Assessing the Robustness of Quantum Ridge Regression on DWave's Quantum Annealers

SANVITO ALESSANDRO STUART K. THUANY
sanvito | tkstuart @kth.se

February 15, 2022

Abstract

Machine learning has become the de-facto standard for modeling complex real-world phenomena, with relevant implications across several industries. However, with the increasing complexity of the models, the greater availability of data, and Moore's law decline have put pressure on researchers to find suitable accelerators to support machine learning computations.

Among the candidates, quantum computers, and quantum annealers, in particular, can provide theoretical speed-ups to machine learning computations, but the current hardware suffers from errors in the results due to thermal noise and structural limitations.

In this paper, we report the results of the empirical analysis of a classical machine learning model, Ridge regression, trained on a DWave's hybrid solver in a realistic scenario. First, we provide a formulation of the training algorithm suitable for the quantum annealer. Then we analyze the results of the comparison between quantum training instances and models trained with the Cholesky closed-form solution to assess the robustness of the training process across different datasets.

Overall, our work finds that training Ridge regression on a quantum annealer does not yield the same reliable results as on classical hardware.

Contents

1 Introduction

1.1 Background

Machine learning has demonstrated its ability to yield relevant predictive results in complex scenarios. However, the computational complexity of this class of models cannot be underestimated and, as Moore's law starts declining and the amount of data available keeps growing, it is imperative to explore alternative platforms to support machine learning computations [1]. Quantum machine learning is a nascent field leveraging the power of quantum computing to speed up machine learning tasks.

Works as [2] [3] [4] demonstrated the theoretical edge of quantum computing over classical computing in optimization settings and the potential for accelerating the solution to NP-hard problems. Moreover, in [5] Google offered empirical proof of the supremacy of quantum computing over classical computing.

However, the quantum computing community has been concerned with the effect of noise on the result of quantum computation [6], and recent works such as [7] assessed the disruptive effect of noise on theoretically sound algorithms.

As recent quantum architectures by DWave have become capable of tackling realistic problems [8], the issue has become even more pressing.

In this paper, we address the issue in the case of a popular linear machine learning model, Ridge regression. First, we formulate the machine learning problem as a Quadratic Unconstrained Binary Optimization problem (QUBO), a form suitable for DWave's quantum annealers. Then we test the robustness of the quantum algorithm by testing the relative distance between the weights found in the quantum solution and the ones found in the analytical solution.

1.2 Problem Discussion

The research builds upon efforts in literature to quantify the quality of quantum machine learning models empirically. In particular, [9] compared a quantum linear regression model with a classical counterpart from the Scikit-learn library [10] in terms of Euclidean error function $E(prediction, groundtruth) = ||prediction - groundtruth||^2$ on synthetic data. Although the authors considered real data for the study, DWave's quantum annealers were deemed too limited to generate meaningful results.

[7], on the other hand, focused on quantum k-means clustering and quantified the quality of the model through $Inertia = \sum_{cluster=1}^{k} \sum_{x \in cluster} ||x - centroid_{cluster}||^2$. The model was tested both on synthetic data and on real data from the Iris benchmark dataset. The comparison was then performed against the Scikit-learn library counterpart.

Finally, [11] assessed the quality of a simple classification algorithm on a small artificial dataset through $Accuracy = number\,of\,correct\,predictions / number\,of\,predictions$. However, no comparison with classical algorithms was provided.

Our work improves on the state of the art by defining a different metric to compare the quantum algorithm with the classical counterpart. The new metric focuses on the model itself and not on the prediction quality, therefore providing a better proxy for robustness. Moreover, our work leverages robust statistical testing to provide significant results. Finally, our work introduces a new formulation for Ridge regression.

1.3 Research question and contributions to the state of the art

Considering the background and the problem discussion, the paper answers the following research question:

RQ1 Does the Ridge regression algorithm running on DWave's quantum annealers reach the same result as their classical counterparts?

The main contributions of our work are:

- The first formulation of Ridge regression as a QUBO problem in literature, to the best of our knowledge.

- A thorough statistical assessment of the robustness of univariate Ridge regression running on DWave's quantum annealers and the impact of noise on the final result of the quantum computation.

The result of the analysis sheds new light on the application of quantum computing in real-life machine learning applications.

1.4 Outline

The rest of the paper is organized as follows. In section 2 we introduce the concepts behind our work. In section 3 we discuss the procedure adopted in performing the experiments. In section 4 we present and analyze the results of the experiments. Finally, in section 5 we draw the conclusions.

2 Theoretical framework

2.1 Adiabatic Quantum Computers

Adiabatic quantum computing is a form of quantum computing leveraging the results of the adiabatic theorem [12] to perform computations. According to the adiabatic theorem, in its simplest case, a quantum system remains in a state of minimal energy provided that:

- There is an energy gap between the quantum state and other states.
- The evolution of the quantum system is sufficiently slow.

In particular, the energy configuration of the system can be represented in the form of the Ising model [13], which is more conveniently represented in DWave's quantum annealers as a QUBO problem:

$$\min_{\forall x \in \mathbb{B}^M} x^T Q x \tag{1}$$

Where $x \in \mathbb{B}^M$ is a binary vector of length M and $Q \in \mathbb{R}^{M \times M}$ is a $M \times M$ real matrix representing the problem.

From an initial state, the quantum system is slowly evolved towards the configuration represented by Q while maintaining the ground state. The ground state, representing the solution to the problem, is then measured with a high probability [14]. Although quantum annealing allows formulating NP problems easily [13], the process is intrinsically stochastic. Moreover, current hardware is subject to noise which can alter the probability distribution [6]. Quantum annealers, therefore, require multiple samples from the underlying probability distribution. Moreover, although quantum annealers can tackle larger realistic problems than traditional quantum computing[8], the size of the problem is still limited by the current technology and might require to perform an embedding of the problem [15], with a consequent loss of quality in the result.

2.2 Linear Regression

Linear regression models the relationship between a dependent variable and one or more independent variables (also called explanatory variables). Given a dataset $D = (y_i, x_{i1}, ..., x_{ip}), i = 1, ..., n$ of n samples and p independent variables, linear models assume that the relationship between the variable y and the set of variables $\mathbf{x}$ is linear, such that:

$$y_i = \beta_0 + \beta_1 x_{i1} + ... + \beta_p x_{ip} + \varepsilon_i = x_i^T \beta + \varepsilon_i, i = 1, ..., n \tag{2}$$

Where the disturbance term, or error variable ε represents the noise to the linear relationship. Writing Equation 2 in a matrix form, we have:

$$y = X\beta + \varepsilon \tag{3}$$

Fitting a regression model means to estimate the parameters β such that the error term ε is minimized. Although there are different approaches to fitting a linear regression model, the most often used is least squares method, which minimizes the sum of square errors $||y - X\beta||$, that is:

$$\min_{\forall \beta \in \mathbb{R}^{p+1}} \varepsilon = ||y - X\beta|| \tag{4}$$

This problem has an analytical solution given by

$$\beta = (X^T X)^{-1} X^T Y \tag{5}$$

If $(X^T X)^{-1}$ does not exist, the pseudo-inverse is computed.

2.3 Ridge Linear Regression

Ridge regression (or L2 regularization) is a method of estimating the parameters of a linear regression model developed to target problems with multi-collinear (highly correlated) independent variables. Ridge regression minimizes a modified version of the error term ε that includes a penalty equivalent to the magnitude of the estimated parameters:

$$\min_{\forall \beta \in \mathbb{R}^{p+1}} \varepsilon = ||y - X\beta|| + \lambda ||\beta|| \tag{6}$$

Where λ is a penalty term. Ridge regression constrains the coefficients β, reducing the model complexity and multi-collinearity. The closed-form solution to Ridge regression is given by

$$\beta = (X^T X + \lambda I)^{-1} X^T Y \tag{7}$$

Where I is the identity matrix and if $(X^T X - \lambda I)^{-1}$ does not exist, the pseudo-inverse is computed.

3 Research methodology

The project uses the empirical method [16], as it provides a direct comparison of the results under control factors allowing replication. Moreover, an analytical analysis does not fit the purpose of the project because it does not consider the empirical noise that affects the current quantum computing implementations.

To verify the hypothesis that quantum Ridge regression produces the same model as the classical implementation, we need to compare the resulting Ridge regression models in many datasets to reach statistical significance and avoid bias due to the characteristics of the data being modeled.

3.1 Implementation Details

3.1.1 Classical Computation

For simplicity, we use the classical implementation of Ridge regression provided by the Scikit-learn [17] package. Scikit-learn Ridge implementation offers several different optimizers including iterative ones that provide faster convergence for large datasets. Nevertheless, we choose to use the *cholesky* solver for a closed-form solution of the Ridge regression problem. This solution is deterministic and, therefore, should yield the same coefficient values at every run for a certain dataset. We rely on this assumption to compare the results of the classical computation to the stochastic results of the quantum computation.

3.1.2 Quantum Computation

Currently, Dwave is the only company providing a commercial quantum annealer solution. Therefore, we will use the Dwave annealer to run the quantum Ridge regression. When eventually more options become available in the market, we suggest that the same methodology is used to assess the noise level in a specific implementation. Nevertheless, because of the differences in practical aspects of different annealers, in our opinion, the studies should be conducted individually for each platform and not across platforms.

The Leap Quantum Application Environment powered by Dwave offers a total of one minute of free access to Dwave's Quantum Processing Unit (QPU) and twenty minutes of free access to the Hybrid Quantum Solver. Since the process underlying the quantum annealing process is stochastic and dominated by random noise, sampling the quantum annealers multiple times might return different results. Therefore, multiple samples are required. In light of the limited computational resources, we limit the number of samples to three samples per model fitting. Moreover, since freely available QPUs support a limited number of bits and would require embedding the problem to a lower dimension, thus limiting the quality of the result, while offering less computation time, we use the Hybrid Quantum Solver.

3.1.3 Ridge Regression Parameter

In the Ridge regression algorithm, the bias-variance trade-off is controlled by the λ parameter. Because we are not interested in optimizing the performance of the algorithm, we do not focus on finding the best value for λ. Nevertheless, if we set the parameter to a fixed value, the results of this research would not be representative and could not be extrapolated for all values of λ. Conversely, if we set λ to a random value, the search space might be limited to a range that would not be not realistic and compromise the analysis. For this reason, we perform a random search with limited resources to define the λ for each dataset.

3.2 Data preparation

Taking into consideration these two aspects:

- There is limited available computing time in the quantum annealer;
- We are not interested in optimizing the performance of the algorithm, but rather compare the results of two models;

We choose to run the experiments with toy datasets, which can be easily collected and present good data quality, which reduces the need for data preprocessing. For simplicity, we will use the toy datasets available in Scikit-learn for regression tasks [18], namely *Iris*, *Boston* and *Diabetes*.

To reduce complexity and computation time, we will evaluate the Ridge regression models using only one explanatory variable, that is, simple linear regression with Ridge regularization. This way, all numerical features of the available datasets can be combined in pairs to generate regression tasks, therefore increasing the number of testing datasets.

Table 1 shows the datasets that are being used, the number of numerical features, and the feature combinations generated from them. The three datasets generate a total of 284 feature combinations, that is, regression tasks that will be used to assess the hypothesis.

Table 1: Dataset feature combinations.

Dataset	Number of Numerical Features	# Feature Combinations	# Feature Combinations Ran
Iris	4	12	12
Boston	14	182	182
Diabetes	10	90	90
Total	**28**	**284**	**284**

Therefore, the data preprocessing consists of creating regression tasks datasets by combining the numeric features of each toy dataset and standardizing the features. The result will be a set of datasets with one explanatory variable and one target variable with zero mean and unit variance. The feature standardization encourages smaller and more balanced weights.

3.3 Testing

To test the robustness of the quantum model, differently from previous literature, we choose to use the relative distance between the weights of the closed-form solution and the sampled model as a proxy for the equivalence between the models. We believe this metric to be a better representative of the robustness of a model since the quantum model, in ideal conditions, should find the same weights as the closed-form solution, which is proven to be unique [19]. Measuring the robustness of the model via a quality metric, such as MSE, would be more dependent on the dataset itself, in that significantly different models might score similarly for certain datasets, thus losing the correlation with the objective of the research.

Once the metric is measured, standard t-testing is used to assess that the mean of the relative error is 0, which means that, on average, the quantum model is the same as the closed-form one. We considered setting a threshold for the error below which the models would be similar. However, the choice of such a threshold would have been arbitrary and dependent on the dataset.

The result of each t-test is considered as a boolean output, true if the models are similar, false otherwise. The collective results across all datasets are then used in a final binomial test to assess if the success rate is close to 100%. Since binomial the threshold for the success rate is arbitrary, we also provide statistics regarding the success rate itself.

3.4 Procedure

For each combination of pairs of features, the following steps will be followed to analyze the data:

1. Find a reasonable value for λ using randomized search [20] with cross validation.
2. Fit a Scikit-learn Ridge regression model on each dataset using the Cholesky solver for a closed-form solution, that is, the classical implementation of Ridge regression.
3. Fit the self-implemented quantum Ridge regression on each dataset. Collect three samples from the quantum annealers using DWave's hybrid solver.
4. Calculate the relative error in terms of l2 norm between the coefficients obtained with the close form solution and the three sets of coefficients obtained with the quantum regressor. This will result in a set of three errors.
5. Generate a binary variable indicating if the mean of the errors is zero or not according to a T-test.

Finally, we apply a Binomial test* to the set of binary variables indicating if the coefficients were the same for each dataset and confirm or refute the hypothesis that the Ridge regression algorithm with a closed-form solution running on DWave's quantum annealers reaches the same result as the Scikit-learn implementation.

In all the statistical tests we consider a threshold $pvalue > 0.05$ to refute $H0$.

The output of the analyses will be the statistics, the results of the statistical test, and visualizations of the model's outputs when meaningful.

4 Results and Analysis

4.1 Ridge regression as a QUBO problem

Starting from Equation 6, we can expand the norm operation as

$$\min_{\forall \beta \in \mathbb{R}^{p+1}} \beta^T X^T X \beta - 2\beta^T X^T Y + \lambda \beta^T \beta \tag{8}$$

* The Binomial test tests the statistical significance of deviations from a theoretically expected distribution of observations into two categories using sample data.

Ignoring the constant term $Y^T Y$. Next, we introduce a precision matrix $P \in \mathbb{R}^{(p+1)\times K(p+1)}$ and a binary vector $\tilde{\beta} \in \mathbb{B}^{K(p+1)}$ with arbitrary precision $K \in \mathbb{N}$ such that $\beta \approx P\tilde{\beta}$. We can then rewrite Equation 8 as

$$\min_{\forall \tilde{\beta} \in \mathbb{B}^{K(p+1)}} \tilde{\beta}^T P^T X^T X P \tilde{\beta} - 2\tilde{\beta}^T P^T X^T Y + \lambda \tilde{\beta}^T P^T P \tilde{\beta} \tag{9}$$

Which is the equivalent QUBO formulation to the aforementioned optimization problem. The optimization problem can now be solved using a quantum annealer. From this very last equation, it can also be inferred that considering $\mathcal{O}(1)$ annealing time, the QUBO formulation has a complexity of $\mathcal{O}(Np^2K^2)$, in which K is the number of binary variables introduced when converting the p + 1 weights in binary form. Therefore, the complexity has a quadratic dependency from K, which was assumed to be a variable in the above analysis. However, by assuming a fixed precision of quantum computers, which is also the case in classical computers, and considering K as a constant, time complexity becomes $\mathcal{O}(Np^2)$, which provides a speedup compared to the classical $\mathcal{O}(Np^2 + p^3)$ [19].

4.2 Robustness assessment

Table 2: Distribution of the error across the datasets.

Dataset	**Mean relative error**	**Standard deviation of the error**	**Accepted ratio**
Iris	7.10	7.54	75%
Boston	0.58	0.69	65.10%
Diabetes	0.89	1.40	63.63%
Overall	0.96	2.40	65.77%

In Table 2, we show the distribution of the error across the original datasets and the ratio of training instances accepted by the t-test.

According to the t-tests, only in 65.77% of the datasets, the quantum Ridge regression reaches the same results as the classical implementation. According to the binomial test, there is strong evidence that the two implementations perform differently.

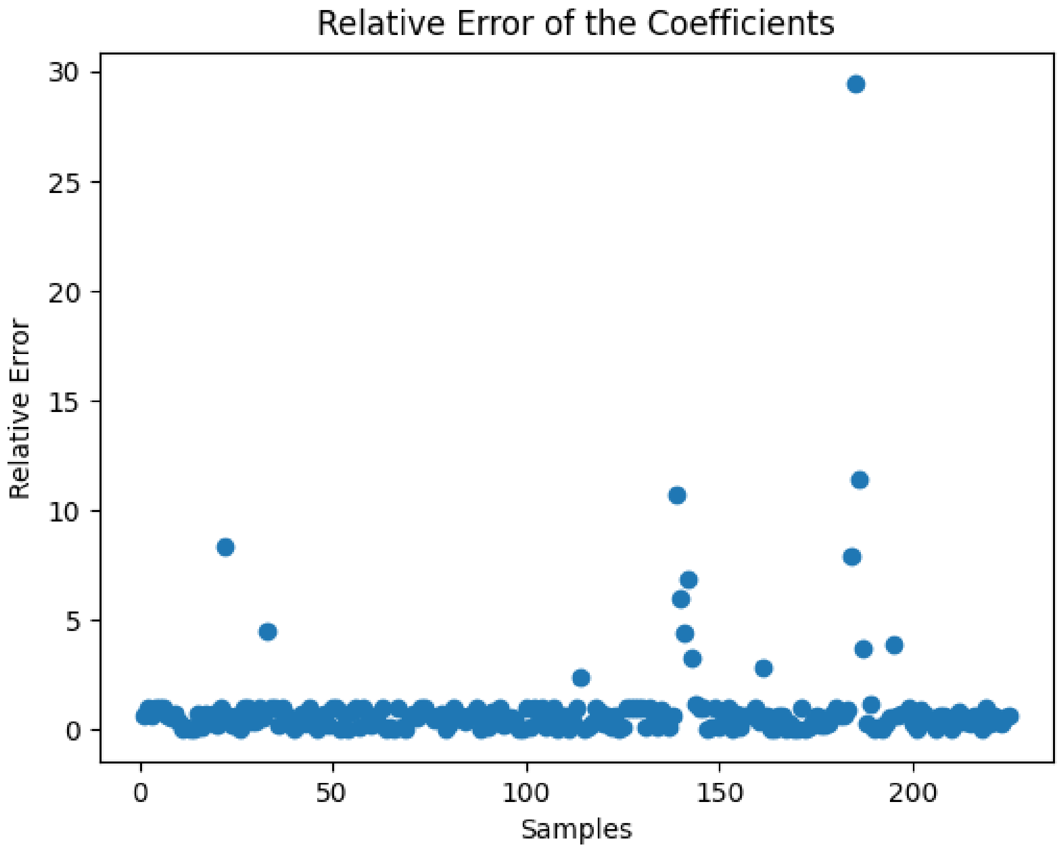

Figure 1: Distribution of the relative error across all the samples.

In Figure 1, we show the distribution of the error across samples from all the datasets. As we could expect from the high standard deviation in Table 2, few outliers show a remarkably higher error.

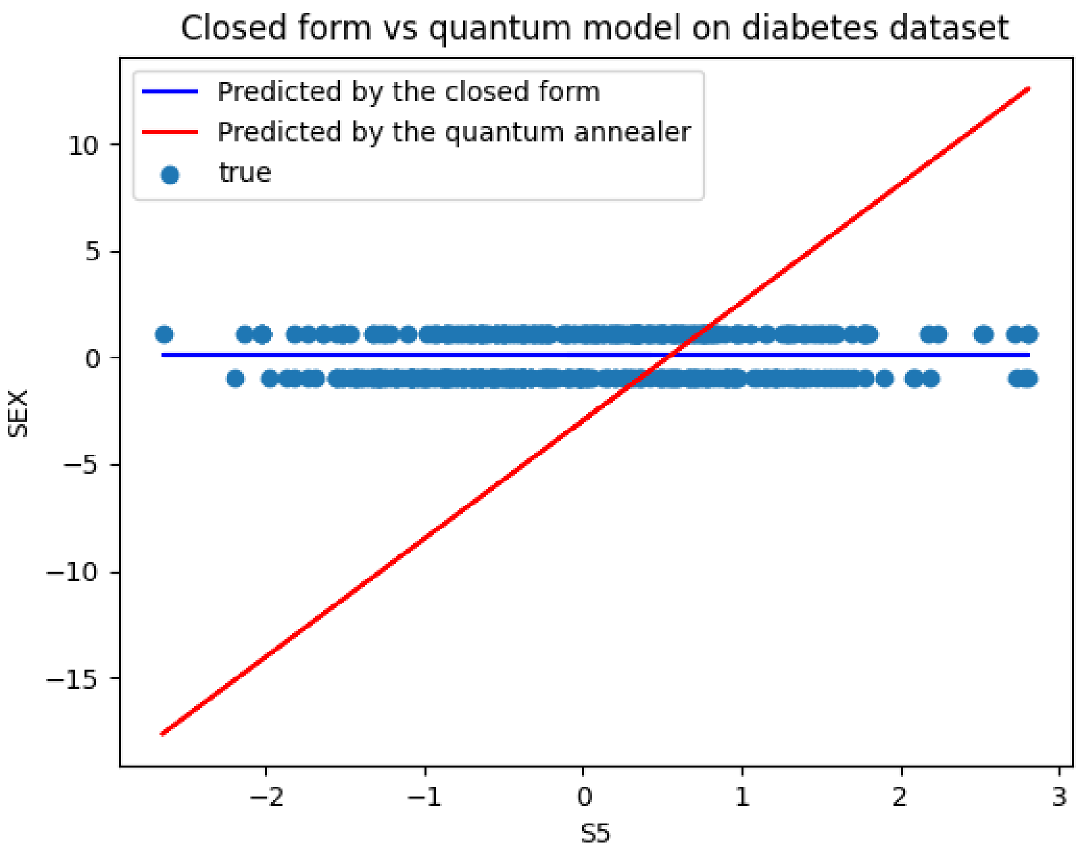

Figure 2: Plot of the predictions of one of the models resulting from quantum training and the closed form solution one on the diabetes dataset.

Finally, a visual inspection of outliers as Figure 2, reveals that not only the distance between quantum and closed form Ridge regression is high, but also that the model resulting from the quantum computation is not reasonable.

5 Discussion

Noise used to be a relevant issue in quantum computing, and our experiments indicate that this is still the case, at least in the case of Ridge regression. Results show, with high significance, that quantum annealers are affected by noise to a point they are not yet able to reach the same solution as classical computation.

However, the risk associated with noise might be mitigated by drawing more samples from the annealer. Moreover, the promise of a consistent speed up encourages reformulating more machine learning models as QUBO problems. Future work might focus on providing such formulation. Furthermore, it might explore the case of multivariate analysis when sufficiently powerful quantum hardware is available. Finally, the same testing methodology might be applied to a wide range of existing quantum machine learning models to assess their robustness.

References

[1] N. Thompson and S. Spanuth, "The decline of computers as a general purpose technology: why deep learning and the end of moore's law are fragmenting computing," *Available at SSRN 3287769*, 2018.

[2] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[3] P. Date, D. Arthur, and L. Pusey-Nazzaro, "QUBO formulations for training machine learning models," *Scientific Reports*, vol. 11, no. 1, p. 10029, May 2021. doi: 10.1038/s41598-021-89461-4 Bandiera_abtest: a Cc_license_type: cc_by Cg_type: Nature Research Journals Number: 1 Primary_atype: Research Publisher: Nature Publishing Group Subject_term: Computer science;Information theory and computation Subject_term_id: computer-science;information-theory-and-computation. [Online]. Available: https://www.nature.com/articles/s41598-021-89461-4

[4] R. Patton, C. Schuman, T. Potok *et al.*, "Efficiently embedding qubo problems on adiabatic quantum computers," *Quantum Information Processing*, vol. 18, no. 4, 2019.

[5] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019.

[6] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, 2018.

[7] D. Arthur *et al.*, "Balanced k-means clustering on an adiabatic quantum computer," *Quantum Information Processing*, vol. 20, no. 9, pp. 1–30, 2021.

[8] "The Advantage™ Quantum Computer | D-Wave." [Online]. Available: https://www.dwavesys.com/solutions-and-products/systems/

[9] P. Date and T. Potok, "Adiabatic quantum linear regression," 2020.

[10] 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.

[11] P. Date, "Quantum discriminator for binary classification," 2020.

[12] M. Born and V. Fock, "Beweis des adiabatensatzes," *Zeitschrift für Physik*, vol. 51, no. 3-4, pp. 165–180, 1928.

[13] A. Lucas, "Ising formulations of many np problems," *Frontiers in physics*, vol. 2, p. 5, 2014.

[14] E. K. Grant and T. S. Humble, "Adiabatic quantum computing and quantum annealing," in *Oxford Research Encyclopedia of Physics*, 2020.

[15] W. Vinci, T. Albash, G. Paz-Silva, I. Hen, and D. A. Lidar, "Quantum annealing correction with minor embedding," *Physical Review A*, vol. 92, no. 4, p. 042310, 2015.

[16] Peter Bock, *Getting it right: R&D methods for science and engineering*. San Diego: Academic Press, 2001. ISBN 978-0-12-108852-1

[17] "Ridge — scikit-learn 1.0.2 documentation," https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html, (Accessed on 01/17/2022).

[18] "7.1. toy datasets — scikit-learn 1.0.1 documentation," https://scikit-learn.org/stable/datasets/toy_dataset.html, (Accessed on 11/29/2021).

[19] C. M. Bishop, "Pattern recognition," *Machine learning*, vol. 128, no. 9, 2006.

[20] "Randomizedsearchcv." [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html