Paper Review
Accelerated dinuclear palladium catalyst identification through unsupervised machine learning

Harsh Poonia

December 2022

1 Mathematics behind unsupervised ML methods used

1.1 Principal Component Analysis (PCA)

Say we have a set of data points in d dimensions, vectors $x^{(1)}, x^{(2)} \dots x^{(m)} \in \mathbb{R}^d$, (the vectors can be thought of as modelling d different attributes of some object). Not each of these attributes carries equal amount of information. Perhaps some of the attributes are strongly correlated, for example age of a vehicle and miles driven, engine size and max acceleration in case of an automobile. There are thus, possibly redundant dimensions in the data points. For the purposes of visualization or efficient storage and computation, perhaps we would like to reduce dimensionality of the vectors without losing too much information. The trick is to find **principal modes of variance**, which are directions $\hat{v}$ along which when the data is *projected* ($x_{\text{proj}} = (x \cdot \hat{v})\hat{v}$) shows maximum dispersion, or in other words retains most of the information in the data. If we take k such principal modes of variation, we can project data points along them, and reduce dimensions from d to k. If the data projected upon principal k components captures most of the variance in the data, it is a good reduction.

$$\begin{aligned}
\mathbf{v} &= \arg\max_{||\mathbf{v}||_2=1} \frac{\sum_i ||\langle \mathbf{x}_i, \mathbf{v} \rangle||_2^2}{n} \\
&= \arg\max_{||\mathbf{v}||_2=1} \mathbf{v}^{\mathrm{T}} C \mathbf{v} \quad \text{where C is the covariance matrix of } \mathbf{x} \\
\mathbf{v} &= \text{Principal Eigenvector (eigenvector corresponding to largest eigenvalue) of } C
\end{aligned}$$

Captured variance $= \lambda_1$ (largest eigenvalue of C)

1.2 k-means Clustering

We are given a training set $x^{(1)}, x^{(2)} \dots x^{(m)} \in \mathbb{R}^d$ and want to group the data into a few cohesive "clusters". There are no labels $y^{(i)}$ to accompany the data vectors, thus *unsupervised*. The k-means clustering algorithm is this -

1. Initialize cluster centroids $\mu_1, \mu_2, \dots \mu_k$ randomly in $\mathbb{R}^d$
2. Repeat until convergence :
   (a) For every i, set $c^{(i)} = \arg\min_j ||x^{(i)} - \mu_j||^2$ (assign each data point to the cluster whose centroid is the closest to it)
   (b) For every j, set
   $$\mu_j = \frac{\sum_{i=1}^n \{c^i == j\} x^j}{\sum_{i=1}^n \{c^i == j\}}$$
   (for each cluster, update the centroid to be the mean of the points assigned to it)

This has the effect of minimising the sum of squared error (sum of sqaures of distances from points to their cluster centroid) and eventually J converges to a local minima of the squared sum of distances.

2 Elementary Appproach and Problems

We start by parametrization of phosphine ligands with a set of descriptors, ranging from certain ligand- specific data, such as proton affinity or highest occupied/lowest unoccupied molecular orbital (HOMO/LUMO) energies, to calculated data that describe the ligand's interaction in model complexes. An elementary approach was this : lower the dimensionality of the ligand knowledge base dataset to 2 and try to gain visual insights from 2D maps, wherein ligands with similar general properties would reside in similar areas. We want to apply this method to predict Pd^0/Pd^{II} monomer vs. Pd^I dimer speciation. Gibbs free energy changes, ΔG° of the reactions fail to provide any explanation for why ligand PCy_3 does not form dimer inspite of similar exergonicity (similarly negative ΔG°). Even on the 2D map constructed earlier, there are no spatial differences between dimer inducing and non-inducing ligands, they reside in the same area. Thus we seek a different approach.

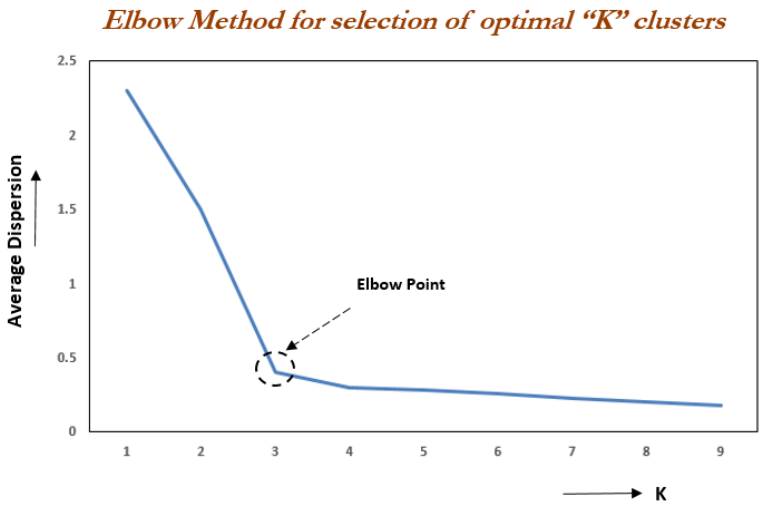
3 Unsupervised Learning to cluster ligands

Supervised learning, used for regression and classification tasks can predict reactivities, conditions and yields in the context of catalysts, but large, labeled training datasets are required as a prerequisite for those, which is not available in our problem. We can still recognise patterns in datasets, using clustering.

3.1 Initial Clustering

We start by using the LKB-P database, and apply the k-means clustering to partition it into different subgroups. The patameter, k was chosen to be 8 using the **elbow method** and **silhoutte scores**.

Elbow Method We chose some parameter k for the number of clusters, and obtain a clustering. Now calculate, for each cluster, the squared sum of distances of data points from the cluster centroid. Sum over all clusters, and we get **WCSS**(within cluster sum of squares). Plot WCSS with k, and we notice that it falls as k rises, but there is a sharp reduction in steepness of falling slope in the middle. That value of k is taken to be optimum.



Silhoutte Scores Silhoutte value is a measure of how similar an object is to its own cluster compared to others, value ranging from -1 to 1. For each datapoint $i \in C_I$, we define numbers $a(i)$ and $b(i)$ akin to similarity and dissimilarity as

$$a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i,j) \quad \text{and} \quad b(i) = \min_{J \neq I} \frac{1}{|C_J|} \sum_{j \in C_J} d(i,j)$$

Here $a(i)$ is the mean distance between the point and others in its cluster, whereas $b(i)$ is the smallest mean distance from the point to points in some other cluster. So the silhoutte value of the point $s(i)$ is calculated as $\frac{b(i)-a(i)}{\max a(i), b(i)}$. The mean of $s(i)$ for cluster k, $\tilde{s}(k)$ is a measure of closeness among points in the cluster. The silhoutte coefficient is $\max_k \tilde{s}(k)$.

An optimal value of $k = 8$ was chosen with the above analyses. This algorithm classified some of the ligands in the same area in the 2D map in different clusters, and also made the clusters overlap in the 2D map, which indicates that the algorithm found differences in ligands that PCA-based visual analysis could not. It successfully differentiated non dimer inducing ligands from dimer-inducing ones (clusters C1 and C4) by grouping them in different clusters. Further subcategorization of the ligands was unfruitful though, because the set of descriptors used in the database were as general as possible, making these ligands quite similar. To answer our specific question, we wish to further refine the clusters C1 and C4, to which 89 (roughly 25% of the original ligand space) ligands were assigned.

3.2 Problem Specific Descriptors

We use a set of new problem related descriptors through DFT calculations. We also keep some from the general descriptors. geometrical, conformational and electronic descriptors were introduced too. To gauge the suitability of the new descriptors for further analysis, we use the

Pearson Correlation Coefficient (PCC) is a measure for the linear relationship between two variables and ranges from -1 to 1 (-1 or 1 = linear relationship; 0 = no linear correlation). A small correlation between descriptors is desired, since heavily realated descriptors encode the same information.

PCA To obtain appropriate visualization of the new problem specific ligand space, we conduct PCA on the new dataset and check if the first few principal components capture most of the variance in the data.

3.3 Problem Specific Clustering

Following the generation of new Pd(I) dimer related data, we proceeded with the k-means clustering and chose $k = 6$ on the basis of the elbow method and silhouette score analyses, described earlier. This problem specific refinement result in further sub-differentiation of the clusters C1 and C4. Of the new clusters, C3' and C0' included known dimer inducing ligands. The ligands in C3' were relatively bulky, resembling already known dimer inducing motif. Experimentally verifying the results, the algorithm was correctly able to distinguish ligands based on their ability to favor an oxidation state (I) and also dihalide bridge geometry. At some places, machine learning predicts similarities that are neither intuitive nor obvious, with seemingly diverse set of ligands grouped into same cluster. Of the 25 ligands predicted by the algorithm to make Pd dimers, 4 were already known to do so, but we discovered several new ones by trying out experiments with the remaining ligands in the lab, which is quite feasible given the small set.