Cardiac Anomaly Detection Using Graph-Based Representation Learning

Demián Fraiman

July 14, 2024

Abstract

This paper presents a computationally efficient approach for real-time cardiac anomaly detection using single-lead electrocardiogram (ECG) signals. We propose a graph-based representation learning framework that combines unsupervised topology learning with probabilistic classification. The method employs either Self-Organizing Maps (SOM) or K-means clustering to construct an embedded graph representation of ECG signals, followed by a Probabilistic Neural Network (PNN) classifier using geodesic distances within the graph. Our approach achieves high accuracy while maintaining low computational complexity ($\mathcal{O}(\log k)$ during prediction), making it suitable for deployment on mobile monitoring devices. Experimental results on both synthetic and real ECG datasets demonstrate the effectiveness of the proposed method, with accuracy up to 98% on synthetic data and 95% on real patient data when incorporating clinical history.

1 Introduction

Currently, cardiac abnormalities represent the leading cause of death worldwide, making their early detection an ongoing challenge. With advances in neural networks and deep learning, effective patient monitoring has been achieved. However, the high computational resource demands and lack of explainability of these models highlight the need to design more appropriate algorithms for this problem.

2 Problem Description

The objective of this work is to develop a real-time patient monitoring system using a single-lead mobile electrocardiographic device. Consequently, it is necessary to design a highly efficient detection algorithm that can continuously process the user's signal.

Furthermore, it is important to consider that diagnosing a sick patient as healthy does not have the same impact as the reverse error, so the algorithm must incorporate strategies that minimize associated risks. Additionally, given the nature of the problem, we can leverage the availability of the user's medical history and lifestyle to dynamically adjust our predictions.

To address this problem, we use a database containing information about various health factors and cardiac pathologies. Additionally, we work with ECG signal measurements, both synthetic and real, which have been annotated and validated by medical specialists.

The following presents the database used:

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
0	0	18393	2	168	62.0	110	80	1	1	0	0	1	0
1	1	20228	1	156	85.0	140	90	3	1	0	0	1	1
2	2	18857	1	165	64.0	130	70	3	1	0	0	0	1
3	3	17623	2	169	82.0	150	100	1	1	0	0	1	1
4	4	17474	1	156	56.0	100	60	1	1	0	0	0	0

Figure 1: Table of user characteristics used.

ECG measurements were taken at a sampling frequency of 140 Hz for one second, so each record is stored in a 140-dimensional vector. Due to the high complexity of the signal and the computational

limitations of the device, one of the main challenges is to find an efficient representation that allows compressing the information without losing consistency.
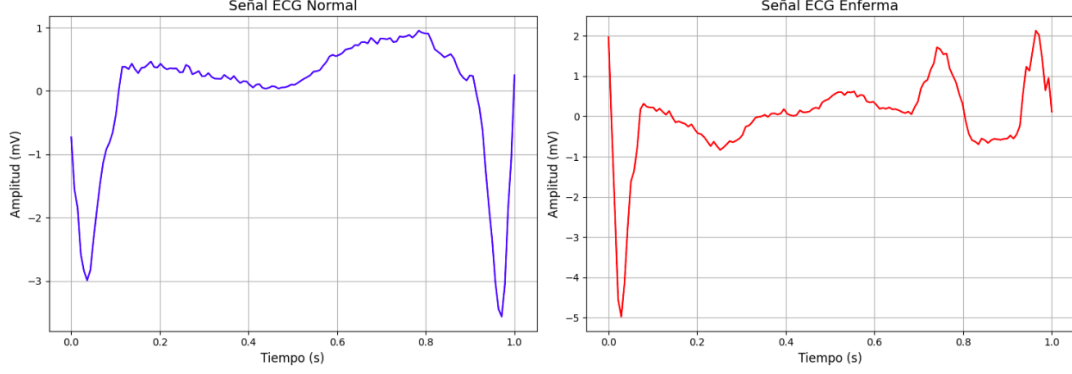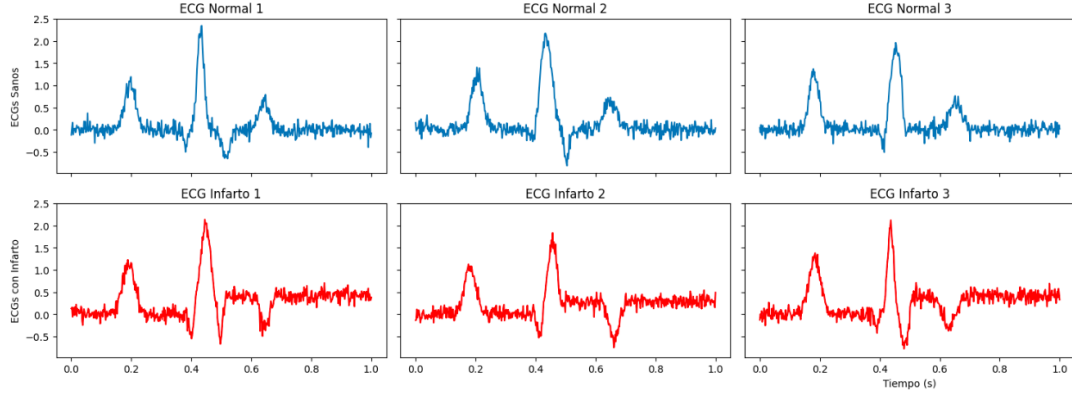

Figure 2: Real samples of one-second signals

Figure 3: Synthetic samples of one-second signals

3 Methodology

3.1 Proposed Model

3.1.1 Estimation of Affinity for Cardiac Conditions

We use a logistic regression model to calculate the probability that a patient develops a cardiac pathology based on basic lifestyle characteristics. In this way, we only store the estimated probability of being healthy or sick, which will serve as a *prior* in the following models.

3.1.2 Topology Learning

It is important to note that Euclidean distance does not adequately represent the difference between two temporal signals, as it does not consider possible time shifts. For this reason, we employ the *Dynamic Time Warping* (DTW) metric.

In this unsupervised learning stage, we seek to find a latent space that faithfully represents the data structure through a graph embedded in the input space. Subsequently, we will use this graph to approximate the output of new data. We propose two distinct approaches for its construction.
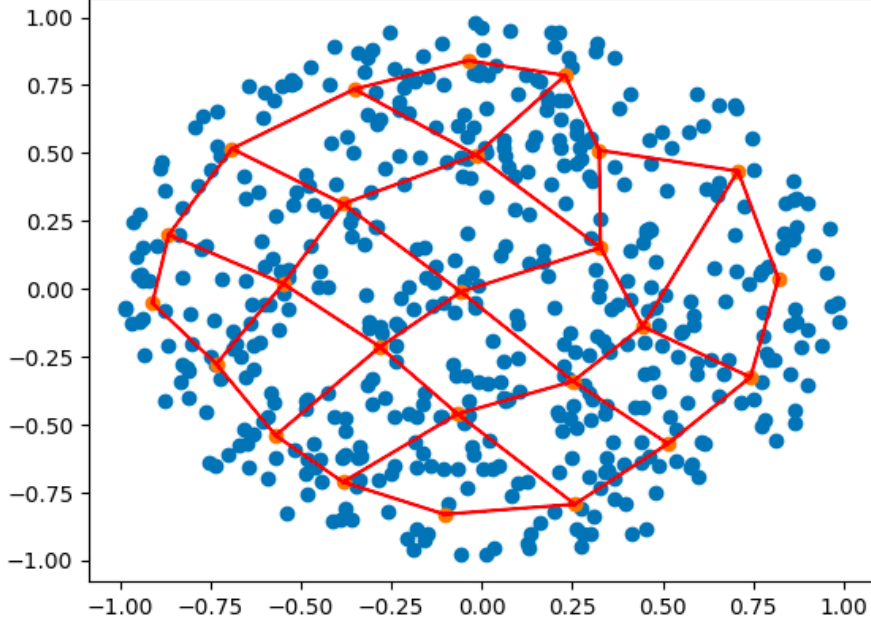
Figure 4: Embedded graph learned using SOM on a sample of uniform distribution on the unit circle.

3.1.3 SOM

The Kohonen network architecture naturally provides the graph structure we seek. By updating each neuron according to its proximity to each data point, these neurons eventually couple in the input space to well represent the sample. In our case, we use a rectangular topology, i.e., a grid-type structure, with two dimensions to enable good visualizations.

3.1.4 K-means

Another alternative consists of applying the K-means algorithm to find k centroids in the data, which will serve as graph nodes. Then, depending on the hyperparameters d and ϵ, nodes can be connected in two ways:

- Connect each node to its d nearest neighbors.
- Connect each node with all others within distance ϵ.

3.1.5 Probabilistic Classifier

Once this graph representation is learned, to predict the output of a new input, we approximate it by its nearest centroid. For classification, we use a *Probabilistic Neural Network* (PNN), where distance is defined by the geodesic distance within the graph. We consider this metric more robust than Euclidean distance due to the *curse of dimensionality*. Thus, we obtain an estimate of the probability of belonging to each class.

From a computational perspective, it is only necessary to store a matrix of class membership probabilities for each centroid. Therefore, the computational cost of making a prediction is reduced to finding the nearest centroid, which, leveraging the *Ball-Tree* data structure, has a cost of $\mathcal{O}(\log k)$.

An advantage of using a generative classifier is that if the loss function does not treat all errors equally, as in our case, we can make the optimal Bayes decision, guaranteeing loss minimization.

From another perspective, this model can be understood as a subdivision of the input space using a Voronoi tessellation based on centroids, where each cell is assigned the probability that its centroid belongs to each class.
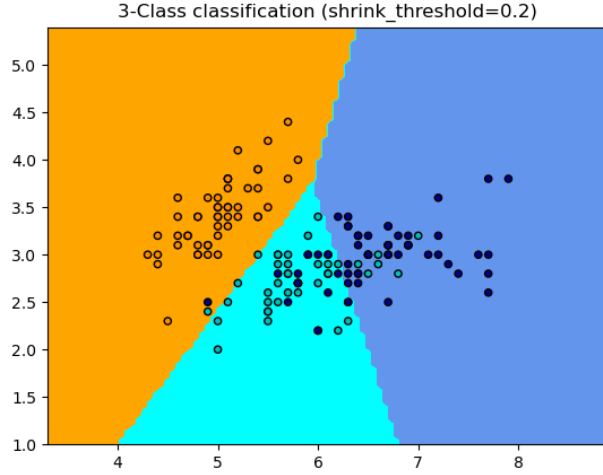
Figure 5: Decision boundary with three illustrative centroids

3.2 Hyperparameter Tuning

Due to the high flexibility of the proposed model, it is important to adjust the hyperparameters for our specific problem. Using cross-validation, we evaluated the following combinations:

For the **K-means** version:

- $\sigma \in \{1, 5, 0.2\}$

- $n\_neighbors \in \{5, 15, 2\}$

For **SOM**:

- $\sigma \in \{0.1, 2, 3, 7\}$

- learning_rate $\in \{0.1, 0.5, 1\}$

- neighborhood_function $\in \{$gaussian, bubble, mexican_hat$\}$

As a result, the optimal values were:

- For **K-means**: $\sigma = 5$ and $n\_neighbors = 2$

- For **SOM**: $\sigma = 7$, learning_rate $= 0.5$ and neighborhood_function $=$ mexican_hat

4 Results

In all cases, we divided the data into train and test sets, using an 80-20 proportion, respectively.

Regarding logistic regression training, the only consideration we had was to normalize the data. Then, we obtained 70 percent effectiveness.

On the other hand, we included as a comparison the result of training a simpler model, which only uses the K-means model and then assigns the majority class of its nearest centroid to each data point to be predicted. We always use the same number of centroids as its complexified version.

4.1 Synthetic Sample

We see that both models show high effectiveness against the artificially generated sample. With only 8 and 16 centroids, the K-means and SOM models, respectively, obtained the following results.
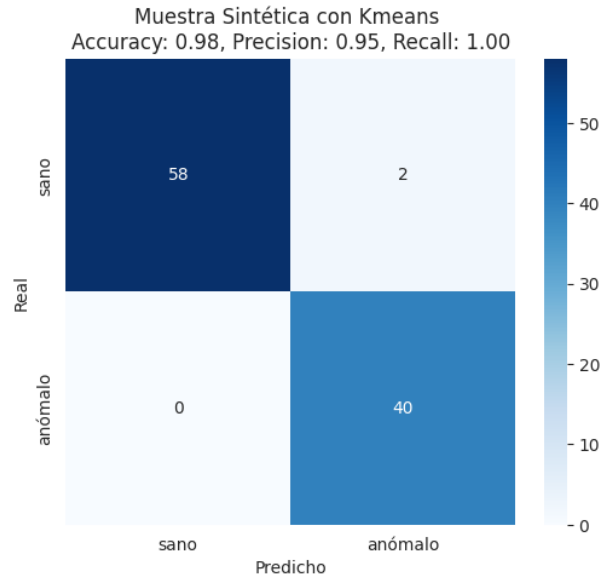
Figure 6: Predictions of the model using K-means on the synthetic sample

It can be seen how the anomalous wave presents ST elevation and an inverted T wave, characteristic of a myocardial infarction.
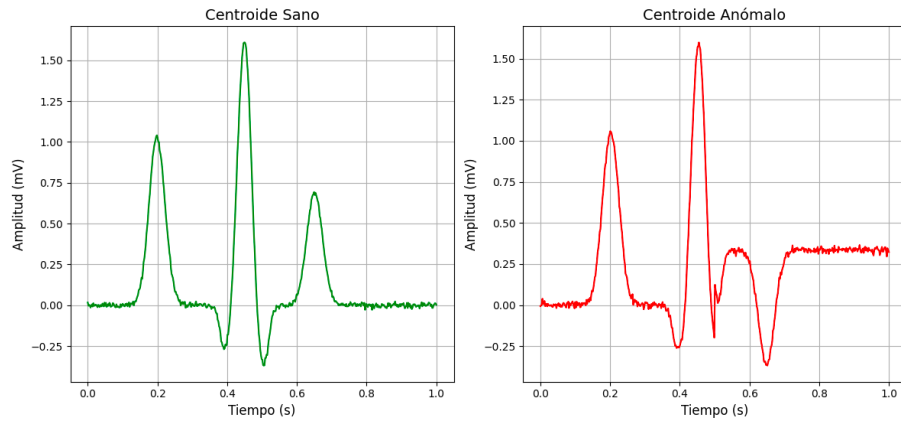

Figure 7: Centroids learned using K-means from the synthetic sample

Using the same representation learned by the model, we can visualize both the data and the centroids in their latent space. The components are the coordinates within the graph, learned with their geodesic distance.
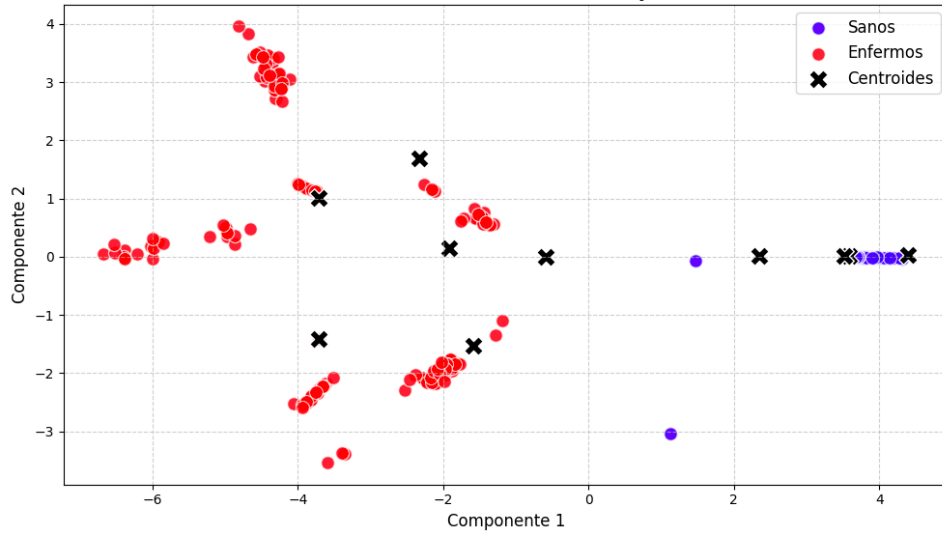
Figure 8: Visualization of the components of the latent space learned with K-means from the synthetic sample

4.2 Real Data

In general, due to the multimodal nature of the data, it was necessary to use more centroids in the real sample to obtain an effective representation. We trained with 10 centroids in the case of K-means and 25 in the case of SOM.
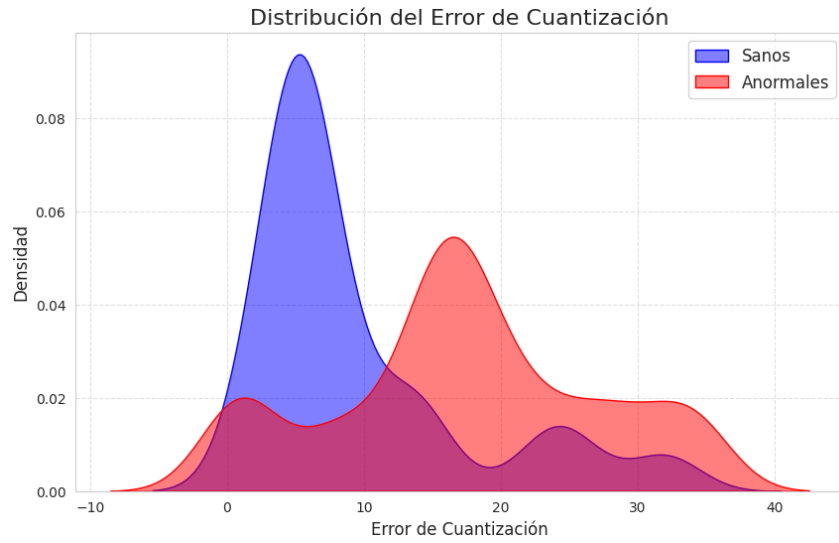

Figure 9: Distributions of approximation errors of data by their nearest centroid

It should be noted that we consistently found differences in the distributions of approximation errors between healthy and anomalous data. This provides evidence of the difference in the marginal distributions of each class.
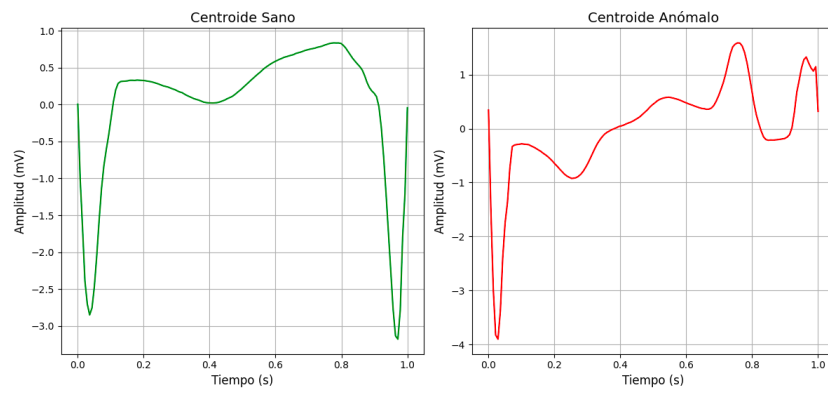
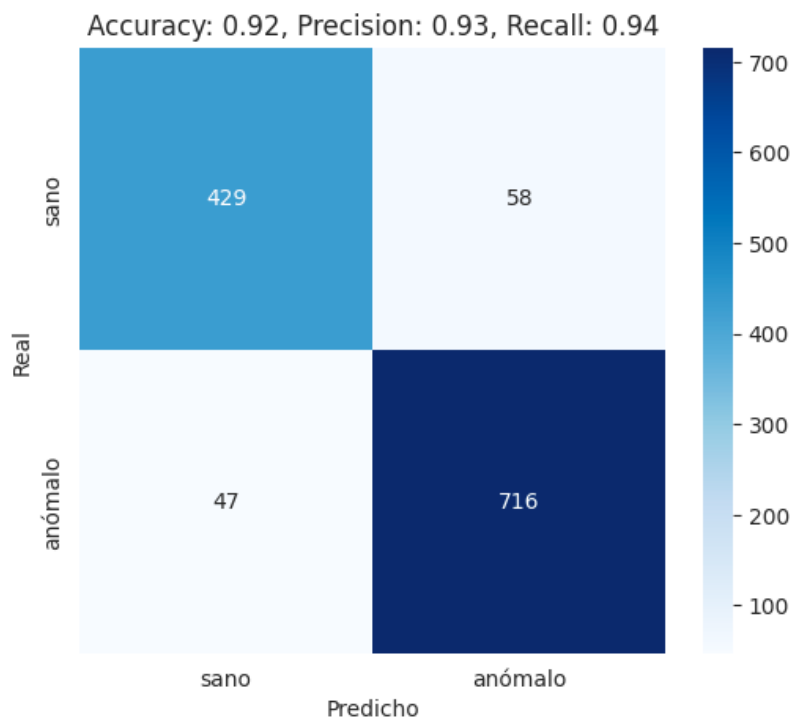Figure 10: Centroids learned using K-means from the real sample

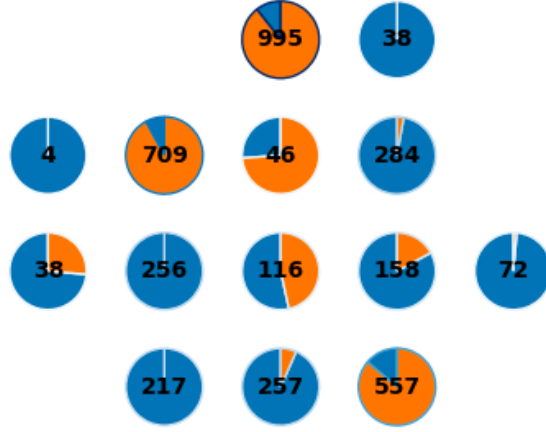Figure 11: Confusion matrix SOM on real sample

Figure 12: Centroids arranged according to the Kohonen network with their numbers of data they approximate, in blue proportion of healthy and in orange proportion of anomalous.

A slight improvement can be observed when incorporating information about people's lifestyle into the model's effectiveness.

Accuracy	Synthetic sample	Real sample	Real sample + history
SOM	0.97	0.92	0.93
K-means	0.98	0.94	0.95
Simple model	0.84	0.92	0.93

Table 1: Accuracy of SOM, K-means models and their simplified version on different samples.

Precision	Synthetic sample	Real sample	Real sample + history
SOM	0.94	0.90	0.94
K-means	0.95	0.92	0.98
Simple model	0.70	0.93	0.94

Table 2: Precision of SOM, K-means models and their simplified version on different samples.

Recall	Synthetic sample	Real sample	Real sample + history
SOM	1.0	0.96	0.94
K-means	1.0	0.98	0.92
Simple model	1.0	0.92	0.91

Table 3: Recall of SOM, K-means models and their simplified version on different samples.

5   Discussions and Conclusions

First, no type of signal preprocessing is being considered. Implementing this stage could help reduce noise and preserve the most relevant attributes of the wave.

In addition to the previous training based on electrocardiograms from different individuals, it would be possible to readjust the model using specific data from each user, which could improve its precision in particular cases.

Finally, another possible improvement would consist of robustifying the model by training it with a database that includes patients in motion, thus allowing better generalization to dynamic variations in signals.

In conclusion, both models demonstrated highly satisfactory performance in terms of precision and classification capacity. We can observe that the simpler model achieves quite comparable performance on the real sample; however, it shows a wide deterioration on the synthetic sample. We believe this is

because, by increasing the number of centroids between one sample and another, all the data sample is sufficiently well represented by its nearest centroids. While, with fewer centroids, it is necessary to leverage the underlying structure generated by the centroid graph.

The generative nature of both models allows the incorporation of different priors and loss functions, making them extremely flexible and adaptable to different clinical scenarios. Additionally, they present high computational efficiency in the prediction phase, with a cost of $O(\log(k))$, without needing to use more than $k = 25$ centroids.

Another key aspect is their high interpretability, which facilitates understanding of the learned patterns and their application in a medical environment. This is especially relevant for cardiac anomaly detection, where model transparency is fundamental for trust and validation by specialists.

References

[1] S. M. M. Martis et al., *ECG beat classification using PCA, LDA, ICA and Discrete Wavelet Transform*, Biomedical Signal Processing and Control, 2013.

[2] T. Kohonen, *Self-Organizing Maps*, Springer Series in Information Sciences, 2001.

[3] H. Sakoe, S. Chiba, *Dynamic programming algorithm optimization for spoken word recognition*, IEEE Transactions on Acoustics, Speech, and Signal Processing, 1978.

[4] D. F. Specht, *Probabilistic neural networks*, Neural Networks, 1990.

[5] S. M. Omohundro, *Five balltree construction algorithms*, International Computer Science Institute, 1989.