Parametric Quantum Feature Selection Methods for Fraud and Default Detection

Sutapa Samanta, Dagen Wang, Todd Hodges, Andras Ferenczi

American Express Co.
(Sutapa.Samanta, Dagen.Wang, Todd.Hodges, Andras.L.Ferenczi1)@aexp.com

Abstract

Feature selection plays an important role in improving the efficiency of machine learning models for credit card fraud and default detection. We formulate the feature selection problem as a Quadratic Unconstrained Binary Optimization (QUBO) problem, which we solve using quantum annealers. We propose three new formulations based on this framework that improve the efficiency and flexibility of machine learning models. We benchmark the proposed methods, existing approaches from the literature, and also compare with classical feature-selection methods such as Random Forest feature importance and a combination of mutual information and Spearman correlation. Extensive experiments show that feature selection using QUBO on quantum computers consistently performs better than the other classical methods. Our experiments show the promise of using quantum computers in machine learning tasks in financial risk assessment applications.

Introduction

Artificial Intelligence (AI) and Machine Learning (ML) are becoming essential components of the business world, finding applications in healthcare, finance, cybersecurity, and marketing. ML models rely on high-dimensional data with noise and correlation. Training models on a big dataset with high dimensional features can be computationally expensive. Moreover, the presence of irrelevant or redundant features in the dataset can lead to overfitting, resulting in models that perform poorly on unseen data.

Feature Selection for Credit Card Fraud Detection

Feature selection(FS) is the crucial process of identifying and selecting a subset of relevant features for the specific problem from a dataset (Guyon and Elisseeff 2003). The FS problem is particularly relevant in transaction fraud detection (Fernandes et al. 2023) and default prediction. Transaction data often involves hundreds of features including but not limited to behavioral biometrics such as keystroke dynamics, mouse movements, touchscreen gestures; network including IP history, VPN/Proxy, ISP reputation; identity such as social media, similarity scores, account age; financial behavior such as credit utilization, loan inquiries, income verification; transaction history such as merchant risk, product anomalies, billing mismatches; AI features such as anomaly scores, synthetic IDs, fraud networks (Howard et al. 2022). These features interact in a non-linear fashion in a high dimensional space where dimensionality can be a limiting factor and redundancy can obscure critical variables.

Selecting a subset of the most relevant features is an NP-hard problem for classical computers (Chen B. 1997). As the number of features in the dataset increases, the solution space grows exponentially, making exhaustive search computationally infeasible for large-scale datasets. This complexity arises from the need to evaluate all possible combinations of features to identify the optimal subset that balances relevance and performance.

There are currently several methods to perform FS (Pudjihartono et al. 2022), which include filter methods, wrapper methods, and embedded methods. Filter methods select the most relevant features as they relate to the training labels. Examples include the Chi-square test, mutual information, correlation-based selection, variance threshold, etc.. Wrapper methods sequentially eliminate irrelevant features by training and testing a model with different subsets. Embedded methods train the model to become less dependent on the less important features by penalizing or ranking feature importance during training. These methods can be combined into hybrid methods that start with filter methods for feature pre-selection, followed by wrapper or embedded methods. Ensemble methods will employ techniques such as voting or weighted aggregation to select the most optimal features. We focus in this paper on leveraging combinatorial optimization for FS, which can be viewed as a hybrid between wrapper and embedded methods. Since the precise FS problem becomes intractable for classical computers as the feature dimensionality increases, recent advancements in quantum computing have shown promise in solving this problem.

Quantum Computers to Solve Optimization Problems

In recent years, quantum computers have gained significant attention for their potential to address problems that are challenging for classical computers alone (Au-Yeung, Chancellor, and Hallmann 2023). The objective is usually to leverage quantum parallelism and entanglement to explore combinatorial solution spaces more efficiently than classical algo-

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

rithms. For example, experiments with quantum simulators and NISQ devices have successfully handled graph-based problems with hundreds of nodes, outperforming classical approaches in scalability (Chatterjee, Bourreau, and Rančić 2024; Au-Yeung, Chancellor, and Hallmann 2023).

Quantum computing typically refers to computations performed on gate-based quantum computers, where quantum states are manipulated through the sequential application of quantum gates. Theoretically, these devices promise substantial speedups over classical computers for certain problems. However, current gate-based quantum computers, often referred to as NISQ (Noisy Intermediate-Scale Quantum) devices, face reliability issues due to noise and limitations in the number of qubits available. Research efforts to address these limitations, such as logarithmic qubit scaling (Mücke et al. 2023; Vlasic, Grant, and Certo 2025) and domain wall encoding (Au-Yeung, Chancellor, and Hallmann 2023; Mücke et al. 2023), are still in their early stages.

An alternative paradigm in quantum computing is quantum annealing (Ray, Chakrabarti, and Chakrabarti 1989; Kadowaki and Nishimori 1998). A quantum annealer is a specialized device designed to find the ground state of an Ising spin glass Hamiltonian. The process begins with the preparation of an initial state which is the ground state of a simple and well-known Hamiltonian. The system then evolves according to the time-dependent Schrödinger equation. At the end of the evolution, the system is expected to reach the ground state of the target Hamiltonian. Quantum annealers have been shown to be less affected by noise and capable of handling significantly larger system sizes compared to gate-based quantum computers. Although quantum annealers lack the versatility of gate-based quantum devices, they exhibit strong potential for solving QUBO problems with greater reliability.

In this paper, we perform FS using QUBO formalism. We use quantum annealers to solve QUBO. We train a ML model with the features selected by quantum computers and find its F1 score and gini on test sets. 'F1 score' provides a balanced measure of a ML model's predictive performance in a binary classification task. This score is widely used for unbalanced dataset. Gini measures the degree of discrimination of a model between the positive and negative classes. Gini is linearly related to AUC (Hastie, Tibshirani, and Friedman 2009). We compare results among different QUBO formalisms. We also compare them with two classical FS methods – (i) Random Forest feature importance, and (ii) FS based on mutual information and Spearman correlation. The first method is efficient for handling very large dataset and the second method is useful if the features have both linear and non-linear dependencies.

Related Research

An early white paper (Milne, Rounds, and Goddard 2017) compares an annealer-based QUBO optimization versus classical Recursive Feature Elimination (RFE) using the Statlog (German Credit Data) dataset to optimize FS for predictive models in credit risk assessment. The authors used a simple logistic regression-based model and concluded that the quantum feature selection (QFS) performed at par with the classical approach albeit with fewer selected features.

The paper (Vlasic, Grant, and Certo 2025) compares classical FS methods (e.g., ANOVA F-test, Chi-Square test) to QFS using quantum annealers by way of QUBO optimization. A comparison of models generated using features selected via classical versus quantum-based approach revealed that the latter outperformed all classical methods. In addition, the QFS-generated features produced more stable models.

Quantum annealing has been used to improve accuracy in recommender systems in (Niu et al. 2024; Nembrini, Ferrari da Crema, and Cremonesi 2021). Another study using the quantum annealing approach with QUBO formulations to identify critical features achieved better performance compared to classical methods (Mücke et al. 2023).

Even on classical hardware, quantum-inspired algorithms have been used to mimic quantum parallel amplitude estimation and amplitude amplification techniques in FS tasks. These methods have shown quadratic speedup in query complexity over their classical counterparts for supervised classification tasks (Chakraborty et al. 2020). In particular, a graph-theoretic quantum-inspired algorithm reduced dimensionality by evaluating feature correlations. Such methods outperform classical approaches in most benchmarks (Chakraborty et al. 2020).

QUBO Formalism

QUBO is an effective and widely used tool to address the FS problem. Let us consider a dataset with N features $f_1, f_2, \ldots, f_N$ and target variable y. The goal is to select a subset of the most important features out of these N features. Let us consider N binary variables $x_1, x_2, \ldots, x_N$ associated with each feature. Each of these binary variables takes values either 0 or 1, i.e., x_i is 1 when i^{th} feature f_i is selected, otherwise it is 0. Then the optimization problem translates to the following minimization problem.

$$\min H = \mathbf{x}^T Q \mathbf{x}\,, \tag{1}$$

where $\mathbf{x}$ is an N-dimensional vector and $\mathbf{x}^T$ is transpose of $\mathbf{x}$, i.e., $\mathbf{x}^T = \begin{pmatrix} x_1 & x_2 & \cdots & x_N \end{pmatrix}$. It is possible to select a desired number of features by adding a penalty term of the form $(\sum_i x_i - k)^2$. This term becomes zero when exactly k features are selected. The modified minimization problem becomes

$$\min H = \mathbf{x}^T Q \mathbf{x} + \left(\sum_i x_i - k \right)^2 . \tag{2}$$

Building the most effective QUBO is still an important area of research. There are several ways to build one. Three of them are widely used in literature, namely Pearson Correlation, SVC (support vector classifier) Boost, and MIQUBO (mutual information QUBO) (Turati, Dacrema, and Cremonesi 2022; Ferrari Dacrema et al. 2022). Among these three methods, MIQUBO performs better for most of our experiments. But it is harder to build, as it involves computing higher order conditional mutual information (CMI). The

computational time and memory requirements increase with the increase in sample size.

In this work, we propose three new methods to build QUBO – (i) the first method is based on mutual information computation; we call it "parametric-MI", (ii) the second method combines mutual information and Pearson correlation; we call it "MI-Pearson", (iii) the third method is using parameterized Distance correlation computation. These new methods use parametric formulation which introduces additional freedom in tuning the ML model.

Methods to Build QUBO

Pearson Correlation Building QUBO using Pearson correlation was originally proposed by (Ferreira and Figueiredo 2012) . This method aims to maximize the correlation between the feature and the target while minimizing the cross-correlations among the features themselves to reduce redundancy. Q-matrix using Pearson correlation is defined as follows.

$$Q_{ij} = \begin{cases} r(f_i, f_j) & \text{for } i \neq j\,, \\ -r(f_i, y) & \text{for } i = j\,. \end{cases} \tag{3}$$

The negative sign in the second line implies that the correlation between the feature and target is maximized when we minimize $\mathbf{x}^T Q \mathbf{x}$.

SVC Boosting This method was proposed in (Neven et al. 2012). First, each feature of a dataset is used to train support vector classifiers. Let the predictions of support vector classifiers are denoted by h_i. Then the Q-matrix is built by computing Pearson correlation as follows.

$$Q_{ij} = \begin{cases} r(h_i, h_j) & \text{for } i \neq j\,, \\ \frac{S}{|N|^2} + \lambda - 2 * r(h_i, y) & \text{for } i = j\,, \end{cases} \tag{4}$$

where S is the number samples and N is the number of features in the dataset. λ is a hyperparameter with a lower bound $\lambda < \frac{1}{N} + \frac{1}{N^2}$.

MIQUBO This Q-matrix is based on work of (Nguyen et al. 2014), and is defined as follows.

$$Q_{ij} = \begin{cases} -CMI(f_i, y|f_j) & \text{for } i \neq j\,, \\ -MI(f_i, y) & \text{for } i = j\,. \end{cases} \tag{5}$$

The first line of the above equation computes the conditional mutual information (CMI) between i^{th} feature and the target if j^{th} feature is selected. The second line computes mutual information (MI) between the feature and the target. Since mutual information is always non-negative, MIQUBO has a trivial solution where all features are selected. MIQUBO requires to specify the desired number of features to be selected by introducing a penalization term as shown in Eq. (2). This method is useful with dataset with interacting features as the second order mutual information or conditional mutual information captures the feature dependencies. But CMI is computationally expensive for very large dataset.

New Proposals to Build QUBO

Parametric-MI We propose the following mutual information based QUBO. This proposal is based on minimal-redundancy-maximal-relevance (mRMR) framework (Peng, Long, and Ding 2005) to minimize redundancy among the features and maximize the relevance of a feature for the expected outcome. The Q-matrix is defined as follows.

$$Q_{ij} = \begin{cases} (1-\alpha) MI(f_i, f_j) & \text{for } i \neq j\,, \\ -\alpha MI(f_i, y) & \text{for } i = j\,. \end{cases} \tag{6}$$

Here $0 \leq \alpha \leq 1$ is a free parameter that can be chosen based on the problem. Due to the above bound on the α value, the first line of Eq. 6 is always positive and the second line is always negative. As a result when we minimize Q, we minimize the mutual information between two features, thus minimizing redundancy, and maximize the mutual information between features and target, thus maximizing the relevance. When $\alpha = 1$, only the diagonal elements are non-zero. In this case, quantum annealing selects all the features in the absence of a penalty term. When $\alpha = 0$, the diagonal elements are zero, and annealer selects no feature. So varying alpha between zero and one is equivalent to changing number of selected features. We can also fix the value of α to fix the importance of diagonal and off-diagonal terms in the Q-matrix, then add a penalty term to select desired number of features. In Fig. 1, we have plotted F1 score and gini from XGBoost ML model as a function of the parameter α and the features constraints k. We see that when we chose to select a small number of features, selecting an optimal α is important for better score. But as we increase the number of selected features (k), ML scores become independent of the choice of α. We have also shown how F1 score and gini

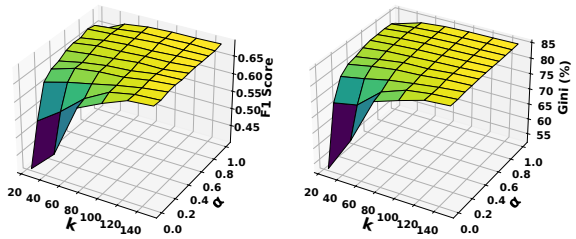

Figure 1: Plot showing how choice of scaling factor α in parametric-MI and number of features k in the penalty term influence the F1 score and gini of XGBoost ML model. When we choose small number of features, the F1 score and gini are heavily influenced by the choice of α. For larger values of k, choice of α becomes less significant. This plot is for the credit card default dataset (Howard et al. 2022).

from parametric-MI compare with other QUBO methods in Fig. 2, 3, and 4. It is evident from Fig. 3, and Fig. 4, that F1 score and gini score from parametric-MI method is comparable with MIQUBO method but without the increased computational complexity.

MI-Pearson Another way to maximize relevance and minimize redundancy is to formulate QUBO by combining the Pearson correlation and mutual information computation. While Pearson correlation captures linear relations between the features, mutual information is utilized to capture any non-linear relation among the features and class. This hybrid approach balances redundancy reduction and information maximization which can be useful for datasets where both linear and non-linear relationships exist.

$$Q_{ij} = \begin{cases} \alpha\, r(f_i, f_j) & \text{for } i \neq j\,, \\ -MI(f_i, y) & \text{for } i = j\,, \end{cases} \tag{7}$$

where $r(f_i, f_j)$ is the Pearson correlation coefficient between pairwise features, and α is a tunable penalty parameter. We set $\alpha = 1$ for all our experiments. As we will see later, this method often performs as per MIQUBO but much simpler to compute.

Parametric Distance Correlation Like mutual information, Distance correlation is a correlation metric which is sensitive to both linear and non-linear correlations. It was first introduced as an alternative to Pearson's correlation coefficient that would not only be sensitive to non-linear correlations but would also guarantee two variables were statistically independent for a coefficient value of 0. Unlike the Distance correlation coefficient, a Pearson coefficient values of 0 does not guarantee that two variables are statistically independent. The Distance correlation coefficient has a range of 0 to 1, where a value of 0 indicates completely independent variables, and a value of 1 indicates perfect correlation between variables.

Given that mutual information and Distance correlation measurements are both sensitive to non-linear correlations, the decision on which to use is primarily dictated by the parameters of the data being analyzed and computational complexity constraints. Relative to conditional mutual information, Distance correlation calculations typically require less computation time and are much better suited at handling continuous variables, as opposed to categorical data. In the context of QUBO problems, a Distance correlation matrix (Q) may be constructed as follows:

$$Q_{ij} = \begin{cases} DC(f_i, f_j) & \text{for } i \neq j\,, \\ -DC(f_i, y) & \text{for } i = j\,. \end{cases} \tag{8}$$

As in previous formulations of the correlation matrix, the off-diagonal elements of the matrix represent the Distance correlation (DC) between the features i and j. The diagonal elements of the matrix represent the negative Distance correlation between the i^{th} feature and the target variable (y).

As with the mutual information correlation matrix, the Distance correlation matrix may be parameterized as shown below.

$$Q_{ij} = \begin{cases} DC(f_i, f_j) & \text{for } i \neq j\,, \\ -\alpha DC(f_i, y) & \text{for } i = j\,. \end{cases} \tag{9}$$

Where $0 \leq \alpha \leq 1$ and, again, acts as a scaling factor that emphasizes more importance on feature-target correlations or feature-feature correlations. Although the implementation of the parameterization is slightly different than that shown for parametric-MI, the general effect is approximately equivalent and leads approximately equal.

Experiments

Quantum Processor

Quantum systems tend to seek the lowest energy configuration. Quantum annealers leverage this property to find solutions to the real world optimization problems. Quantum annealing is a method of quantum computing that uses the combination of thermal fluctuations and quantum tunneling to find the lowest energy state or ground state of the system Hamiltonian. When we map a real-life optimization problem to a Hamiltonian and perform quantum annealing, the resulting ground state is the solution to our optimization problem.

We use quantum annealers (Condello 2022) for our experiments. Current quantum annealers (McGeoch and Farré 2022) have limited connectivity and can only solve small or sparsely connected QUBO. Since we use large real world datasets containing hundreds of features and densely connected QUBO, quantum processors (QPU) are not suitable for our work. We use Hybrid Solver for our experiments. Hybrid solver uses both classical and quantum solver (Raymond et al. 2023). It decomposes larger optimization problem into smaller ones to run them on QPU.

Classical Data

We perform experiments on three publicly available credit card datasets from Kaggle[1]. The first one is a credit card fraud dataset which contains transactions made by credit cards in September 2013 by European cardholders. This dataset comes with 30 features and about 300K samples (ULB 2017).

The second data is a larger credit card fraud dataset with 433 features and 144K samples. The data comes from Vesta's real-world e-commerce transactions and contains a wide range of features from device type to product features (Howard et al. 2019).

The third one is the credit default dataset (Howard et al. 2022). The credit default dataset contains thirteen months of credit default data denoted by 'monthly-0', 'monthly-1' and so on. Each month's data contains various credit default related profiles. The target variable is binary, representing whether a customer defaults. This dataset also contains separate data files for training and testing. We use only the last month data 'monthly-12' for our experiments.

We remove features with non-numeric values, zero variance and all NAN values before building QUBO. We build QUBO using the methods described in the last section and optimize them in a quantum annealer. For the first two datasets, we split them into 80% - 20% train and test data. For the credit default dataset, we use train data 'monthlytrain-12' for the training and test data 'monthlytest-12' for the testing. We then train XGBoost (Chen and Guestrin 2016) ML model with the selected features and compute there scoring indices 'F1' and 'gini'. These two are widely used metrics in the financial services industry. We

[1]https://www.kaggle.com/

also perform FS from two classical methods – (i) Random Forest with feature importance, and (ii) mutual information + Spearman correlation and compare their F1 score and gini with the quantum features. For the completing, we also train the model with the full dataset and compute their F1 score and gini.

Results

In this section, we present the results from our FS experiments using different QUBO formulations across three credit card datasets.

Small Credit Card Fraud Data (ULB 2017)

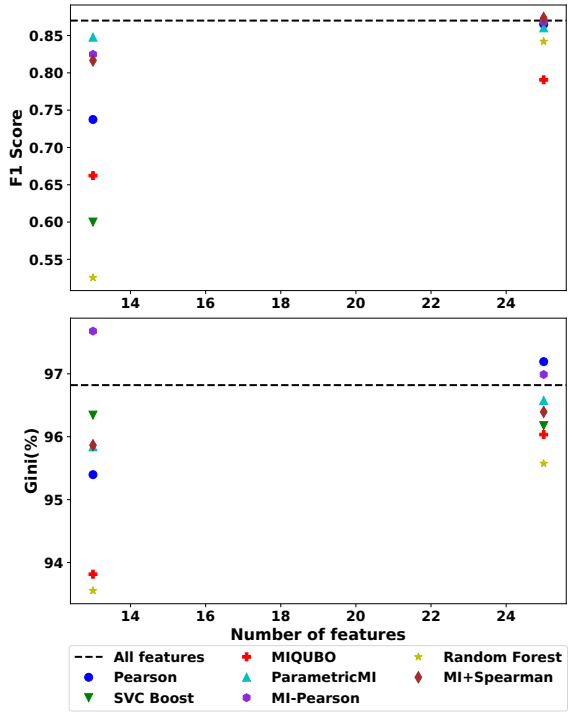

Figure 2: Plot of F1 score and gini with number of features for different methods for the small credit fraud dataset (ULB 2017). Numerical value of all features F1 score and gini is represented by the black dashed line. With $k = 13$ selected features, parametric-MI performs best for F1 score and MI-Pearson works best for gini. With $k = 25$ selected features, classical MI-Spearman dominates for F1 score and Pearson QUBO dominates for gini.

Metric	k	best value	method
F1	25	0.87356	MI-Spearman
Gini	13	97.67777	MI-Pearson

Table 1: Best XGBoost scores for each metric and corresponding FS methods.

Experimental result for the Kaggle credit card fraud data (ULB 2017) is illustrated in Fig. 2. The best ML scores are also tabulated in Table 1. We consider subsets of size $k = 13$ and $k = 25$ out of a total of 30 features. These results were compared against the baseline model trained using all 30 features. Interestingly, the newly proposed parametric-MI QUBO has F1 score better than the conventional MIQUBO for this dataset. Most notably, gini score is the best with newly proposed MI-Pearson QUBO method with only 13 selected features. In fact, gini score for MI-Pearson QUBO with only 13 selected features outperforms all other methods including the performance by the full dataset. The reason for this achievement can be explained as follows. This dataset contains features with both linear and non-linear dependencies among themselves and with the target. MI-Pearson QUBO, by virtue of its construction, was able to capture these relationships. As a result, it was able to select most important subset of features and eliminate features those are harmful for ML performance. Gini score with 25 quantum features selected by Pearson correlation QUBO is higher than the gini of the full dataset. MI-Pearson QUBO formulation also performs better than the full dataset. MIQUBO is the worst performing technique for this data, likely because MIQUBO excels in capturing higher order interactions but fails to capture the linear relationship among the features. Among the classical methods, MI-Spearman performs well when large number of features are selected.

Large Credit Card Fraud Data (Howard et al. 2019)

Metric	k	best value	method
F1	204	0.77935	Parametric-MI
Gini	198	92.71444	MI-Pearson

Table 2: Best XGBoost scores for each scoring metric and corresponding methods.

Fig. 3 summarizes the FS results on a more complex fraud dataset (Howard et al. 2019). The best ML scores are also tabulated in Table 2. This original dataset has a total of 433 features. After the pre-processing of data, e.g., removal of features with zero variance, large NAN fractions etc., the working dataset contains 280 features. We evaluate subsets of the features with number of selected features (k) ranging from 25 to 200 in increments of 25. For smaller number of selected features, MIQUBO formulation performs better in terms of both F1 score and gini. This suggests that this dataset has features with higher order interactions, which are best captured using MIQUBO. Thus MIQUBO was able to selected the best features for the ML. As we increase the subsystem size, other methods approach and sometimes slightly exceed the baseline. Specifically, MIQUBO, MI-Pearson, and parametric-MI outperform the baseline by a little margin.

Credit Card Default Data (Howard et al. 2022)

FS result for the credit card default data from Kaggle (Howard et al. 2022) is illustrated in Fig. 4. The best scores are listed in Table 3. The dataset for this experiment contains 182 features. After removing non-numeric

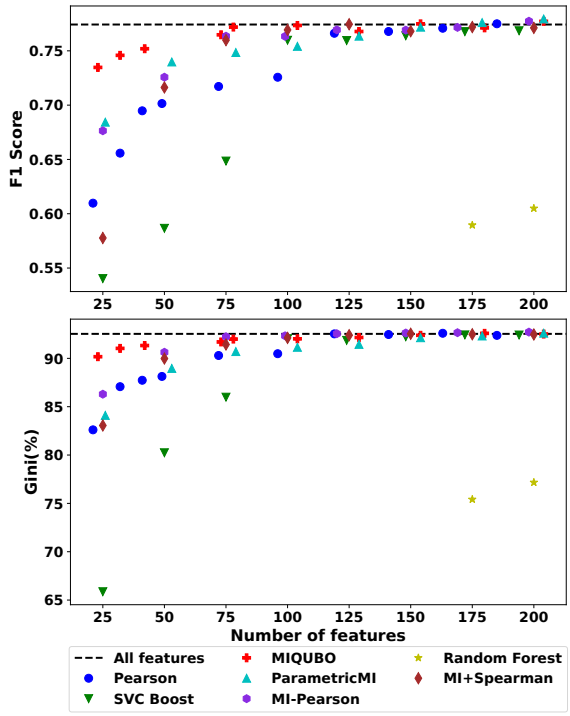

Figure 3: Plot of F1 score and gini with number of features for different methods for the dataset (Howard et al. 2019). Quantum features selected using MIQUBO performs better than all other methods, especially for small number of features.

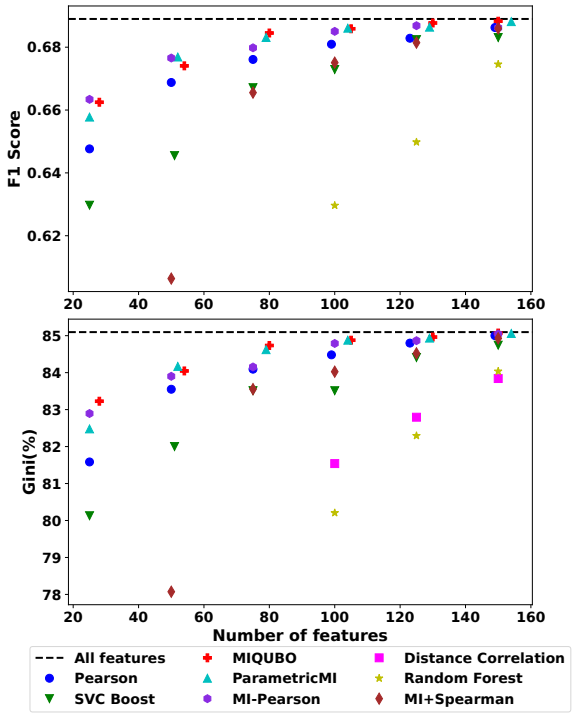

Figure 4: Plot of F1 score and gini with number of features for different methods for the dataset (Howard et al. 2022). Features selected using MIQUBO and parametric-MI show better performances than all other methods.

Metric	k	best value	method
F1	150	0.68827	MIQUBO
Gini	150	85.07098	MIQUBO

Table 3: Best XGBoost scores for each scoring metric and corresponding methods.

features and features with zero variance, 174 features remained. We select subsets of feature with k ranging from 25 to 150 in the interval of 25. For smaller subsets of selected features, MIQUBO, parametric-MI QUBO, and MI-Pearson QUBO have similar ML performance. This is encouraging as both the newly proposed parametric-MI QUBO and MI-Pearson QUBO have lower computational complexity than traditional MIQUBO, making them more practical to use for large dataset. As we increase the number of selected features, other methods also converge with the baseline given by full dataset trained ML model. This is expected because, as we approach the feature count of the full dataset, the significance of feature selection reduces.

QUBO on Classical Simulator

We performed FS on a simulated annealing(SA) using all the QUBO formulations discussed above using the credit default dataset (Howard et al. 2022). We find that selected features using simulated annealing has similar F1-score and gini for all the QUBO formulations. Fig. 5 shows the comparison of ML performance on feature selected by quantum annealing and simulated annealing. Other QUBO formulations show similar pattern (See Appendix).

Conclusion

We find that the performance of QUBO-based feature selection method is dataset dependent. The MI-Pearson QUBO excels in datasets where features have both linear and nonlinear relations as we saw for the small credit fraud dataset. In contrast, MIQUBO has an advantage when the features have more complex and higher order interactions as seen from the large credit fraud data. Meanwhile parametric-MI QUBO offers a balance between performance and computational efficiency, making it an alternative to MIQUBO when the resources are limited. The results highlight the importance of selecting QUBO formulations tailored to the characteristics of the dataset to maximize ML performance.

Discussions

Our results have shown that FS using QUBO on a Quantum annealer consistently performs better than popular classical features selection methods. Also, for almost all the cases, training ML with quantum selected features performs better than training it with the full dataset. Even if the margin of improvement compared to baseline is small, it can result in significant revenue growth for financial institutions. These

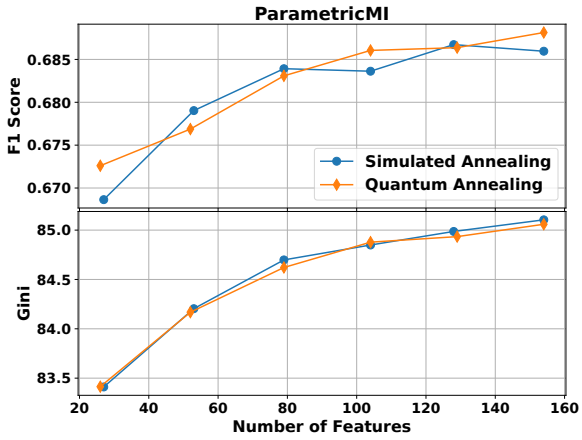

Figure 5: Comparison of XGBoost performance between FS using QA and SA using Parametric-MI QUBO on credit-default dataset.

achievements with current noisy and limited qubit quantum computer presents opportunity for further improvements as quantum computers continue to improve.

An additional point of interest is the increase in overall F1 score and gini scores when introducing parameterization; especially when the number of allowable features for selection is constrained to a relatively small number in comparison to features available. One possible explanation is that the observed increase in performance stems from the initial changes made to the diagonal elements of the correlation matrices used. For the diagonal elements of the correlation matrices to be used within the context of a minimization problem, it was necessary to make them all negative, as seen in the Q-matrix equations throughout the paper. In doing so, the mathematical distance (absolute value of difference) between feature-target correlations and feature-feature correlations is increased artificially. When a large number of features are selected, these small distance increases become negligible and have little to no effect on the final results. However, when constrained to a small number of features, the added distance has a more significant impact. As a result, it is possible that the scaling factor provides a form of correction for the artificial distance increase. Given that experimental observations show an increase in F1 score when the feature-target correlation values are reduced at low feature numbers, the interpretation seems plausible.

The features selected using hybrid quantum annealing (QA) and simulated annealing (SA) exhibit comparable performance across all tested QUBO formulations. This similarity arises because both approaches employ analogous mechanisms for escaping local minima in search of global optima. Specifically, SA exploits thermal fluctuations, whereas QA leverages quantum fluctuations and tunneling. While quantum tunneling can provide an advantage to QA in certain problem instances, such benefits are less pronounced when the energy landscape is relatively simple, characterized by wide and shallow barriers, where SA performs equally well. In practice, current quantum annealers operate within finite annealing times, often violating the strict adiabatic condition, and are subject to environmental noise. Decoherence effects can suppress coherent quantum tunneling, causing the resulting dynamics to resemble classical thermal hopping. Nevertheless, for problem instances that map to energy landscapes with tall and narrow barriers, and provided that coherence in quantum annealers is significantly improved, QA-based QUBO formulations are expected to surpass classical techniques in performance.

Acknowledgments

We wish to express our gratitude to Mariya Bessonov, and Adel Dayarian, who provides great insights on validating quantum algorithm results and many helpful discussions. We recognize Krishna Sunkara for helping setting up computing environment. We would also like to acknowledge Amol Salunkhe for the discussion on higher order of correlations.

Appendix

We present the result to show how FS using SA compare with FS using QA using different QUBO formalism.

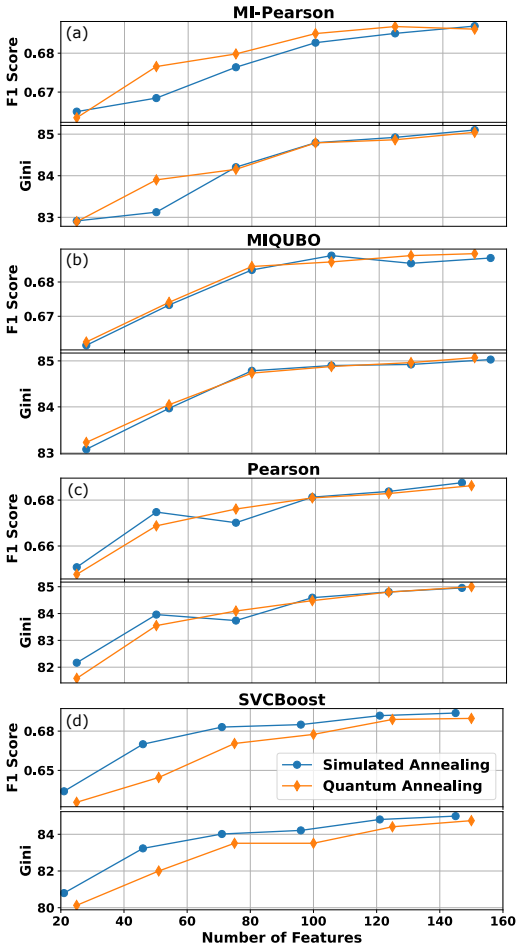

Figure 6: The comparison between performance of QA and SA feature selection shown for (a) MI-Pearson QUBO, (b) MIQUBO, (c) Pearson correlation, and (d) SVC Boost.

References

Au-Yeung, R.; Chancellor, N.; and Hallmann, P. 2023. NP-hard but no longer hard to solve? Using quantum computing to tackle optimization problems. *Frontiers in Quantum Science and Technology*, 2.

Chakraborty, S.; Shaikh, S. H.; Chakrabarti, A.; and Ghosh, R. 2020. A hybrid quantum feature selection algorithm using a quantum inspired graph theoretic approach. *Applied Intelligence*, 50(5): 1775–1793.

Chatterjee, Y.; Bourreau, E.; and Rančić, M. J. 2024. Solving various NP-Hard problems using exponentially fewer qubits on a Quantum Computer. arXiv:2301.06978.

Chen, T.; and Guestrin, C. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, 785–794. New York, NY, USA: ACM. ISBN 978-1-4503-4232-2.

Chen B., H. J. . W. Y. 1997. The minimum feature subset selection problem. *J. of Comput. Sci. & Technol*, 12: 145–153.

Condello, A. 2022. contributors. dwave-system; 2018.

Fernandes, A. A. A.; et al. 2023. Feature Engineering Strategies for Credit Card Fraud Detection. *arXiv preprint*.

Ferrari Dacrema, M.; Moroni, F.; Nembrini, R.; Ferro, N.; Faggioli, G.; and Cremonesi, P. 2022. Towards Feature Selection for Ranking and Classification Exploiting Quantum Annealers. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, 2814–2824. New York, NY, USA: Association for Computing Machinery. ISBN 9781450387323.

Ferreira, A. J.; and Figueiredo, M. A. 2012. Efficient feature selection filters for high-dimensional data. *Pattern Recognition Letters*, 33: 1794–1804.

Guyon, I.; and Elisseeff, A. 2003. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3(null): 1157–1182.

Hastie, T.; Tibshirani, R.; and Friedman, J. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2 edition. ISBN 978 - 0387848570.

Howard, A.; Aritra; Xu, D.; Vashani, H.; inversion; Negin; and Dane, S. 2022. American Express - Default Prediction. https://kaggle.com/competitions/amex-default-prediction. Kaggle.

Howard, A.; Bouchon-Meunier, B.; CIS, I.; inversion; Lei, J.; Vesta, L.; Marcus; and Abbass, H. 2019. IEEE-CIS Fraud Detection. https://kaggle.com/competitions/ieee-fraud-detection. Kaggle.

Kadowaki, T.; and Nishimori, H. 1998. Quantum Annealing in the Transverse Ising Model. *Physical Review E*, 58(5): 5355–5363.

McGeoch, C.; and Farré, P. 2022. Advantage processor overview. *D-Wave systems*, 14–105.

Milne, A.; Rounds, M.; and Goddard, P. 2017. Optimal Feature Selection in Credit Scoring and Classification Using a Quantum Annealer. White Paper.

Mücke, S.; Heese, R.; Müller, S.; Wolter, M.; and Piatkowski, N. 2023. Feature Selection on Quantum Computers. *Nature Quantum Machine Intelligence*, 5(11).

Nembrini, R.; Ferrari da Crema, M.; and Cremonesi, P. 2021. Feature Selection for Recommender Systems with Quantum Computing. *Entropy*, 23(8): 970.

Neven, H.; Denchev, V. S.; Rose, G.; and Macready, W. G. 2012. QBoost: Large Scale Classifier Training with Adiabatic Quantum Optimization. In *Proceedings of the Asian Conference on Machine Learning*, volume 25 of *Proceedings of Machine Learning Research*, 333–348. PMLR.

Nguyen, X. V.; Chan, J.; Romano, S.; and Bailey, J. 2014. Effective global approaches for mutual information based feature selection. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, 512–521. New York, NY, USA: Association for Computing Machinery. ISBN 9781450329569.

Niu, J.; Li, J.; Deng, K.; and Ren, Y. 2024. CRUISE on Quantum Computing for Feature Selection in Recommender Systems. *arXiv preprint arXiv:2407.02839*.

Peng, H.; Long, F.; and Ding, C. 2005. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8): 1226–1238.

Pudjihartono, N.; Fadason, T.; Kempa-Liehr, A. W.; and O'Sullivan, J. M. 2022. A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction. *Frontiers in Bioinformatics*, 2: 927312.

Ray, P.; Chakrabarti, B. K.; and Chakrabarti, A. 1989. Sherrington-Kirkpatrick model in a transverse field: Absence of replica symmetry breaking due to quantum fluctuations. *Phys. Rev. B*, 39(16): 11828.

Raymond, J.; Stevanovic, R.; Bernoudy, W.; Boothby, K.; McGeoch, C. C.; Berkley, A. J.; Farré, P.; Pasvolsky, J.; and King, A. D. 2023. Hybrid Quantum Annealing for Larger-than-QPU Lattice-structured Problems. *ACM Trans. Quant. Comput.*, 4(3): 17.

Turati, G.; Dacrema, M. F.; and Cremonesi, P. 2022. Feature Selection for Classification with QAOA. In *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 782–785.

ULB. 2017. Credit Card Fraud Detection. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data. Kaggle.

Vlasic, A.; Grant, H.; and Certo, S. 2025. Feature selection through quantum annealing. *The Journal of Supercomputing*, 81(147).