ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2.
Available from http://www.i6doc.com/en/.

Approximating Archetypal Analysis Using Quantum Annealing

Sebastian Feld, Christoph Roch, Katja Geirhos, Thomas Gabor

LMU Munich - Mobile and Distributed Systems Group
Oettingenstr. 67, 80538 Munich - Germany

Abstract. Archetypes are those extreme values of a data set that can jointly represent all other data points. They often have descriptive meanings and can thus contribute to the understanding of the data. Such archetypes are identified using archetypal analysis and all data points are represented as convex combinations thereof. In this work, archetypal analysis is linked with quantum annealing. For both steps, i.e. the determination of archetypes and the assignment of data points, we derive a QUBO formulation which is solved on D-Wave's 2000Q Quantum Annealer. For selected data sets, called *toy* and *iris*, our quantum annealing-based approach can achieve similar results to the original R-package "archetypes".

1 Introduction

Consider a database storing characteristics of soccer players such as speed, accuracy or height. An extreme data point could represent a player that is not fast, but big and strong. Intuitively, these characteristics can represent a defender. Other extreme data points may be interpreted as striker, goalkeeper or midfielder. All remaining data points can either be assigned to one of these categories or a combination thereof. Archetypal Analysis (AA) describes how to find such extreme data points. These so-called archetypes should be chosen such that all data points can be represented as a convex combination of them. A representation is good if the approximation is similar to the original data point. Existing implementations of AA use an iterative approximation algorithm [1, 2]. Thus, there is interest in studying further approaches to AA, as it may help finding better solutions or solutions faster. With this paper, we combine AA with Quantum Annealing (QA). QA is a metaheuristic for solving optimization problems that incorporates quantum effects and that is available as specialized hardware requiring the input to be in the form of a quadratic unconstrained binary optimization (QUBO) problem. The functional form of a QUBO problem is: minimize $x^T Qx$ with $x_i \in \{0, 1\}^n$ being a binary vector of size n representing spins in a quantum system and Q being an $n \times n$ real-valued matrix describing the variables' relationship. Given matrix Q, the annealing process tries to find binary variable assignments to minimize the objective function. QA can be used to solve various optimization problems. [3] presents Ising formulations for Karp's 21 NP-complete problems, as this is another input format of current QA hardware. [4] uses a QUBO formulation for portfolio optimization based on Markowitz's modern portfolio theory. [5] describes prime factorization as an optimization problem while [6] proposes a method to transform any factorization problem into an Ising model.

2 Archetypal Analysis (AA)

AA is a data analysis method focussing on a dataset's extreme points [1]. It identifies archetypes such that all data points can be represented as a convex combination thereof. Archetypes are not necessarily data points, but are limited to being convex combinations of them. This can be stated as following constrained minimization problem [2, 7]: $RSS = \left\|X - \alpha Z^T\right\|_F$ with $Z = X^T\beta$ s.t. $\sum_{j=1}^{k} \alpha_{ij} = 1$ and $\sum_{i=1}^{n} \beta_{ij} = 1$ with $\alpha_{ij}, \beta_{ij} \geq 0$, $i = 1, \ldots, n$ and $j = 1, \ldots, k$.

The $n \times m$-matrix X holds n observations with m attributes each, $m \times k$-matrix Z represents the k archetypes and $n \times k$-matrix α assigns the data points to the archetypes. $\left\|\cdot\right\|_F$ is a suitabe matrix norm, in our case the Frobenius norm [2]. The RSS to be minimized can be regarded as solution quality. The first constraint states that data points are represented as convex combinations of archetypes, i.e. $\tilde{X} = \alpha Z^T$ with $\tilde{X}$ being the representation of data points. The second constraint states that archetypes are convex combinations of data points, i.e. $Z = X^T\beta$ with β being an $n \times k$-matrix of coefficients.

Since α_{ij} and β_{ij} are real numbers between zero and one, there are unlimited value assignments in general resulting in the impracticality to try all combinations to find the best one. For this, [1] have developed an approximation algorithm for AA. Based on that, [2] have implemented AA in programming language R and introduced a corresponding package. After initial selection of archetypes Z, they alternately repeat the following steps: (1) calculate best assignment matrix α for given archetypes Z and (2) calculate best archetypes Z for given matrix α. This process is repeated until a maximum number of iterations is reached or the decrease in RSS is below a threshold [2]. This algorithm is nondeterministic as the result depends on the initial archetypes and there is no guarantee to find optimal archetypes or assignments [2].

AA is used to analyze athletes [7], scientists [8], head shapes [1] or alternative routes [9]. Extensions to AA focus on robustness against outliers [10] or handling missing values [11] or nominal data [12].

3 Concept

AA provides optimal archetypes and assignments. Thus, there are two optimization problems that can either be solved using a single QUBO form or two separate ones. Preliminarily, we represented AA as a single QUBO only by neatly transforming the minimization function with all constraints. However, numerous auxiliary variables were introduced making this approach unfeasible. Instead, a vivid consideration of what makes good archetypes led to a simpler and smaller QUBO where archetypes and assignments are calculated separately.

3.1 Part 1: Archetypes

When choosing $k > 1$ archetypes for given data set X, [1] showed that optimal archetypes lie on the data set's convex hull. We also observe that archetypes are not too close to each other, but distributed over the convex hull. Assuming the

ESANN 2020 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Online event, 2-4 October 2020, i6doc.com publ., ISBN 978-2-87587-074-2. Available from http://www.i6doc.com/en/.

high distances between archetypes as a good choice in general, finding archetypes can be rephrased as: choose k data points as archetypes so that their distances are maximal. Note that assuming that high distances make up good choices of archetypes highly simplifies the problem. While AA approximates the convex hull through k points, we look for k data points with maximal distance to each other. However, these two problems seem related, as the respective solutions are very similar in many cases (see Sec. 4).

As stated, we restrict ourselves so that only data points can be archetypes and not, as originally defined, also convex combinations thereof. This limitation leads to a lower number of decision variables and thus to a simpler and faster calculation. With this restriction, we propose to use the following minimization function:

$$\textit{minimize} \;\; y = -\sum_{i=1}^{n}\sum_{j=1}^{n} B_i \cdot B_j \cdot d\left(P_i, P_j\right) + C_1 \cdot \left(\sum_{i=1}^{n} B_i - k\right)^2 \tag{1}$$

Here, vector B is defined holding n binary decision variables with n being the number of data points. $B_i = 1$ means that data point i is an archetype and $B_i = 0$ if not. $d(P_i, P_j)$ is the Euclidean distance between data points P_i and P_j. However, the first term of Eq. 1 is minimal if all data points are selected as archetype. Thus, an additional constraint is needed that enforces that exactly k archetypes are selected. Since QUBO formulations do not allow constraints, this condition must be transformed into a penalty added to objective function y. It is zero iff the constraint is met, otherwise a positive value worsens the solution quality such that the solution is no longer the best. The resulting penalty is depicted as the second term of Eq. 1. Constant C_1 weights the penalty term and must be chosen appropriately for each data set. It must be high enough so that the solution quality for all assignments that do not meet the condition will severely be degraded.

Finally, Eq. 1 can be rewritten into an $n \times n$-matrix Q where $d\left(P_i{,}P_j\right)$ is replaced by the actual distances between the respective data points and parameters C_1 and k are accordingly assigned with specific values. Then, the final minimization problem is in QUBO form of $y = B^T Q B$.

3.2 Part 2: Assignments

This part represents each data point as a convex combination of the k archetypes, e.g. "P_1 is 0.75 times Z_1 plus 0.25 times Z_2 plus 0 times Z_3". The goal is to find an assignment matrix α for data set X with given archetypes Z where α_{ij} specifies to what degree data point i is assigned to archetype j. The following must apply [2]: entry α_{ij} is a real number $0 \leq \alpha_{ij} \leq 1$ and $\sum_{j=1}^{k} \alpha_{ij} = 1$ for all $1 \leq i \leq n$.

The assignment of individual data points is independent and can therefore be optimizied separately using a QUBO formulation. Since QUBO is suitable only for combinatorial optimization problems and finding the mapping is continuous, real-valued α_{ij} must first be discretized. For a dynamic fit we introduce precision

g, an input parameter that specifies how many decision variables are used per data point per archetype. This results in $g \cdot k$ decision variables needed for each data point. Thus, we introduce the following minimization function:

$$\textit{minimize}\; y_i = \sum_{j=1}^{m} \left(X_{ij} - \sum_{l=1}^{k} \alpha_{il} \cdot Z_{lj}^T \right)^2 + C_2 \cdot \left(\sum_{s=1}^{g \cdot k} B_s - g \right)^2 \qquad (2)$$

A data point's representation as a convex combination of archetypes is good if it is as similar as possible to the data point, i.e. the sum of squared deviations (RSS) is minimal. This results in the first term of Eq. 2 for each data point i. Vector B of binary decision variables B_i is defined as $B = (B_1, \ldots, B_{g \cdot k})$ where each block of g variables has the same meaning, i.e. we count the number of variables per block having value 1, divide it by g and interpret this value as the proportion of the associated archetype j for data point i. The condition that the sum of α values for a data point equals one must also apply. The combination of the two conditions and a conversion to a penalty term modeled as [13] yields the second term of Eq. 2. C_2 weights the penalty term and must be chosen appropriately. This defines the framework conditions for the assignments, but the actual optimization is still pending. Again, the two parts (enforcing convex combination and minimizing deviation) of Eq. 2 are rewritten into a $g \cdot k \times g \cdot k$ matrix Q where corresponding entries of Z^T and X are used for a specific data set and parameter g and weight C_2 must be chosen appropriately.

In summary, our approach requires to solve multiple QUBO formulations. Finding archetypes results in an $n \times n$ matrix Q with n being the number of data points (Eq. 1). For the assignments we solve n QUBO formulations each having size $g \cdot k \times g \cdot k$ (Eq. 2). Thus, a total of $n + 1$ QUBO formulations must be solved for a data set with n data points.

4 Evaluation

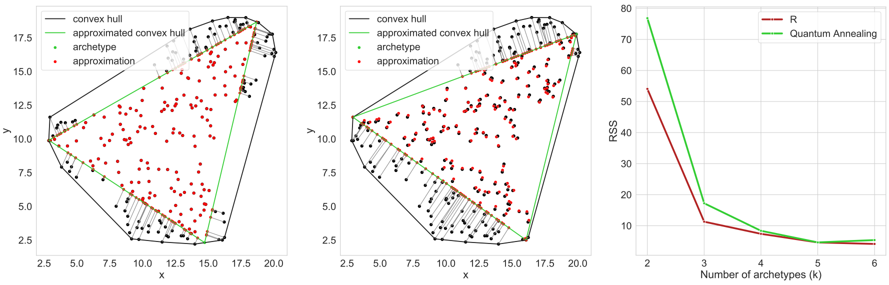
We evaluate our approach against R-package "archetypes". Based on the $2D$ dataset *toy* [2] it is shown how both methods behave for different values of $2 \leq k \leq 6$. Since both methods are non-deterministic, the calculation is performed ten times for each k. The average solution quality $\varnothing RSS$ and standard deviation σ are calculated from these results. In addition, RSS_{min} as the smallest value obtained is given. The results are presented in Fig. 1a and Fig. 1b. The average solution quality shows for all values of k that the method of R-package "archetypes" performs better. However, for most values of k, the differences are insignificantly small, especially for $k = 5$. Also regarding the best found solutions, the differences in quality are not particularly large. For most values of k, RSS_{min} is smaller with R. However, our QA approach provides a slightly better result for $k = 4$. Interestingly, the standard deviation of R for $k \leq 3$ is exactly 0. In these cases the same result was always found. For larger values of k the standard deviation increases. This is different regarding our QA heuristic: Here, the standard deviation is relatively large only for $k = 2$ and otherwise quite small. Overall, the standard deviations are in similar orders of magnitude.

k	RSS_{min}	$\varnothing RSS$	σ
2	54.074	54.074	0.000
3	11.307	11.307	0.000
4	6.844	7.394	1.119
5	2.525	4.627	2.243
6	1.692	4.138	1.977

(a) *RSS with R*

k	RSS_{min}	$\varnothing RSS$	σ
2	67.732	76.837	5.960
3	17.237	17.238	0.001
4	6.828	8.376	1.542
5	4.568	4.642	0.045
6	5.316	5.387	0.051

(b) *RSS with QA*



(c) *Convex hull with R* (d) *Convex hull with QA* (e) *Scree plot of R and QA*

Fig. 1: Results for the dataset *toy*.

For determining how many archetypes describe data set *toy* best, a scree plot is shown for both methods. For this, the number of archetypes k was plotted against the respective average value of *RSS* (see Fig. 1e). It shows that for both methods the average value of *RSS* decreases strongly for $k \leq 3$. These "elbows" at $k = 3$ indicate that data set *toy* can best be described by three archetypes. For comparing the best solutions for $k = 3$, they are visually shown in Fig. 1c and Fig. 1d. It can be seen that the chosen archetypes from R are slightly better. However, the assignments seem to be similarly good for both methods.

Besides data set *toy*, we also used dataset *iris* [14] to evaluate our QA approach. The data set consists of 150 measurements of iris blossoms, all of which belong to one of three different species. An interesting feature of the dataset is that species 1 is well distinguishable from the others, while species 2 and 3 are more similar [15]. This challenges methods trying to predict species affiliation based on the measurement data. From the descriptive meaning one expects that dataset *iris* can best be described by three archetypes, whereby each archetype should represent one species.

The analysis showed that for both methods the average value of *RSS* is similarly good (R 6.366 and QA 6.437), however, R performs slightly better. Nevertheless, the best *RSS* value was obtained with our QA method (QA 6.192 and R 6.366). Thus, regarding only solution quality, both methods perform equally good. Regarding the results w.r.t number of archetypes, R performed as expected, since all three archetypes belonged to different species. This was not the case with our QA method. Two archetypes belonged to species 1, one archetype to species 2 and species 3 was not considered. This contradicts the vivid meaning. However, it is astonishing that nevertheless the solution quality is not worse than the results from *R*.

5 Conclusion

This paper proposes to approximate archetypal analysis using quantum annealing. For this, QUBO formulations were provided and compared to the R-Package "archetypes" for data sets with different dimensions. Overall, the results of our QA method are comparable to those from R. Even though the average solution quality of R was usually better, there were some cases for which a slightly better result could be achieved using our QA method.

Regarding ongoing and future research we want to extend our QA method in order to find the number of archetypes which describe the data best in advance. That implies to include the so-called "elbow criterion" into our QUBO formulation. Another application of our QA approach would be to find the farthest points in a point cloud, which is also a quite complex problem and a vital topic in computational science.

References

[1] Adele Cutler and Leo Breiman. Archetypal analysis. *Technometrics*, 36(4):338–347, 1994.

[2] Manuel Eugster and Friedrich Leisch. From spider-man to hero-archetypal analysis in R. 2009.

[3] Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2:5, 2014.

[4] Davide Venturelli and Alexei Kondratyev. Reverse quantum annealing approach to portfolio optimization problems. *Quantum Machine Intelligence*, 1(1-2):17–30, 2019.

[5] Catherine C McGeoch. Adiabatic quantum computation and quantum annealing: Theory and practice. *Synthesis Lectures on Quantum Computing*, 5(2):1–93, 2014.

[6] Shuxian Jiang, Keith A Britt, Alexander J McCaskey, Travis S Humble, and Sabre Kais. Quantum annealing for prime factorization. *Scientific reports*, 8(1):17667, 2018.

[7] Manuel JA Eugster. Archetypal athletes. *arXiv preprint arXiv:1110.1972*, 2011.

[8] Christian Seiler and Klaus Wohlrabe. Archetypal Analysis: Ein neuer Ansatz zur Klassifizierung von Wissenschaftlern. *ifo Schnelldienst*, 65(22):7–12, 2012.

[9] Sebastian Feld, Martin Werner, Mirco Schönfeld, and Stefanie Hasler. Archetypes of alternative routes in buildings. In *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10. IEEE, 2015.

[10] Manuel JA Eugster and Friedrich Leisch. Weighted and robust archetypal analysis. *Computational Statistics & Data Analysis*, 55(3):1215–1225, 2011.

[11] Irene Epifanio, M Victoria Ibáñez, and Amelia Simó. Archetypal analysis with missing data: see all samples by looking at a few based on extreme profiles. *The American Statistician*, pages 1–28, 2019.

[12] Sohan Seth and Manuel JA Eugster. Archetypal analysis for nominal observations. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):849–861, 2015.

[13] Fred Glover, Gary Kochenberger, and Yu Du. A tutorial on formulating and using QUBO models. 2019.

[14] Edgar Anderson. The irises of the gaspe peninsula. *Bull. Am. Iris Soc.*, 59:2–5, 1935.

[15] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.