Quantum Advantages in Galaxy Classification

Nick De Marchi
Electrical and Computer Engineering
University of Waterloo
n2demarc@uwaterloo.ca
20848920

***Abstract*—This study addresses the challenge of efficiently classifying galaxy morphology by using various machine learning methods applied to large-scale astronomical datasets, particularly focusing on the Sloan Digital Sky Survey Data Release 7 catalog. Leveraging various statistical parameters derived from galaxy images, classical and quantum logistic regression and support vector machine models are developed to classify galaxies. We posses two main morphological class types: spiral and elliptical. Results indicate high accuracy in predicting morphology, with parameters such as Concentration (C) and Entropy (H) identified as crucial predictors, along with quantum machine learning algorithms performing slightly better than their classical counterparts in terms of time complexity and accuracy. These models offer a practical tool for automated classification, overcoming the limitations of manual inspection in handling vast datasets. By capturing key aspects of galaxy structure and evolution, this approach aligns with existing knowledge of classical and quantum machine learning methods for binary classification.**

***Index Terms*— Quantum Machine Learning, Machine Learning, Logistic Regression, Support Vector Classification, Galaxy Morphology, Sloan Digital Sky Survey, Galaxy Catalogs**

I. Introduction

A. Galaxy Catalogs and Classes

THE advancements in astronomical instrumentation have enabled the collection of vast galaxy catalogs, revealing the evolution and composition of galaxies in our Universe. These catalogs showcase the diverse range of galactic structures, shedding light on their developmental stages over time and within cluster environments. A challenge in astrophysics is being able to detect morphological classes of galaxies without visual inspection. Various morphological classes exist, including dwarf, spiral, barred, non-barred, and elliptical galaxies. Efficiently knowing which state of the evolutionary sequence a galaxy is in can be a challenge and is typically studied within a cluster environment (i.e. The VERTICO survey [1]). Many classification catalogs have heavily relied on visual inspection [2], which is a very time-consuming method. These classifications were made possible by inviting the general public to visually inspect and classify these galaxies (via the internet). This issue was not as pressing in the past since older catalogs were more sparse, however, recent state-of-the-art instruments such as the Sloan Digital Sky Survey (SDSS) [3] can benefit from more efficient methods of classification due to the sheer size of these catalogs.

Over the years, astronomers have developed various methods to classify galaxies based on their observable features, known as morphological parameters. One approach to characterizing galaxy morphology is through the use of quantitative parameters derived from imaging data. For example, the CAS system developed by R.J. Abraham in 1996 [4], quantifies the visual appearance of galaxies using three key parameters: Concentration, Asymmetry, and Smoothness (CAS). In recent years, with the advent of large-scale galaxy surveys including the SDSS, the study of galaxy morphology has entered a new era. The SDSS, in particular, has revolutionized the field by cataloging millions of galaxies and providing high-quality images and spectra for each object (see figure 1).

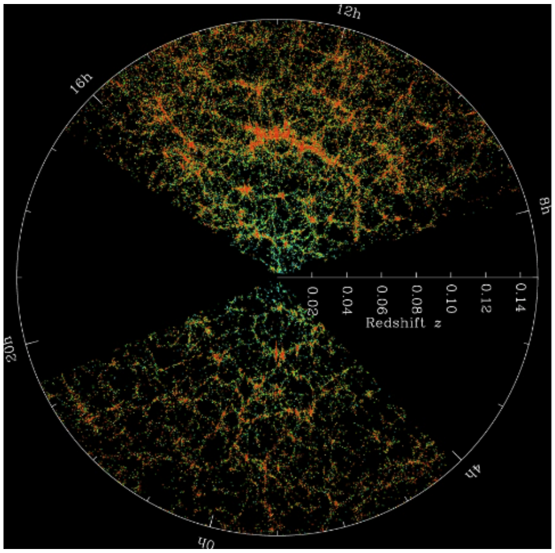

Fig. 1: SDSS galaxy map coverage. Redshift (z) is a measure of cosmological distance where $z = 0$ is the local Universe and $z > 0$ are distant galaxies. Every point is a detected galaxy [3].

By analyzing these morphological parameters derived from large galaxy surveys, astronomers can gain valuable insights into the physical processes driving galaxy evolution. In this paper, the aim is to explore the relationship between galaxy class and key statistical morphological parameters derived from the analysis of large galaxy surveys [5], [6]. We will focus on the application of **machine learning** methods to classify galaxies based on their morphological properties and infer the key characteristics behind identifying their galaxy type.

B. Advantages of Quantum Computing

Quantum computing harnesses the principles of quantum mechanics, such as superposition and entanglement to process information in ways that classical computing is unable to do. Unlike classical bits, which exist as either 0 or 1, quantum bits (qubits) can represent multiple states simultaneously, enabling massively parallel computation. This unique capability positions quantum computing as a transformative technology for solving complex problems in optimization, cryptography, and machine learning, where large datasets and high-dimensional parameter spaces pose significant challenges.

Quantum machine learning (QML) leverages quantum algorithms to enhance the speed and efficiency of data processing tasks. For binary classification, such as the one presented in this paper of detecting spiral or elliptical galaxies from galaxy survey data, QML can offer potential advantages including but not limited to:

- faster training times
- improved handling of high-dimensional parameters
- faster hyper-parameter searching using quantum search algorithms

Quantum circuits can encode and process complex patterns more effectively than traditional methods, making them suitable for datasets like those from the SDSS. Furthermore, hybrid quantum-classical approaches allow for integrating quantum speedups into existing machine learning pipelines, potentially yielding more accurate and computationally efficient models for galaxy classification. We will explore the latter in a hybrid quantum logistic regression model.

II. Methods

A. Approach to classifying galaxies

The main objective will be to utilize various statistical parameters of galaxies to build four models to calculate which class a particular galaxy belongs to. For this problem we will strictly focus on the two 'base-classes' of galaxy types: **spiral** or **elliptical**. In order to achieve this, we will utilize two galaxy catalogs from SDSS Data Release 7 [7] (here-forth referred to as SDSS-DR7). The first considers various statistical quantities of interest for 670,560 galaxies [6]. This dataset will then be combined with another galaxy catalog from SDSS-DR7 which provides the inferred morphological classes for the same set of galaxies[1] [2]. To achieve this we will stitch the two datasets together based on their unique ID's and perform some data cleaning as discussed in section II-B. I will then utilize two classical and two quantum machine learning algorithms in order to model the aforementioned classes of spiral (class-1) or elliptical (class-0) ad discussed in section II-C and II-D.

B. Data Preparation

The statistical morphological parameters provided by our first SDSS-DR7 catalog are: Concentration (C), Asymmetry (A), Smoothness (S), Gradient Pattern Analysis (G_2) and Entropy (H). Detailed definition of our 6 parameters are provided in Appendix A. This dataset also provides various error flags for the calculation of the parameters. Our data will be cleaned to utilize the parameter sets that contain no errors in their calculations (i.e. Error = 0 as outlined in Appendix B). In terms of our dataset that contains the classifications, we are provided the classes 'Elliptical', 'Spiral' and 'Uncertain'. We will also filter out any galaxies that have been flagged as 'Uncertain'. Once we apply these filters, the number of galaxies in our sample reduces to **213,338**.

The final two steps in our data preparation will be to **standardize** our features and develop an algorithm to **sub-sample our dataset with stratification**. Standardization is a common technique used in machine learning analyses as it offers many benefits such as faster convergence, higher interoperability of coefficients, and to ensure all variables are on the same scale to equalize importance. In terms of sub-sampling, this is strictly for measuring the time efficiency of our algorithms for various subset sizes N_s. By stratifying our sampled dataset we ensure that we have the same percentage of classes in our sampled dataset as we did in the original dataset. We will sample between 500 to 200,000 galaxies from our initial 213,338 while maintaining the underlying distributions (see figures 8, 9, and 10 in appendix C for the distributions and correlation matrix of our parameters).

There are several factors that can affect the direct relationship between our morphological parameters and the galaxy type. In particular, potential **confounders** in our dataset are galaxy colour and galaxy size. These variables are correlated with *both* the morphological parameters used and the galaxy type. As discussed in [5] and [6], galaxy size correlates with morphological parameters such as concentration and asymmetry, potentially affecting the interpretation of the relationship between these parameters and galaxy type, and ultimately can degrade our model.

In order to handle these factors we may consider a few techniques such as feature engineering or making catalog cuts based on colour and size distributions. For our case, we will simply consider removing the parameter K which is a size measurement since we will not be making colour-magnitude cuts.

C. Classical Algorithms

For this study we will use two main types of algorithms for binary classification. These are **logistic regression** (LR) and **support vector machines** (SVM). We will build both the classical and quantum model for each and conduct a comparative study between them. The goal is to outline potential performance based advantages in utilizing QML for a binary classification task.

To understand classical LR, let's first consider the basic linear regression model which maps input features $\mathbf{x}_i \in \Re^d$, to a continuous output variable $y_i \in \Re$ by the following equation:

$$y_i = b + \mathbf{w}^T \mathbf{x}_i,$$

where $\mathbf{w} \in \Re^d$ is the parameter vector and $b \in \Re$ is the bias.

With LR, we apply the sigmoid function $\sigma(y)$ to constrain the output y_i to the range $[0, 1]$. The sigmoid function is defined as:

$$\sigma(y_i) = \frac{1}{1 + e^{-y_i}}.$$

[1] this is the catalog that relied on visual inspection for classification. More than 10,000 participants were needed

Thus, given our input features, we can interpret $\sigma(y_i)$ as the probability of belonging to the desired class (e.g., a spiral galaxy), and we predict the positive class when $\sigma(y_i) > 0.5$.

To find the optimal parameters $\mathbf{w}$, we minimize the following cost function, which represents the negative log-likelihood [2]:

$$J(\mathbf{w}) = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right],$$

where $\hat{y}_i = \sigma(\mathbf{w}^T\mathbf{x}_i + b)$.

In classical support vector classification (SVC), we aim to find a hyperplane that separates data points belonging to two classes with maximal margin. Given our data and binary class labels shifted to be $y_i \in \{-1, 1\}$ instead of $\{0, 1\}$, the optimization problem is formulated as:

$$\min_{\mathbf{w},b,\xi} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i,$$

subject to the constraints:

$$y_i(\mathbf{w}^\top\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i,$$

Here, $\phi(\mathbf{x}_i)$ is a feature transformation function, ξ_i are slack variables that allow violations of the margin, and $C > 0$ is a regularization parameter that controls the trade-off between maximizing the margin and penalizing violations.

To efficiently map the input data to a higher-dimensional feature space, we use the kernel trick, which computes a kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^\top\phi(\mathbf{x}_j)$, avoiding explicit computation in the transformed space.

The dual form of the optimization problem leveraging our kernel is:

$$\max_{\boldsymbol{\alpha}} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j),$$

subject to:

$$\sum_{i=1}^{n}\alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad \forall i.$$

Here, $\boldsymbol{\alpha}$ are the Lagrange multipliers, and the support vectors correspond to the data points with non-zero α_i. To implement both LR and SVM, the Scikit-Learn library is used [8].

D. Quantum Algorithms

We will utilize two corresponding QML algorithms to compare against the classical case. They are quantum logistic regression (QLR) and quantum support vector machines (QSVM). The QLR approach involves encoding classical data and performing quantum operations to model the logistic regression function. This was performed using the following steps to build a circuit for our quantum linear model. First, each classical vector $\mathbf{x}_i$ is encoded into quantum states by applying a rotation $R_x(x_i)$ to the i-th qubit:

$$|\psi_i\rangle = R_x(x_i)\,|0\rangle$$

where $R_x(x_i)$ is a rotation operator parameterized by x_i, and $|0\rangle$ represents the initial state of the qubit. Each feature x_i is thus mapped to a corresponding rotation on the respective qubit. Next, a second rotation is applied to each qubit, parameterized by the weights w_i that the model learns:

$$R_Y(w_i)\,|\psi_i\rangle$$

Here, w_i corresponds to the weights learned by the model during training. Then, a Controlled-Z (CZ) gate is applied to entangle the qubits, allowing interactions between the features. This step captures the correlations between the features, with the entanglement represented as:

$$C_Z\,|\psi_i\rangle\,|\psi_j\rangle$$

where C_Z denotes the Controlled-Z gate, which applies a phase shift to the state when both qubits are in the $|1\rangle$ state. Finally, the expectation value of the state vector is measured to form a quantum linear model. The result is a real-valued outcome. To map the output to a binary classification, we aply the sigmoid function $\sigma(y)$ to the real-part of the expectation value. A sample of the circuit built for the quantum linear model is shown in figure 11 in appendix D. We then minimize the logistic loss using classical optimization with SciPy's minimize function to update the weights w_i [9].

In terms of QVSM, this algorithm builds upon the classical SVM by incorporating quantum kernels, which can enable more efficient computations in high-dimensional feature spaces. QSVM operates by embedding input data $\mathbf{x}_i \in \Re^d$ into a quantum Hilbert space using a quantum feature map $\phi_q(\mathbf{x}_i)$. The kernel function $K_q(\mathbf{x}_i, \mathbf{x}_j)$ is then computed as the inner product of these quantum-embedded states:

$$K_q(\mathbf{x}_i, \mathbf{x}_j) = \langle\phi_q(\mathbf{x}_i)|\phi_q(\mathbf{x}_j)\rangle$$

where $\phi_q(\mathbf{x}_i)$ is a quantum state generated by applying a sequence of parameterized quantum gates to an initial quantum state (e.g., $|0\rangle^{\otimes n}$).

The QSVM optimization problem mirrors the dual formulation of the classical SVM, with the quantum kernel replacing the classical one. The objective is to maximize:

$$\max_{\boldsymbol{\alpha}} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K_q(\mathbf{x}_i, \mathbf{x}_j),$$

subject to the same constraints as in the classical SVM case. Quantum kernels $K_q(\mathbf{x}_i, \mathbf{x}_j)$ can encode complex relationships between data points that may not be possible with classical SVM. For example, certain quantum kernels exhibit exponential separation between quantum and classical models, enabling QSVM to potentially solve problems that are not yet possible classically. To implement QSVM we will utilize IBM's Qiskit library and its built in QSVM function [10].

[2] The minimization process is performed using gradient descent

III. Results

To analyze the performance of these models, we will split our dataset into two components, training and testing sets. The latter helps us determine how well we generalize to unseen data. Our split for this analysis is 80/20 in terms of train/test sets. The metric utilized to score the performance is the classification accuracy, precision, and recall. The classification accuracy is described by:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} = \frac{\sum_i^n C_i}{n}$$

where,

$$C_i = \begin{cases} 1 & \text{if } y_{\text{pred},i} = y_{\text{true},i} \\ 0 & \text{otherwise} \end{cases}$$

The precision and recall describe the models sensitivity in accurately predicting true instances and the models ability to predict all possible true instances respectively. Essentially, these metrics focus on the correctness and completeness of our predictions. For each model, we perform hyper-parameter tuning using k-fold cross validation and the resulting best-fit model parameters are utilized [3]. We tabulate the classification accuracies in figure 6 and will utilize the confusion matrix to visualize the precision and recall of each model as shown in figures 2, 3, 4, and 5. Using our confusion matrix, the precision and recall are calculated as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

From our results, we notice that the quantum algorithms out-perform their classical counterparts for this task, however, the quantum algorithms were trained on a subset of the data, whereas the classical algorithms utilized the full training dataset. This was due to the fact that considerable overhead time was experienced while running our algorithms on IBM's quantum computing resources. Thus, we were only efficiently able to train our QML models up to dataset sizes of 2,000 or less. Therefore, in figure 6 we have included in blue for the classical algorithms what their performance is on the same subset of 2,000 galaxies. We notice the QML algorithms still perform better in terms of their binary classification accuracy, precision, and recall, which is indicating QML may be suitable for this type of classification task. In terms of the precision and recall, both classical algorithms appear to be relatively balanced in their correctness and completeness with SVM performance slightly better (0.95 recall and 0.94 precision for SVM compared to 0.94 recall and 0.90 precision for LR). From the confusion matrix we also notice that the precision and recall is better when predicting instances of spiral galaxies. This is likely the result of a large class imbalance in our dataset and so our model favors spiral galaxies.

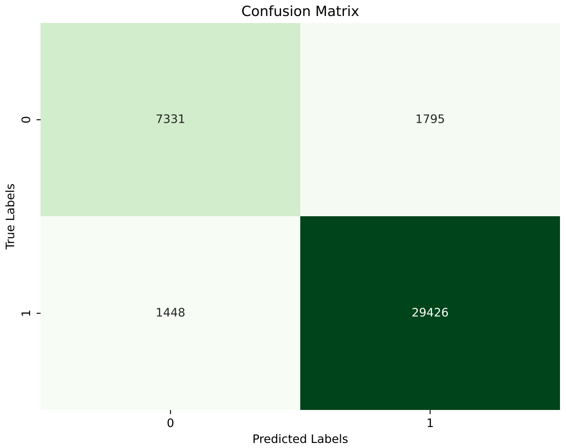

Fig. 2: SVM Confusion Matrix

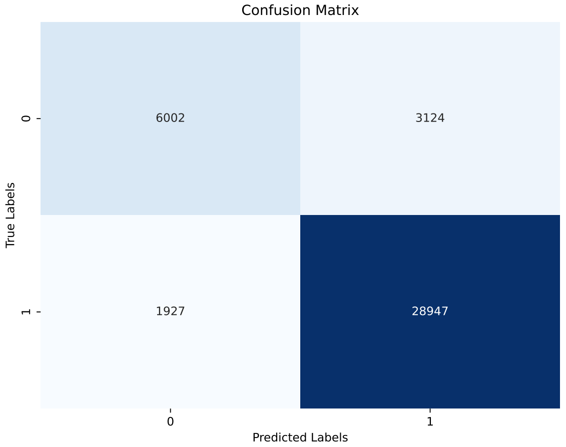

Fig. 3: LR Confusion Matrix

Moving to our QML models, in terms of their confusion matrices in figures 4 and 5, the first thing that stands out is that we have perfect precision for QLR and perfect recall for QSVM when fitting these models. Although on small datasets, this is not inherently "bad", however, it indicates an imbalance or potential performance trade-off in our models. Achieving perfect precision often comes at the cost of a lower recall, meaning the model is overly conservative and misses many actual positive cases. Alternatively, achieving perfect recall often comes at the cost of low precision, meaning the model is overly aggressive and predicts many instances as positive. We can safely say that spiral galaxy predictions made by QLR are essentially guaranteed to be correct, while QSVM will not miss any instances of correctly labeling a spiral galaxy.

[3] For SVM/QSVM we tried linear and non-linear kernels. Ultimately we found the RBF kernel to perform best for SVM, and the circular kernel for QSVM, indicating the decision boundary may be non-linear in our data

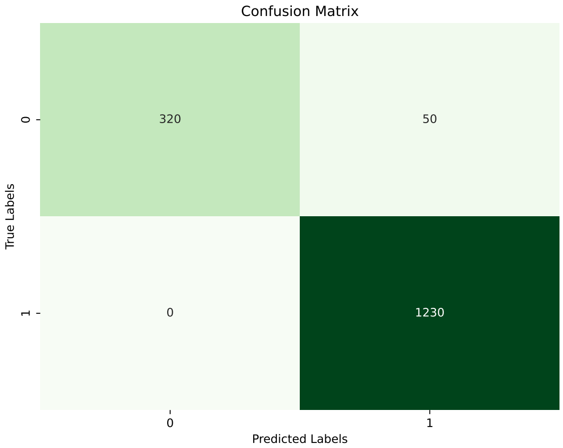

Fig. 4: QSVM Confusion Matrix

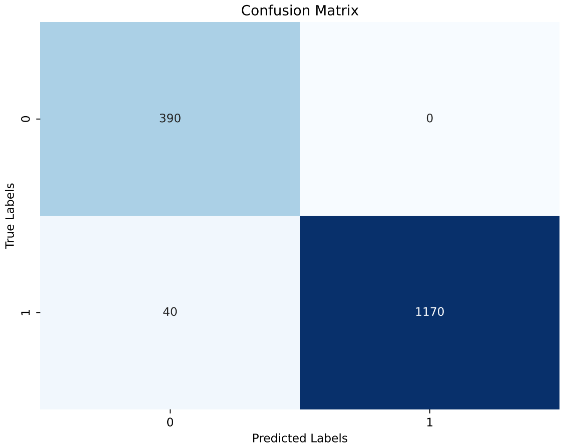

Fig. 5: QLR Confusion Matrix

Method	**LR**	**SVM**
Classical-Train	0.88 (0.90)	0.92 (0.94)
Classical-Test	0.87 (0.84)	0.91 (0.88)
Quantum-Train	0.98	0.96
Quantum-Test	0.90	0.92

Fig. 6: Train and Test data accuracy for classical and quantum versions of LR and SVM classifiers.

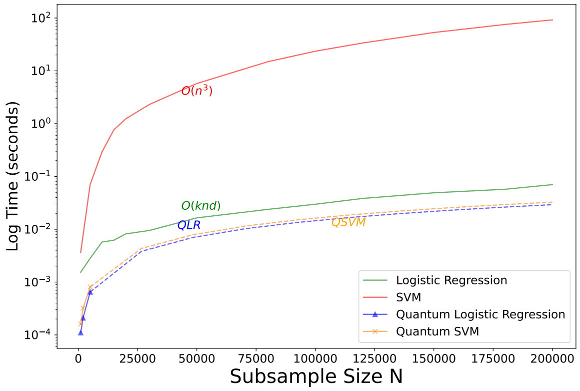

Fig. 7: Plot of sample size vs training time. The training time is logarithmically transformed to fit all instances more clearly. Here red is the classical SVM time, green is classical LR, the 3 orange 'X' points are the time to train 500, 1000, and 2000 galaxy objects using QSVM, whereas the 3 blue 'triangles' are the time for QLR training. Respective dotted orange and blue lines are extrapolated using univariate spline.

While SVM and QSVM appear to perform better than LR and QLR, one of the advantages of using the latter is it's interpretability of coefficients. Using the absolute magnitudes of the coefficients from our model output, we find that the two most important features in predicting our classes will be the entropy H and the concentration C. Referring to equation 1 in appendix E, we find that as entropy increases the log-odds of the outcome increases by 1.72 units. Concentration works in reverse, in that as concentration increases the log-odds of the outcome decreases by 1.23 units. This is **highly interpretable** with our parameters because as the concentration of light in a galaxy decreases (lower values of C), the entropy or disorder increases (higher values of H) and we are more likely to be a spiral galaxy (class-1). Conversely, as the concentration increases (higher values of C), the entropy decreases (lower values of H) we are more likely to be an elliptical (class-0). Thus our models follow the physics of galaxy types and we can assert that C and H are the two **most important features** for classifying galaxies.

The final metric analyzed to compare between classical and QML algorithms was the time efficiency of each algorithm for various subsets of our galaxy catalog. We consider sub-sampled sizes ranging from 500 galaxies to the full 200,000 galaxies and meaure the time to fit each model. Figure 7 displays the result in the time taken to fit each model. The y-axis contains the time in seconds, logarithmically scaled, to avoid visual scaling issues (i.e. polynomial time algorithms will dominate the y-axis). Over-plotted are the time efficiency analyses for each algorithm which are found to be $O(n^3)$ and $O(knd)$ for classical SVM and LR respectively, along with $\approx O(log(n))$ for QSVM and QLR. One challenge faced when utilizing IBM's quantum computing resources was the large overhead time in submitting a job (queue time) along with large overhead time for the run-time of the code on the resource[4]. Thus, given the time constraints the solution was to grab a few data-points for small sub-samples sizes (500,1000, and 2000) and then attempt to extrapolate the remaining time estimates for larger sizes for our quantum algorithms. For reference, to fit QSVM or QLR using 2,000 galaxies, the actual run-time on IBM's quantum computing was approximately 0.1 milliseconds, however, the total session run time locally was close to 30 minutes. This overhead is much to long to properly analyze for larger and larger datasets as the overhead scales

[4] IBM-Sherbrooke, IBM-Brisbane, and IBM-Kyiv were all considered in a queue and the shortest wait-time was the resource picked

exponentially (i.e. for 500 galaxies the overhead dropped down to only 17 seconds, whereas for 5,000 galaxies the job timed out after one hour).

IV. Discussion and Future Work

This study explored galaxy morphology classification using various machine learning methods applied to the large-scale astronomical SDSS-DR7 catalog. Leveraging classical and quantum machine learning, four models were trained to classify galaxies into spiral or elliptical types. Our results indicate high accuracy in predicting morphology, with a slight advantage to the quantum algorithms, with entropy H, and concentration C, identified as crucial predictors of morphology type. These parameters reflect key aspects of galaxy structure and evolution and align with existing knowledge of these galaxy types, demonstrating the effectiveness of our approaches in understanding galaxy morphology. These models provide a practical tool for automated classification, addressing the limitations of manual inspection in large datasets. Future research could refine the model by incorporating additional parameters[5], exploring alternative classification techniques, and training our quantum models more efficiently on full catalogs. Potential improvements in this study primarily center around time-efficiency and training our QML models on larger sized datasets. Future work can also consider interaction terms based on the correlations between parameters including S, G_2 and H. Overall, the study contributes to advancing our ability to classify large datasets of galaxies and underscores the potential advantages of leveraging quantum computing with machine learning methods for classification tasks.

References

[1] T. Brown, C. D. Wilson, N. Zabel, T. A. Davis, A. Boselli, A. Chung, S. L. Ellison, C. D. P. Lagos, A. R. H. Stevens, and L. Cortese, "Vertico: The virgo environment traced in co survey," *The Astrophysical Journal Supplement Series*, vol. 257, no. 2, p. 21, 2021.

[2] C. J. Lintott, K. Schawinski, A. Slosar, K. Land, S. Bamford, D. Thomas, M. J. Raddick, R. C. Nichol, A. Szalay, D. Andreescu, P. Murray, and J. van den Berg, "Galaxy zoo: Morphologies derived from visual inspection of galaxies from the sloan digital sky survey," *Monthly Notices of the Royal Astronomical Society*, vol. 389, pp. 1179–1189, 2008.

[3] "Sloan Digital Sky Survey (SDSS)," https://www.sdss.org/, accessed: [March 21st 2024].

[4] R. G. Abraham, N. R. Tanvir, B. X. Santiago, R. S. Ellis, K. Glazebrook, and S. van den Bergh, "Galaxy morphology to i=25 mag in the hubble deep field," *Monthly Notices of the Royal Astronomical Society*, vol. 279, no. 3, pp. L47–L52, 1996.

[5] M. M. Pawlik, V. Wild, C. J. Walcher, P. H. Johansson, C. Villforth, K. Rowlands, J. Mendez-Abreu, and T. Hewlett, "Shape asymmetry: a morphological indicator for automatic detection of galaxies in the post-coalescence merger stages," *Monthly Notices of the Royal Astronomical Society*, vol. 456, no. 1, pp. 303–318, 2016. [Online]. Available: https://arxiv.org/abs/1512.02000

[6] P. H. Barchi, R. R. de Carvalho, R. R. Rosa, R. Sautter, M. Soares-Santos, B. A. D. Marques, E. Clua, T. S. Gonçalves, C. de Sá-Freitas, and T. C. Moura, "Machine and deep learning applied to galaxy morphology – a comparative study," *Astronomy and Computing*, vol. 30, p. 100334, 2019. [Online]. Available: https://arxiv.org/abs/1901.07047

[7] K. N. Abazajian, J. K. Adelman-McCarthy, M. A. Agüeros, S. S. Allam, C. Allende Prieto, D. An, K. S. J. Anderson, S. F. Anderson, J. Annis, N. A. Bahcall, *et al.*, "The seventh data release of the sloan digital sky survey," *Astrophysical Journal Supplement Series*, vol. 182, p. 543, 2009.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[9] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İlhan Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. . Contributors, "SciPy 1.0: Fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[10] Q. D. Team, "Qiskit: An open-source framework for quantum computing," https://qiskit.org/, 2023, accessed: 2024-11-30. [Online]. Available: https://qiskit.org/

[11] D. J. Eisenstein, D. H. Weinberg, E. Agol, H. Aihara, C. A. Prieto, and et al., "Sdss-iii: Massive spectroscopic surveys of the distant universe, the milky way, and extra-solar planetary systems," *Astrophysical Journal*, vol. 142, p. 72, 2011.

[5] We may also wish to include other galactic parameters related to morphological type including star formation rates, gas-to-dust and gas-to-star ratios, cluster densities and cluster-centric distances to improve model performance.

Appendix A
Statistical parameter definitions

Below are how each of our 6 statistical morphological parameters are calculated:

1) **Concentration**:

$$C = 5log(\frac{R_{80}}{R_{20}})$$

Here R_{80} and R_{20} are the radii which encompass 80 percent ans 20 percent the total flux respectively. Concentration is a statistic that tells you how concentrated the light from a galaxy is. Larger values of C indicate the light is more centrally concentrating, and thus we are considered an elliptical galaxy (class 0).

2) **Asymmetry**:

$$A = \frac{\Sigma|I - I_\theta|}{2\Sigma|I|} - A_{bg}$$

Here I is the intensity per pixel of our image and I_θ is the same but for an image rotated by 180 degrees. Asymmetry is a statistic that tells us about the shape of a galaxy by utilizing this rotation of our galaxy image by 180 degrees and subtracting this from our original image. The more spherical an object, the smaller the asymmetry and more elliptical we are (class 0).

3) **Smoothness**:

$$S = \frac{\Sigma|I - I_\sigma|}{\Sigma|I|}$$

Similar to asymmetry, however now we smooth our image with a Gaussian filter and consider the difference with our original image, Thus, S is referred to as the clumpiness parameter and picks out clumpy regions. The smaller S the more elliptical we are (class 0).

4) **Gradient Pattern Analysis**:

$$G_2 = \frac{N_A}{N}\left(2 - \frac{|\Sigma_i v_i|}{\Sigma_i|v_i|}\right)$$

Here N_A and N are the number of asymmetric gradient vectors and the number of symmetric plus asymmetric gradient vectors respectively. In terms of v_i the numerator is the asymmetrical vector sum and the denominator is the norm. If we have misaligned vectors then the numerator tends to 0, whereas, if our vectors are aligned then our second term in the brackets tends to 1 and we are more symmetric and thus more elliptical.

5) **Entropy**:

$$H = \frac{1}{2\bar{X}n(n-1)}\Sigma_j\left(2j - n - 1\right)|X_j|$$

Here H is the entropy and measures the inequalities in light distribution per pixel X_j for n pixels in our image. The higher our entropy, the more disturbed our galaxy is, the lower our entropy, the more elliptical we are (class 0).

6) **Petrosian Ellipse Area**:

$$K = (\frac{R_p}{\mathrm{FWHM}/2})^2$$

Where R_p is the Petrosian radius (see [11] for information on calculation of R_p). Essentially, K is the area of the galaxy's Petrosian ellipse divided by the area of the Full Width at Half Maximum(FWHM). his parameter gives us information of the galaxy sizes.

Appendix B
Error in SDSS-DR7

Below is a list of the Error encountered with the data collection:

- Error = 0: success (no errors);
- Error = 1: many objects of significant brightness inside 2 R_p of the galaxy;
- Error = 2: not possible to calculate the galaxy's R_p;
- Error = 3:problem calculating G_2;
- Error = 4: problem calculating H;
- Error = 5: problem calculating C;
- Error = 6: problem calculating A;
- Error = 7: problem calculating S.

Appendix C
Distribution Plots

Below in figure 8 we plot the distributions for our parameters over our entire dataset and in figure 9 we plot the distributions for our sub-sampled and standardized dataset. We notice that the underlying distributions have remained.

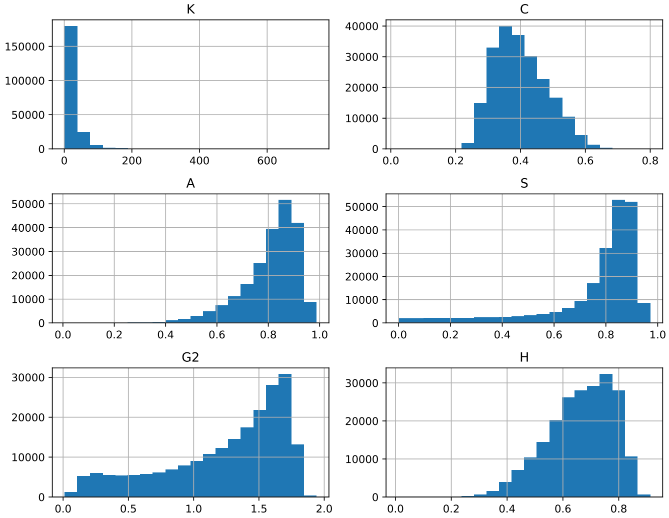

Fig. 8: Plot of the distribution for each of our 6 parameters in the full SDSS-DR7 dataset.

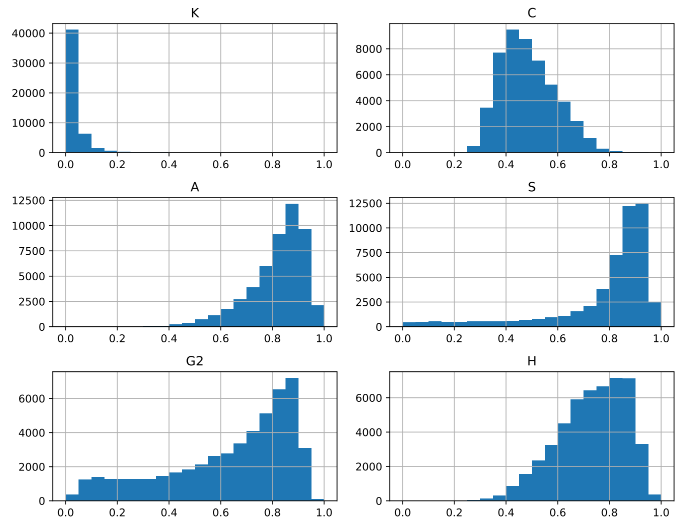

Fig. 9: Plot of the distribution for each of our 6 parameters in the SDSS-DR7 dataset after sub-sampling 50,000 galaxies and standardizing our data.

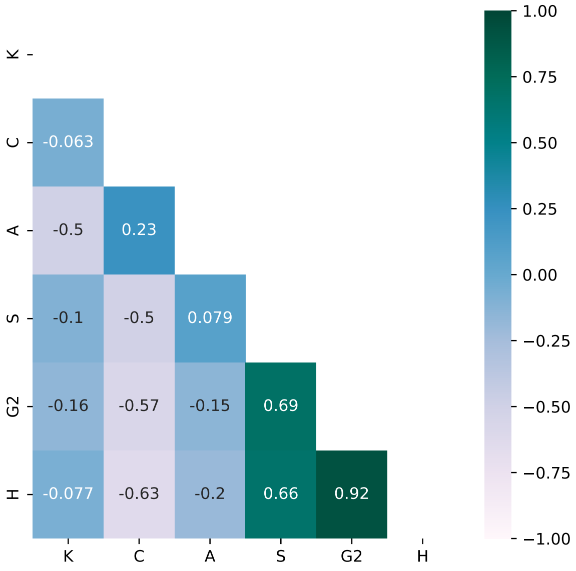

Fig. 10: The correlation matrix for our morphological parameters. We notice that $S, G2$ and H are highly correlated. These 3 parameters also negatively correlate with C.

Appendix D
Quantum Linear Circuit

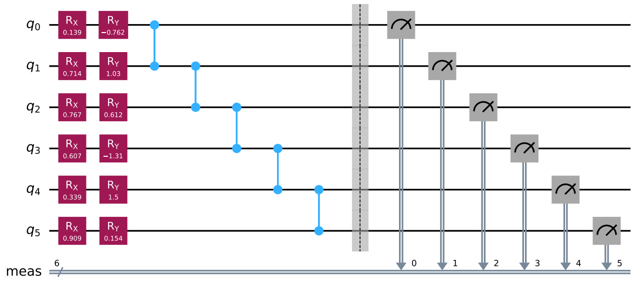

Fig. 11: Sample quantum linear model circuit. The sigmoid function is applied upon measurement of the real-part of our expectation value

Appendix E
Logistic Regression Equation

The following equation is the product of using logistic regression with gradient descent to find the optimal coefficients for our statistical parameters. **LR Model Result:**

$$2.81 - 1.23C - 1.09A + 0.39S - 0.73G_2 + 1.72H \quad (1)$$