Setting Vector Quantizer Resolution via Density Estimation Theory

Josh Taylor(✉) and Stella Offner

The University of Texas at Austin, Austin, TX, USA
joshtaylor@utexas.edu

Abstract. We introduce a framework for selecting the number of codebook vectors in a vector quantizer based on local characteristics of the data density, the degree to which the process of VQ distorts the representation of this density, and the theoretical efficiency of estimators of these densities. In our analysis, L^2 theory from kernel density estimation relates the number of VQ prototypes to observed sample size, dimension, and complexity, all of which intuitively influence codebook sizing.

Keywords: Vector Quantization · Variable Kernel Density Estimation

1 Introduction

Vector Quantization (VQ, [11]) is a technique for data encoding and compression used for a variety of pattern matching tasks in many domains. Given a d-dimensional sample of data $X = \{x_s \in \mathbb{R}^d\}_{s=1}^N$ drawn from density $f(x)$, VQ algorithms learn a set of *prototype* or *codebook* vectors $W = \{w_i \in \mathbb{R}^d\}_{i=1}^M$, where typically $M << N$, to approximate $f(x)$. The quantization function $Q(x)$ represents x by its **B**est **M**atching **U**nit (BMU) in W, which is the element of W with minimum r-th order distortion (distance to x, also called quantization error). Voluminous VQ research (see [12] for a thorough overview) has focused primarily on theoretical considerations and implications of VQ design for either *a fixed or asymptotically increasing number of prototypes*. That is, given M, much is known about optimal W, the behavior of the resulting quantization error, the geometries of the resulting Voronoi partition, etc. By contrast, this body of research has very little to say about specifying M in practical settings: VQ theorists often assume M "large enough" to invoke limiting behaviors in proofs; VQ practitioners may select M to meet constraints such as ensuring a maximum quantization error, or size of the encoded signal. While situational constraints may be used to set M in some domains, we are interested in VQ as a tool to build *prototype-based models in machine learning* [2], which include, e.g., the Self-Organizing Map (SOM, [14]), Neural Gas (NG, [20]), Learning Vector

Authors acknowledge partial support from NSF AAG 2107942 and seed funding from NSF's Institute for Foundations in Machine Learning.

T. Villmann et al. (Eds.): WSOM+2024, LNNS 1087, pp. 107–117, 2024.
https://doi.org/10.1007/978-3-031-67159-3_13

Quantization (LVQ, [14]), and even K-means [19]. These algorithms are regularly applied to domains where no hard constraints guide the selection of M.

Various rules-of-thumb and algorithmic enhancements have appeared over the years to address this issue. In what follows, we present a framework for choosing M based on concepts from density estimation theory. In Sect. 3 we review methods for selecting M in practice; Sect. 4 presents a review of the statistical theory of kernel density estimation, curated for our current needs; Sect. 5 outlines our density-based framework, which is exercised on synthetic data in Sect. 6.

2 VQ Theory

Optimal VQ design seeks the best codebook W to represent $f(x)$, as measured through minimum expected distortion, also called Quantization Error: [10]:

$$E[D_r] = \int ||x - w_{b(x)}||^r \, f(x)\, dx \qquad (1) \qquad\qquad W = \underset{\{w_i\}_{i=1}^M}{\arg\min}\, E[D] \qquad (2)$$

While (1) has motivated the study of many theoretical aspects of VQs over the last half-century, we briefly present two theoretical results first presented by Zador [26] germane to this work. Both involve a crucial quantity $\alpha = \frac{d}{d+r}$, which we call the *magnification exponent*. **Z1:** Evaluated at its theoretical optimal W (2), distortion (1) tends to 0 at rate $\propto ||f||_\alpha M^{-\frac{r}{d}}$ as $M \to \infty$. Here, $||f||_p$ denotes the L^p function norm. **Z2:** The prototypes $W = \{w_i\}_{i=1}^M$ (2) are joint minimizers of (1) if and only if the point density of prototypes, which we denote $g(w)$, obeys the following power law with respect to the data density f:

$$g(w) = cf(w)^\alpha, \quad c = 1/\int f(w)^\alpha \, dw \qquad (3)$$

The consequence of **Z1** is that an optimal VQ with larger M *always* results in lower quantization error, which may suggest over-specifying M. Since prototype-based ML models can lose pertinence or efficacy as M increases and the VQ mapping tends toward the identity function, we have reason to prefer a more parsimonious approach. One implication of **Z2** is that prototypes resulting from an optimal VQ can be thought of as a sample of size M from $g = cf^\alpha$. As defined, $\alpha < 1$ *always* with $\lim_{d\to\infty} \alpha = 1$. Consequently, the density surface defined by g magnifies parts of input space which have low measure according to f, and usually ensures g a smoother density than f, which will be important in Sect. 5 In practice, we are not given $f(x)$ but rather X, a sample of size N assumed to be drawn from it, so the expectation of (1) is replaced by its empirical mean. Unfortunately, this guarantees that minimization of (1) is NP hard. Nevertheless, a collection of practical algorithms [8,16,17,19] have been developed by both engineers and statisticians to fit VQ models to observed data X. These classical algorithms above are all based on *competitive* learning, indicating that the prototypes "compete" to represent (quantize) each x_s. Neural VQs, such as the SOM [14] and Neural Gas [20], add a *cooperative* element to their iterative update rules where, at each learning step, prototype updates are influenced both by their own receptive fields (like classical VQ) and by the RFs of neighboring prototypes.

However, cooperation is known [6,14] to impart a new magnification exponent α^* on the resulting prototype density g. While several works have derived theoretical expressions for α^* in restricted settings, no form exists for a neural VQ with arbitrary but fixed M in general dimension d, although it is "certainly less than unity" [14, Chapter 3]. While we will not attempt to add any further insight to α^* in this work, we have highlighted the different magnification behaviors of neural VQs to stress that the results of Sect. 5 will differ for them.

3 Existing Methods to Select M

In practice, VQ sizing is often done via trial-and-error, where M is specified along a grid, VQ codebooks of size M are built, some measure of fit (e.g., quantization error) computed, and the process repeated until a "good" codebook size is determined (or time/computational constraints are met); we stress the absence of any universally acceptable definition of "good." Various rules of thumb abound, mostly in practical handbooks (e.g., [15]) and documentation for VQ algorithms, with a wide range of suggested starting values for M. Letting $M = \mathcal{O}(\sqrt(N))$ is a popular starting point for grid searching, which we note has analogs in Silverman's rule of thumb for selecting the resolution in KDEs [23]. Less ad-hoc methods to select M exist, but most require that multiple VQs be fit [7], shift the burden of specifying M to that of specifying alternate (possibly more obtuse) parameters [9] or require additional information such as labeled data [24]. We prefer a more self-supervised and formal approach. To do so, in the next section we invoke theory from kernel density estimation where researchers have battled the dilemma of optimal data resolution for over half a century.

4 Variable Kernel Density Estimation

The advantage of VQ, and the crux of our methodology, depends on its density matching properties governed by the power law (3). As such, we pause now to introduce relevant parts of density estimation theory of use in this work. Specifically, we focus on variable KDEs, as they are better suited to higher-dimensional settings [25], harkening the space filling advantage of VQs [18].

Kernel density estimation utilizes smoothing kernel functions K centered at observed data $X = \{x_s\}_{s=1}^N$ to approximate the density $f(x)$ from which X is assumed to be arise. Variable kernel density estimators (vKDEs) prescribe different orientations and levels smoothing amounts across $\mathbb{R}^d$, whereas *fixed* KDEs use the same smoothing parameters everywhere. Fixed KDEs are more commonly employed, mostly because their construction is simpler but vKDEs are capable of better performance in high-dimensional settings. We note that most of the theory below is summarized from [25], but we borrow some notation from [4] for mathematical brevity. A multivariate vKDE has the form

$$f_y \approx \tilde{f}_y = \frac{1}{N}\sum_{s=1}^{N} K_{H_y}(y - x_s), \quad K_{H_y} = |H_y|^{-1}K(H_y^{-1}y), \tag{4}$$

where $y \in \mathbb{R}^d$ is an arbitrary point and subscripts y denote function evaluation at y. The kernel parameter H_y is called the *bandwidth* matrix prescribing the scale and rotation of smoothing K asserts amongst the sample points to produce the estimate $\hat{f}$. In contrast with fixed estimators, H_y varies with y.

H_y is the most important tuneable parameter of (4); we re-parameterize it as $H_y = h_y A_y$, where $h_y > 0$ is a scalar controlling the kernel's size and $A_y : |A_y| = 1$ is a $d \times d$ rotation matrix controlling its elliptical shape. To achieve consistency, we require that $h_y \to 0$ as $N \to \infty$. The exact functional form of K is less crucial to the success of (4) [23], but K is generally constrained to be a probability density in its own right satisfying the following moment conditions:

$$(i) \int K(y)\, dy = 1; \quad (ii) \int yK(y)\, dy = 0; \quad (iii) \int yy^T K(y)\, dy = I_d \tag{5}$$

where I_d is the d-dimensional identity matrix. For computational simplicity, K is often assumed to be the pdf of standard multivariate Gaussian distribution.

4.1 L^2–Error for vKDEs

We now give an overview of squared error analysis of (4), which is crucial to the theory of KDEs and motivates the methodology presented in Sect. 5. Detailed derivations of these results are found in, e.g., [25], [23, Chapter 6] and [4]. At an arbitrary evaluation point y, pointwise squared error of (4) with bandwidth parameter $H_y = h_y A_y$ is given by $(\hat{f}_y - f_y)^2$. Because $\hat{f}_y$ is a random variable (via its dependence on our sample X), it is more appropriate to consider pointwise Mean Squared Error $MSE\left[\hat{f}_{y|X}\right] = E_X\left[(\hat{f}_{y|X} - f_y)^2\right]$ where we have used the notation $f_{y|X}$ to remind us that the $\hat{f}_y$ is a function of the random sample X. Pointwise *bandwidth selection* in kernel density estimation seeks H_y as the arg min of $MSE\left[\hat{f}_y\right]$, which is often intractable (even if true f is known). To circumvent this, *asymptotic* error analysis replaces f with its Taylor series expansion in a window of shape/size H_y about y. Ignoring order-3 terms and above in the series results in the following Asymptotic Variance (AV) and Squared Bias (ASB) decomposition (from [23], with notation from [4]):

$$AMSE[\hat{f}_y] = AV[\hat{f}_y] + ASB[\hat{f}_y] \tag{6}$$

$$AV[\hat{f}_y] = \frac{f_y R[K]}{h_y^d N} \tag{7}$$

$$ASB[\hat{f}_y] = h_y^{2p_y}\beta_y \tag{8}$$

$$\beta_y = \left[\frac{1}{p_y!}\mu_{p_y}^T(K)\, A_y^{\otimes p_y}\, \mathcal{D}^{\otimes p_y} f_y\right]^2 \tag{9}$$

In the above, $R[K] = \int K(z)^2\, dz$ denotes the statistical *roughness* of K and $p_y \in \{2, 4\}$ is the *order* of the estimate, based on the curvature of f_y ($p = 2$ when the Hessian $\nabla^2 f_y$ is positive or negative definite, $p = 4$ when it is indefinite). μ_{p_y} is the vectorized p_y-th moment of the kernel, defined as $\mu_p(K) = \int z^{\otimes p} K(z)\, dz$ where $z^{\otimes p}$ is the p-th fold Kronecker product of the vector $z \in \mathbb{R}^d$ with itself (a d^p length vector, after vectorization). $A_y^{\otimes p_y}$ is the p_y-th Kronecker product of A with itself (a $d^{p_y} \times d^{p_y}$ matrix), and $\mathcal{D}^{\otimes p_y} = \frac{\partial f}{\partial y^{\otimes p_y}}$ is the vectorized p_y-th fold Kronecker power of the differential operator applied to f_y whose d^{p_y} elements contain all mixed partials of f up to order p_y. More explanation of vectorized higher-order derivatives can be found in [5].

Pointwise bandwidth selection minimizes $AMSE[\hat{f}_y]$ over h_y and A_y, which is separable in these parameters as the latter enters only via β_y. [25] describes this minimization over A_y in great detail as its procedure various based on p_y; for brevity we reproduce the optimal bandwidth results below. Letting A_y^* be the minimizer of β_y and β_y^* its minimum value, the optimal bandwidth scale h_y and resulting AMSE is given by

$$h_y^* = \left(\frac{d\, f_y\, R[K]}{2p_y\, \beta_y^* N} \right)^{1/(d+2p_y)} \tag{10}$$

$$AMSE^* \left[\hat{f}_y \right] = \epsilon[\hat{f}_y]\, N^{-2p_y/(d+2p_y)} \tag{11}$$

$$\text{where } \epsilon[\hat{f}_y] = \left(1 + \frac{d}{2p_y} \right) \left(f_y R[K] \right)^{2p_y/(d+2p_y)} \left(\frac{2p_y}{d} \beta_y^* \right)^{d/(d+2p_y)} .$$

(11) tells us that vKDE error decreases with sample size N at a rate $\mathcal{O}(N^{-2p_y/(d+2p_y)})$, which is faster than fixed KDEs most everywhere [25]. We harness this improvement in the methodology for sizing VQ codebooks described below.

5 Methodology

5.1 Motivation

Our goal is to deduce a relationship between the number of VQ prototypes M, the complexity of our data X, and the observed sample size N. This is motivated by the assumption that encoding a large amount of complicated data demands more prototypes – but how many more? The answer depends on how one characterizes the loose notion of data complexity, which we formalize here.

Zador's results summarized in Sect. 2 reveal that the M prototypes of an optimal VQ follow the density $g = cf^\alpha$. Thus, fitting a VQ in practice with sample X of size N results in a prototype sample W (from density g) of size M. Density estimation theorists have spent decades formally characterizing how (i.e., the process of kernel bandwidth selection), and to what degree of accuracy (i.e., the resulting minimal AMSE), one can infer the underlying density from which a sample arises. Deeper scrutiny of their main conclusions, as encapsulated in the terms of the optimal AMSE (11), reveal several analogues to data considerations germane to VQ practitioners when specifying M, such as sample size, dimension, and data complexity (terms involving f, $\mathcal{D}f$, etc.). **Our observation is that these M prototypes, if optimally derived, should represent *their* density g *as well as* the data X represent *their* density f**. That is, we propose selecting M to best represent g, as measured via AMSE, modulo the constraints on this process inherent in the observed sample X from which W arises.

5.2 An Equivalent Sample Size Analysis for *M*

Statisticians compare estimators in terms of their relative *efficiency*, as measured by their error. At evaluation point y,, we let $\tilde{f}_y$ and $\tilde{g}_y$ denote the theoretically optimal variable density estimates of f_y and g_y based on samples X and W, respectively. That is, $\tilde{f}_y$ and $\tilde{g}$ have functional form given by (4) with sample sizes N and M, respectively. The *efficiency* of $\tilde{g}$ to $\tilde{f}$ can be characterized by quantity η, which reports the ratio of their minimal errors:

$$\eta = \frac{AMSE^*[\tilde{g}]}{AMSE^*[\tilde{f}]} = \frac{\epsilon[\tilde{g}_y]\, M^{-2q_y/(d+2q_y)}}{\epsilon[\tilde{f}_y]\, N^{-2p_y/(d+2p_y)}} \tag{12}$$

where, for clarity, we have let q_y denote the order of g_y (the analog of p_y for f_y). For given values of its RHS (including M and N), η conveys how much better ($\eta \leq 1$) or worse ($\eta > 1$) $\tilde{g}_y$ is at estimating g_y than $\tilde{f}_y$ is at estimating f_y; we now have a functional form relating data complexity (the admittedly complex ratio $\epsilon[\tilde{g}_y]/\epsilon[\tilde{f}_y]$), the observed sample size N, and the number of prototypes M. Prescribing an acceptable efficiency η allows us to solve for the number of prototypes required to achieve such efficiency at y, which is known as *equivalent sample size analysis* in density estimation theory:

$$M_y^* = \Big(1/\eta \times \epsilon[\tilde{g}_y]/\epsilon[\tilde{f}_y] \times N^{2p_y/(d+2p_y)}\Big)^{(d+2q_y)/(2q_y)} . \tag{13}$$

Taking the expectation of M_y^* over our sample gives an estimate of the aggregate number of prototypes required to achieve efficiency η overall:

$$M^* = \frac{1}{N} \sum_{s=1}^{N} M_{x_s}^* . \tag{14}$$

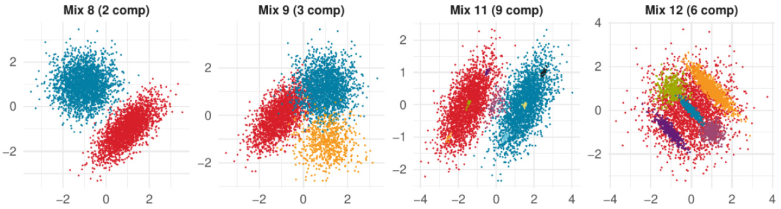
η is the only tuneable parameter in our methodology and has a direct, interpretable meaning; e.g., specifying $\eta = 2$ results in a number of prototypes which represent $g = cf^{\alpha}$ *twice* as badly as f could be represented with its N observations. Thus, a practitioner only needs to choose their relative error tolerance prior to constructing a VQ codebook.

5.3 Practical Considerations

Evaluating M^* in practice is not trivial for various reasons and we pause now to address computational roadblocks. Assuming f is known (for now), Zador's conclusions relate the prototype density $g = cf^\alpha$ but we still must compute its normalizing constant c. Empirical estimation of $c = E_X[f(X)^{\alpha-1}]$ is an option, but this will suffer from the "curse of dimensionality" unless increasingly large sample sizes are available. Instead, for now, we propose restricting f to the class of Gaussian Mixture Models ($f = \sum_k w_k \phi(\mu_k, \Sigma_k)$) and approximating true g by the straightforward method of [1]. When true f is not known (most practical settings), we propose using a preliminary *pilot* estimate $\hat{f}_0$ directly from X using Gaussian kernels, which is common among modern kernel smoothing methods [4]. Finally, because the components of AMSE involve terms whose size grows exponentially with data dimension d, computation of (11) demands the careful algorithms of [5] for feasibility in arbitrary dimension.

6 Experiments and Discussion

We selected four Gaussian mixtures f from density estimation literature [3] to showcase AMSE-modulated M^*. These mixtures, abbreviated Mix. $\{8, 9, 11, 12\}$, have varying number of components and component-wise covariance structures resulting in different complexities. The mixtures in the literature are all $2d$; because we would like to showcase M^* across a range of d we have expanded their dimensionality by replicating each component-wise covariance across the range $d \in \{2, 4, 6, 8, 10, 20, 40, 60, 80, 100\}$. Visuals of these mixtures in $2d$ are given below, where colors correspond to the various mixture components.



Each mixture and dimension listed above corresponds to a known data density f from which we estimate the prototype density g according to [1]. From each f we sample $X \sim f$ of size $N = 10,000$ and compute the number of prototypes M^* according to (14). To assess the suitability of M^* we view it in relation to [1] an information-theoretic characterization of data complexity, and [2] the Mean Quantization Error resulting from K-means across a large range of M.

For measure [1], we compute the overall mean $\bar{\mu}$) and covariance $\bar{\Sigma}$ for each $f(m, d)$. Because the multivariate Gaussian density has maximum entropy for continuous unbounded data with a given first and second moment, the KL-Divergence from $N(\bar{\mu}, \bar{\Sigma})$ to $f(m, d)$ represents how far, in an information-theoretic sense, $f(m, d)$ is from maximum entropy; i.e., it is *one* measure of data complexity that is unitless (comparable across varying d). We posit that higher-entropy mixtures should require more prototypes, which is generally confirmed in Fig. 1. For each mixture KL increases with d, which is not unexpected as it is known that higher-dimensional data tend to form "hubs" (clumps) in space more strongly [21]. While M^* generally decreases with KL it is not a completely monotonic trend which is likely due to several factors: (a) we have empirically estimated both KL and M^* from our sample of size $N = 10,000$ so both are subject to typical statistical estimation errors; (b) likely more influential is the suitability of KL as the only oracle for data complexity.

In practice, it is common to build several VQs for the same data by varying M and observing how the quantization error (distortion) behaves in response. So-called knee-finding algorithms (e.g., [22]) are often used to help guide the eye to points of regime switching along the QE vs. M curve. These curves, resulting from fitting K-means for M in the range $[10, 9000]$ are shown in Fig. 2 for our two most visually complicated mixtures (11 and 12). In each case, the M^* selected via (14) is shown on the curve (salmon points), along with the curve's knee (dark blue points) identified by [22].

Comparing the two, we see in Fig. 1 that M^* appears a conservative recommendation for the number of prototypes, at least when we demand $\eta = 1$. We also note the trend that as d increases $M^* \rightarrow M_{knee}$, suggesting M^* may signal regime changes along the QE curve *without* the burden of constructing many different quantizers, at least in higher-d cases.

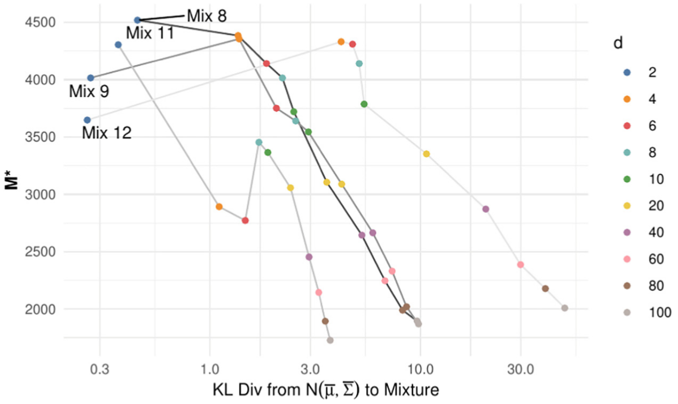


Fig. 1. M^* from (14) with $\eta = 1$ vs. KL-Divergence from a $N(\bar{\mu}, \bar{\Sigma})$ density to each mixture of noted dimension. A larger KL divergence indicates the mixture is less complicated, from an Information Theory perspective, relative to its overall scale.

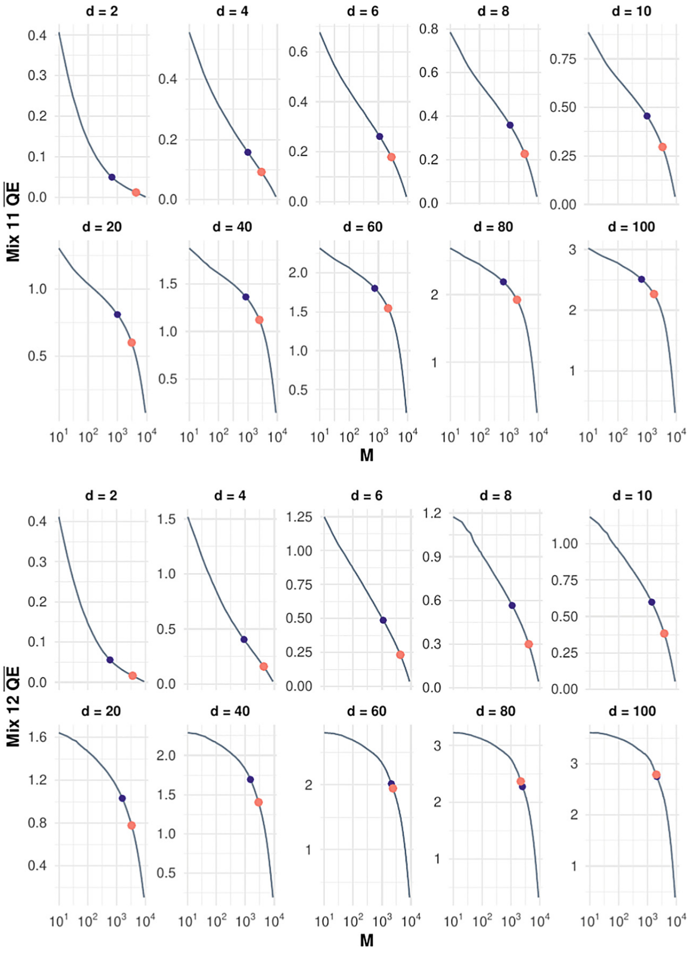


Fig. 2. Mean quantization error for Mixtures 11 and 12 (dark lines) produced by K-means along a range of M. The salmon points indicate the M^* selected via our method, and the dark blue points are the result of the knee-finding Kneedle algorithm [22].

7 Conclusions and Further Work

We have presented a framework for selecting the number of prototypes (codebook vectors) in a vector quantizer modulated by squared-error theory of kernel density estimation. The link between these two relies on a subset of VQ theory defining the prototype density of an optimal quantizer. Crucially, our equivalent sample size analysis, common in analyses of statistical estimators, replaces ad-hoc or rule-of-thumb methods for sizing codebooks with a formalism whose only parameter η is a meaningful measure of relative estimator efficiency. Specification of M^* is completely *a-priori*, alleviating the need to fit multiple vector quantizer(s). Of further benefit, our methodology can be inverted to reveal the (relative) impact of a given codebook size on the ability for resulting prototypes to to represent *their* density (and, in turn, the data density).

For this introductory work we have restricted experimentation to synthetic cases where the data density f is known; in the future we will remove this restriction by prescribing a pilot estimation scheme such as in [4] to select M in more general settings. Inspired by [13], we also believe uniformly demanding $\eta = 1$ (as done in our initial experiments) may err on a conservative estimate M^*; further experiments with a range of η efficiencies are planned.

References

1. Ajgl, J., Šimandl, M., Duník, J.: Approximation of powers of gaussian mixtures. In: 18th International Conference on Information Fusion, pp. 878–885. IEEE (2015)
2. Biehl, M., Hammer, B., Villmann, T.: Prototype-based models in machine learning. Wiley Interdisc. Rev. Cogn. Sci. **7**(2), 92–111 (2016)
3. Chacón, J.E.: Data-driven choice of the smoothing parametrization for kernel density estimators. Can. J. Stat. **37**(2), 249–265 (2009)
4. Chacón, J.E., Duong, T.: Multivariate kernel smoothing and its applications. Chapman and Hall/CRC (2018)
5. Chacón, J.E., Duong, T.: Higher order differential analysis with vectorized derivatives. arXiv preprint arXiv:2011.01833 (2020)
6. Cottrell, M., Fort, J.C., Pagès, G.: Theoretical aspects of the SOM algorithm. Neurocomputing **21**(1–3), 119–138 (1998)
7. De Bodt, E., Cottrell, M., Verleysen, M.: Statistical tools to assess the reliability of self-organizing maps. Neural Netw. **15**(8–9), 967–978 (2002)
8. Forgey, E.: Cluster analysis of multivariate data: efficiency vs. interpretability of classification. Biometrics **21**(3), 768–769 (1965)
9. Fritzke, B.: A growing neural gas network learns topologies. In: Advances in Neural Information Processing Systems, vol. 7 (1994)
10. Gersho, A.: Asymptotically optimal block quantization. IEEE Trans. Inf. Theory **25**(4), 373–380 (1979)
11. Gersho, A., Gray, R.M.: Vector Quantization and Signal Compression, vol. 159. Springer, New York (2012)
12. Gray, R.M., Neuhoff, D.L.: Quantization. IEEE Trans. Inf. Theory **44**(6), 2325–2383 (1998)
13. Gray, R.M., Olshen, R.A.: Vector quantization and density estimation. In: Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171), pp. 172–193. IEEE (1997)
14. Kohonen, T.: Self-Organizing Maps, vol. 30. Springer, Heidelberg (2012)
15. Kohonen, T., et al.: Matlab implementations and applications of the self-organizing map. Unigrafia Oy, Helsinki, Finland **177** (2014)
16. Linde, Y., Buzo, A., Gray, R.: An algorithm for vector quantizer design. IEEE Trans. Commun. **28**(1), 84–95 (1980)
17. Lloyd, S.: Least squares quantization in PCM. IEEE Trans. Inf. Theory **28**(2), 129–137 (1982)
18. Lookabaugh, T.D., Gray, R.M.: High-resolution quantization theory and the vector quantizer advantage. IEEE Trans. Inf. Theory **35**(5), 1020–1033 (1989)
19. MacQueen, J.: Classification and analysis of multivariate observations. In: 5th Berkeley Symp. Math. Statist. Probability, pp. 281–297 (1967)

20. Martinetz, T.M., Schulten, K.J.: A "neural gas" network learns topologies. In: Kohonen, T., Mäkisara, K., Simula, O., Kangas, J. (eds.) Proceedings of the International Conference on Artificial Neural Networks 1991 (Espoo, Finland), pp. 397–402. Amsterdam; New York: North-Holland (1991)
21. Radovanovic, M., Nanopoulos, A., Ivanovic, M.: Hubs in space: popular nearest neighbors in high-dimensional data. J. Mach. Learn. Res. **11**(sept), 2487–2531 (2010)
22. Satopaa, V., Albrecht, J., Irwin, D., Raghavan, B.: Finding a "kneedle" in a haystack: Detecting knee points in system behavior. In: 2011 31st International Conference on Distributed Computing Systems Workshops, pp. 166–171. IEEE (2011)
23. Scott, D.W.: Multivariate Density Estimation: Theory, Practice, and Visualization. Wiley, Hoboken (2015)
24. Silva, L.A., de Vasconcelos, B.P., Del-Moral-Hernandez, E.: A model to estimate the self-organizing maps grid dimension for prototype generation. Intell. Data Anal. **25**(2), 321–338 (2021)
25. Terrell, G.R., Scott, D.W.: Variable kernel density estimation. Ann. Stat. 1236–1265 (1992)
26. Zador, P.: Asymptotic quantization error of continuous signals and the quantization dimension. IEEE Trans. Inf. Theory **28**(2), 139–149 (1982)