Master Thesis Lepke

Dario Lepke

8.3.2022

Contents

In this chapter we will first summarize the main ideas of clustering and then apply it to the precipitation data. If not indicated otherwise the information is taken from Elements of Statistical Learning.

1 Main Idea Clustering

We can describe an object by a set of measurements or its similarity to other objects. Using this similarity we can put a collection of objects into subgroups or clusters. The objects in the subgroups should then be more similar to one another than to objects of different subgroups. This means inside the clusters we aim for homogeneity and for observations of different clusters for heterogeneity. With the clustering analysis applied to the precipitation data we want to study if there are distinct groups (regions) apparent in the CAB. So that if we later apply the regression models we predict the precipitation for each group and not for the whole region.

To explore the grouping in the data we need a measure of (dis)similarity. This measure is central and depends on subject matter considerations. We construct the dissimilarities based on the measurements taken for each month. We interpret this as a multivariate analysis where, each month is one variable. So given the area in the CAB (resolution 5°x5°), we have 612 cells and 432 months, resulting in a 612×432 data matrix. we want to cluster cells into homogen groups.

2 Clustering Methods

2.1 K-means

In the following we briefly describe the K-means procedure. Beforehand we have to specify a number of clusters C we believe exist in our data. Then we randomly initialize C cluster centers in the feature space. Now two steps are repeated until convergence:

1. For each center we identify the points that are the closest to this center.
   These points “belong” now to a cluster C.
2. In each cluster we compute the mean of each variable and get a vector of means. This mean vector is now the new center of the cluster.

As a measure of dissimilarity we use the Euclidean distance:

$$d(x_i, x_{i\acute{t}}) = \sum_{i=1}^{p} (x_{ij} - x_{i\acute{t}j})^2 = ||x_i - x_{i\acute{t}}||^2$$

Meaning for the points i and $i\acute{t}$ we compute the squared difference for each variable and sum them up. As stated above we are searching for clusters that are themselves compact, meaning homogeneous. We do so by minimizing the mean scatter inside the clusters. We summarize this scatter as within-sum-of-squares

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i\acute{t})=k} ||x_i - x_{i\acute{t}}||^2 = \sum_{k=1}^{K} N_k \sum_{C(i)=k} ||x_i - \bar{x}_k||^2$$

where $\bar{x} = (\bar{x}_{1k}, ..., \bar{x}_{pk})$ stands for the mean vectors of the kth cluster and $N_k = \sum_{i=1}^{N} I(C(i) = k)$.

2.1.1 Kmeans characteristics

variance of each distribution of each attribute (variable) is spherical, variance is symmetrical? all variables have same variance, not the case in our example, therefore scaling or pca equal number of observations in each clusters, we don‘t know

Since we use the Euclidean distance the similarity measures will be sensitive to outliers and scale. K-means assumes that the variance of a variable´s distribution is spherical, meaning it wight not work well in situations that violate this assumptions (f.e non-spherical data). Further assumptions are same variance of the variables, and equally sized clusters. Now how “large” these violations have to be so that k-means does not work well anymore has no clear-cut answer.

2.2 K-medoids

We can adjust k-means procedure so that we can use other distances than the Euclidean distance. The only part of the k-means algorithm that uses Euclidean distance is when we compute the cluster centers. We can replace this step by formalizing an optimization with respect to the cluster members. For example so that each center has to be one of the observations assigned to the cluster. K-medoids is far more computationally intensive than K-means.

2.2.1 K-medoids characteristics

K-medoids is Less sensitive to outliers, because it minimizes sum of pairwise dissimilarities instead of sum of squared Euclidean distances. As stated above it is also more computationally intensive.

2.3 PCA (Multivariate Verfahren und ESL2)

Goal is reduction of correlated and eventually large number p variables to a few. We accomplish this by creating new variables that are linear combinations of the original ones. We call these new variables principle components. The new variables are not correlated any more and ordered according to the variance they explain. The first $k < p$ principal components then contain the majority of variance. (Multivariate

Verfahren) As they are ordered they also provide a sequence of best linear approximations of our data. Let $x_1, x_2, ..., x_N$, be our observations and we represent them by a rank-q linear model

$$f(\lambda) = \mu + V_q \lambda$$

with μ a location vector in $\mathbb{R}^p$ and V_q is a $p \times q$ matrix with columns being orthogonal unit vectors as columns. λ is a q vector of parameters. In other words we are trying to fit a hyperplane of rank q to the data. If we fit this model to minimize reconstruction error using least squares we solve

$$\min_{\mu, \{\lambda_i\}, V_q} \sum_{i=1}^{N} ||x_i - \mu - V_q \lambda_i||^2$$

If we partially optimize for μ and λ_i we obtain

$$\hat{\mu} = \bar{x},$$
$$\hat{\lambda_i} = V_q^T (x_i - \bar{x})$$

Therefore we need to search for the orthogonal matrix V_q

$$\min_{V_q} \sum_{i=1}^{N} ||(x_i - \bar{x}) - V_q V_q^t (x_i - \bar{x})||^2$$

We can assume here that $\bar{x}$ is 0, if this not the case we can simply center the observations $\tilde{x}_i = x_i - \bar{x}$. $H_q = V_q V_q^T$ projects each observation x_i from the original feature space onto the subspace that is spanned by the columns of V_q. $H_q x_i$ is then the orthogonal projection of x_i Hence H_q is also called the *projection matrix*. Stackoverflow post (https://math.stackexchange.com/questions/3982195/what-are-left-and-right-singular-vectors-in-svd) We find V_q then by constructing the *singular value decomposition* of our data matrix X. X contains the (centered) observations in rows, giving a $N \times p$ matrix. The SVD is then:

$$X = UDV^T$$

Where U is an orthogonal matrix containing the *left singular vectors* u_j as columns, and V contains the *right singular vectors* v_j. The columns of U span the columns space of X and the columns of V span the row space. D is diagonal matrix which contains *singular values*, $d_1 \leq d_2 \leq ... \leq d_p \leq 0$. *Singular values* are square roots of non-negative eigenvalues. The columns of UD are the principal components of X. So the SVD gives us the matrix V (the first q columns give the solution to the minimization problem above) as well as the principal components from UD.

2.4 Gap statistic

(see gap statistic in plain english stackoverflow) (see clusgap paper) As stated above we usually measure how compact our clusters are by assessing $W(C)$ or $log(W_c)$. Where are a low value indicates compact clusters. To compare the value we need a reference. We therefore want to estimate how large W_c were if there were no clusters present in our data. The larger the difference between the W_c from the data and the one from the reference the more likely we are to say that the found number of clusters is indeed correct. We construct reference data by sampling from a uniform distribution based on our data. Say we have p variables. We sample n times from each of the p uniformly distributed variables, where maximum and minimum are obtained from our data. We then cluster the reference data in the same we cluster our observed data and compute W_c. We repeat this process several times and compute the average of W_c, $E\{log(W_c)\}$. The gap statistic is then the difference

$$E\{log(W_c)\} - log(W_c)$$

. So in cases where our data is formed of clusters we would expect a high gap statistic. As Tibshirani et al. note and as it is also done in the used R function clusgap, doing a PCA on the data and compute the gap statistic on the PCA scores can improves the results of gap statistic.