A REGIONAL ANALYSIS OF THE SOCIO-ECONOMICAL PROPERTIES OF THE TURKEY CITIES

Muhammed Emre KESKİN*

Alınış Tarihi: 06 Ağustos 2018 **Kabul Tarihi: 15 Eylül 2018**

Abstract: Turkey is divided into 7 regions depending on the cities' geographic locations. Since the geographic properties of the cities belonging the same region are the same, socio-economical properties like populations, migration rates, annual incomes per person are expected to be similar. Some cities may not possess the same socio-economic structure with the rest of the cities that are from the same region but are assigned to the region anyway just because of geographical proximity. This study aims to find the cities which are in a sense exceptional in their regions. In order to eliminate the effect of the geographical proximity of the cities, not exact locations of the cities but the estimate locations obtained from multi-dimensional scaling are used. At the first hand, a k-means clustering algorithm which only depends on the geographical locations of the cities are used to form 7 clusters. Then, a decision tree analysis is used to form the clusters using both coordinates of the cities and socio-economical properties. Clusters obtained by k-means and decision tree analysis are then compared by themselves and with the real regions of Turkey and discussed.

Keywords: Multi-Dimensional Scaling, K-Means Clustering, Decision Tree Analysis

TÜRKİYE ŞEHİRLERİNİN SOSYO-EKONOMİK ÖZELLİKLERİNİN BÖLGESEL ANALİZİ

Öz: *Türkiye şehir yerlerine bağlı olarak 7 bölgeye ayrılmıştır. Aynı bölgedeki şehirlerin coğrafi özellikleri aynı olduğundan, nüfus, göç oranı, kişi başına düşen yıllık gelir gibi sosyo-ekonomik göstergelerinin de benzer olması beklenir. Bazı şehirler bulunduğu bölge içindeki diğer şehirlerden sosyo-ekonomik yapı bakımından farklı olmasına rağmen coğrafi yakınlık sebebiyle bulunduğu bölgeye atanmış olabilirler. Bu çalışma, bölgelerinde bir anlamda aykırı olan şehirleri tespit etmeyi amaçlamaktadır. Şehirlerin coğrafi yakınlığının etkisini ortadan kaldırmak için şehirlerin gerçek yerleri değil çok-boyutlu ölçeklendirme yönteminin verdiği şehir yerleri kullanılmaktadır. Başlangıçta sadece coğrafi yer bilgisine dayanan k-ortalama gruplandırma yöntemiyle şehirler 7 ayrı gruba bölünmüştür. Ardından, şehir yerleri ve sosyo-ekonomik göstergeleri beraber kullanan karar ağacı yöntemi grup oluşturmak için kullanılmıştır. K-ortalama ve karar ağacı yöntemlerinin verdiği gruplar birbirleriyle ve ardından gerçek Türkiye bölgeleriyle kıyaslanmış ve tartışılmıştır.*

Anahtar Kelimeler: *Çok-Boyutlu Ölçeklendirme, K-Ortalama Gruplaması, Karar Ağacı Analizi*

* Dr. Öğretim Üyesi, Atatürk Üniversitesi, Mühendislik Fakültesi, Endüstri Mühendisliği Bölümü

I. Introduction

Turkey is divided into 7 regions depending on the cities' geographic locations. Distribution of the regions mostly depends on the geographical natural barriers like the North Anatolian mountain ranges in the north and Toros mountain range in the south. Since the geographic properties of the cities belonging the same region are the same, socio-economical properties like populations, migration rates, annual incomes per person are expected to be similar. Although this is generally true, some cities may not be alike the other cities. Some cities may not possess the same socio-economic structure with the rest of the cities that are from the same region but are assigned to the region anyway just because of geographical proximity. Such cities are probably suit more to other regions since they are more similar to the cities of some other regions. This study aims to find the cities which are in a sense exceptional in their regions and to point out the differences of and properties of these studies which differentiates them from the rest of the region cities. In other words, the cities which are not alike the other cities that are in the same region are going to be found in this study. Hence, a similarity rating for the cities of the regions would be obtained as a result.

First of all, estimated locations of the cities are found by a technique named multi-dimensional scaling. In order to eliminate the effect of the geographical proximity of the cities, not exact locations of the cities but the estimate locations are used. Hence, results of the multidimensional scaling are used as input for some further research. We apply a k-means clustering algorithm on the approximated location data, so as to make a comparison between the formed clusters and the real existing seven regions of Turkey. To have a meaningful comparison, 7 clusters were formed. It should be noted that the cities which are close to boundaries of the regions in real does not have to be so in clusters since the locations used in clustering algorithms are approximated values. Still it is possible to extract some interpretations in terms of the wrongly placed cities. Hence we obtain confusion matrices as the result of the algorithm. Since the k-means clustering algorithm only depends on the geographical locations of the cities, conclusions extracted from it would not be so meaningful in terms of socio-economic properties of the cities. Hence, a decision tree analysis is also conducted in order to form the clusters using both coordinates of the cities and socio-economical properties. We took the regions of the cities as their classes. On the other hand, populations, cumulative migrations, annual incomes per person, areas, first and second coordinates of locations of the cities (obtained from multidimensional scaling) are used as inputs for the algorithm. Note that mostly due to geographical natural barriers, distributions of the regions are not balanced in terms of the north-south and east-west dimensions. For instance, Karadeniz and Akdeniz regions lie mostly in east-west dimension, while Ege or Doğu Anadolu has also considerable north-south dimension lengths. This is why first and second dimensions are

treated as different inputs for the algorithm. Still, it should be underlined that the 2D coordinates of the multidimensional scaling does not actually fit to the east-west and north-south real dimensions, since these locations are just approximations, and distances between cities are the only criteria that are preserved while constructing the locations. Therefore, one city may be at the north compared to another in real, but multidimensional scaling does not care about it, but it only puts them on a map so the distance between them is as close as possible to the real distance. Then, multidimensional scaling may point that the city which is at the north in real is in the south according to the approximated locations. However, because there are 81 cities and the multidimensional scaling put all of them in 2D dimensions, one may expect that the cities which are close in one dimension in the real, should be close in one of the approximated dimensions. Therefore, using the multidimensional scaling location results, and treating dimensions separately as inputs to the decision tree algorithm makes sense. We use 50 random cities for training and remaining 31 as validation set in order to obtain the best "theta", that is the tolerable node entropy, leading minimum validation "error". Error here is the number of the misplaced cities. Confusion matrices were also constructed to observe the mistakes made during classification.

To initiate the multidimensional scaling algorithm, we need the knowledge of between city distances, and we make use of the distance table existing at the web site of General Directorate of Highways of Turkey (http://www.kgm.gov.tr/). To have the populations, cumulative migrations, annual incomes per person, and the areas of the cities, we refer to the web site of Turkish Statistical Institute (http://www.tuik.gov.tr/). All the data reflect the year 2016, and unfortunately there is no more up to date data in all of the fields. For at least one of the populations, migrations and the annual income per person, the newest data were to the year 2016. Therefore, we stick to year 2016 for the analysis. We rearrange the data in all the tables so that the cities are sorted in the same order in all of the tables, and all of the names are written exactly the same. For instance, the name of the Sakarya was as Adapazarı in some of the tables and we made all of them as Adapazarı. Similar change was made for Mersin which was named as İçel in some tables. Finally, all the Turkish characters are changed with the English ones in order the codes run properly. After that step, the data were appropriate to be used in the algorithms.

Rest of the paper is organized as follows. We provide a brief literature about multi-dimensional scaling, k-means clustering and decision tree analysis section 2. Later, we give the details of multi-dimensional scaling algorithm employed for our study in section 3. Next, we give k-means application details in section 4 and we provide decision tree analysis in section 5. Finally, we conclude with discussions and future research directions in section 6.

II. Literature Review

We use multi-dimensional scaling for approximating the locations of the cities at the beginning. Hence, we provide a brief literature review of the subject first. Multi-dimensional scaling is mostly used for approximating the locations of some points given that the distances between the nodes are known (Bronstein et al. (2006)). Details of the multi-dimensional scaling, variants of the algorithm and implementations of it can be found in Borg and Gruenen (2003). It has also been elaborated by Alpaydın (2009) especially with a computer science and machine learning point of view. It has a broad range of application. To count some, it has been employed for construction of a self-organizing map of Turkey cities by Altınel et al. (2003) similar to what is done in this study. However, we do not confine with multi-dimensional scaling but enrich it with k-means clustering and decision tree analysis. In an interesting study by Kandoğan (2001), multi-dimensional scaling is used for approximating 3 dimensional locations of the stars. It is even employed in electromagnetic tracking in high dose rate brachytherapy in a recent work by Götz et al. (2017).

The next thing after multi-dimensional scaling is to employ a k-means algorithm for clustering the cities. Therefore, we give a brief literature review of k-means algorithm in the following. The k-means algorithm is a widely used clustering algorithm based on division method. Its procedure is simple and efficient, suiting for clustering analysis of big data sets. It uses distance or similarity measure to divide the sample into several clusters. Delias et al. 2015 suggest a model of clustering event logs model for supporting healthcare management decisions in flexible environments. Parekh and Saleena (2015) present a cloud based framework with clustering techniques to determine region wise diagnosis. Clustering analysis is also used for elderly patient subgroups to identify medication related readmission risks (Olson et al. 2016). A fuzzy clustering approach is used by Ben-Arieh and Gullipalli (2012) through data envelopment analysis with spars input and output data. Moreover, Tsumoto et al. (2015) propose a method for the construction of a clinical pathway based on attribute and sample clustering, called dual clustering. Within the same cluster, the similarity among samples is higher, and the dissimilarity among samples in different clusters is higher. Since k-clustering is a very well-known method with so many applications we direct the interested readers to pioneering work by Jain et al. (1999) about data clustering.

Besides, a decision-tree analysis is conducted as a supervised clustering method. Hence, we provide a short literature review also. A decision tree is a hierarchical model for supervised learning (Brodley and Utgoff (1995)) It is a popular classification algorithm that is commonly used in a broad range of areas (Yıldız and Alpaydın (2001)). It is even used as a main method for multi-labeled classification in which nodes can be member of more than one classes (Vens et al. (2008)). It is accepted as the most successful supervised machine learning methods together with support vector machines (Rokach and Maimon

(2008)). One may find many applications of decision trees but there is no decision tree application for analysis of socio-economic properties of cities of Turkey. Therefore, we confine with these studies and direct the interested readers to the seminal study by Breiman (2017).

Finally, we give a short review of the studies that concentrate on the socio-economic properties of the cities of Turkey. There are several studies elaborating the socio-economic situations of the cities of Turkey such as the study by Gürbüz and Karabulut (2008) which elaborates the relationship between the crime rates and the socio-economic determinants, and the study of Cömertler and Kar (2007) which works on the dependency of rural migration rates on socio-economic properties. Erkip (2005) provides another study that focuses on the number of malls depending on the social welfare of the cities. On the other hand, Korte and Ayvalioglu (1981) study on the relationship between the social welfare and the hospitality rate of the Turkey cities in their interesting work. Last, Uzun et al. (2010) put light on the illegal settlement rates in the cities of Turkey while keeping an eye on the socio economic situation of them. However, to our knowledge there is no study that seeks for the exceptional cities within regions in terms of socio-economic properties. Moreover, there is no study that employs tools of machine learning such as multi-dimensional scaling, k-means and decision tree analysis for evaluation of the socio-economic properties of the cities of Turkey.

III. Multidimensional Scaling

As previously mentioned in the introduction part, multidimensional scaling algorithm is a method of approximating the locations of the points in a lower dimensional space, for instance in two dimensional space as done in this study, by making use of the distances between the points. It should be noted that, in real the points may not be in a 2D dimensional scale. Therefore, multidimensional scaling is also a method for dimensionality reduction. However, the main aim is to preserve the between city distances as much as possible.

Before giving the formulations used in the algorithm, we provide the notation used in the algorithm in Table 1.

Table 1: *Parameters Used in Multi-dimensional Scaling*

Parameter	**Definition**
N	Number of the cities
X	Matrix of the locations of cities
b_{rs}	$\frac{1}{2}(d_{r\bullet}^2 + d_{\bullet s}^2 - d_{\bullet\bullet}^2 - d_{rs}^2)$
B	Matrix containing b_{rs} values

d_{rs}	Real distance between the cities r and s
C	The matrix whose columns are the Eigen vectors of B
D	A diagonal matrix containing the Eigen values of B in the diagonal

Mathematical details of the algorithm are as follows. Suppose the matrix $B = XX^T$ where X is the matrix of the locations of cities that is to be approximated. Each row of the matrix X corresponds to one city and contains the dimensional knowledge of the city which is wanted to be obtained. On the other hand B is an $N \times N$ matrix where N is the number of cities and whose instances are given according to following formula;

$$b_{rs} = \frac{1}{2}(d_{r\bullet}^2 + d_{\bullet s}^2 - d_{\bullet\bullet}^2 - d_{rs}^2)$$

where d_{rs} is the real distance between the cities r and s which is already known, $d_{r\bullet}^2 = \frac{1}{N}\sum_s d_{rs}^2$, $d_{\bullet s}^2 = \frac{1}{N}\sum_r d_{rs}^2$ and $d_{\bullet\bullet}^2 = \frac{1}{N^2}\sum_r\sum_s d_{rs}^2$. Observe that the matrix B can be constructed by making use of the between city real distances. Then, $X = CD^{1/2}$ can be used as an approximation for X where C is the matrix whose columns are the Eigen vectors of B, and D is a diagonal matrix containing the Eigen values of B in the diagonal. Then matrix C and D can also be constructed and an approximation for the X matrix is obtained thereafter. When C is constructed from two Eigen vectors of B corresponding to the highest Eigen values of B, and D is a 2×2 diagonal matrix having those two Eigen values in its diagonal, we obtain two dimensional approximated locations for the cities where between cities Euclidean distances are as close as possible to the real distances. The followings given in Table 2 and Figure 1 are the approximated locations for the cities and the plot of the cities in two dimensional scale for that approximated location data. Columns of Table 2 stands for the plate number of the cities, name of the cities, and approximated x and y coordinates of the cities, respectively.

Table 2: *Approximated 2D locations of the cities of the Turkey*

#	**City**	**x coordinate**	**y coordinate**
1	ADANA	-373.03	-75.597
2	ADIYAMAN	-302.3	-417.02
3	AFYON	-167.89	465.28
4	AGRI	244.06	-766.56
5	AMASYA	208.54	-17.446

6	ANKARA	27.582	274.09
7	ANTALYA	-487.05	457.85
8	ARTVIN	447.22	-636.79
9	AYDIN	-308.53	769.07
10	BALIKESIR	-50.632	770.87
11	BILECIK	51.706	572.94
12	BINGOL	-24.044	-622.15
13	BITLIS	-81.806	-817.95
14	BOLU	209.26	392.22
15	BURDUR	-291.99	542.68
16	BURSA	69.268	638.85
17	CANAKKALE	82.464	912.76
18	CANKIRI	150.78	218.97
19	CORUM	175.9	66.414
20	DENIZLI	-279.51	643.52
21	DIYARBAKIR	-196.01	-627.35
22	EDIRNE	315.8	872.11
23	ELAZIG	-99.532	-486.89
24	ERZINCAN	202.98	-392.11
25	ERZURUM	219.65	-589.79
26	ESKISEHIR	-68.288	500.03
27	GAZIANTEP	-361.5	-298.68
28	GIRESUN	348.17	-255.18
29	GUMUSHANE	321.79	-434.48
30	HAKKARI	-320.49	-1053.9
31	HATAY	-440.79	-211.01
32	ISPARTA	-311.32	497.19
33	MERSIN	-412.64	-23.461
34	ISTANBUL	264.29	648.52
35	IZMIR	-217.44	790.19
36	KARS	307.37	-767.91
37	KASTAMONU	268.85	207.14
38	KAYSERI	-88.653	-32.538
39	KIRKLARELI	319.8	848.8
40	KIRSEHIR	-55.589	102.23
41	KOCAELI	214.32	556.98
42	KONYA	-242.24	242.31
43	KUTAHYA	-100.77	553.01
44	MALATYA	-131.16	-377.52
45	MANISA	-209.9	774.35
46	KAHRAMANMARAŞ	-284.15	-247.12

47	MARDIN	-317.14	-664.58
48	MUGLA	-382.35	749.36
49	MUS	-11.057	-734.27
50	NEVSEHIR	-115.58	45.288
51	NIGDE	-180.22	36.261
52	ORDU	349.23	-205.38
53	RIZE	470.68	-435.79
54	ADAPAZARI	202.42	521.98
55	SAMSUN	329.39	-35.395
56	SIIRT	-216.41	-818.91
57	SINOP	418.92	78.222
58	SIVAS	39.843	-186.45
59	TEKIRDAG	294.09	777.81
60	TOKAT	138.72	-111.14
61	TRABZON	411.58	-386.86
62	TUNCELI	119.25	-527.35
63	SANLIURFA	-359.1	-457.41
64	USAK	-191.14	575.24
65	VAN	-30.328	-962.49
66	YOZGAT	46.113	51.76
67	ZONGULDAK	239.96	438.26
68	AKSARAY	-148.76	117.66
69	BAYBURT	296.26	-481.22
70	KARAMAN	-327.28	177.23
71	KIRIKKALE	40.435	192.15
72	BATMAN	-214.48	-727.14
73	SIRNAK	-351.77	-850.99
74	BARTIN	291.26	372.39
75	ARDAHAN	390.52	-761.53
76	IGDIR	292.62	-869.35
77	YALOVA	204.33	624.62
78	KARABUK	257.1	328.46
79	KILIS	-412.8	-308.36
80	OSMANIYE	-386.13	-172.34
81	DUZCE	207.83	443.41

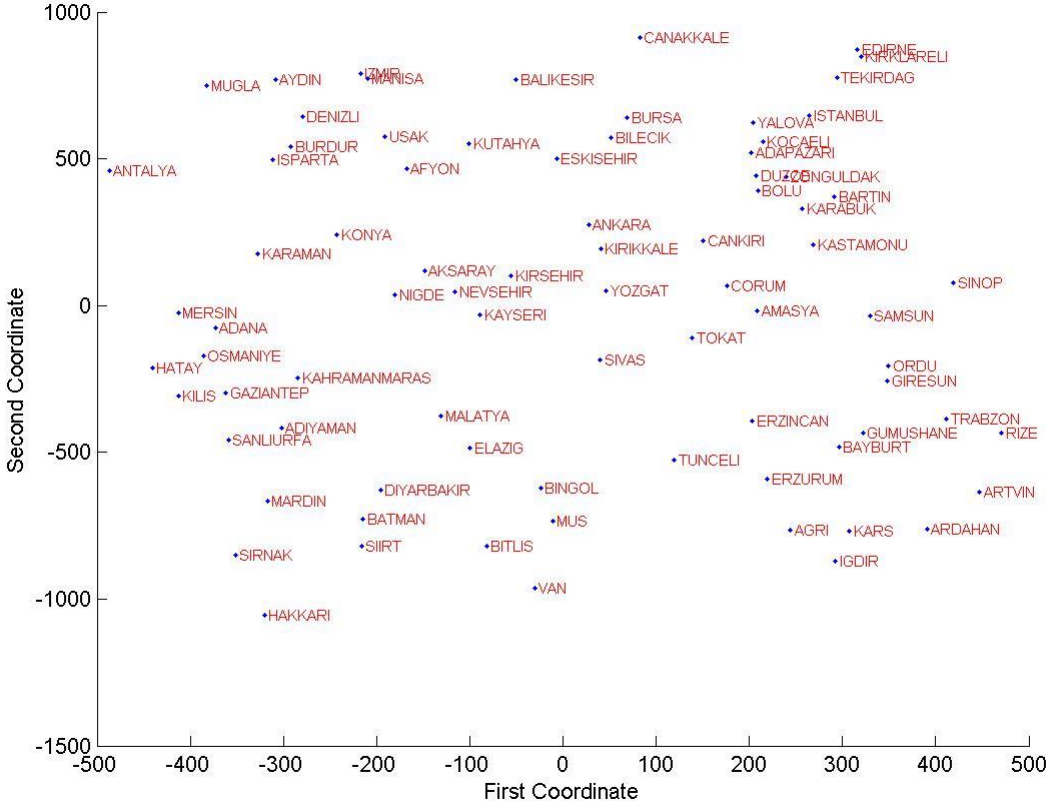

Figure 1: Plot of the cities of Turkey according to approximated 2D locations

IV. K-Means Clustering

k-means clustering algorithm depends on k reference points in the data. These reference points actually represent *k* different clusters. The other points are referred to nearest reference point. In other words, each point is regarded in the cluster formed by the nearest reference point. After referring each point to the reference points, the reference points are recalculated as the mean of the vectors belonging to each cluster. Then since the reference points have changed, a new calculation is required for each data point to determine possible changes in the clusters. This procedure is iterated until the reference points converge.

There are three different methods of initializing the *k*-means algorithm offered in the literature. First one is simply taking *k* random point as the initial reference vectors. Second one suggests adding small random vectors to the overall mean. Third one depends on dividing the principal component into *k* parts. Initially, we start the algorithm with 7 random data points as suggested in the first approach. However, it should be noted that even after reference points converge we are not guaranteed that each of them would belong to different regions in real. Moreover, the first reference point is regarded as the first region even if it probably belongs to some other region in real. Therefore, actually confusion matrices do not make very sense in that approach and one should be careful while interpreting the confusion matrices. Here is the obtained confusion matrix;

confusion_matrix =

9	0	0	0	0	0	2
3	5	0	0	0	0	0
8	0	0	0	0	0	0
0	0	0	7	4	0	7
0	0	4	1	6	3	0
0	2	3	0	0	4	0
1	7	0	1	0	0	4

As noted above, one should be careful about interpreting this matrix. For instance, one should not say that all of the cities of the third region are labeled as in the first region. Third row of the matrix simply says that, the first reference point is the nearest reference vector for all of the cities of the third region. Then one may comment that, most probably this reference point is actually somewhere in the third region. Then, on the contrary to classical confusion matrices, we suggest reading that row as it is the first row. Then, for instance the second row may remain as second because most probably this reference point is somewhere in the second region. This approach is a little bit problematic since one may easily see that more than one row can be interpreted as belonging to one region, while another region may not be referred by any of the reference vectors. Hence, initiating the *k*-means algorithm with random data points does not lead very meaningful results in terms of region-cluster comparisons.

In order to overcome this problem, one may come up with the idea that all of the reference vector should belong to different regions. Then, we decide to use the mean of the city locations belonging to each region in real as the initial reference vectors, and name this method as true region based *k*-means algorithm. The following is the formed confusion matrix;

confusion_matrix =

10	0	1	0	0	0	0
0	5	3	0	0	0	0
0	0	8	0	0	0	0
5	0	0	8	5	0	0
0	1	0	0	7	6	0
0	4	0	0	0	5	0
1	0	0	1	0	0	11

The problems we face while handling previous confusion matrix resolves here, because each row is now represented by a reference vector which most probably belongs to the region corresponding to the row. Therefore, one may comfortably make conventional confusion matrix interpretations here. For instance, there are 3 cities which are labeled to be in region three while they are indeed in region two. Note that number of wrongly placed cities is 27, which is one third of all the cities. The followings are the wrongly placed cities by the algorithm;

'ADIYAMAN is assigned to Akdeniz while it is indeed in Guneydogu Anadolu'

'ANTALYA is assigned to Ege while it is indeed in Akdeniz'

'ARTVIN is assigned to Dogu Anadolu while it is indeed in Karadeniz'

'BALIKESIR is assigned to Ege while it is indeed in Marmara'

'BINGOL is assigned to Guneydogu Anadolu while it is indeed in Dogu Anadolu'

'BITLIS is assigned toGuneydogu Anadolu while it is indeed in Dogu Anadolu'

'BOLU is assigned to Marmara while it is indeed in Karadeniz'

'BURDUR is assigned to Ege while it is indeed in Akdeniz'

'ELAZIG is assigned to Guneydogu Anadolu while it is indeed in Dogu Anadolu'

'ESKISEHIR is assigned to Marmara while it is indeed in Ic Anadolu'

'GAZIANTEP is assigned to Akdeniz while it is indeed in Guneydogu Anadolu'

'GUMUSHANE is assigned to Dogu Anadolu while it is indeed in Karadeniz'

'HAKKARI is assigned to Guneydogu Anadolu while it is indeed in Dogu Anadolu'

'ISPARTA is assigned to Ege while it is indeed in Akdeniz'

'MALATYA is assigned to Akdeniz while it is indeed in Dogu Anadolu'

'MUS is assigned to Guneydogu Anadolu while it is indeed in Dogu Anadolu'

'RIZE is assigned to Dogu Anadolu while it is indeed in Karadeniz'

'SIVAS is assigned to Karadeniz while it is indeed in Ic Anadolu'

'TRABZON is assigned to Dogu Anadolu while it is indeed in Karadeniz'

'SANLIURFA is assigned to Akdeniz while it is indeed in Guneydogu Anadolu'

'VAN is assigned to Guneydogu Anadolu while it is indeed in Dogu Anadolu'

'ZONGULDAK is assigned to Marmara while it is indeed in Karadeniz'

'BAYBURT is assigned to Dogu Anadolu while it is indeed in Karadeniz'

'BARTIN is assigned to Marmara while it is indeed in Karadeniz'

'KARABUK is assigned to Marmara while it is indeed in Karadeniz'

'KILIS is assigned to Akdeniz while it is indeed in Guneydogu Anadolu'

'DUZCE is assigned to Marmara while it is indeed in Karadeniz'

V. Decision Tree Analysis

In this part of the paper, we implement a decision tree algorithm for classification of the cities while the regions of the cities are regarded as the class labels. As a reminder for the mainstreams of the algorithm, the data is graded according to its entropy which is a measure for the homogeneity of the data in terms of number of classes that the data belongs.

Before giving the formulations used in the algorithm, we provide the notation used in the algorithm in Table 3.

Table 3: *Parameters used in Decision Tree Analysis*

Parameter	Definition
N_m	Total number of data points at hand
N_m^i	The number of class i data points
p_m^i	Probability for a data point belong to class i
Γ_m	The entropy of the data
K	The total number of labels

Let the data points at hand has a total number of N_m, then probability for a data point to be in a class i is simply the rate of number of class i data points over N_m. In other words, if N_m^i denotes the number of class i data points within the data at hand, the probability for a data point belong to class i is $p_m^i = \frac{N_m^i}{N_m}$. The entropy of the data, on the other hand, is given as $\Gamma_m = -\sum_{i=1}^{K} p_m^i \log_2 p_m^i$ where K is the total number of labels. If this entropy is under a tolerable level, then the data at hand is labeled according to the label that the majority of the data points belongs to. If not, then the data is split so that the decrease in the total level of the entropy is the maximum. To be able to find the best split, one needs to calculate the entropy of each possible split. The entropy of each part after split is also calculated according to above formula. However, when they are summed up to construct the total entropy of the split, each data part is multiplied with weights which is the rate of number of data points in the data part over the number of the points in the parenting data. For instance, if the number of data points in parenting data is N_m and the data is split among n data parts, entropy of the split is given as $\Gamma_m^{'} = -\sum_{j=1}^{n} \frac{N_{mj}}{N_m} \sum_{i=1}^{K} p_m^i \log_2 p_m^i$ where N_{mj} is the number of data points in the j^{th} child of the parenting data. The split that leads the minimum entropy is selected and implemented. After that, each child data node is treated as if it were the root node and the process continues until all of the data nodes are labeled.

The inputs for the algorithm, those are the dimensions on which the data points are divided through, are populations, total migration rates, annual incomes per person, areas, first and second coordinates (comes from multidimensional scaling) of the cities. 50 cities are randomly selected as the training instances and the remaining 31 cities are treated as the validation set. Error for the classification tree is the total number of misplaced instances for the validation set. Resulting confusion matrices are provided for three different values of "theta", namely for 0.1, 0.5 and 1. Note that theta stands for the hyper-parameter that indicates the complexity of the implied model that is used to predict the data. If a high parameter value is used the issue of over-learning occurs while under-learning occurs for low level of theta.

theta = 0.1,
confusion_matrix =

3	0	1	0	0	0	0
0	2	1	0	0	0	0
0	0	3	0	0	0	0
0	0	0	5	3	0	0
0	0	0	0	4	1	0
0	0	0	0	0	4	0
0	0	1	1	0	0	2,

theta = 0.5,
confusion_matrix =

3	0	1	0	0	0	0
0	2	1	0	0	0	0
0	0	3	0	0	0	0
0	0	0	7	1	0	0
0	0	0	0	4	1	0
0	0	0	0	0	4	0
0	0	1	1	0	0	2,

theta = 1
confusion_matrix =

3	0	1	0	0	0	0
0	0	3	0	0	0	0
0	0	3	0	0	0	0
0	0	0	8	0	0	0
0	0	0	2	0	3	0
0	0	0	0	0	4	0
0	0	0	1	0	0	3.

Observe that if we think the error as the total number of misplaced cities, the errors are 8, 6 and 10 for the theta values 0.1, 0.5 and 1, respectively. Then 0.1 and 1 values for theta causes relatively higher errors due to overfitting and model bias, respectively. Then the optimum theta value should be in between. The plot of theta versus error rate given in Figure 2 may help in finding the optimum theta which is as follows;

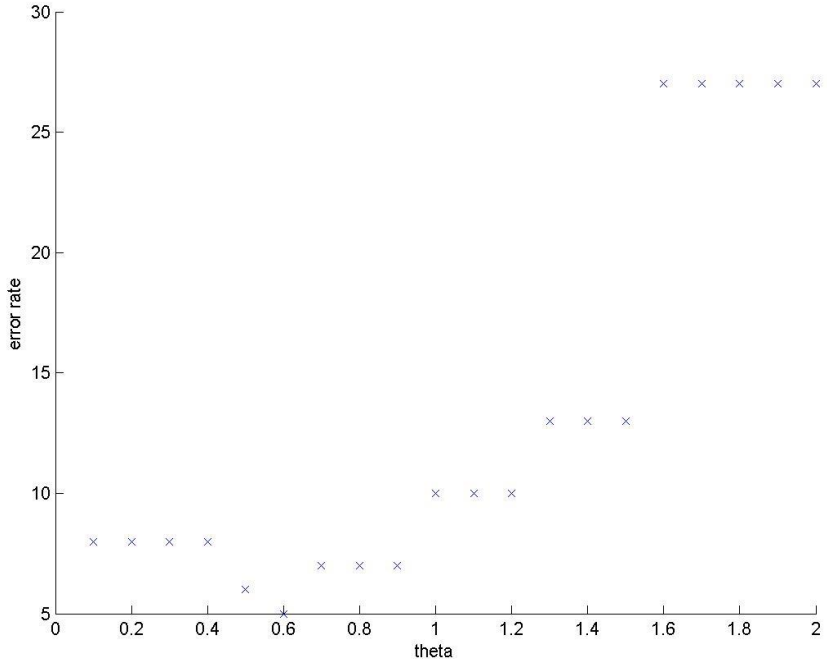

Figure 2: Plot of the Theta Versus Error Rate

As can be seen from Figure 2, error rate first decreases with theta and then increases starting from 0.6 level. Therefore, the best value for theta value leading minimum error rate is 0.6, and the resulting confusion matrix, misplaced cities for the validation set are as follows;

theta = 0.6

'ARTVIN is assigned to Dogu Anadolu while it is indeed in Karadeniz'

'BALIKESIR is assigned to Ege while it is indeed in Marmara'

'BURDUR is assigned to Ege while it is indeed in Akdeniz'

'CANKIRI is assigned to Karadeniz while it is indeed in Ic Anadolu'

'HAKKARI is assigned to Guneydogu Anadolu while it is indeed in Dogu Anadolu'

confusion_matrix =

3	0	1	0	0	0	0
0	2	1	0	0	0	0
0	0	3	0	0	0	0
0	0	0	7	1	0	0
0	0	0	0	4	1	0
0	0	0	0	0	4	0
0	0	0	1	0	0	3.

We also give the constructed decision tree for the optimum value of theta in Figure 3 below. As can be seen from Figure 3 that first two criteria while placing the cities into clusters are y and x coordinates of the cities. Then, population and migration rates come in.

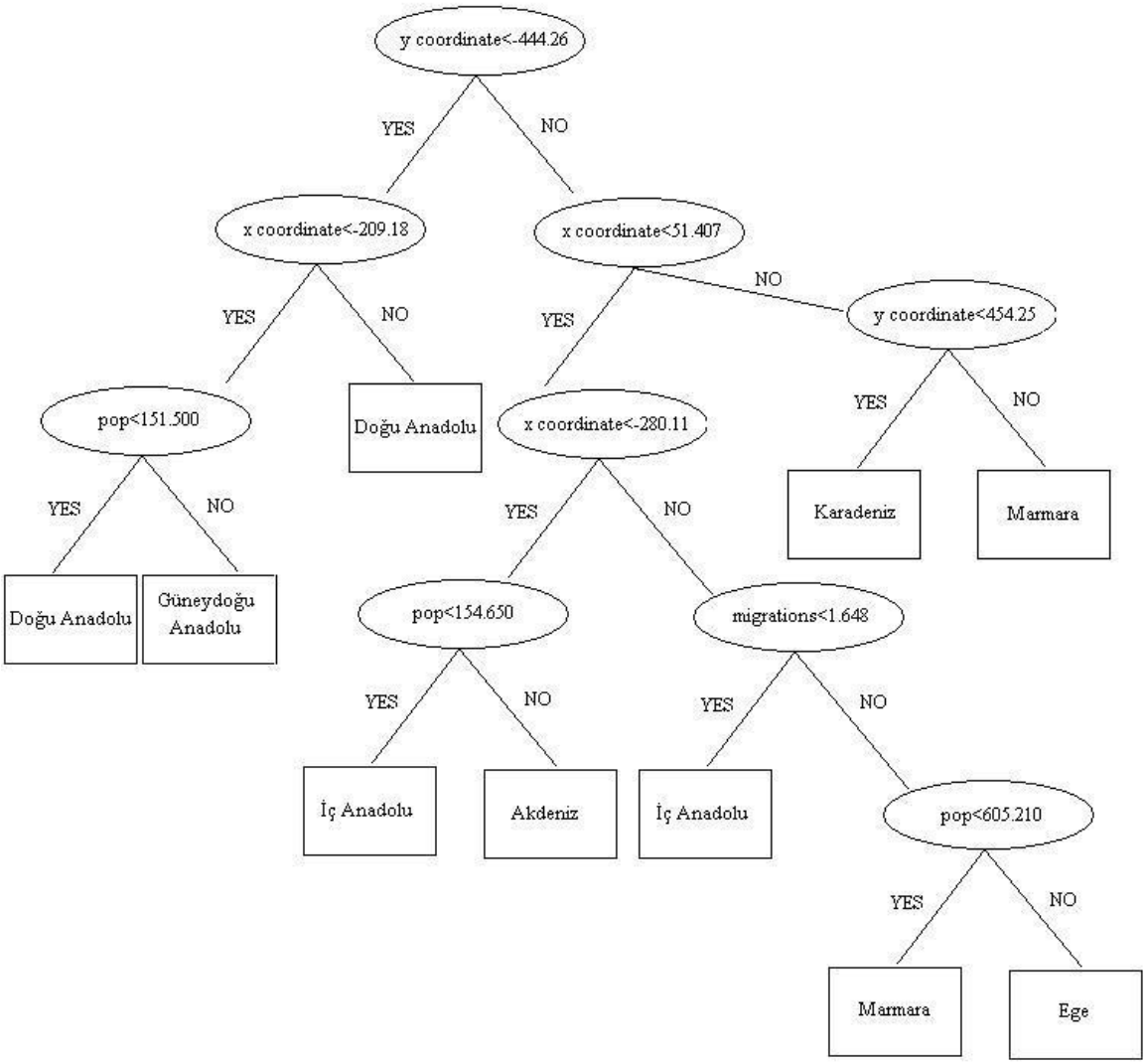

Figure 3: Plot of the decision tree for theta=0.6

VI. Conclusions and Discussions

One may see that results of the clustering algorithm which is run on the location data that is obtained from multidimensional scaling, has an error rate of 33.3% which is quite high. However, it should be noted that clustering is not a classification algorithm. Its procedures are totally independent of the class label knowledge, except the initiation phase in which the means of the cities of the regions are used as the initial reference vectors. Hence actually this error rate can even be considered as a success. Our intention was also not running a classification algorithm, but to see the differences between the results of clustering algorithm and the real regions. The correct comment here is that 33% of the cities of Turkey are closer to one of the neighboring region centers than the region centers of the regions that they belong to.

On the other hand, the error rate for the decision tree algorithm reduces to around 16% with the choice of correct theta value which is 0.6. 16% error rate comes from the rate 5/31, where 5 is the number of misplaced cities in the validation set, while there are 31 cities in the validation set. Then if we also take the inefficient number of data points into consideration, we can comfortably assert that the algorithm is an efficient one. Usage of first (x) and second (y) dimensions obtained from multidimensional scaling in decision tree algorithm is also justified because they are many times used while splitting the data even more than the other criteria, as can be seen from Figure 3. Finalizing comments about the decision tree algorithm may depend on the if-and rules that can be extracted from the decision tree. One can extract conclusive remarks about the social and economic situations of the regions in very little amount of times by just making use of these if-and rules.

Multi-dimensional scaling can be used in many areas, such as GPS-positioning, surface matching and dimension reduction. Besides, k-means clustering is one of the most frequent techniques used in classification. Classification is a special type of problem that can be faced in numerous disciplines such as location theory, artificial intelligence, etc. Finally, decision tree is a supervised artificial learning technique which is especially referred in machine learning and regression disciplines.

References

Alpaydin, E. (2009), "Introduction to machine learning", MIT press, Cambridge, London.

Altınel, K., Aras, N., and Oommen, J. B. (2003, September) "A self-organizing method for map reconstruction", Neural Networks for Signal Processing, 2003. NNSP'03. 2003 IEEE 13th Workshop on (pp. 677-687)

Ben-Arieh, D. and Deep K. G. (2012). "Data Envelopment Analysis of clinics with sparse data: Fuzzy clustering approach.", Computers & Industrial Engineering 63 (1):13-21.

Borg, I. and Groenen, P. (2003) "Modern multidimensional scaling: theory and applications", Journal of Educational Measurement, 40(3), 277-280.

Breiman, L. (2017). Classification and regression trees. Routledge.

Brodley, C. E. and P. E. Utgoff. (1995) "Multivariate Decision Trees.", Machine Learning 19: 45-77.

Bronstein, A. M., Bronstein, M. M. and Kimmel, R. (2006) "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching", Proceedings of the National Academy of Sciences, 103(5), 1168-1172.

Cömertler, N. and Kar, M. (2007) "Türkiye'de suç oranının sosyo-ekonomik belirleyicileri: yatay kesit analizi", Journal of the Faculty of Political Sciences, Ankara University , Vol. 62, No. 2 (2007): pp. 37-57.

Delias, P., Michael D., Evangelos G., Panagiotis M. and Nikolaos M. (2015) "Supporting healthcare management decisions via robust clustering of event logs", Knowledge-Based Systems 84:203-13.

Erkip, F. (2005). The rise of the shopping mall in Turkey: the use and appeal of a mall in Ankara. Cities, 22(2), 89-108.

Götz, T. I., Ermer, M., Salas-González, D., Kellermeier, M., Strnad, V., Bert, C., ... and Lang, E. W. (2017) "On the use of multi-dimensional scaling and electromagnetic tracking in high dose rate brachytherapy", Physics in Medicine & Biology, 62(20), 7959.

Gürbüz, M. and Karabulut, M. (2008). "Kırsal göçler ile sosyo-ekonomik özellikler arasındaki ilişkilerin analizi", Türk Coğrafya Dergisi, (50), 37-60.

Kandogan, E. (2001, August) "Visualizing multi-dimensional clusters, trends, and outliers using star coordinates", Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, (pp. 107-116). ACM.

Korte, C., & Ayvalioglu, N. (1981). Helpfulness in Turkey: Cities, towns, and urban villages. Journal of Cross-Cultural Psychology, 12(2), 123-141.

Jain A.K. , Murty M.N. and Flynn P.J. (1999), "Data Clustering: A Review", ACM Computing Surveys, Vol. 31, No. 3, pp. 264-323

Olson, C. H., Sanjoy D., Vipin K., Karen A. M., and Bonnie L. W. (2016) "Clustering of elderly patient subgroups to identify medication-related readmission risks.", International journal of medical informatics 85 (1):43-52.

Parekh, M. and Saleena B. (2015) "Designing a cloud based framework for healthcare system and applying clustering techniques for region wise diagnosis.", Procedia Computer Science 50:537-42.

Rokach, L. and Maimon, O. Z. (2008), Data mining with decision trees: theory and applications (Vol. 69). World scientific publishing, Singapore.

Tsumoto, S., Shoji H. and Haruko I. (2015) "Mining Schedule of Nursing Care Based on Dual-Clustering.", Procedia Computer Science 55:1203-12.

Uzun, B., Çete, M., & Palancıoğlu, H. M. (2010). Legalizing and upgrading illegal settlements in Turkey. Habitat International, 34(2), 204-209.

Vens, C., Struyf, J., Schietgat, L., Džeroski, S. and Blockeel, H. (2008) "Decision trees for hierarchical multi-label classification", Machine learning, 73(2), 185.

Yildiz, C. T. and Alpaydin, E. (2001) "Omnivariate decision trees", IEEE Transactions on Neural Networks, 12(6), 1539-1546.