerance parameter σ, which is proportional to the number of test particles populating a single disc. An example of the graphical interface can be seen in Fig. 2.4.1.

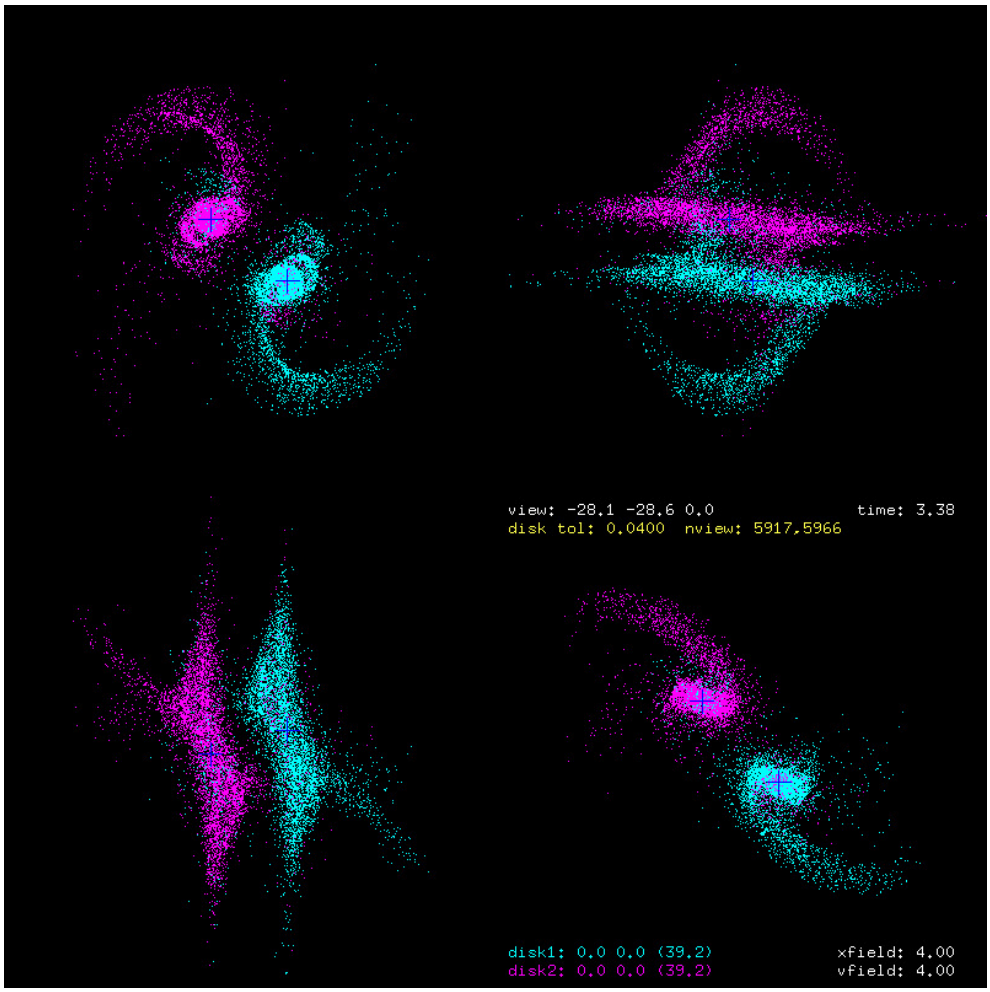

Figure 2.4.1: Example of an IDENTIKIT simulation displayed in the interactive graphical interface (specifically, the r121 simulation listed in Table 2.4.1). The four quadrants shown correspond to: (*top-left*) (X, Y) plane; (*top-right*) (V, Y) plane; (*bottom-left*) (X, V) plane; and (*bottom-right*) (X, Z) plane. The colors differentiate test particles from each disc, and blue crosses indicate the center of each galaxy.

Chapter 3

Data description

3.1 Synthetic galaxies

From the IDENTIKIT simulations, we draw two samples of galaxies: a sample comprising only major mergers, categorized into galaxy pairs and post-merger phases, and a sample of non-interacting galaxies as control sample. Details on the selection criteria for both samples are described in Sec. 3.1.1 and Sec. 3.1.2. Subsequently, we detail the properties of the final imaging set obtained and the preprocessing performed before to the experiments conducted in this study.

3.1.1 Major-merger selection

All our samples are selected from the simulations in Table 2.4.1 with a mass ratio $\mu \geq 1/4$ (specifically, r12, r13, and r15 simulations). For each merger simulation, we select snapshots to construct images, differentiating the merger stage based on the following temporal definitions:

$$\text{snap}_{\text{merger}}(t_i) \begin{cases} \text{Galaxy pair (pair)} & \text{if } t_i \in [t_p - 100\text{Myr}, t_c - 100\text{Myr}], \\ \text{Post-merger (post)} & \text{if } t_i \in [t_c + 100\text{Myr}, t_c + 600\text{Myr}], \end{cases}$$

where t_p and t_c refer to the time of the initial pericentric separation (p) and the time of coalescence, respectively. The initial pericentric separation is part of our initial conditions in each merger system, so the definition of this specific time comes naturally. For the time of coalescence, we track the kinematics of the inner

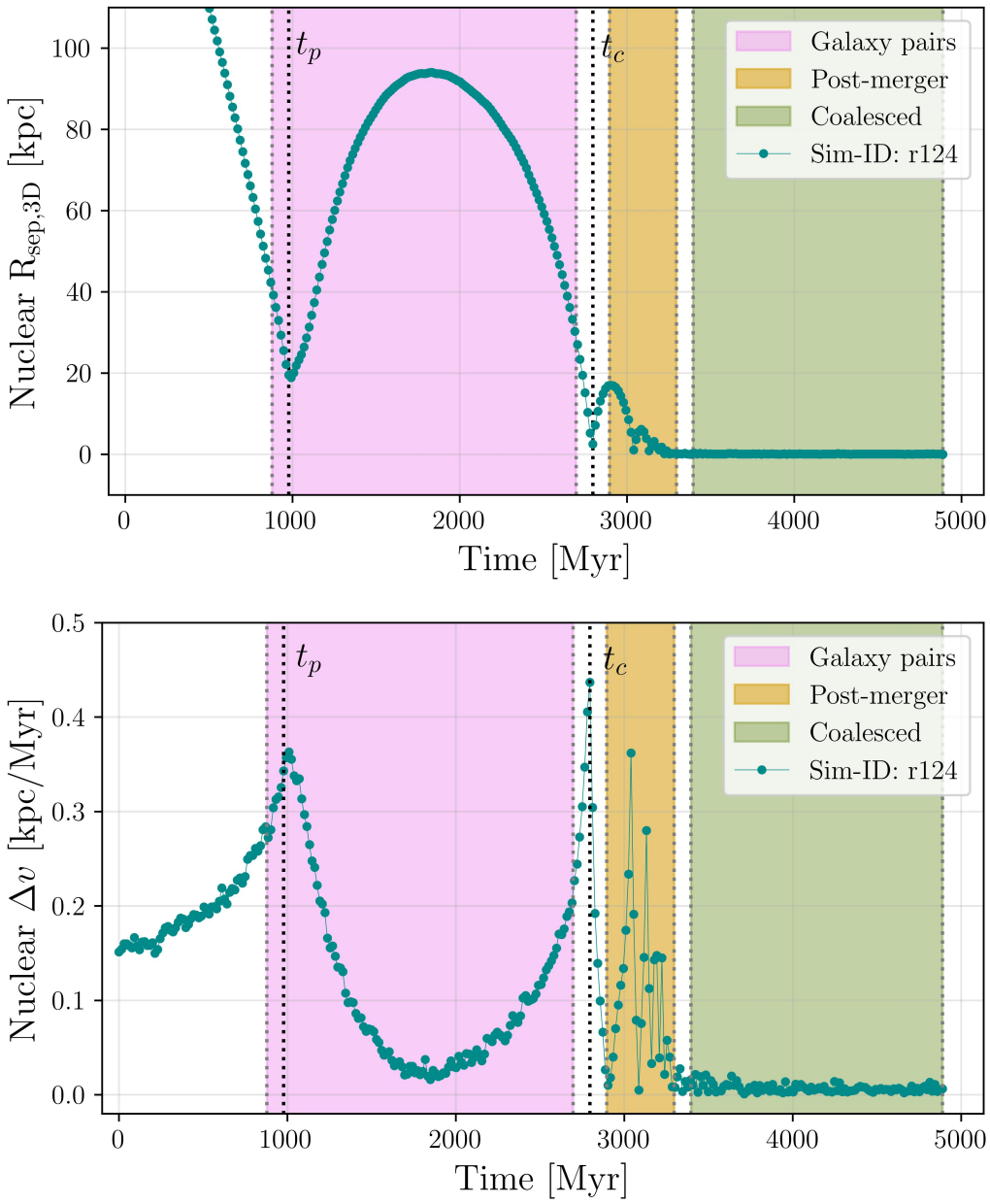

Figure 3.1.1: Example of the class selection criteria for galaxy mergers compared to the kinematic information extracted from each snapshot in the r124 simulation: (*top*) Evolution of the relative separation between the centers of the two galaxies over time; (*bottom*) Evolution of the relative nuclear velocity over time. The dotted black lines represent the time of the initial pericentric separation and the time of coalescence, respectively. The shaded areas indicate the different merger stages.

particles at the center of each galaxy and, as in Moreno et al. (2019), define t_c as the last time the central black holes are more than 0.5 kpc apart. Fig. 3.1.1 shows an example of how our selection criterion is applied to the evolution of the relative nuclear separation between the two galaxies and their relative nuclear velocity. We note that various criteria and cutoffs, based on gas kinematics, star formation values of both galaxies or timescales computed from cosmological simulations, can be used to differentiate galaxy pairs and post-mergers (e.g., Moreno et al., 2019; Bottrell et al., 2019a; Ferreira et al., 2020). However, for simplicity, we limit our analysis to the information available in our simulations, using time ranges consistent with previous studies in the literature and with the perturbations observed in our data. The galaxy-pair phase is defined to begin 100 Myr before the first pericentric passage and end just before coalescence. Some studies also propose a cut in the projected relative velocity (e.g., Ellison et al., 2008; Patton et al., 2013); however, this can introduce biases due to the omission of snapshots around the first and second pericenter, depending on the line of sight of the snapshot. The post-merger phase is determined to last 500 Myr after coalescence, which is long enough to include disturbances produced by the collision without achieving equilibrium in the system, as morphological features after a collision fade over time, with higher merger mass ratios causing these features to fade more quickly (Lotz et al., 2010).

We space the area of snapshot selection by 200 Myr for two reasons: First, we aim to reduce inter-class confusion during training of the ML classifier by avoiding snapshots that are too close to the point of differentiation between galaxy pairs and post-mergers. Second, several analyses suggest a delay between mergers and black hole growth on the order of 100 Myr, a property that is often used in theoretical analysis as the beginning of the post-merge phase (e.g., Di Matteo et al., 2007; Wild et al., 2010; Moreno et al., 2019).

Table 3.1.1: Summary of the different angles explored in this study. The configuration of each system is constructed by leaving the first galaxy in its fixed position (i.e., $(i_1 = 0°, \omega_1 = 0°)$), while varying the inclination (i_2) and azimuth (ω_2) of the second galaxy according to the combination of the displayed parameters.

Disc position	$\boldsymbol{i_2}[°]$	$\boldsymbol{\omega_2}[°]$
Parallel	0.0	{0.0}
Upside	45.0	{0.0; 90.0; 180.0; 270.0}
Flat-side	90.0	{0.0; 90.0; 180.0; 270.0}
Downside	135.0	{0.0; 90.0; 180.0; 270.0}
Anti-parallel	180.0	{0.0}

3.1.2 Non-merger selection

A sample of non-mergers is necessary for our model to learn how to distinguish major mergers from other types of galaxies. The construction of this sample considers different scenarios where a snapshot is categorized as an *isolated* (iso) galaxy:

$$\mathrm{snap_{iso}(t_i)} \begin{cases} \text{Isolated model realization,} & \\ \text{Major-merger simulation} & \text{if } t_i \in [t_c + 700\mathrm{Myr}, t_{\mathrm{end}}], \\ \text{Minor-merger simulation} & \text{if } t_i \in [t_c + 700\mathrm{Myr}, t_{\mathrm{end}}], \end{cases}$$

where t_{end} is the final time of the simulation. We explore different types of disc and elliptical galaxies in this set. We include perfect discs from the realizations of isolated massive galaxy models (i.e., the rFM, rHM, and rQM simulations), excluding the EM model due to its low mass. Elliptical galaxies produced as a result of major merger simulations are selected once both galaxies have reached coalescence and the resulting system no longer suffers major perturbations (see Fig. 3.1.1, green area). Finally, we also include the result of a minor merger case, represented by the r181 simulation. Minor mergers do not cause major disruptions in morphology, preserving the structure and properties of the most massive galaxy while the satellite galaxy is completely absorbed. However, these interactions can induce changes in the disc of the massive galaxy (Villalobos & Helmi, 2008; Stewart et al., 2009; Barnes, 2016).

The construction of the synthetic dataset is as follows: For all major mergers, we explore 14 different angles between the colliding galaxy discs (see Table 3.1.1).

From each of these configurations, we select six random viewing angles and extract three snapshots for each class according to their corresponding time in the simulation. A schematic representation of this process is provided in Fig. 3.1.2. From the combination of these parameters we obtain 168 unique interacting systems and a total of 3024 snapshots per class. As previously stated, we include 300 snapshots from the isolated galaxy model realizations and 300 snapshots from the minor-merger simulation to the isolated class, yielding a total of:

$$\mathcal{D}_{\rm org} = \{N_{\rm iso},\ N_{\rm pair},\ N_{\rm post}\} = \{3624,\ 3024,\ 3024\} \tag{3.1.1}$$

samples. We visualize these snapshots through the IDENTIKIT interface and convert their projected positions into grayscale images (i.e., one-channel images) of pixel size (80, 80). We rescale the pixel values of each image to the range between 0 and 1, normalizing by the maximum and minimum pixel value of each image. We are aware that intrinsic brightness can be a classification criterion by itself. However, we are only interested in the structure and not in other properties that might correlate with a galaxy class or a dominant stellar population, such as surface brightness. To mitigate overfitting durnig the training of the classifiers and enhance confidence in predictions, we randomly divided our dataset into two subsets: training (7254 images) and testing (2418 values). As further explained in Sec. 4.2, the Modulo-AutoML platform automatically divides the input dataset into training and validation subsets, with 80% used for training and 20% for validation.

We generate augmented images for the training set by applying zoom, rotation, and small translational transformations to the original set to prevent *imbalanced classes*[1] in the dataset and achieve rotational invariance in the models (Dieleman et al., 2015). The final synthetic training set then consist of

$$\mathcal{D}^{\rm train}_{\rm aug} = \{N^*_{\rm iso},\ N^*_{\rm pair},\ N^*_{\rm post}\} = \{10000,\ 10000,\ 10000\} \tag{3.1.2}$$

[1]Class imbalance is a common issue in classification tasks when one class in the dataset is more prevalent than the others. Imbalanced datasets can significantly impact ML models, leading to classifiers that favor the dominant class (Buda et al., 2018). Various solutions have been proposed to address this problem, such as upsampling, downsampling, or weighting the cost function according to the proportion of objects in each class relative to the total. Additionally, in certain cases, it is possible to use a known distribution of objects to condition the classifier on this prior distribution. For this study, we upsampled the images to equalize the number of objects per class.

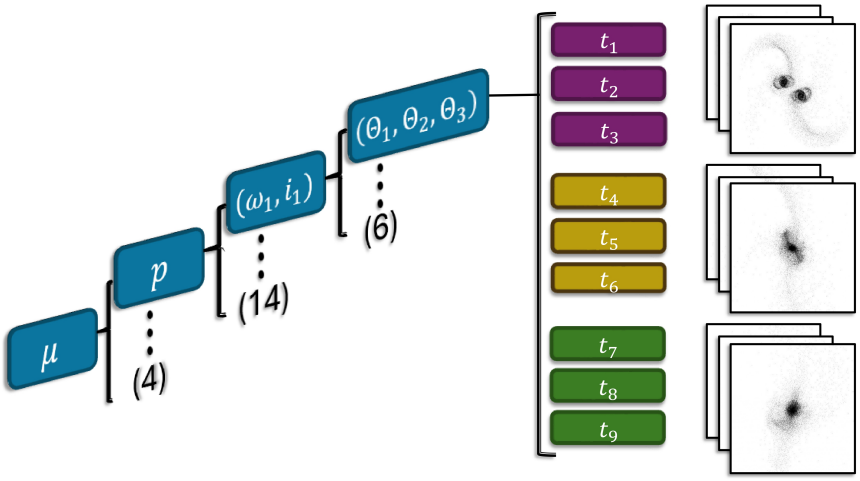

Figure 3.1.2: Schematic representation of the snapshot selection for galaxies in major-merger simulations. Given the mass ratio and the initial pericentric separation of the system, the angles between the galactic disks are selected from the list in Table 3.1.1. A random viewpoint is considered, from which three snapshots are chosen within the time ranges specified for each class. The numbers in parentheses indicate the total number of possible values for each parameter.

samples. Finally, although the IDENTIKIT simulations also give us a visualization of the velocity projections for the particles that make up both galaxies, we will leave this information out of the training for the classifiers since performing spectroscopy to obtain data cubes for a large sample of observational galaxies is not feasible.

3.2 Observed galaxies

Galaxy mergers are often identified by low surface brightness and fast-fading features. We require a larger sample of galaxies and deep imaging to capture these major disturbances and test our models. DECaLS is ideal for this purpose. DECaLS uses the Dark Energy Camera (DECam; Flaugher et al., 2015) at the 4m Blanco Telescope at Cerro Tololo Inter-American Observatory near La Serena, Chile. DECam has a roughly hexagonal 3.2 deg^2 field of view with a pixel scale of 0.262" per pixel. The median point spread function full width at half maximum (FWHM) is 1".18, 1".18, and 1".11 for g, r, and z, respectively. DECaLS covers an area of 10480 deg^2 with 5σ depths of $g \sim 24.65$, $r \sim 23.61$, and $z \sim 22.84$[2] in a

[2]See https://www.legacysurvey.org/dr5/description/ and related pages.

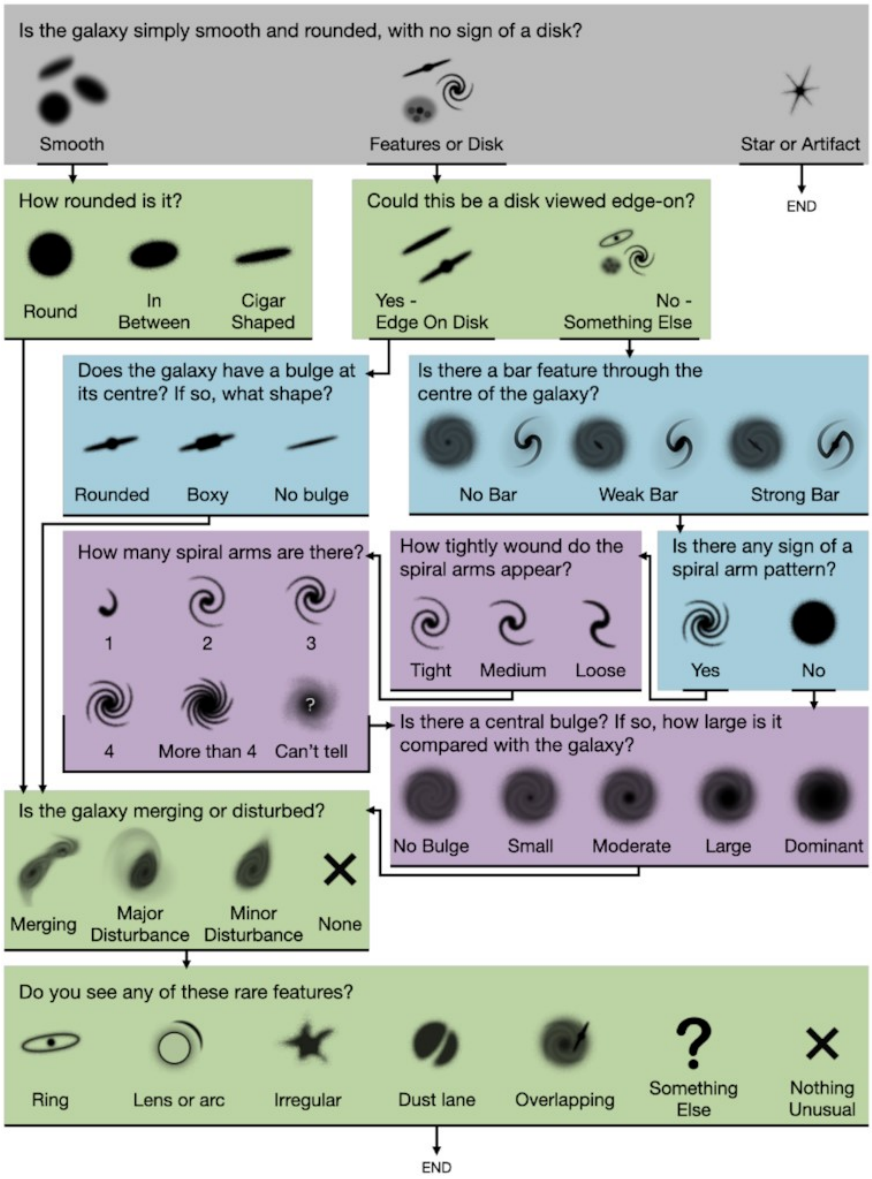

Figure 3.2.1: Classification decision tree for GZ5. Questions shaded with the same colours are at the same level of branching in the tree; grey have zero-depended questions, green one, blue two, and purple three. Figure from Walmsley et al. (2022).

0.45" aperture (Dey et al., 2019). The survey completed observations in March 2019.

Walmsley et al. (2022) presented the catalog Galaxy Zoo DECaLS, which combines the morphological classifications from Galaxy Zoo Data Release 5 (GZ5) volunteers with predictions of a Bayesian neural network on a sample of 314000 galaxies from DECaLS imaging included in the NASA-Sloan Atlas v1.0.0 (NSA). The volunteer classifications of RGB images of galaxies with a pixel size of (424, 424) employed more complex classification criteria than previous versions (see Fig. 3.2.1), aimed at improving the classification of weak bars and minor mergers. Both the catalog and the images are available to the community via the Zenodo[3] platform, from which we compiled the information used in this study. Following the indications for the use of the catalog (for further details, see Sec. 6 Walmsley et al., 2022), we delete artifacts, wrong-size images, and images with aberrations. We focus our attention on the vote fractions that represent mergers or disturbances of the galaxies. To identify massive galaxies and major mergers within the GZ5 catalog, we used stellar mass measurements from the Max Planck Institute for Astrophysics and the Johns Hopkins University (MPA-JHU) catalog. The measurements of MPA-JHU contain galaxy spectra and physical parameters of galaxies in the SDSS Data Release 8. Stellar masses are calculated using the Bayesian methodology and model grids described in Kauffmann et al. (2003) for the optical photometric measurements (i.e, observations in u, g, r, i, and z filters) of galaxies. Star formation rates (SFRs) are computed within the galaxy fiber aperture using the nebular emission lines as described in Brinchmann et al. (2004). SFRs outside of the fiber are estimated using the galaxy photometry following Salim et al. (2007). They supply emission line classifications based on the BPT diagram (Baldwin et al., 1981; Brinchmann et al., 2004). Galaxies are divided into *Star Forming*, *Composite*, *AGN (optical)*, *Low S/N Star Forming*, *Low S/N LINER*, and *No-class* categories. As previously mentioned in Sec. 1.2, the AGN selection criteria may introduce biases regarding the presence of mergers. However, studying the AGN population within galaxy mergers is beyond the scope of this research. Therefore, the information available in the MPA-JHU catalog will be used only to approximately characterize our sample of galaxies.

We match the coordinates of the GZ5 imaging provided by the NSA to those

[3]GZ5-DECaLS: https://zenodo.org/records/4573248

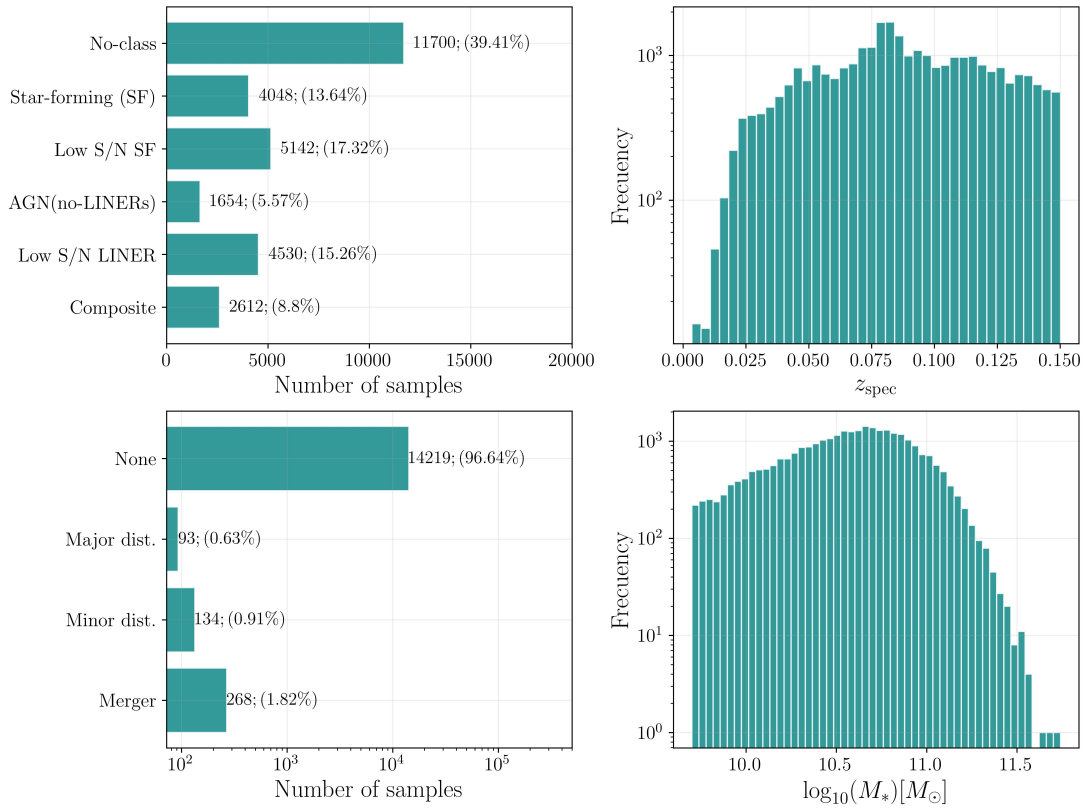

Figure 3.2.2: Properties of the sample of galaxies collected from GZ5 used in this study: (*top-left*) Emission line classifications based on the BPT diagram. (*top-right*) Distribution of spectroscopic redshift values. (*bottom-left*) Classifications by merger or perturbation type according to GZ5 fractions; galaxies with fraction votes $f_i \geq 0.7$ where i corresponds to the respective category, were counted. (*bottom-right*) Distribution of stellar mass values.

in the MPA-JHU catalog within a radius of 5" and filter for galaxies with a stellar mass of $\log(M_*/M_\odot) \geq 9.7$. The final sample contains 29686 galaxies. The distribution of properties such as stellar mass, redshift, BPT classification, and GZ5 merger vote fraction is shown in Fig. 3.2.2. Predominantly observed at low redshift and with low nuclear activity (at least in the optical range), the GZ5 visual classifications predict a sample of ∼300 massive galaxies that are experiencing or have experienced a merger event. Similar to the synthetic dataset, the GZ5/MPA-JHU dataset is converted into grayscale images and resized to a pixel size of (80, 80). The pixel values of each image are rescaled to a range between 0 and 1, normalizing by the maximum and minimum pixel values of each image.

Chapter 4

Experimental setup

4.1 Problem definition

The objective of this study is to analyze whether an ML model ($f_{\boldsymbol{\theta}}$) with parameters $\boldsymbol{\theta}$ can estimate the merger stage between two colliding galaxies using only morphological features. Given a dataset $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), ..., (\boldsymbol{x}_N, y_N)\}$, consisting of N pairs of images $\boldsymbol{x}_i \in \mathbb{R}^{n \times n}$ and labels $y_i \in Y = \{0, 1, 2, ..., K\}$, where $n \times n$ is the pixel size and K represents the number of classes, the supervised multi-class classification problem can be defined as determining a discriminant learning function,

$$f_{\boldsymbol{\theta}} : \mathbb{R}^{n \times n} \rightarrow Y \quad | \quad \hat{y}_i = f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \tag{4.1.1}$$

where $\hat{y}_i$ corresponds to the class prediction of the respective image $\boldsymbol{x}_i$, and the discriminant function is learned from the input data by updating the $\boldsymbol{\theta}$ parameters according to a cost function $\mathcal{L}(y, \hat{y})$ which compares the known actual values with the model predictions.

We proceed to explain the platform used for model selection, describe how the optimization process is performed, outline the classification algorithms employed, and detail how each classifier is evaluated. Finally, the experiments conducted in this study are presented.

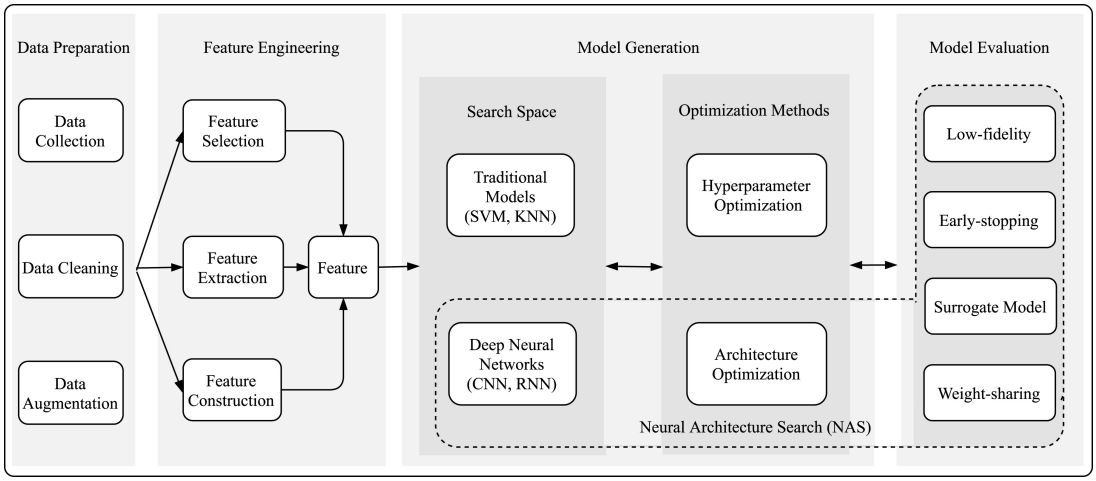

Figure 4.2.1: A general overview of an AutoML pipeline covering data preparation, feature engineering, model generation and model evaluation. Figure from He et al. (2021).

4.2 The AutoML modulos platform

The selection of the most suitable ML model is a computationally expensive task, often influenced by the nature and dimensionality of the data. To reduce these development costs, the concept of automating the entire ML pipeline has emerged: automated machine learning (AutoML). AutoML is a comprehensive end-to-end system that dynamically combines various techniques to create an easy-to-use ML pipeline (for a review see He et al., 2021). This process typically involves several steps, summarized in Fig 4.2.1: preliminary preparation of input features (e.g., cleaning or augmentation), feature engineering, selection and optimization of the ML algorithm, and evaluation of the resulting model.

We use an AutoML workflow from the `Modulos-ai`[1] service. The Modulos-AutoML platform (version 1.1.2) is designed for automated model selection and supervised learning in regression, classification, and forecasting tasks. The platform detects the type of uploaded data, randomly splits the input dataset into training and validation sets, proposes a range of ML models for the task, and generates a *candidate*[2]. The candidate is then trained, validated, and delivered to the user, with this cycle repeating until a user-defined endpoint is reached.

[1]Modulos-AutoML: https://www.modulos.ai/

[2]Hereafter, we refer to the combination of an ML model and the hyperparameters chosen by the platform as a *candidate*. This candidate is subsequently trained, validated, and delivered to the user as a ready-to-use *solution*.

4.2.1 Bayesian optimization

In ML, hyperparameters are predefined settings that control the training process and model structure. Unlike model parameters ($\boldsymbol{\theta}$), which are learned during training, hyperparameters are set before the training begins. Their selection is crucial for model performance. The Modulos-AutoML platform uses Bayesian optimization (e.g., Bergstra et al., 2011; Snoek et al., 2012; Srinivas et al., 2012) to iteratively adjust the hyperparameters of each ML algorithm, thereby reducing the computational costs associated with hyperparameter optimization. Bayesian optimization employs a surrogate model, often a Gaussian process, to approximate the objective function (i.e., the function that quantifies the performance of the model) $g(h_1, h_2, ..., h_n)$, where $\{h_1, h_2, ..., h_n\}$ are the hyperparameters. Each combination of hyperparameters corresponds to a different model. Although evaluating the actual objective function is costly, Bayesian optimization can efficiently find a set of hyperparameters that optimizes the performance of the classifier by updating the posterior distribution with each evaluation and using it to make informed predictions about the next hyperparameter combination.

Modulos-AutoML considers different quantities as hyperparameters depending on the selected ML algorithm, something we discuss in more detail in Sec. 4.2.2.

4.2.2 Machine learning and deep learning models

Given the number and the size of images in our synthetic dataset, Modulos-AutoML proposes random forest, extreme gradient boosting, and convolutional neural networks as potential classifiers. We detail each algorithm and the respective hyperparameters considered during the optimization.

4.2.2.1 Random forest

Random forest (RF) is an ensemble learning method introduced by Breiman (2001) that synthesizes the outputs of multiple individual decision trees to produce a final classification. Each tree within the RF is trained on a bootstrap sample (i.e., a randomly drawn subset of the training data with replacement), a technique known as bootstrap aggregating or bagging (Ho, 1998). For classification tasks, the final prediction of the model is determined by majority voting among the individual trees (e.g., Khaled Fawagreh & Elyan, 2014). Despite its relatively straightforward

design and long-standing presence, RF remains one of the most robust ML models in astronomy due to its capability to capture complex nonlinear relationships and interactions (e.g., Dubath et al., 2011; Cheng et al., 2020; Sánchez-Sáez et al., 2021; Rose et al., 2023). It is frequently employed as a benchmark model for comparing the performance of more sophisticated algorithms.

Modulos-AutoML optimizes the hyperparameters in RF that control the number of estimators (i.e., the total number of trees), the minimum number of samples required for a leaf node, the function used to measure the quality of each split and the number of features considered when splitting a node.

4.2.2.2 Extreme gradient boosting

Extreme gradient boosting (XGBoost; Chen & Guestrin, 2016) is a scalable machine learning algorithm based on gradient tree boosting[3] (Friedman, 2001) designed to address the limitations of traditional boosting methods. It improves predictive performance by optimizing a series of weak learners[4], typically decision trees, which are sequentially trained to minimize residual errors from previous iterations.

XGBoost employs the gradient descent algorithm to optimize the following objective,

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(\boldsymbol{x}_i)) + \Omega(f_t) \tag{4.2.1}$$

Here l is a differentiable convex loss function that measures the difference between the prediction y_i and the target $\hat{y}_i$ at an instant i and iteration t. This objective adds the term f_t that most improves its performance, adjusting parameters at each step, thus creating new trees and minimizing the errors of the previous ones. $\Omega(f_t)$ is a regularization term that incorporates Lasso (L1) and Ridge (L2) penalties to reduce model complexity and prevent overfitting. For computational efficiency, XGBoost employs second-order Taylor expansion to approximate the loss function,

[3] In the literature, gradient tree boosting is also known as gradient boosting machine (GBM) or gradient boosted regression tree (GBRT).

[4] A *weak learner* or *base learner* is an algorithm that performs slightly better than random guessing on a given problem. However, when combined with other weak learners in a larger model known as an *ensemble*, they collectively contribute to creating robust models capable of making accurate predictions. Examples of weak learners include decision trees (with a low deep of the tree), Naive Bayes classifiers, single-layer perceptron, and k-nearest neighbors (with a small k), among others.

enabling faster convergence and more accurate predictions (for more details, see Chen & Guestrin, 2016, Sec. 2). Gradient tree algorithms are not common in image-based analysis in astronomy; however, they have been successfully implemented in the estimation of physical parameters through regression from photometric and spectroscopic data (e.g., Tamayo et al., 2016; Bethapudi & Desai, 2018; Tarsitano et al., 2022; Andrae et al., 2023).

Modulos-AutoML sets the number of trees to 100 and optimizes the hyperparameters that control the step size in the gradient descent optimization process (i.e., the learning rate), the minimum loss reduction required to make a further partition on a leaf node and the depth to which a tree can grow.

4.2.2.3 Convolutional neural networks

Deep learning (DL) is a subarea of machine learning based on artificial neural networks (ANNs). The fundamental building unit of an ANN is the perceptron, which aims to mathematically represent the behavior of a human neuron (Rosenblatt, 1958). The perceptron is defined as follows:

$$\hat{y} = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b), \tag{4.2.2}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ represents the input vector containing n values, $\boldsymbol{w} \in \mathbb{R}^n$ represents the parameters of the perceptron, b the bias, and $\hat{y}$ the predicted output of the model. The function σ is called activation function, its a nonlinear function that transforms the result of the linear transformation and determines the ouput of the perceptron.

The combination of several perceptrons, with interconnected hidden layers, forms a multilayer perceptron (MLP; Rumelhart et al., 1986). The use of multiple layers allows the abstraction of complex nonlinear relationships, making MLPs universal approximators. Mathematically, the MLP can be defined as:

$$\hat{\boldsymbol{y}} = \sigma(\boldsymbol{W}^T \boldsymbol{x} + \boldsymbol{b}), \tag{4.2.3}$$

where $\boldsymbol{W} \in \mathbb{R}^{mxn}$ represents the matrix of parameters called weights, $\boldsymbol{x} \in \mathbb{R}^n$ is the input of the layer, $\hat{\boldsymbol{y}} \in \mathbb{R}^m$ is the output of a layer of m neurons, $b \in \mathbb{R}^m$ is a vector containing the bias of each neuron. Training a MLP involves updating the weights associated with each connection among neurons to minimize the loss

function that measures the difference between predicted and real outputs, in a process called backpropagation. This process is performed using optimization algorithms, such as gradient descent (e.g., Sun et al., 2020). In this framework, the most frequent loss function for classification is the categorical cross-entropy loss function (e.g., Wang et al., 2020),

$$\mathcal{L}_{\mathrm{CE}} = -\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic}\log\left(\hat{y}_{ic}\right), \tag{4.2.4}$$

where $\hat{y}_i$ corresponds to the model output, y_i corresponds to the actual desired output, C is the total number of classes, and N is the total number of data points used to compute the loss function.

While MLPs can theoretically approximate any continuous function, their practical application often requires a large number of neurons and layers, leading to increased computational complexity and a higher risk of issues such as overfitting and vanishing or exploding gradients. This often renders MLPs less efficient and computationally unfeasible for complex tasks. Furthermore, standard MLPs lack the ability to capture spatial relationships within the data, which is particularly crucial in tasks involving image pattern recognition, where local and hierarchical features are essential. Convolutional neural networks (CNNs; Fukushima, 1980; Lecun et al., 1998) have become the base models for tackling tasks involving matrix-structured data with DL. CNNs are a type of feed-forward neural network with multiple layers specifically designed for image processing, allowing better feature extraction than traditional ANNs.

CNNs comprise three principal components: convolutional layers, pooling layers, and fully connected layers. Convolutional layers apply filters to the input (typically images), performing matrix multiplications at each location to produce feature maps of reduced dimensionality. The filter values are optimized during training. Pooling layers reduce the dimensionality of the data by using sliding windows that select either the maximum value (max pooling) or the average value (average pooling). Convolutional layers are typically followed by pooling layers, with fully connected layers at the end to process the extracted features and perform the final task. Vanilla CNNs and their modifications offer a significant reduction in computational cost due to the use of fewer model parameters and have proven highly useful in astronomy for image-based analysis (e.g., Dieleman et al., 2015;

Huertas-Company et al., 2015; Pérez-Carrasco et al., 2019; Bottrell et al., 2019a; Ferreira et al., 2020; Medina-Rosales et al., 2024).

Modulos-AutoML proposes an architecture consisting of two convolutional blocks with kernel sizes of (3×3), using 32 and 64 filters, respectively, followed by a max pooling layer with a kernel size of (2×2), and a dropout layer (Srivastava et al., 2014) as a regularization method. Hidden layers are included up to the output layer, with the number of neurons matching the number of classes. Trainable weights are optimized using adaptive moment estimation (ADAM; Kingma & Ba, 2017). The hyperparameters to optimize include the batch size used during training, the dropout rate applied after the convolutional operations, the number of fully connected layers before the output layer, and the number of neurons in each of these layers. Each layer in the network employs rectified linear unit activation functions (ReLU; Krizhevsky et al., 2012), while the output layer uses a softmax activation function to provide the probabilities for each class. The softmax function can be seen as a generalization of the sigmoid function for multi-class classification, and its formaly describe as:

$$\text{Softmax}(x_i) = \frac{\exp(x_i)}{\sum_{j=1}^{n} \exp(x_j)}, \tag{4.2.5}$$

which rescales the elements x_i of the n-dimensional input $\boldsymbol{x}$ so that their values lie in the range [0, 1] and sum to 1.

4.2.3 Metrics and objective

The evaluation of a classifier involves assessing how accurately it assigns objects to their respective classes. This requires metrics that provide insights into the frequency of correct predictions, error rates, and how well the predictions reflect the entire dataset. In a binary classification problem, we typically examine the key concepts of *true negative* (TN), *true positive* (TP), *false positive* (FP), and *false negative* (FN), which are essential for defining these metrics[5] (e.g., Naidu et al., 2023). In a multi-class classification problem (such as the one studied

[5]In binary classification, true positives refer to correctly identified positive instances, while true negatives are correctly identified negative instances. False positives, or Type I errors, occur when negative instances are incorrectly classified as positive, and false negatives, or Type II errors, occur when positive instances are incorrectly classified as negative. These concepts are foundational for evaluating the performance of a classification model in terms of both correctness and error.

here), the concepts of TP, FP, and others are computed for each class individually. This is done by treating each class as the positive class while considering all other classes as negative. This method, known as *one-vs-all* or *one-vs-rest*, allows binary classification metrics to be extended to multi-class problems by breaking the problem down into multiple binary comparisons.

The AutoML platform supports a range of metrics for classification tasks, which serve as objectives for improvement during training. We select the Receiver Operating Characteristic curve (ROC curve; e.g., Fawcett, 2006) to examine the performance of each method and dataset. On a ROC curve, the y-axis represents the true positive rate (TPR, also known as *recall* or *sensitivity*),

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{4.2.6}$$

which reflects the ability of the classifier to detect positive instances. The x-axis represents the false positive rate (FPR),

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}, \tag{4.2.7}$$

which indicates the rate at which negative instances are mistakenly classified as positive. A well-performing model will have a ROC curve that approaches the top-left corner, indicating high TPR (correctly identifying positives) and low FPR (minimizing false alarms). Additionally, we compute the area under the ROC curve (AUC) as a performance metric for machine learning models (e.g., Bradley, 1997; Fawcett, 2006). The AUC quantifies the probability that a classifier will correctly rank a randomly selected positive instance higher than a randomly selected negative instance. This measure reflects the ability of the model to distinguish between positive and negative classes, indicating its separability and discriminative power.

We also compute various classification metrics to provide a more detailed evaluation of the effectiveness of the classifiers. Accuracy measures the proportion of correct predictions, considering both TP and TN, relative to the total number of elements in the sample, offering a general overview of the performance. Precision focuses on the correctness of predicting positive instances, indicating the proportion of TP out of all elements predicted as positive. Precision is useful when minimizing FP is crucial. Recall or TPR, as we mentioned, evaluates the model's ability to

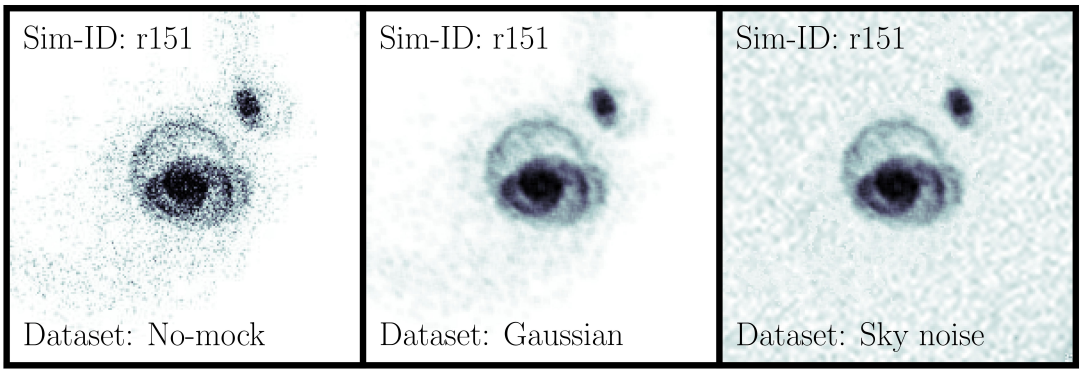

Figure 4.3.1: Preprocessing performed on images obtained from galaxy merger simulations. (*left*) Control images with no additions. (*center*) Convoluted images with a Gaussian point spread function (PSF) with a full width at half maximum (FWHM) of 1" and a pixel scale of 0.262", reproducing the scale of DECaLS-GZ5 (Walmsley et al., 2022). (*right*) Convoluted images with a Gaussian PSF together with the addition of random cutoff of the sky noise present in DECaLS-GZ5 imaging. For visualization purposes, the image shown has not been rescaled or cropped from its original size.

correctly identify positive instances, representing the proportion of TP among the actual positive instances in the dataset. Recall becomes particularly relevant when FN carries significant consequences. Lastly, the F1-score is the harmonic mean of precision and recall, providing a balanced metric when both measures are equally important. It is especially useful for imbalanced datasets, where accuracy alone might offer a misleading assessment.

4.3 Experiments

In addition to comparing ML models in galaxy merger classification, we created different levels of observational realism to evaluate model performance based on the type of images used during training. Three noise levels were defined for the synthetic galaxies in both the training set and test set: (1) *No-mock*: highly idealized images without any additional treatment beyond what is explained in Sec. 3.1, used as a control; (2) *Gaussian*: images convolved with a Gaussian point spread function (PSF) matching that reported by DECaLS for GZ5 imaging (Walmsley et al., 2022); and (3) *Sky noise*: images convolved with the Gaussian PSF and supplemented with background noise slices randomly selected from GZ5. Bottrell et al. (2019a) demonstrated that the presence of neighboring sources during training is crucial for the success of their DL model. However, Ferreira

et al. (2020) show that crowded sky regions negatively affect DL models. For the latter, point external sources in the background were removed by replacing the corresponding pixels with values drawn from a normal distribution, using the mean and standard deviation of the image. An example of the different noise levels applied to the images is shown in Fig. 4.3.1.

ML models often experience degradation in performance when applied to datasets derived from distributions different from those used during training, a problem commonly referred to as domain shift (e.g., Saenko et al., 2010; Sun & Saenko, 2016). Furthermore, morphological features in galaxy mergers exhibit high sensitivity to the available instrumental resolution (e.g., Bottrell et al., 2019b; Rose et al., 2023). To assess both phenomena in our study, we employ each synthetic dataset with varying noise levels to train the ML models proposed by Modulos-AutoML, which are subsequently tested on images from each noise level category[6]. Finally, the model demonstrating the best performance on the Sky noise dataset (i.e., the dataset with the highest level of observational realism) is applied to the observational sample collected from GZ5.

[6]Since there are three machine learning algorithms, each trained on three distinct training sets, this results in a total of nine classifiers. Each classifier is tested on three different test sets.

Chapter 5

Results

We run the experiments described in Sec. 4.2 as follows: We select the input dataset from the list of noise levels described in Sec. 4.3, run one AutoML workflow per ML model available for classification (i.e., three workflows per dataset) with the finalization criterion that if there is no further improvement in the AUC score (macro) using the respective validation split after 250 candidates, the workflow terminates. We select the best candidate for each workflow as our optimal classifier and subsequently evaluate it with the different test sets available (for more information about the selected classifiers and full results, see Appendix A). The results of the classifiers on different test sets can be seen in Table 5.0.1, Table 5.0.2, and Table 5.0.3 for models trained with No-mock, Gaussian, and Sky-noise datasets respectively, where we summarize different metrics besides AUC for completeness. A visual comparison of the experiments performed in this study is shown in Fig. 5.0.1.

As shown in Table 5.0.1, when evaluating the ML models trained on the No-mock dataset with data of the same type, the CNN emerges as the best classifier, achieving an AUC of 97.2% and an accuracy of 86.7%. XGBoost and RF yield comparable results, but with a reduction of $\sim 3\%$ and $\sim 8\%$, respectively. This order in the effectiveness of the models is maintained when evaluating the Gaussian dataset; however, the CNN experiences an $\sim 8\%$ decrease in accuracy, which is reflected in its predictions, as it suffers from difficulty in distinguishing isolated galaxies from the other classes. XGBoost shows a reduction in accuracy of $\sim 4\%$, increasing the confusion between isolated and post-merger galaxies. RF

Table 5.0.1: Summary of the test stage for the different models trained with the No-mock dataset. In addition to AUC (macro-average), we include additional metrics to evaluate the results: accuracy, f1-score, recall, and precision. All listed metrics are maximized at 1 (i.e., perfect predictions). Note that the best-performing ML method is bold faced.

ML-Model	AUC	Accuracy	F1-Score	Recall	Precision
Test: No-mock					
RF	0.911	0.769	0.766	0.769	0.768
XGBoost	0.942	0.828	0.826	0.827	0.826
CNN-torch	**0.972**	**0.867**	**0.866**	**0.867**	**0.867**
Test: Gaussian					
RF	0.930	0.770	0.763	0.769	0.770
XGBoost	0.943	0.784	0.777	0.783	0.800
CNN-torch	**0.960**	**0.793**	**0.792**	**0.793**	**0.835**
Test: Sky noise					
RF	0.779	0.481	0.400	0.481	0.362
XGBoost	**0.844**	**0.634**	**0.640**	**0.634**	**0.681**
CNN-torch	0.842	0.615	0.607	0.615	0.697

shows no significant changes, maintaining similar metrics across both datasets. The evaluation of images with added background noise leads to a significant deterioration in the performance of all ML models, with a decrease in accuracy of $\sim 24\%$. RF is the worst-performing model, achieving an AUC of 77.9% and an accuracy of 48.1%. Its predictions are heavily biased toward post-mergers, with a precision of 48.1%, indicating a high rate of false positives, exceeding the number of true positives, as observed by the 36.2% in recall. The CNN follows with an AUC of 84.2% and an accuracy of 61.5%. It also struggles particularly with misclassifications in post-mergers, and shows a notable decline in its ability to identify galaxy pairs. XGBoost demonstrates the smallest reduction in performance, with an AUC of 84.4% and an accuracy of 63.4%. Although it shares with the other models a tendency to overpredict post-merger cases, it retains the ability to differentiate between classes, as reflected in its metrics, which remain relatively balanced.

ML models trained on highly idealized images experience a significant decline in their inference capabilities when faced with noisy data distributions. The added Gaussian PSF immediately makes it more difficult to distinguish isolated galaxies from post-mergers, which is expected, as the morphological perturbations become less pronounced compared to images with perfect resolution. Background

Table 5.0.2: Summary of the test stage for the different models trained with the Gaussian dataset. In addition to AUC (macro-average), we include additional metrics to evaluate the results: accuracy, f1-score, recall, and precision. All listed metrics are maximized at 1 (i.e., perfect predictions). Note that the best-performing ML method is bold faced.

ML-Model	AUC	Accuracy	F1-Score	Recall	Precision
Test: No-mock					
RF	0.940	0.736	0.721	0.737	0.780
XGBoost	**0.954**	**0.756**	0.744	**0.759**	**0.818**
CNN-torch	0.939	0.752	**0.745**	0.745	0.789
Test: Gaussian					
RF	0.960	0.851	0.850	0.851	0.850
XGBoost	**0.971**	**0.868**	**0.868**	**0.868**	**0.868**
CNN-torch	0.959	0.860	0.860	0.860	0.859
Test: Sky noise					
RF	0.868	0.463	0.374	0.464	0.620
XGBoost	**0.891**	**0.673**	**0.672**	**0.673**	**0.764**
CNN-torch	0.850	0.598	0.600	0.598	0.717

noise further obscures these perturbations, making even galaxy pairs nearly indistinguishable from other classes for both RF and CNN. Notably, XGBoost can mitigate this degradation more effectively than CNN and RF.

According to Table 5.0.2, all ML models trained with images of the Gaussian dataset perform well when evaluated with images from the same type, even improving the overall metrics scores compared with the No-mock trained classifiers. XGBoost takes a slight lead over RF and CNN with 97.1% in AUC and 86.8% in accuracy. The rest of the metrics remain in balance for all three algorithms. When evaluated with No-mock images, all algorithms experience a decrease in accuracy of $\sim 11\%$, maintaining the same order of effectiveness as before. Additionally, the three ML models favor higher precision over recall, meaning they avoid false positives at the expense of missing real positive cases. This bias is reflected in the post-merger class, where predictions favor this class over isolated galaxies. RF and CNN classify 20.3% and 22.1% of No-mock post-mergers as galaxy pairs, while XGBoost only 8.0%. For the Sky noise dataset, the worst-performing classifier is RF, with an AUC of 86.8% and an accuracy of 46.3%, followed by CNN, with an AUC of 85.0% and an accuracy of 59.8%. Finally, XGBoost achieves an AUC of 89.1% and an accuracy of 67.3%. Similar to the predictions on the No-mock image dataset, all three models prioritize precision over recall, favoring the prediction

Table 5.0.3: Summary of the test stage for the different models trained with the Sky noise dataset. In addition to AUC (macro-average), we include additional metrics to evaluate the results: accuracy, f1-score, recall, and precision. All listed metrics are maximized at 1 (i.e., perfect predictions). Note that the best-performing ML method is bold faced.

ML-Model	AUC	Accuracy	F1-Score	Recall	Precision
Test: No-mock					
RF	**0.890**	**0.736**	**0.732**	**0.736**	**0.763**
XGBoost	0.880	0.690	0.694	0.690	0.711
CNN-torch	0.841	0.467	0.394	0.468	0.540
Test: Gaussian					
RF	**0.970**	**0.888**	**0.887**	**0.887**	**0.889**
XGBoost	0.964	0.803	0.802	0.802	0.834
CNN-torch	0.968	0.848	0.847	0.848	0.851
Test: Sky noise					
RF	0.930	0.791	0.788	0.791	0.795
XGBoost	0.935	0.796	0.792	0.796	0.795
CNN-torch	**0.952**	**0.820**	**0.818**	**0.820**	**0.821**

of post-mergers over isolated galaxies. RF experiences a significant decline in identifying galaxy pairs, correctly predicting only 2% of the total galaxy pairs in the Sky noise test set. CNN and XGBoost retain some ability to differentiate between the stages of merging pairs, with 43.7% and 67.1% correct predictions from the total galaxy pair population, respectively.

ML models trained with images distorted by our simulated PSF exhibit diverse behaviors when evaluated on images from different distributions. A slight deterioration is observed in all three models when evaluated on highly idealized images, with all three maintaining similar results. However, when evaluated on the dataset with background noise, RF and CNN perform similarly to how they did when trained on No-mock images, while XGBoost shows an improvement of approximately 4% in accuracy. No significant differences are observed between the Gaussian and No-mock training sets when predicting galaxies in the Sky noise test set. Nevertheless, RF and CNN continue to show a tendency to lose their ability to recognize galaxy pairs (arguably the most distinctive class among the three under study).

The ML models trained on the Sky noise dataset, as shown in Table 5.0.3, achieve the best results on the Sky noise test set across the three experiments. As such, we show their ROC curves in Fig. 5.0.2 and their normalized confusion matrix

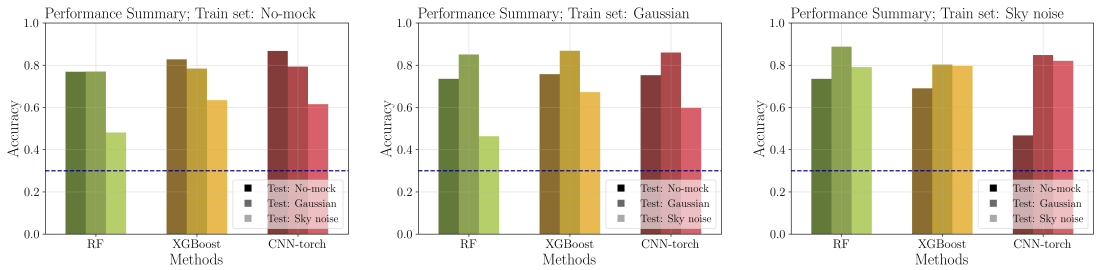

Figure 5.0.1: Accuracy of the three experiments performed in this study across all test sets. (*left*) ML models trained with No-mock images. (*center*) ML models trained with Gaussian images. (*right*) ML models trained with Sky noise images. The opacity of the colors highlights the implemented test set. The dashed blue line represents the accuracy score of a random classifier.

in Fig. 5.0.3. CNN reaches 95.2% AUC and 82% accuracy. XGBoost obtains 93.5% AUC and 79.6% accuracy, while RF achieves 93% AUC and 79.1% accuracy. The remaining metrics for all three models maintain similar values, indicating a balanced performance in their classifications. The evaluation of the No-mock test set yields positive results for RF, achieving 89% AUC and 73.6% accuracy. XGBoost shows a small decline compared to its versions trained on No-mock and Gaussian images, obtaining 88.8% AUC and 69% accuracy. CNN has the worst performance among the three ML models, with 84.1% AUC and 46.7% accuracy. The latter is unable to distinguish between mergers and non-mergers, suffering from a bias toward the galaxy pairs category. In contrast, all three ML models perform better, to varying degrees, on the Gaussian test set compared to the Sky noise set. RF is the best-performing model, achieving 97% AUC and 88.8% accuracy. CNN achieves 96.8% AUC and 82% accuracy, while XGBoost shows the smallest improvement, with 96.4% AUC and 80.3% accuracy.

All three ML models trained on the Sky noise dataset perform well when evaluated on images of the same type, successfully distinguishing the three classes. However, they struggle with the post-merger class as in previous experiments. Taking a deeper look into the predictions on the Sky noise test set, we observe that all models have difficulties with images of post-mergers extracted from simulations with a mass ratio of $\mu = 1/4$ (i.e., r15 simulations in Table 2.4.1). In these cases, the morphological perturbations are subtler and fade more quickly than in other cases, making them resemble isolated galaxies (see Fig. 5.0.4). Overall, CNN shows a slight advantage over RF and XGBoost. Yet, CNN suffers a 35.3% drop in accuracy when inferring on the No-mock image set. Decision tree-based models

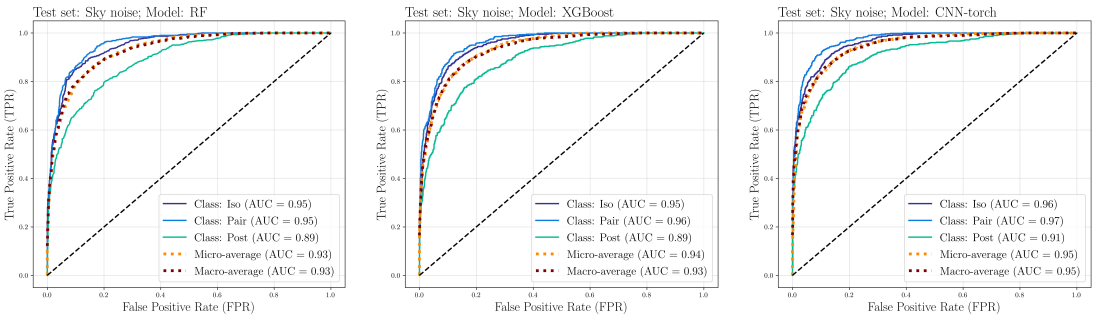

Figure 5.0.2: ROC curves of best-performing ML models in the Sky noise test set. (*left*) RF, trained with Sky noise imaging. (*center*) XGBoost, trained with Sky noise imaging. (*right*) CNN, trained with Sky noise imaging. We compute the macro-average, micro-average, and one-class vs. rest cases. We show the AUC score for each case respectively.

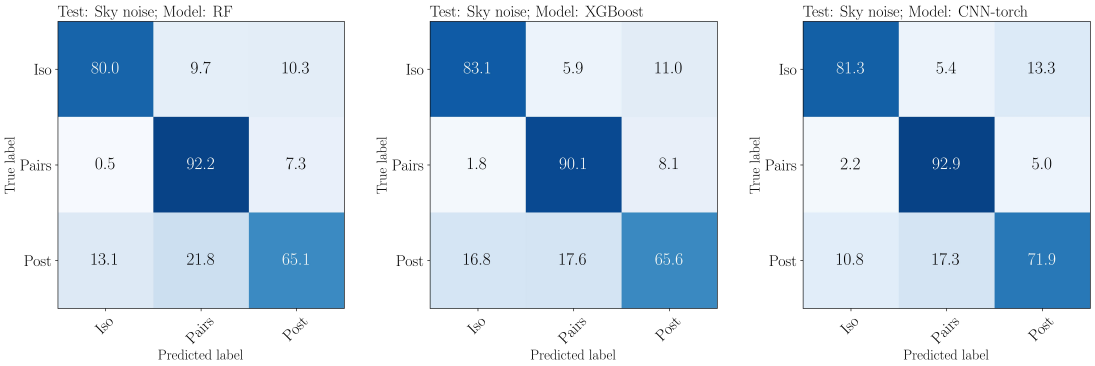

Figure 5.0.3: Normalized confusion matrixes of best-performing ML models in the Sky noise test set. (*left*) RF, trained with Sky noise imaging. (*center*) XGBoost, trained with Sky noise imaging. (*right*) CNN, trained with Sky noise imaging.

experience a more gradual decline in effectiveness when transitioning to highly idealized images. Additionally, all three models improve their performance on the Gaussian test set, likely due to the background noise in the training dataset, which acts as a form of regularization. This makes the learning process more challenging during training but enhances the performance during inference.

The selection of the most optimal classifier for evaluation with the observational dataset could be limited to choosing the ML model with the best performance metrics on the Sky noise test set. However, our results indicate the following: While CNNs tend to achieve the best results on metrics with data derived from distributions similar to those used during training, they experience a significant decline in performance when making predictions on out-of-distribution data. This decline is also observed, to varying degrees, with RF. Only XGBoost smooths the transition from one dataset to another with similar characteristics, maintaining a

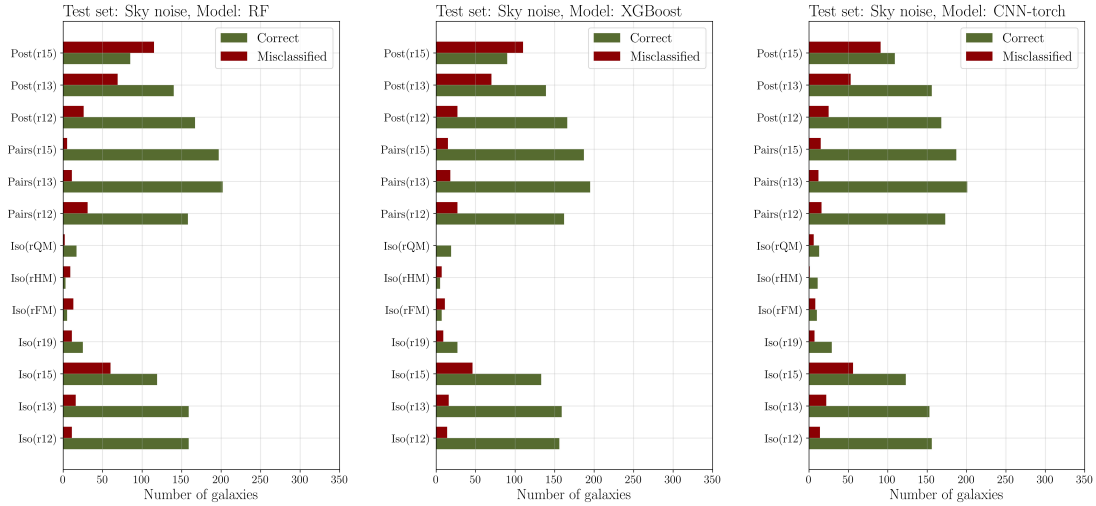

Figure 5.0.4: Analysis of the predictions made by the best-performing ML models in the Sky noise test set. (*left*) RF, trained with Sky noise imaging. (*center*) XGBoost, trained with Sky noise imaging. (*right*) CNN, trained with Sky noise imaging.

clear distinction among the three classes analyzed in this study. Given that the results on the Sky noise test set only differ by $\sim 3\%$, XGBoost trained on Sky noise will be used as our selected classifier for subsequent analysis. As an example, Fig. 5.0.5 shows correct predictions and their associated probabilities. Similarly, Fig. 5.0.6 presents examples of incorrect predictions for all possible cases.

We applied our selected classifier to the sample of massive galaxies extracted from GZ5. A random sample of the images and their respective predictions can be seen in Fig. 5.0.7. A visual inspection of the predictions made by XGBoost reveal two main findings: First, point sources in the background noise of the images negatively impact our predictions. Sources located at a short distance from the main system are misidentified by XGBoost as part of the observed galaxy (or galaxies), introducing uncertainty during inference. Second, the primary difference between the simulations used to train the classifier and the real galaxies from GZ5 lies in the behavior and distribution of the gas. Since our simulations lack a hydrodynamic treatment, they cannot accurately represent the gas distributions that arise after a merger event, which negatively affects the performance of the classifier. Regions with a disc exhibiting pronounced spiral arms or a bulge surrounded by a broad halo are mistakenly identified as morphological features of a post-merger galaxy rather than an isolated galaxy. However, XGBoost correctly identified strong morphological features like bridges and tails with high certain.

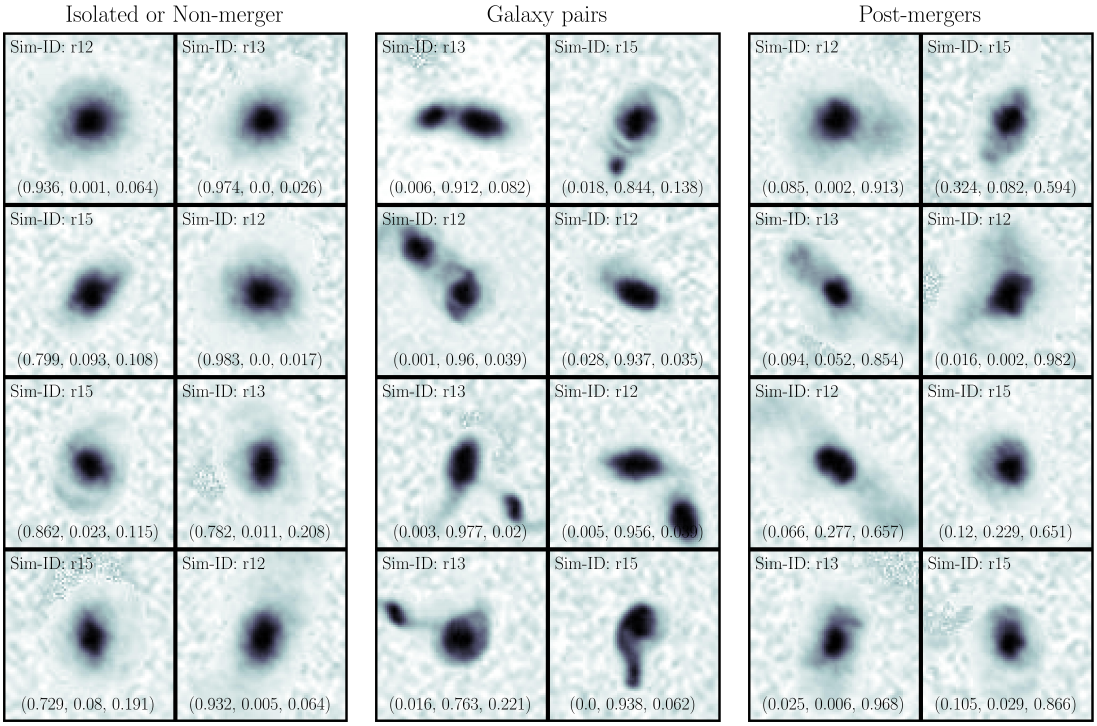

Figure 5.0.5: Examples from the Sky noise test set that were correctly classified using XGBoost (trained on the Sky noise dataset) are presented. The simulation from which each image was extracted is indicated, along with the probabilities provided by XGBoost, sorted as *(p(iso), p(pair), p(post))*.

For further analysis, we compute the fraction of galaxy pairs, post-mergers, and mergers in general in our data sample and contrast it with the fractions reported by Walmsley et al. (2022) in the GZ5 catalog (see Fig. 5.0.8). We only consider those galaxies with class probabilities $p(\text{class}) \geq 0.85$. For GZ5 fraction votes, we consider as galaxy pairs galaxies with $f_{\text{merger}} \geq 0.7$, and as post-mergers galaxies with $f_{\text{major-dist}} \geq 0.7$ or $f_{\text{minor-dist}} \geq 0.7$. We use the probabilities given by XGBoost and the fractions of GZ5 to compute multiclass binomial errors for both cases (Cameron, 2011). We include merger fractions reported by Conselice et al. (2009), computed for galaxies in the Cosmic Evolution Survey (COSMOS) derived with CAS selection criteria, and Casteels et al. (2014) for galaxies in the Galaxy And Mass Assembly (GAMA) survey derived with CAS selection criteria combined with close-pair detection. Finally, we calculate merger fractions and errors according to the best fits found for massive galaxies in the EAGLE simulation (Qu et al., 2017) and CANDELS (Duncan et al., 2019; Ferreira et al., 2020) for a redshift $z \leq 0.15$.

We observe that the galaxy merger fraction obtained in this study falls within the limits provided by the EAGLE simulation for the selected redshift range and with the value reported by Casteels et al. (2014) for their sample in the GAMA survey. Additionally, the number of galaxy pairs identified by XGBoost closely aligns with the values reported by GZ5. Also, our propagation of binomial errors associated with the probabilities of each class is lesser than those associated with fraction votes of GZ5. The merger fraction reported by Casteels et al. (2014), obtained through CAS selection criteria, displays low errors, similar to our value. However, it may be premature to attribute this solely to the robustness of the CAS parameters, as their study includes a smaller sample of galaxies (1,470 galaxies from GAMA) compared to other studies.In fact, in Conselice (2003), which examines a sample of approximately 20,000 galaxies, the errors closely resemble those reported by Duncan et al. (2019), Ferreira et al. (2020), and Walmsley et al. (2022). However, our merger fraction is higher than those reported by Conselice (2003), Duncan et al. (2019), and Ferreira et al. (2020). This overestimation can be attributed to the number of post-mergers identified by our ML model, which is six times the amount reported by the GZ5 catalog. As previously discussed, our algorithm encounters difficulties when processing images of galaxies with extensive gas distributions. In the experiments with synthetic data and in the visual inspection of predictions for our DECaLS dataset, XGBoost tends to classify galaxies with broad gas distributions as post-mergers, even when they would typically be considered isolated galaxies.

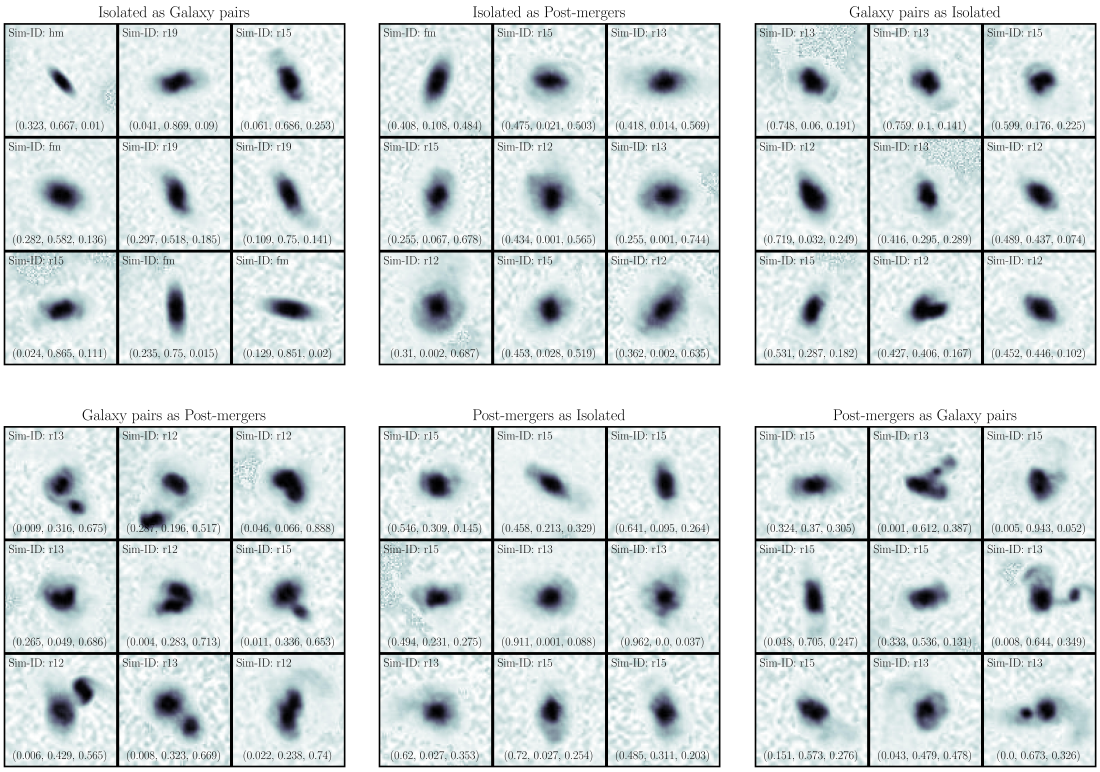

Figure 5.0.6: Examples from the Sky noise test set that were misclassified in all possible cases using XGBoost (trained on the Sky noise dataset) are presented. The simulation from which each image was extracted is indicated, along with the probabilities provided by XGBoost, sorted as *(p(iso), p(pair), p(post))*.

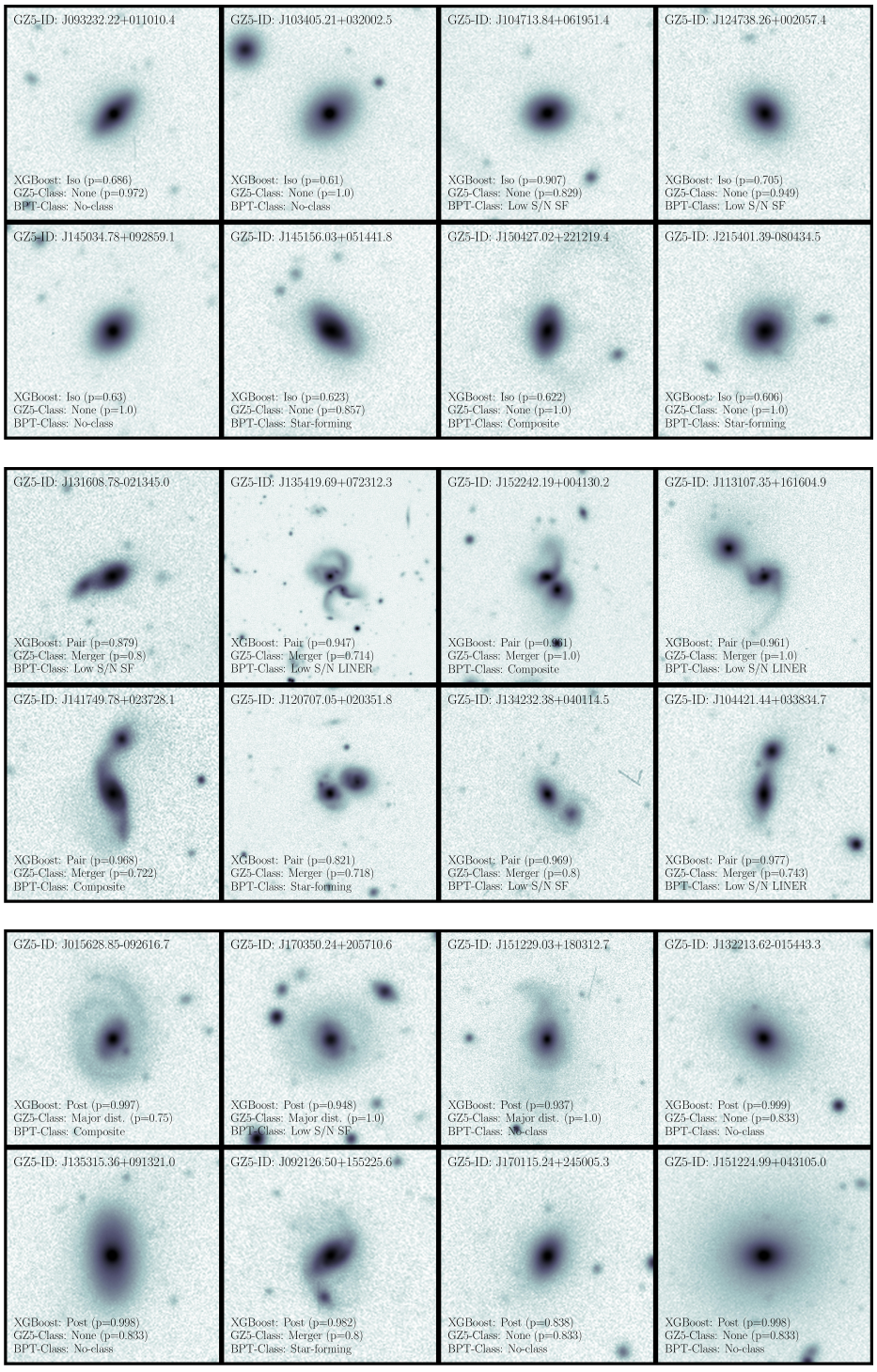

Figure 5.0.7: Examples of XGBoost predictions for massive galaxies extracted from GZ5 are presented. (*top*) Isolated galaxies or non-mergers. (*center*) Galaxy pairs. (*bottom*) Post-mergers. We display the NSA ID for each galaxy, the probability of the respective class given by XGBoost, the highest merger-fraction vote from the GZ5 catalogue, and the galaxy type according to the MPA-JHU. For visualization purposes, we show the original images of the galaxies (i.e., without rescaling or cropping).

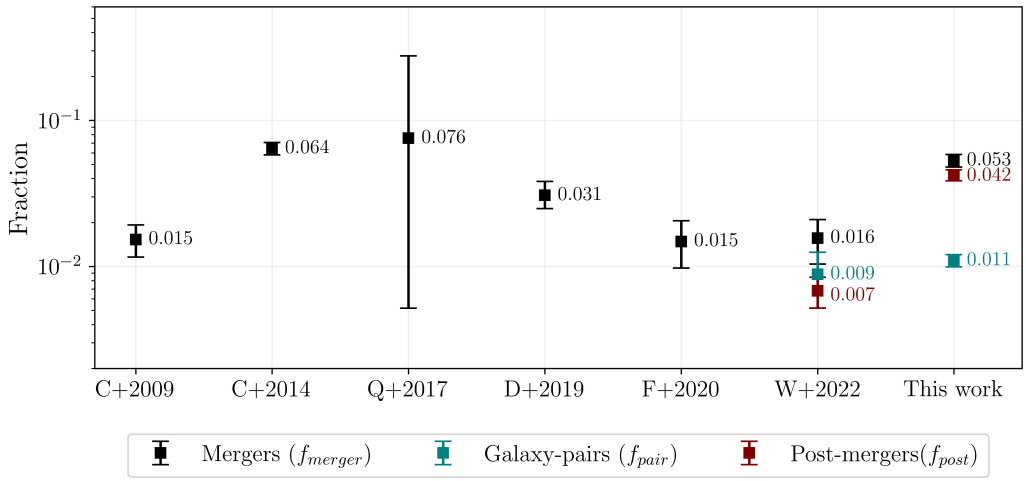

Figure 5.0.8: Comparison of literature values for the fraction of galaxy mergers up to redshift $z \leq 0.15$. We include values based on the reported fits and errors for massive galaxies from the following sources: Measurements of galaxies in the Cosmic Evolution Survey (COSMOS) derived with *CAS* selection criteria (C+2009; Conselice et al., 2009), measurements of galaxies in the Galaxy And Mass Assembly (GAMA) survey derived with *CAS* selection criteria combined with close-pair detection, (C+2014; Casteels et al., 2014), the EAGLE simulation (Q+2017; Qu et al., 2017), photometric detections from CANDELS (D+2019; Duncan et al., 2019), and deep learning predictions on CANDELS (F+2020; Ferreira et al., 2020). We compute the fraction of galaxy pairs and post-mergers based on galaxies with high-fidelity fraction votes from the GZ5 catalogue (i.e., fractions $f_{\mathrm{GZ5\,class}} \geq 0.7$; W+2022; Walmsley et al., 2022), summing these to obtain the merger fraction according to the GZ5 catalogue. For our own values, we count galaxies with high-probability predictions for the respective merger stage ($p(\mathrm{class}) \geq 0.85$), and then sum them to obtain the total merger fraction. Errors for W+2022 and our work correspond to 1σ multiclass binomial errors (Cameron, 2011).

Chapter 6

Discussion and conclusions

6.1 Discussion

Regarding the objectives of this investigation, two central questions must be addressed to fully characterize our results. The first concerns the synthetic dataset implemented for training, and the second pertains to the ML models.

6.1.1 *Observational realism or detailed physics?*

We revisit this question asked by Bottrell et al. (2019a). Their results stated that observational realism is more important than the detailed physical treatment of the images obtained in a hydrodynamic simulation (such as radiative transfer or maps of stellar mass) when training a CNN for galaxy merger classification. Our results extend this statement one step further: ML models trained with images obtained from N-body simulations can recognize galaxy pairs in a merger event relying solely on morphological features if instrumental effects and background sky noise are taken into account. At the same time, we found a ceiling, as morphological features alone canot completely separate isolated galaxies from post-mergers.

The performance issues of our best classifier on the sample of observational data can be attributed to three aspects related to the training data: First, our galactic models do not include internal structures; we work with perfect discs, which evolve as the merger process progresses. Structures such as bars and spirals could introduce a greater diversity of galaxies than currently addressed (Barnes, 2016), thus achieving better prediction performance. Second, in agreement

with Bottrell et al. (2019a), our results indicate that the absence of external sources or crowded regions during training negatively affects our classifier when inferring on real galaxies. This contrasts with the findings of Ferreira et al. (2020); however, we attribute this to differences between the images from CANDELS and DECaLS, particularly in image size and the resolution of the observation cameras. Galaxies surrounded by external sources are uncommon in the centered images from CANDELS. Third, gas distribution: Although our N-body treatment is computationally efficient, its treatment of the gas is very approximated. This limitation presents a significant drawback for our classifier, as XGBoost struggles to differentiate between the extended gas halos of elliptical galaxies and major disturbances present in post-mergers.

6.1.2 *Traditional ML models or higher model complexity?*

Computer sciences have an arguably faster advance than other fields of natural science. Classical models are frequently replaced by new algorithms that are more complex[1] and, in theory, offer better performance. However, model complexity often goes in hand with a higher risk of overfitting, an increase in computational costs, and the need of bigger training datasets to achieve better result (e.g., Hu et al., 2021).

Similar to Cheng et al. (2020), we tested various traditional ML algorithms for our classification task and found that CNNs yielded the best performance metrics when applied to data extracted from the same distribution used for training. However, when presented with data from different distributions, CNNs exhibited a notable decline in performance, demonstrating their inability to generalize effectively. We note that the vanilla CNN architecture proposed by Modulos-AutoML is relatively small in size compared to similar research (e.g., Bottrell et al., 2019a; Pearson et al., 2019; Ferreira et al., 2020, 2022). However, we do not find evidence to suggest a different behavior with a bigger architecture than that observed in our results. In DL, it is a regular practice to use techniques such as fine-tuning[2]

[1]Model complexity refers to factors such as model type, model size, optimization processes, and data complexity in ML or DL algorithms. These factors influence performance, computational costs, and the reliability of predictions.

[2]Fine-tuning refers to the process of taking a pre-trained model and adapting it to a specific task by continuing training on a new dataset. This typically involves using a model that has been previously trained on a large dataset and then adjusting the weights of the model by training it on a smaller, task-specific dataset.

(Yosinski et al., 2014) to facilitate transfer learning[3] (e.g., Weiss et al., 2016), but in a supervised learning regime, this requires having sufficient labeled data from the distribution of interest.

Rose et al. (2023) have already demonstrated the capability of RF in differentiating galaxy mergers using structural parameters of galaxies as training data. Our results support this finding, showing that traditional ML models can achieve robust performance when provided with the appropriate conditions. XGBoost emerges as our overall best classifier. The complexity penalization and the construction of each tree, accounting for errors during XGBoost training, create a simple yet effective classifier that performs well with image input-data outside the training distribution.

Model complexity should be handled with care, with generalization as a key objective. Ensemble models are robust ML algorithms capable of capturing highly non-linear behavior, but DL models can leverage hierarchical feature extraction and learn complex representations directly from raw data, usually achieving better results than traditional ML models. Ferreira et al. (2024) presented a methodology that combines traditional ML models with a higher level of DL model complexity: the integration of an attention-based model (Vaswani et al., 2017) for image analysis and encoding of the information (i.e., visual transformers, Dosovitskiy et al., 2021), followed by the use of ensemble decision tree models for galaxy merger classification, achieving high performance when applying their model to galaxies in the SDSS DR7, demonstrating an interesting prospect for galaxy merger classification.

6.2 Conclusions

Galaxy mergers are a fundamental aspect of galaxy formation and evolution. However, traditional detection methods lack the effectiveness and efficiency required to handle large datasets. In this thesis, we compared different machine learning (ML) models for classifying major galaxy mergers and their merger stages relying solely on morphological image-based information. Additionally, we analyzed the effects of noise levels added to the images during the training and

[3]Transfer learning refers to the practice of leveraging knowledge gained from training a model on one task to improve performance on a different but related task

testing of the models.

To create a dataset large enough to train the classifiers in a supervised manner without risking overfitting, we used IDENTIKIT simulations (Barnes & Hibbard, 2009; Barnes, 2011a) along with the ZENO programming framework (Barnes, 2011b) to obtain a comprehensive library of galactic encounters. We successfully created four stable galactic models with different masses and 13 unique galaxy-galaxy interaction systems. Due to the principal feature of IDENTIKIT simulations, the simultaneous calculation of all possible collision angles between galaxies, the snapshots used to create the synthetic dataset represent 169 different collision events. We tracked the kinematics of the inner particles of the galaxies in all our simulations to define their merger stage (e.g., Moreno et al., 2019; Bottrell et al., 2019a). For further evaluation, we retrieved a sample of ∼30000 massive, low-redshift ($z \leq 0.15$) galaxies from DECaLS-GZ5 (Walmsley et al., 2022) with their spectroscopic measurements according to the MPA-JHU catalogs.

Model selection, training, and hyperparameter optimization were pursued through the Modulos-AutoML platform. This allowed us to compare the effectiveness of classifiers such as random forest (RF; Breiman, 2001), extreme gradient boosting (XGBoost; Chen & Guestrin, 2016), and convolutional neural networks (CNNs; Fukushima, 1980; Lecun et al., 1998). Using our synthetic dataset, we defined three noise levels for training and testing the available models: (1) *No mock*: highly idealized images, without any alteration beyond rescaling to the input size of the models. (2) *Gaussian*: images altered by a Gaussian point spread function (PSF), which simulates the observational properties performed by DECam (Flaugher et al., 2015), present in the DECaLS-GZ5 images. (3) *Sky noise*: images altered by the Gaussian PSF and with added slices of background observational noise randomly selected from DECaLS-GZ5 imaging. We trained all available ML models using data from the three noise level training partitions, thus producing nine classifiers. These classifiers were further tested on the three noise level test partitions to evaluate their performance under different noise conditions.

We found that ML models with the best performance on the Sky noise test set (i.e., the synthetic test set with the highest observational realism) are those trained on data from the same distribution. CNN achieved 95.2% AUC (macro), while XGBoost and RF obtained 93.5% and 93.0%, respectively. Training with No-mock or Gaussian images results in poor predictions on the Sky noise test

set. No significant differences were found between the two training regimes when evaluating on Sky noise. However, experiments with models trained on the No-mock and Gaussian sets showed that both CNN and RF experience significant performance degradation as the noise level in the images during inference increases. Both classifiers struggle to differentiate isolated galaxies from post-mergers and overpredict the latter when facing data from outer distributions (i.e., images extracted from distributions different from those used during the ML model training). Although XGBoost is also affected by the domain shift, it shows the least drastic changes among the three algorithms, maintaining its ability to distinguish all categories.

Therefore, XGBoost trained on Sky noise was selected as the classifier to evaluate the sample of galaxies extracted from DECaLS-GZ5. A visual inspection of the predictions made by XGBoost revealed its ability to recognize the morphological characteristics of mergers with a high degree of certainty. However, the presence of external sources or crowded regions in the images negatively affects the classifier, as it confuses these elements with parts of the galaxy (or galaxies) under study. Additionally, XGBoost tends to classify galaxies with strong morphological features (such as bars or spirals) or extended gas distributions (such as massive elliptical galaxies) as ongoing post-mergers. We further confirm these assumptions by calculating our galaxy pair, post-merger, and merger fractions and comparing them to those expected according to the GZ5 fraction votes, values reported from galaxy mergers selected through a CAS criteria (Conselice et al., 2009; Casteels et al., 2014), the best fits obtained for massive galaxies on CANDELS imaging (Duncan et al., 2019; Ferreira et al., 2020), and the EAGLE simulation (Qu et al., 2017) up to a $z \leq 0.15$. Our results fall within the range of mergers expected from the EAGLE simulation and show a considerable improvement in the certainty of the classifications. Additionally, our galaxy pair fraction falls close to that reported by GZ5. However, we note an increase in the merger fraction when compared with observational studies, something explained by an overpopulation of post-mergers predicted by our ML model, as our post-merger fraction is six times greater than the one reported by the GZ5 catalog.

Our results indicate that the morphological features present in a galaxy merger event are a solid basis for training an ML model for classification. However, this must be complemented with a physical treatment capable of reproducing the

internal structure of galaxies and the gas distributions resulting from a collision. This sustains the results of Rose et al. (2023) on the importance of the structural asymmetry parameter (A/A_0) to correctly identify galaxy mergers at low redshift using an RF trained on multiple structural parameters. Our experiments agreed with prior research (e.g., Bottrell et al., 2019a; Pearson et al., 2019; Ferreira et al., 2020) that all simulation-based ML models must consider instrumental and observational errors specific to the survey under study to maximize their performance. Finally, the inclusion of galaxies with internal structures such as bars and spirals seems to be a key addition to distinguish morphological features produced by a recent merger event from those features present in an isolated galaxy.

Bibliography

Abazajian, K. N., Adelman-McCarthy, J. K., Agüeros, M. A., et al. 2009, ApJS, 182, 543

Andrae, R., Rix, H.-W., & Chandra, V. 2023, ApJS, 267, 8

Baldwin, J. A., Phillips, M. M., & Terlevich, R. 1981, PASP, 93, 5

Barnes, J. & Hut, P. 1986, Nature, 324, 446

Barnes, J. E. 1988, ApJ, 331, 699

Barnes, J. E. 1990, Journal of Computational Physics, 87, 161

Barnes, J. E. 2011a, MNRAS, 413, 2860

Barnes, J. E. 2011b, ZENO: N-body and SPH Simulation Codes, Astrophysics Source Code Library, record ascl:1102.027

Barnes, J. E. 2012, MNRAS, 425, 1104

Barnes, J. E. 2016, MNRAS, 455, 1957

Barnes, J. E. & Hibbard, J. E. 2009, AJ, 137, 3071

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. 2011, in Advances in Neural Information Processing Systems, ed. J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, & K. Weinberger, Vol. 24 (Curran Associates, Inc.)

Bethapudi, S. & Desai, S. 2018, Astronomy and Computing, 23, 15

Bottrell, C., Hani, M. H., Teimoorinia, H., et al. 2019a, MNRAS, 490, 5390

Bottrell, C., Simard, L., Mendel, J. T., & Ellison, S. L. 2019b, MNRAS, 486, 390

Bradley, A. P. 1997, Pattern Recognition, 30, 1145

Breiman, L. 2001, Machine Learning, 45, 5

Brinchmann, J., Charlot, S., White, S. D. M., et al. 2004, MNRAS, 351, 1151

Buda, M., Maki, A., & Mazurowski, M. A. 2018, Neural Networks, 106, 249

Cabrera-Vives, G., Miller, C. J., & Schneider, J. 2018, AJ, 156, 284

Cameron, E. 2011, Publications of the Astronomical Society of Australia, 28, 128–139

Casteels, K. R. V., Conselice, C. J., Bamford, S. P., et al. 2014, MNRAS, 445, 1157

Chen, T. & Guestrin, C. 2016, in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16 (New York, NY, USA: Association for Computing Machinery), 785–794

Cheng, T.-Y., Conselice, C. J., Aragón-Salamanca, A., et al. 2020, MNRAS, 493, 4209

Cisternas, M., Jahnke, K., Inskip, K. J., et al. 2011, ApJ, 726, 57

Conselice, C. J. 2003, ApJS, 147, 1

Conselice, C. J. 2014, ARA&A, 52, 291

Conselice, C. J., Yang, C., & Bluck, A. F. L. 2009, MNRAS, 394, 1956

Cox, T. J., Jonsson, P., Somerville, R. S., Primack, J. R., & Dekel, A. 2008, MNRAS, 384, 386

Darg, D. W., Kaviraj, S., Lintott, C. J., et al. 2010a, MNRAS, 401, 1552

Darg, D. W., Kaviraj, S., Lintott, C. J., et al. 2010b, MNRAS, 401, 1043

Dey, A., Schlegel, D. J., Lang, D., et al. 2019, AJ, 157, 168

Di Matteo, P., Combes, F., Melchior, A. L., & Semelin, B. 2007, A&A, 468, 61

Dieleman, S., Willett, K. W., & Dambre, J. 2015, MNRAS, 450, 1441

Domínguez Sánchez, H., Huertas-Company, M., Bernardi, M., Tuccillo, D., & Fischer, J. L. 2018, MNRAS, 476, 3661

Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. 2021, An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

Drlica-Wagner, A., Sevilla-Noarbe, I., Rykoff, E. S., et al. 2018, ApJS, 235, 33

Dubath, P., Rimoldini, L., Süveges, M., et al. 2011, MNRAS, 414, 2602

Duncan, K., Conselice, C. J., Mundy, C., et al. 2019, ApJ, 876, 110

Eddington, A. S. 1916, MNRAS, 76, 572

Ellison, S. L., Mendel, J. T., Patton, D. R., & Scudder, J. M. 2013, MNRAS, 435, 3627

Ellison, S. L., Patton, D. R., Simard, L., & McConnachie, A. W. 2008, AJ, 135, 1877

Ellison, S. L., Viswanathan, A., Patton, D. R., et al. 2019, MNRAS, 487, 2491

Farouki, R. T. & Shapiro, S. L. 1982, ApJ, 259, 103

Fawcett, T. 2006, Pattern Recognition Letters, 27, 882, rOC Analysis in Pattern Recognition

Ferreira, L., Bickley, R. W., Ellison, S. L., et al. 2024, MNRAS, 533, 2547

Ferreira, L., Conselice, C. J., Duncan, K., et al. 2020, ApJ, 895, 115

Ferreira, L., Conselice, C. J., Kuchner, U., & Tohill, C.-B. 2022, ApJ, 931, 34

Flaugher, B., Diehl, H. T., Honscheid, K., et al. 2015, AJ, 150, 150

Freeman, K. C. 1970, ApJ, 160, 811

Freeman, P. E., Izbicki, R., Lee, A. B., et al. 2013, MNRAS, 434, 282

Friedman, J. H. 2001, The Annals of Statistics, 29, 1189

Fukushima, K. 1980, Biological Cybernetics, 36, 193

Genel, S., Vogelsberger, M., Springel, V., et al. 2014, MNRAS, 445, 175

Grogin, N. A., Conselice, C. J., Chatzichristou, E., et al. 2005, ApJ, 627, L97

Grogin, N. A., Kocevski, D. D., Faber, S. M., et al. 2011, ApJS, 197, 35

He, X., Zhao, K., & Chu, X. 2021, Knowledge-Based Systems, 212, 106622

Heckman, T. M., Smith, E. P., Baum, S. A., et al. 1986, ApJ, 311, 526

Hernquist, L. 1990, ApJ, 356, 359

Hewlett, T., Villforth, C., Wild, V., et al. 2017, MNRAS, 470, 755

Ho, T. K. 1998, IEEE transactions on pattern analysis and machine intelligence, 20, 832

Hopkins, P. F. 2015, MNRAS, 450, 53

Hopkins, P. F. 2017, MNRAS, 466, 3387

Hopkins, P. F., Hernquist, L., Cox, T. J., et al. 2006a, ApJS, 163, 1

Hopkins, P. F., Hernquist, L., Cox, T. J., & Kereš, D. 2008, ApJS, 175, 356

Hopkins, P. F., Hernquist, L., Cox, T. J., Robertson, B., & Springel, V. 2006b, ApJS, 163, 50

Hopkins, P. F., Wetzel, A., Kereš, D., et al. 2018, MNRAS, 480, 800

Howard, S., Keel, W. C., Byrd, G., & Burkey, J. 1993, ApJ, 417, 502

Hu, X., Chu, L., Pei, J., Liu, W., & Bian, J. 2021, Knowledge and Information Systems, 63, 2585

Huertas-Company, M., Gravet, R., Cabrera-Vives, G., et al. 2015, ApJS, 221, 8

Ji, I., Peirani, S., & Yi, S. K. 2014, A&A, 566, A97

Kartaltepe, J. S., Mozena, M., Kocevski, D., et al. 2015, ApJS, 221, 11

Kartaltepe, J. S., Sanders, D. B., Scoville, N. Z., et al. 2007, ApJS, 172, 320

Kauffmann, G. & Haehnelt, M. 2000, MNRAS, 311, 576

Kauffmann, G., Heckman, T. M., White, S. D. M., et al. 2003, MNRAS, 341, 33

Khaled Fawagreh, M. M. G. & Elyan, E. 2014, Systems Science & Control Engineering, 2, 602

Khochfar, S. & Burkert, A. 2006, A&A, 445, 403

Kingma, D. P. & Ba, J. 2017, Adam: A Method for Stochastic Optimization

Klimontovich, Y. L. 1967, The Statistical Theory of Non-Equilibrium Processes in a Plasma, Vol. 9 (Elsevier)

Kocevski, D. D., Brightman, M., Nandra, K., et al. 2015, ApJ, 814, 104

Koekemoer, A. M., Faber, S. M., Ferguson, H. C., et al. 2011, ApJS, 197, 36

Kormendy, J. & Ho, L. C. 2013, ARA&A, 51, 511

Krizhevsky, A., Sutskever, I., & Hinton, G. E. 2012, in Advances in Neural Information Processing Systems, ed. F. Pereira, C. Burges, L. Bottou, & K. Weinberger, Vol. 25 (Curran Associates, Inc.)

Lambrides, E. L., Watts, D. J., Chiaberge, M., et al. 2021, ApJ, 919, 43

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. 1998, Proceedings of the IEEE, 86, 2278

Lin, L., Koo, D. C., Weiner, B. J., et al. 2007, ApJ, 660, L51

Lin, L., Koo, D. C., Willmer, C. N. A., et al. 2004, ApJ, 617, L9

Lintott, C. J., Schawinski, K., Slosar, A., et al. 2008, MNRAS, 389, 1179

Lotz, J. M., Jonsson, P., Cox, T. J., et al. 2011, ApJ, 742, 103

Lotz, J. M., Jonsson, P., Cox, T. J., & Primack, J. R. 2010, MNRAS, 404, 575

Lotz, J. M., Primack, J., & Madau, P. 2004, AJ, 128, 163

Lynds, R. & Toomre, A. 1976, ApJ, 209, 382

Mantha, K. B., McIntosh, D. H., Brennan, R., et al. 2018, MNRAS, 475, 1549

Medina-Rosales, E., Cabrera-Vives, G., & Miller, C. J. 2024, MNRAS, 531, 52

Mihos, J. C. & Hernquist, L. 1994, ApJ, 437, L47

Mihos, J. C. & Hernquist, L. 1996, ApJ, 464, 641

Moreno, J., Torrey, P., Ellison, S. L., et al. 2019, MNRAS, 485, 1320

Mortazavi, S. A., Lotz, J. M., Barnes, J. E., Privon, G. C., & Snyder, G. F. 2018, MNRAS, 474, 3423

Mortazavi, S. A., Lotz, J. M., Barnes, J. E., & Snyder, G. F. 2016, MNRAS, 455, 3058

Mundy, C. J., Conselice, C. J., Duncan, K. J., et al. 2017, MNRAS, 470, 3507

Naab, T. & Burkert, A. 2003, ApJ, 597, 893

Naidu, G., Zuva, T., & Sibanda, E. M. 2023, in Artificial Intelligence Application in Networks and Systems, ed. R. Silhavy & P. Silhavy (Cham: Springer International Publishing), 15–25

Nair, P. B. & Abraham, R. G. 2010, ApJS, 186, 427

Navarro, J. F., Frenk, C. S., & White, S. D. M. 1996, ApJ, 462, 563

Nelson, D., Springel, V., Pillepich, A., et al. 2018, arXiv e-prints, arXiv:1812.05609

Nelson, D., Springel, V., Pillepich, A., et al. 2019, Computational Astrophysics and Cosmology, 6, 2

Patton, D. R., Torrey, P., Ellison, S. L., Mendel, J. T., & Scudder, J. M. 2013, MNRAS, 433, L59

Pearson, S., Privon, G. C., Besla, G., et al. 2018, MNRAS, 480, 3069

Pearson, W. J., Wang, L., Trayford, J. W., Petrillo, C. E., & van der Tak, F. F. S. 2019, A&A, 626, A49

Peng, C. Y., Ho, L. C., Impey, C. D., & Rix, H.-W. 2002, AJ, 124, 266

Pérez-Carrasco, M., Cabrera-Vives, G., Martinez-Marin, M., et al. 2019, PASP, 131, 108002

Perlmutter, S., Aldering, G., Goldhaber, G., et al. 1999, ApJ, 517, 565

Planck Collaboration, Ade, P. A. R., Aghanim, N., et al. 2016, A&A, 594, A13

Postman, M., Coe, D., Benítez, N., et al. 2012, ApJS, 199, 25

Privon, G. C., Barnes, J. E., Evans, A. S., et al. 2013, ApJ, 771, 120

Qu, Y., Helly, J. C., Bower, R. G., et al. 2017, MNRAS, 464, 1659

Ricci, C., Bauer, F. E., Treister, E., et al. 2017, MNRAS, 468, 1273

Riess, A. G., Filippenko, A. V., Challis, P., et al. 1998, AJ, 116, 1009

Robotham, A. S. G., Driver, S. P., Davies, L. J. M., et al. 2014, MNRAS, 444, 3986

Rodriguez-Gomez, V., Genel, S., Vogelsberger, M., et al. 2015, MNRAS, 449, 49

Rose, C., Kartaltepe, J. S., Snyder, G. F., et al. 2023, ApJ, 942, 54

Rosenblatt, F. 1958, Psychological review, 65, 386

Rossa, J., Laine, S., van der Marel, R. P., et al. 2007, AJ, 134, 2124

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. 1986, Nature, 323, 533

Saenko, K., Kulis, B., Fritz, M., & Darrell, T. 2010, in Computer Vision – ECCV 2010, ed. K. Daniilidis, P. Maragos, & N. Paragios (Berlin, Heidelberg: Springer Berlin Heidelberg), 213–226

Salim, S., Rich, R. M., Charlot, S., et al. 2007, ApJS, 173, 267

Sánchez-Sáez, P., Reyes, I., Valenzuela, C., et al. 2021, AJ, 161, 141

Sanders, D. B., Soifer, B. T., Elias, J. H., et al. 1988, ApJ, 325, 74

Sartori, L. F., Schawinski, K., Trakhtenbrot, B., et al. 2018, MNRAS, 476, L34

Satyapal, S., Ellison, S. L., McAlpine, W., et al. 2014, MNRAS, 441, 1297

Schaye, J., Crain, R. A., Bower, R. G., et al. 2015, MNRAS, 446, 521

Shah, E. A., Kartaltepe, J. S., Magagnoli, C. T., et al. 2020, ApJ, 904, 107

Snoek, J., Larochelle, H., & Adams, R. P. 2012, in Advances in Neural Information Processing Systems, ed. F. Pereira, C. Burges, L. Bottou, & K. Weinberger, Vol. 25 (Curran Associates, Inc.)

Spergel, D. N., Bean, R., Doré, O., et al. 2007, ApJS, 170, 377

Spergel, D. N., Verde, L., Peiris, H. V., et al. 2003, ApJS, 148, 175

Springel, V. & White, S. D. M. 1999, MNRAS, 307, 162

Srinivas, N., Krause, A., Kakade, S. M., & Seeger, M. W. 2012, IEEE Transactions on Information Theory, 58, 3250

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. 2014, Journal of Machine Learning Research, 15, 1929

Stewart, K. R., Bullock, J. S., Barton, E. J., & Wechsler, R. H. 2009, ApJ, 702, 1005

Sun, B. & Saenko, K. 2016, in Computer Vision – ECCV 2016 Workshops, ed. G. Hua & H. Jégou (Cham: Springer International Publishing), 443–450

Sun, S., Cao, Z., Zhu, H., & Zhao, J. 2020, IEEE Transactions on Cybernetics, 50, 3668

Szegedy, C., Liu, W., Jia, Y., et al. 2015, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Tamayo, D., Silburt, A., Valencia, D., et al. 2016, ApJ, 832, L22

Tarsitano, F., Bruderer, C., Schawinski, K., & Hartley, W. G. 2022, MNRAS, 511, 3330

Theys, J. C. & Spiegel, E. A. 1977, ApJ, 212, 616

Toomre, A. 1977, in Evolution of Galaxies and Stellar Populations, ed. B. M. Tinsley & D. C. Larson, Richard B. Gehret, 401

Toomre, A. & Toomre, J. 1972, ApJ, 178, 623

Treister, E., Schawinski, K., Urry, C. M., & Simmons, B. D. 2012, ApJ, 758, L39

van der Kruit, P. C. & Searle, L. 1981, A&A, 95, 105

Vaswani, A., Shazeer, N., Parmar, N., et al. 2017, in Advances in Neural Information Processing Systems, ed. I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett, Vol. 30 (Curran Associates, Inc.)

Ventou, E., Contini, T., Bouché, N., et al. 2017, A&A, 608, A9

Villalobos, Á. & Helmi, A. 2008, MNRAS, 391, 1806

Villforth, C., Hamann, F., Rosario, D. J., et al. 2014, MNRAS, 439, 3342

Villforth, C., Hamilton, T., Pawlik, M. M., et al. 2017, MNRAS, 466, 812

Vlasov, A. A. 1961, Many-particle theory and its application to plasma. (Gordon and Breach)

Vogelsberger, M., Genel, S., Springel, V., et al. 2014, MNRAS, 444, 1518

Wallin, J. F. & Stuart, B. V. 1992, ApJ, 399, 29

Walmsley, M., Lintott, C., Géron, T., et al. 2022, MNRAS, 509, 3966

Wang, Q., Ma, Y., Zhao, K., & Tian, Y. 2020, Annals of Data Science, 1

Weiss, K., Khoshgoftaar, T. M., & Wang, D. 2016, Journal of Big Data, 3, 9

Weston, M. E., McIntosh, D. H., Brodwin, M., et al. 2017, MNRAS, 464, 3882

White, S. D. M. 1978, MNRAS, 184, 185

White, S. D. M. & Rees, M. J. 1978, MNRAS, 183, 341

Wild, V., Heckman, T., & Charlot, S. 2010, MNRAS, 405, 933

Willett, K. W., Lintott, C. J., Bamford, S. P., et al. 2013, MNRAS, 435, 2835

York, D. G., Adelman, J., Anderson, John E., J., et al. 2000, AJ, 120, 1579

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. 2014, in Advances in Neural Information Processing Systems, ed. Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Weinberger, Vol. 27 (Curran Associates, Inc.)

Appendix A

ML models & hyperparams

In this section we detail each solution obtained at the end of the workflows discussed in Chapter. 5, including hyperparameters and full test results.

- **Random forest (RF)**: The hyperparameters selected for RF classifiers are highlighted in Table A0.1. These include the number of estimators, the minimum number of samples required for a leaf node, the function used to measure the quality of each split, and the number of features considered when splitting a node. All RF classifiers in this study use *Shannon entropy* as a measure of impurity for each tree,

$$H(Q_m) = -\sum_k p_{mk} \log(p_{mk}), \tag{A0.1}$$

  where p_{km} is the proportion of class k observations in node m. The impurity of a node in the RF can then be represented as the sum of the Shannon entropies computed for each leaf, weighted by the number of data points.

  The ROC curves for the RF classifiers trained with the No-mock, Gaussian, and Sky-noise dataset can be found in Fig. A0.1. Similarly, the normalized confusion matrixes can be visualized in Fig. A0.2.

- **Extreme gradient boosting (XGBoost)**: The hyperparameters selected for XGBoost classifiers are highlighted in Table A0.2. These include the step size in the gradient descent optimization process (i.e., the learning rate), the minimum loss reduction required to make a further partition on a leaf node, and the depth to which a tree can grow. All XGBoost classifiers in this study

have a fixed number of estimators equal to 100, and L1 and L2 regularization parameters are also fixed to $\lambda = 1.05$ and $\alpha = 0.05$, respectively.

The ROC curves for the XGBoost classifiers trained with the No-mock, Gaussian, and Sky-noise dataset can be found in Fig. A0.3. Similarly, the normalized confusion matrixes can be visualized in Fig. A0.4.

- **Convolutional neural network (CNN)**: The hyperparameters selected for CNN classifiers are highlighted in Table A0.3. These include the batch size used during training, the dropout rate applied after the convolutional operations, the number of fully connected layers before the output layer, and the number of neurons in each layer. All CNN classifiers in this study are trained for up to 70 epochs, using early stopping as a regularization method with patience of 7. Learning rate and beta parameters for the ADAM optimizer are fixed to default settings, that is, $\gamma = 0.001$, and $(\beta_1 = 0.9, \beta_2 = 0.999)$. The cost function used to adjust the parameters via backpropagation is categorical cross-entropy (see Eq. [4.2.4]).

  The evolution of CNNs during training is shown through learning curves in Fig. A0.5. The ROC curves for the CNN classifiers trained with the No-mock, Gaussian, and Sky-noise dataset can be found in Fig. A0.6. Similarly, the normalized confusion matrixes can be visualized in Fig. A0.7.

The summary of the metrics relative to the predictions of the classifiers with all test sets can be found in Table 5.0.1, Table 5.0.2 and Table 5.0.3. All the solutions are programmed in Python 3 (version: 3.12.2) under the framework of AutoML, which is executed automatically. However, the implementations of the models come from open-source Python packages, including scikit-learn[1], Faiss[2], PyTorch[3], and XGBoost[4]. The trained models and respective statistical weights shown in this article will be shared upon reasonable request.

[1]Scikit-Learn: https://scikit-learn.org/stable/index.html
[2]Faiss: https://faiss.ai/
[3]PyTorch: https://pytorch.org
[4]XGBoost: https://xgboost.readthedocs.io/en/

Table A0.1: Full list of hyperparameters selected for RF classifiers by AutoML through Bayesian optimization.

Training set	n-trees	min-leaf	criterion	max-feats
No-mock	908	0.0104	entropy	square root
Gaussian	603	0.0188	entropy	square root
Sky-noise	748	0.0156	entropy	square root

Table A0.2: Full list of hyperparameters selected for XGBoost classifiers by AutoML through Bayesian optimization.

Training set	learning rate	gamma	max-depth
No-mock	0.526	1.160	3
Gaussian	0.492	0.230	5
Sky-noise	0.250	0.625	5

Table A0.3: Full list of hyperparameters selected for CNN classifiers by AutoML through Bayesian optimization.

Training set	batch size	drop-rate	fc-hidden-layers	neurons
No-mock	256	0.251	3	[102, 179, 64]
Gaussian	64	0.306	2	[104, 192]
Sky-noise	64	0.329	2	[114, 198]

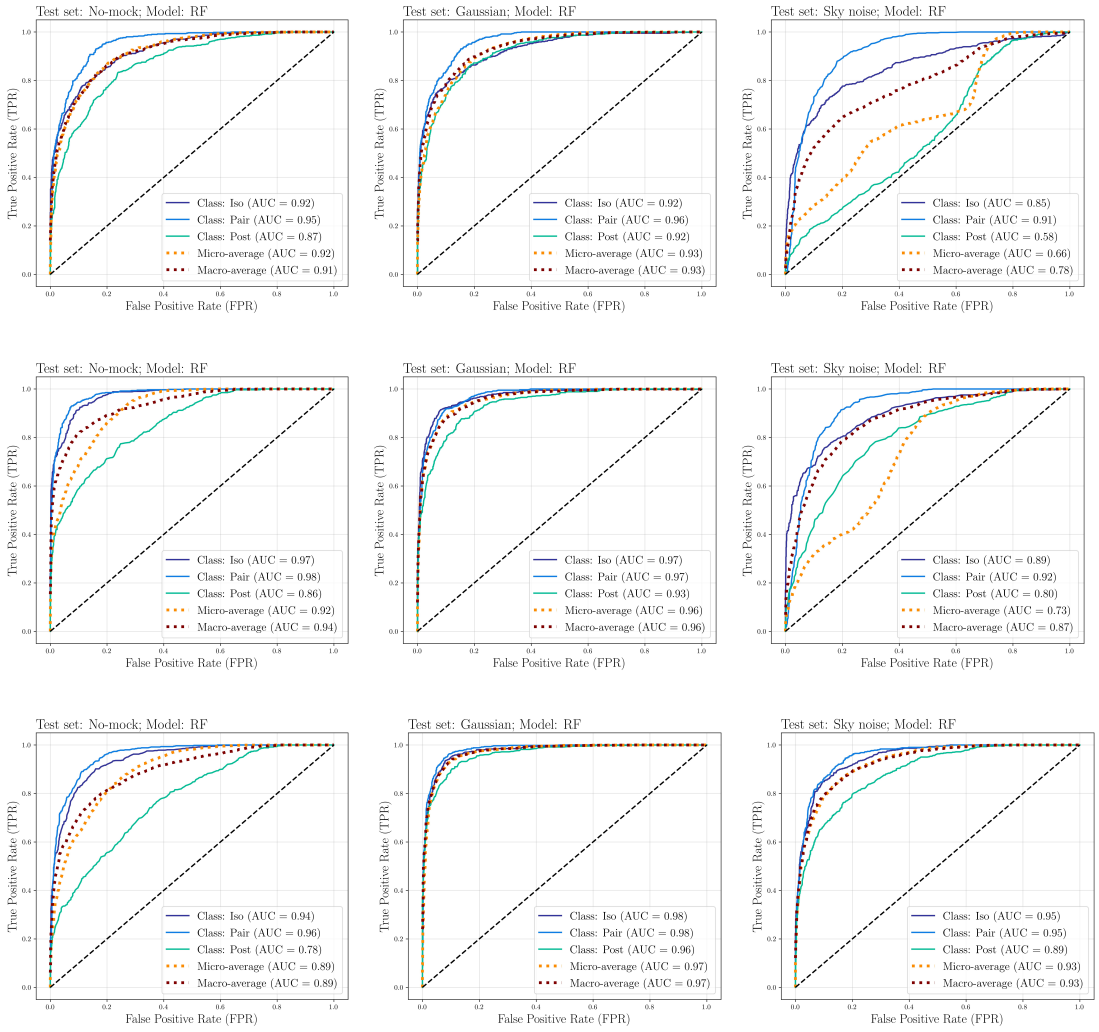

Figure A0.1: ROC curves of all experiments performed in this study with RF. (*top row*) Classifiers trained with the No-mock dataset. (*center row*) Classifiers trained with the Gaussian dataset. (*bottom row*) Classifiers trained with the Sky-noise dataset. From left to right we show the results with No-mock, Gaussian, and Sky-noise test sets. We compute the macro-average, micro-average and one-class vs. rest cases. We show the AUC score for each case, respectively.

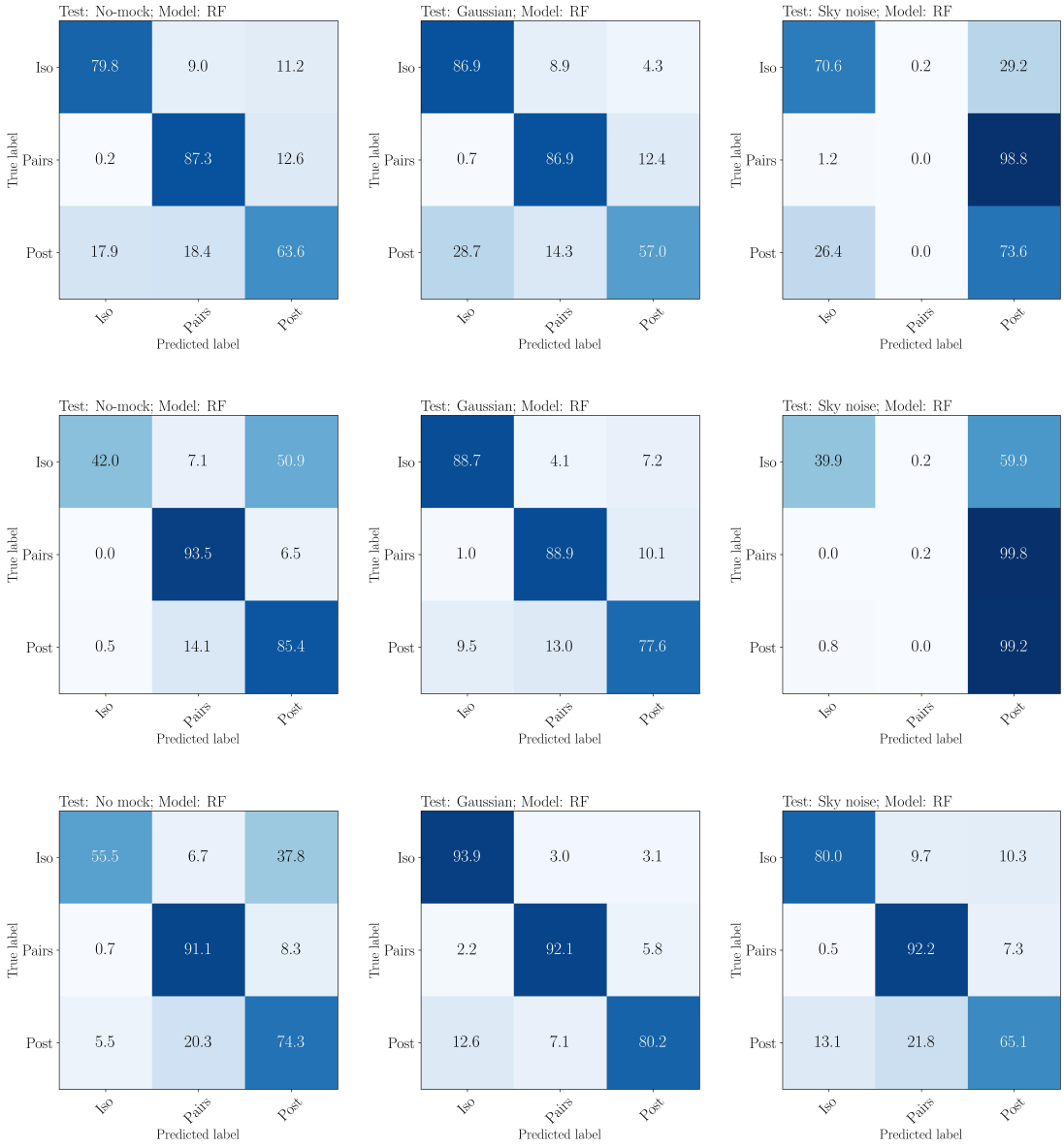

Figure A0.2: Normalized confusion matrixes of all experiments performed in this study with RF. (*top row*) Classifiers trained with the No-mock dataset. (*center row*) Classifiers trained with the Gaussian dataset. (*bottom row*) Classifiers trained with the Sky-noise dataset. From left to right we show the results with No-mock, Gaussian, and Sky-noise test sets.

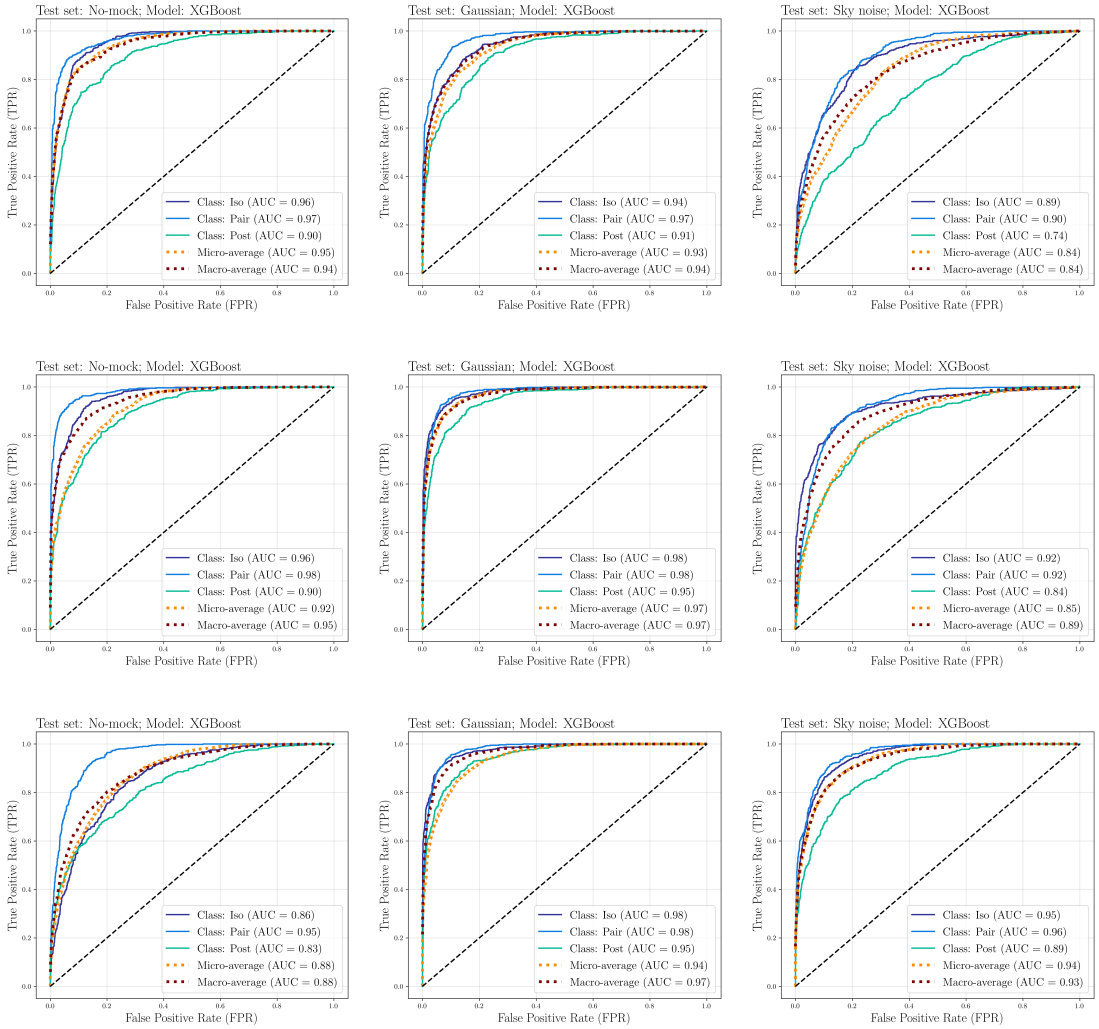

Figure A0.3: ROC curves of all experiments performed in this study with XGBoost. (*top row*) Classifiers trained with the No-mock dataset. (*center row*) Classifiers trained with the Gaussian dataset. (*bottom row*) Classifiers trained with the Sky-noise dataset. From left to right we show the results with No-mock, Gaussian, and Sky-noise test sets. We compute the macro-average, micro-average and one-class vs. rest cases. We show the AUC score for each case, respectively.

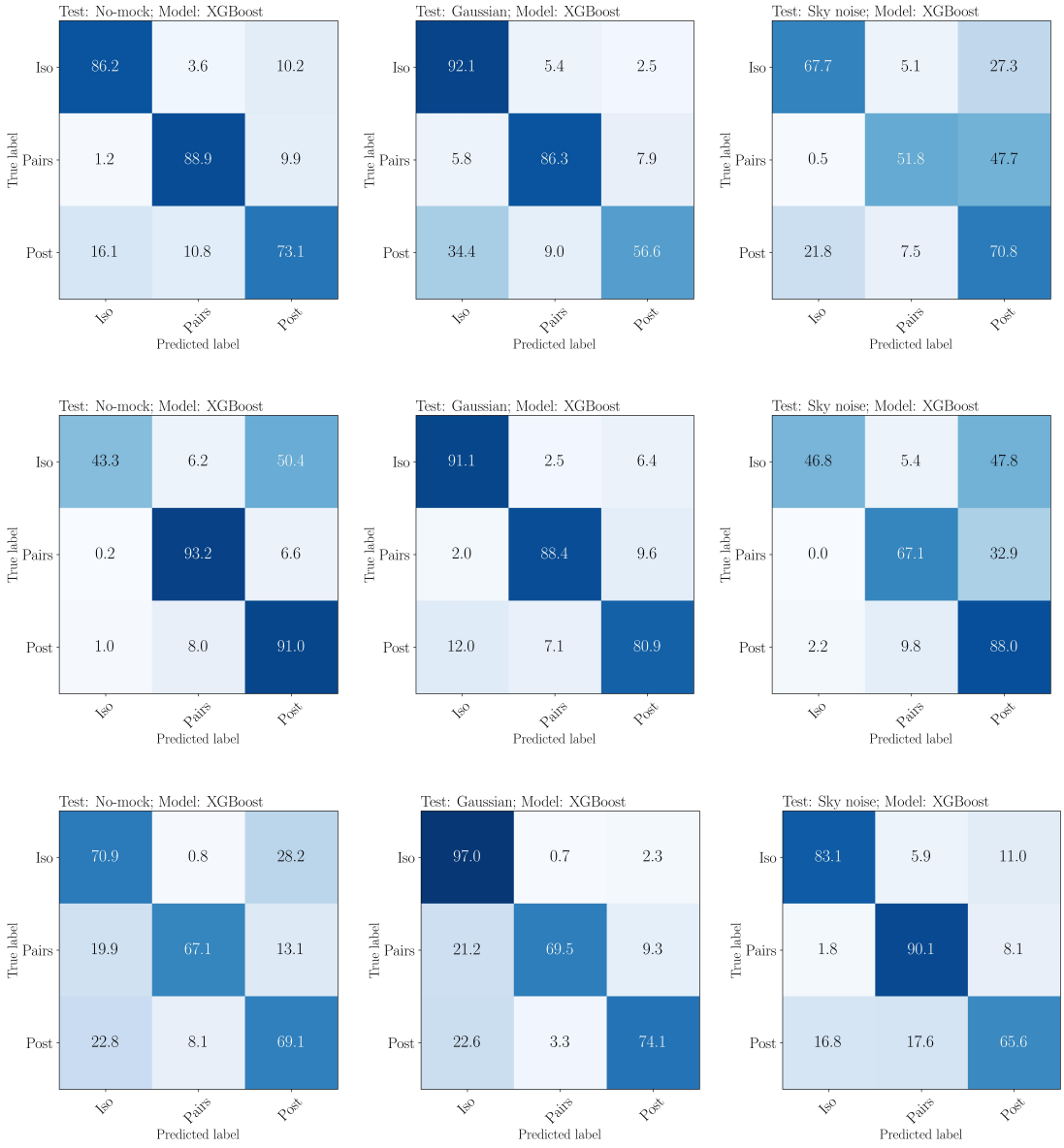

Figure A0.4: Normalized confusion matrixes of all experiments performed in this study with XGBoost. (*top row*) Classifiers trained with the No-mock dataset. (*center row*) Classifiers trained with the Gaussian dataset. (*bottom row*) Classifiers trained with the Sky-noise dataset. From left to right we show the results with No-mock, Gaussian, and Sky-noise test sets.

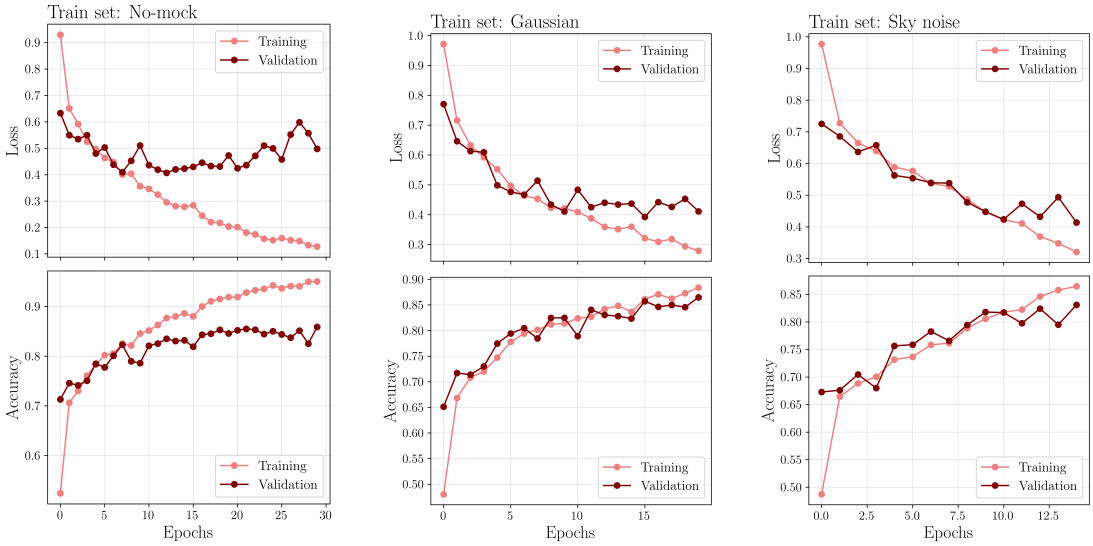

Figure A0.5: Learning curves for the CNN classifiers used in this study: (*left*) No-mock training, (*center*) Gaussian training, and (*right*) Sky noise training. The upper plots show the evolution of the cost function, while the lower plots show the evolution of a classification metric (accuracy in this case). The final value in each figure corresponds to the epoch selected by early stopping as the most optimal model achieved during training.

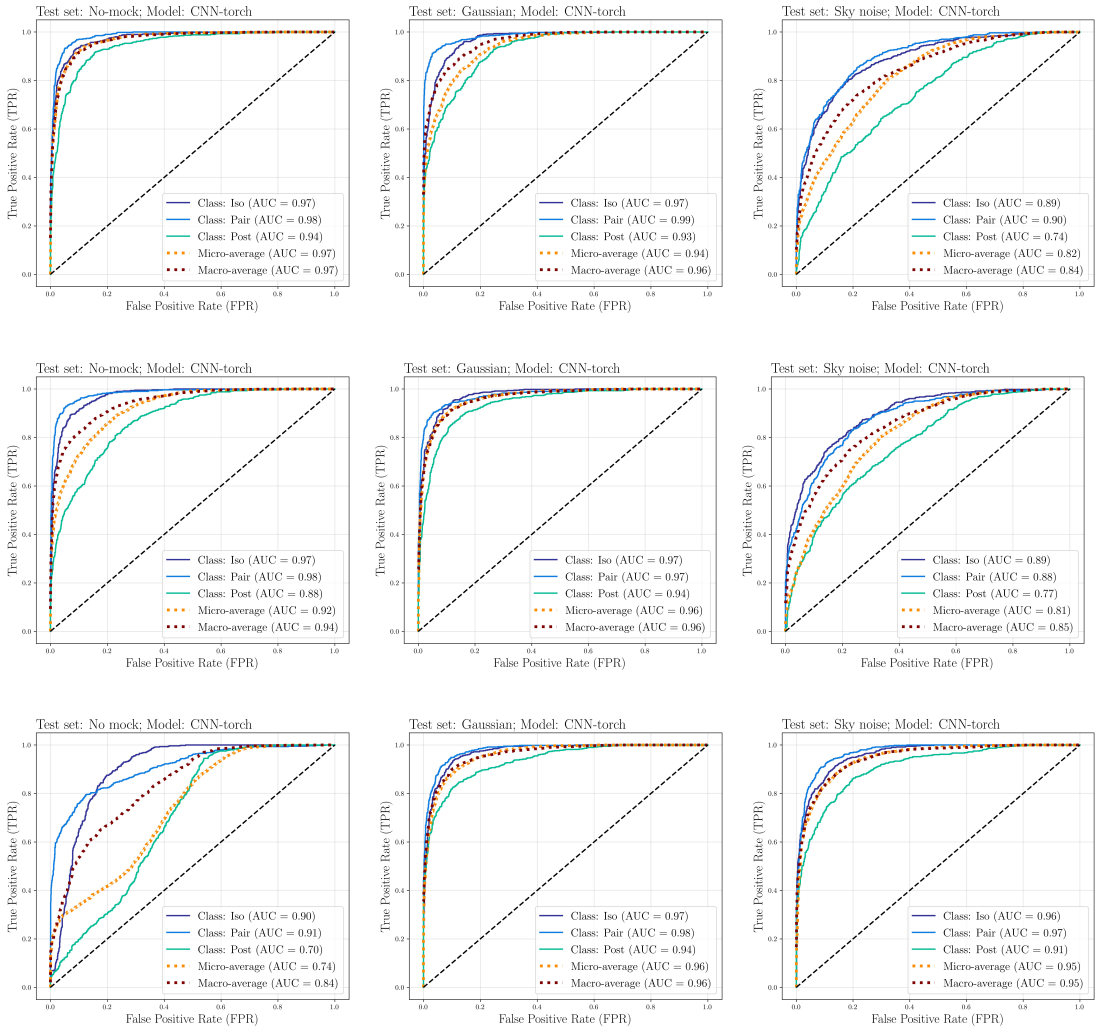

Figure A0.6: ROC curves of all experiments performed in this study with CNNs. (*top row*) Classifiers trained with the No-mock dataset. (*center row*) Classifiers trained with the Gaussian dataset. (*bottom row*) Classifiers trained with the Sky-noise dataset. From left to right we show the results with No-mock, Gaussian, and Sky-noise test sets. We compute the macro-average, micro-average and one-class vs. rest cases. We show the AUC score for each case, respectively.

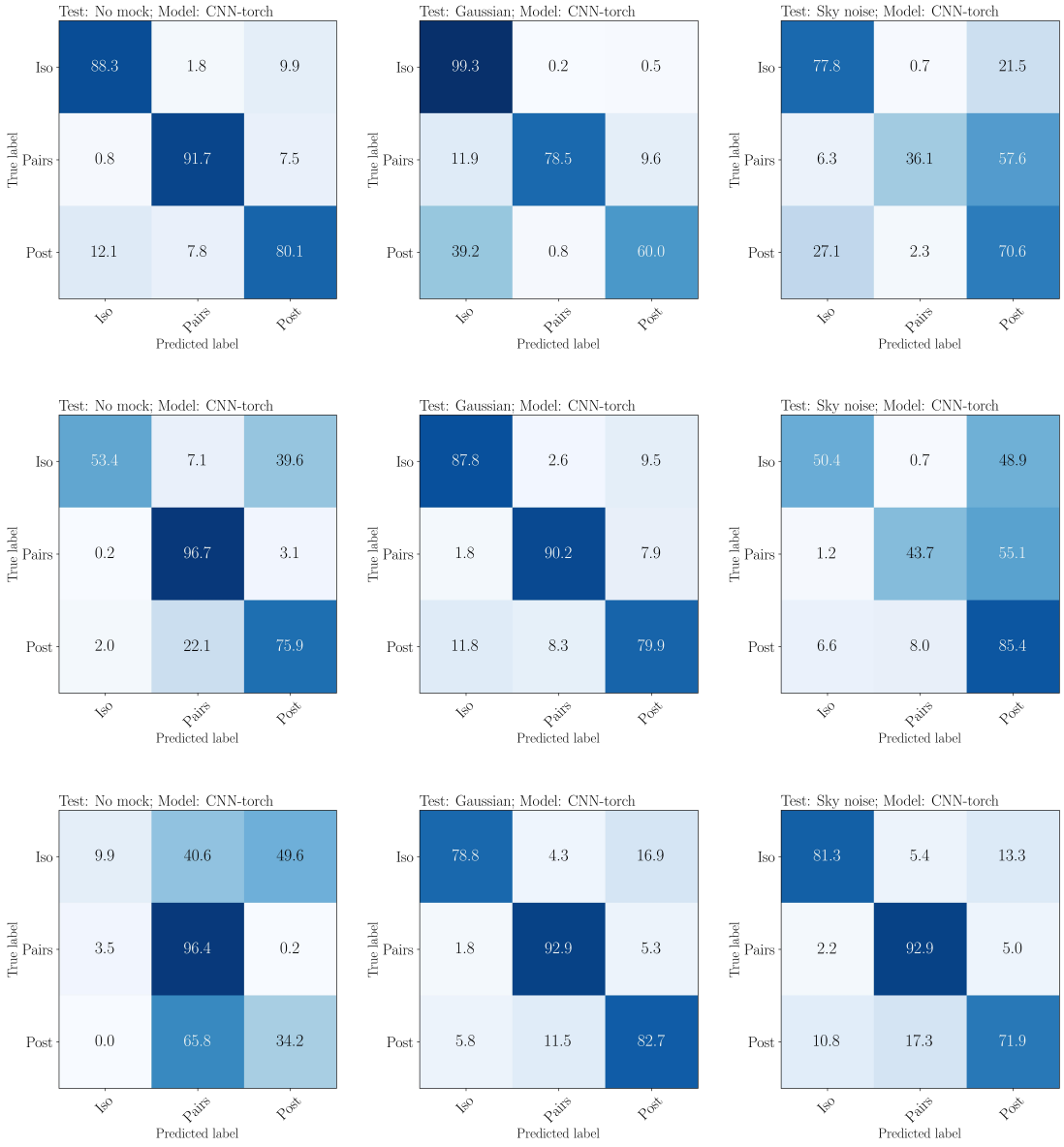

Figure A0.7: Normalized confusion matrixes of all experiments performed in this study with CNNs. (*top row*) Classifiers trained with the No-mock dataset. (*center row*) Classifiers trained with the Gaussian dataset. (*bottom row*) Classifiers trained with the Sky-noise dataset. From left to right we show the results with No-mock, Gaussian, and Sky-noise test sets.