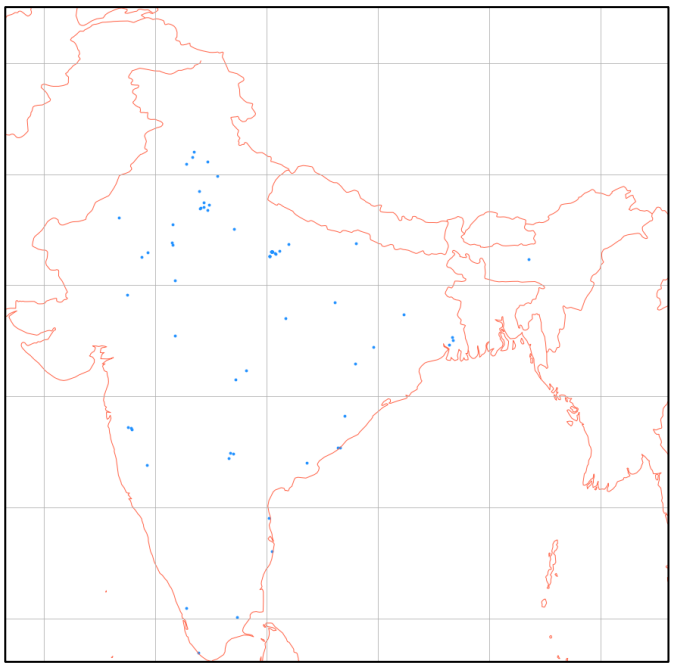
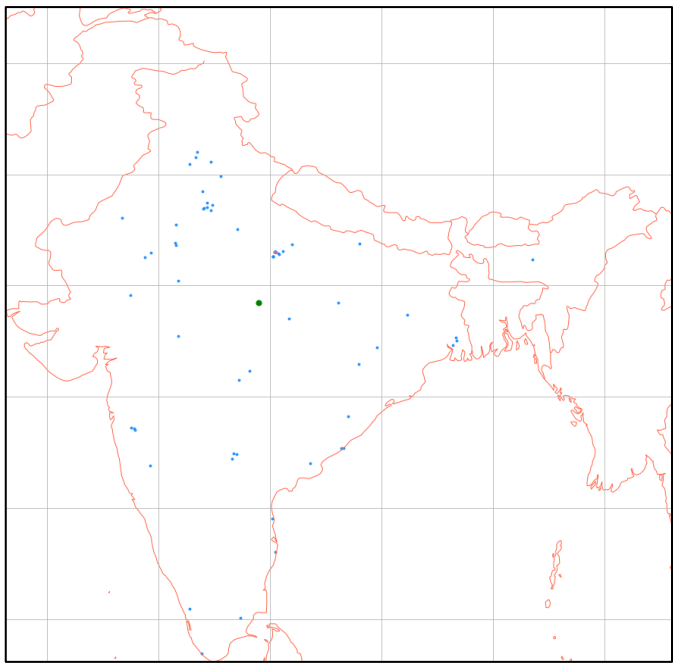
WAREHOUSE ALLOCATION

Data collected by the retailer was widely spread all over India. To understand the data better and make conclusions more evident, I plotted the points on India's map. Also, in the first stage, I tried to look out for a single option, i.e., if the online retailer can set up a single distribution centre and sell all products from that centre only, highlighted with a green mark on the map.





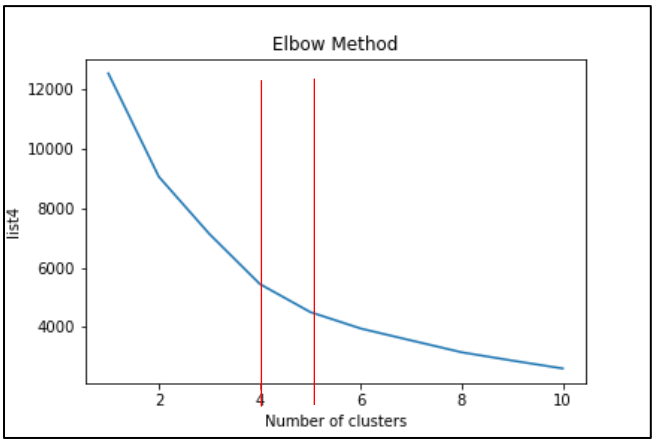
Total number of customers – 79; Total number of product categories – 4

Product	Apparel	Books	Electronics	Grocery	Total
Requirement (in kgs)	*405.24*	*589.91*	*309.02*	*806.34*	*2110.51*

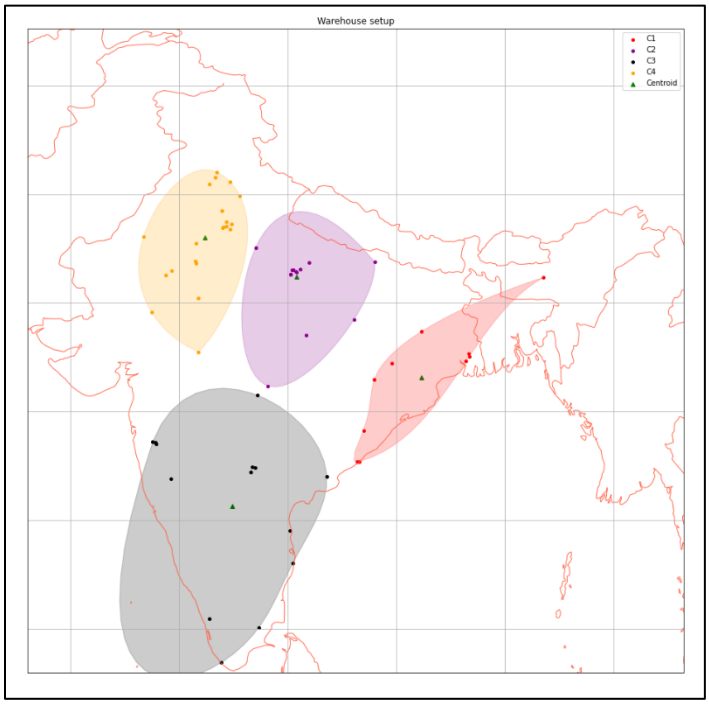
Given: Transportation Cost is proportional to Euclidean distance. I made a fair assumption of transportation cost being proportional to geographical distance as this is the case in real life. Also, Euclidean distance would be proportional to geographical distance.

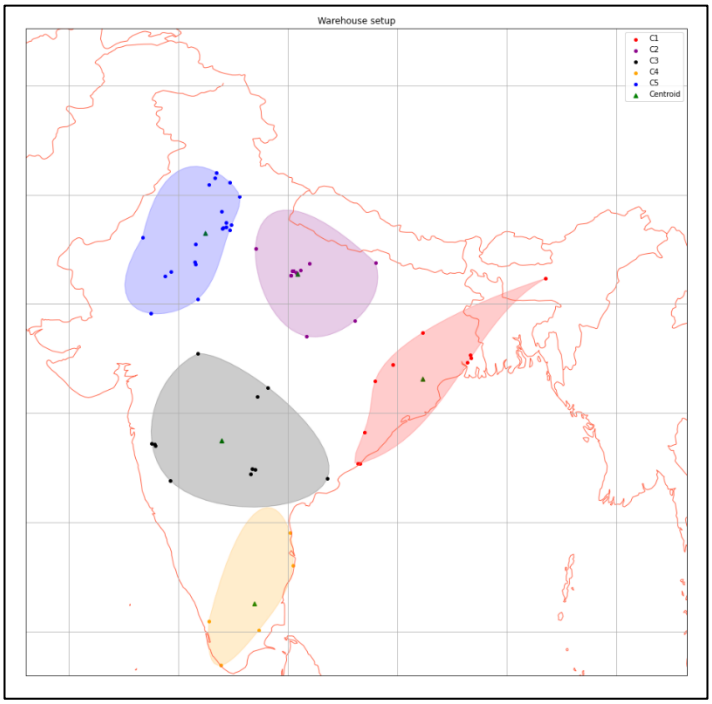
Considering warehouse setup cost one-time cost / fixed cost, I have assumed if there is no significant difference in transportation cost after setting up a new warehouse, then there is no use of that warehouse setup.

Using the elbow method, I suggest setting up either 4 or 5 warehouses as there is not much change in slope after those points, as shown below.



I chose to check for each case, i.e., 4 and 5 warehouses setup. Below are the clusters found in each case. Warehouses would be set up at the centroid of each cluster.





Let's also look at the cluster wise supply and total transportation cost born by each warehouse.

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*50.16*	*109.19*	*58.68*	*62.14*	*280.17*
Cluster 2	*191.88*	*270.3*	*95.44*	*339.79*	*897.41*
Cluster 3	*59.66*	*82.08*	*64.55*	*162.44*	*368.73*
Cluster 4	*103.54*	*128.34*	*90.35*	*241.97*	*564.2*

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*50.16*	*109.19*	*58.68*	*62.14*	*280.17*
Cluster 2	*191.17*	*250.74*	*93.49*	*333.59*	*868.99*
Cluster 3	*58.3*	*59.18*	*55.41*	*126.51*	*299.4*
Cluster 4	*11.95*	*42.61*	*13.75*	*61.09*	*129.4*
Cluster 5	*93.66*	*128.19*	*87.69*	*223.01*	*532.55*

No. of warehouses	1	4	5
Transportation Cost	*32068.62*	*17500.63*	*14696.68*

As there is a significant difference of around 3000 in transportation cost, it is always good to set up an extra warehouse and meet the demand. For initial setup, or when there is no constraint on quantity, five warehouses will be set up at the following locations:

Centroids	Latitude	Longitude
Cluster 1	21.562635	86.163641
Cluster 2	26.372268	80.425056

Cluster 3	18.769526	76.982077
Cluster 4	11.315807	78.462551
Cluster 5	28.260735	76.212671

The problem arises when we have a constraint on the maximum quantity of each product available in any warehouse is 100 kgs. In the above cases, multiple warehouses were dealing with more than 100 kgs of any products as the maximum requirement is grocery, i.e., 806.34 kgs. The minimum number of warehouses must be 806.34/100 = 8.0634, which will turn into a minimum requirement of 9 warehouses. Let's start building multiple clusters using the k-means algorithm and check the best reasonable possibility.

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*66.67*	*139.81*	*99.69*	*178.7*	*484.87*
Cluster 2	*338.57*	*450.1*	*209.33*	*627.64*	*1625.64*

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*59.79*	*98.23*	*43.87*	*68.26*	*270.15*
Cluster 2	*279.38*	*371.27*	*174.54*	*562.98*	*1388.17*
Cluster 3	*66.07*	*120.41*	*90.61*	*175.1*	*452.19*

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*50.16*	*109.19*	*58.68*	*62.14*	*280.17*
Cluster 2	*191.88*	*270.3*	*95.44*	*339.79*	*897.41*
Cluster 3	*59.66*	*82.08*	*64.55*	*162.44*	*368.73*
Cluster 4	*103.54*	*128.34*	*90.35*	*241.97*	*564.2*

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*50.16*	*109.19*	*58.68*	*62.14*	*280.17*
Cluster 2	*191.17*	*250.74*	*93.49*	*333.59*	*868.99*
Cluster 3	*58.3*	*59.18*	*55.41*	*126.51*	*299.4*
Cluster 4	*11.95*	*42.61*	*13.75*	*61.09*	*129.4*
Cluster 5	*93.66*	*128.19*	*87.69*	*223.01*	*532.55*

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*93.66*	*128.19*	*87.69*	*223.01*	*532.55*
Cluster 2	*19.11*	*46.91*	*40.49*	*38.01*	*144.52*
Cluster 3	*191.17*	*250.74*	*93.49*	*333.59*	*868.99*
Cluster 4	*57.77*	*50.89*	*53.09*	*107.62*	*269.37*
Cluster 5	*31.58*	*70.57*	*20.51*	*43.02*	*165.68*
Cluster 6	*11.95*	*42.61*	*13.75*	*61.09*	*129.4*

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1	*70.82*	*92.99*	*71.7*	*130.06*	*365.57*
Cluster 2	*19.11*	*46.91*	*40.49*	*38.01*	*144.52*

Cluster 3	*11.95*	*42.61*	*13.75*	*61.09*	*129.4*
Cluster 4	*182.41*	*237.05*	*93.09*	*330.92*	*843.47*
Cluster 5	*31.58*	*70.57*	*20.51*	*43.02*	*165.68*
Cluster 6	*47.89*	*50.74*	*50.43*	*88.66*	*237.72*
Cluster 7	*41.48*	*49.04*	*19.05*	*114.58*	*224.15*

One observation found after multiple times clustering is that there is always one cluster around UP that requires more products. The reason is straightforward, i.e., density around the area. So, the idea is to set multiple warehouses near the centroid of that cluster to fulfil the requirement. Now, let's set up multiple warehouses with internal clustering in the case of 4 and 7 clusters after carefully analyzing all other options.

Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1.1	34.94	40.81	38.88	26.35	140.98
Cluster 1.2	15.22	68.38	19.8	35.79	139.19
Cluster 2	191.88	270.3	95.44	339.79	897.41
Cluster 3.1	17.62	38.01	36.15	78.7	170.48
Cluster 3.2	7.79	24.47	13.48	51.86	97.6
Cluster 3.3	34.25	19.6	14.92	31.88	100.65
Cluster 4.1	24.45	37.08	19.12	61.54	142.19
Cluster 4.2	30.03	24	15.58	83.99	153.6
Cluster 4.3	14.8	37.03	3.88	38.27	93.98
Cluster 4.3	34.26	30.23	51.77	58.17	174.43

Total number of warehouses after setting up multiple warehouses at cluster 2 centroid = 9+4 =13

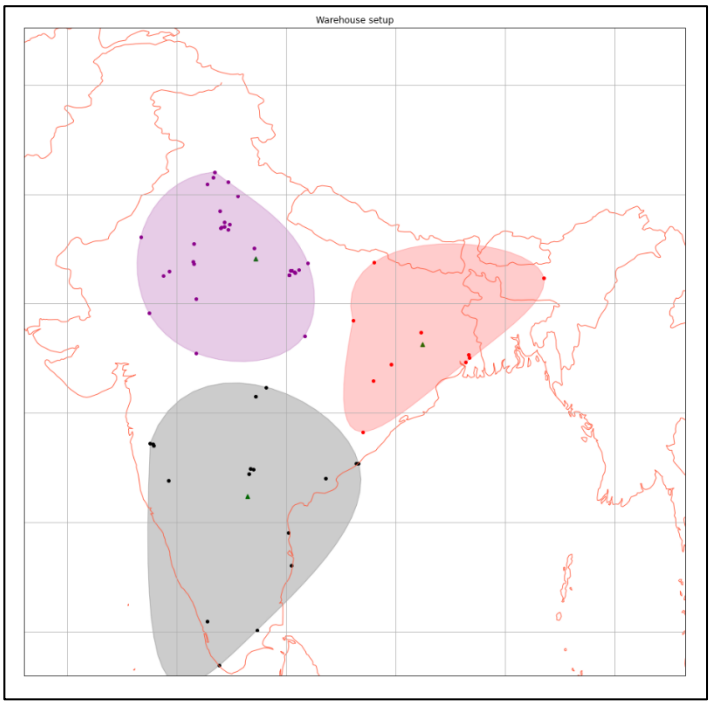
Product	Apparel	Books	Electronics	Grocery	Total
Cluster 1.1	46.37	55.91	52.58	68.52	223.38
Cluster 1.2	24.45	37.08	19.12	61.54	142.19
Cluster 2	19.11	46.91	40.49	38.01	144.52
Cluster 3	11.95	42.61	13.75	61.09	129.4
Cluster 4	182.41	237.05	93.09	330.92	843.47
Cluster 5	31.58	70.57	20.51	43.02	165.68
Cluster 6	47.89	50.74	50.43	88.66	237.72
Cluster 7.1	11.57	18.35	3.27	28.84	62.03
Cluster 7.2	29.91	30.69	15.78	85.74	162.12

Total number of warehouses after setting up multiple warehouses at cluster 4 centroid = 8+4 =12

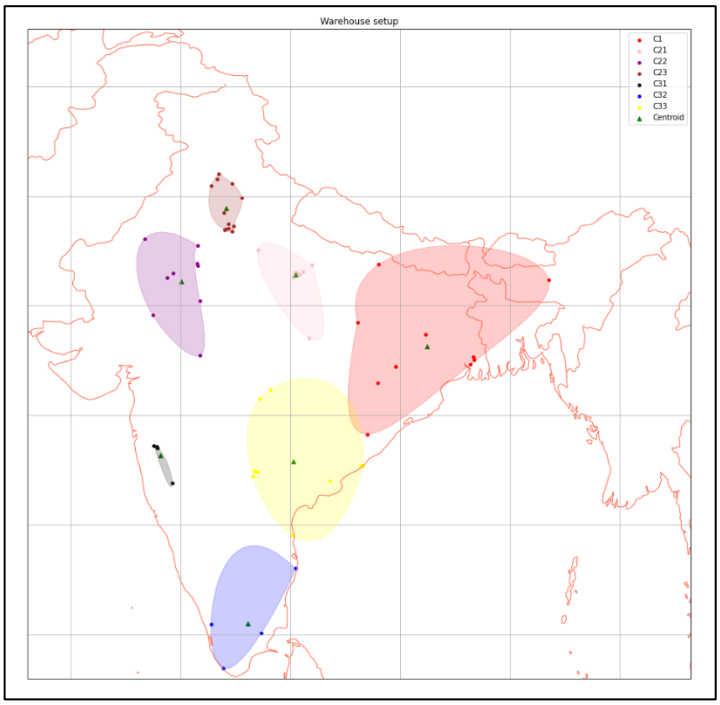
No. of warehouses	1	2	3	4	5	6	7	12	13
Total Cost	32068	27094	24077	17500	14696	12724	11140	9872	8960

In all the above tables, marked yellow options are not feasible as they don't satisfy the maximum limit of 100 kgs. Over here, a number of warehouses 12 and 13 is feasible as there will be an option to open multiple warehouses in the marked yellow cluster.

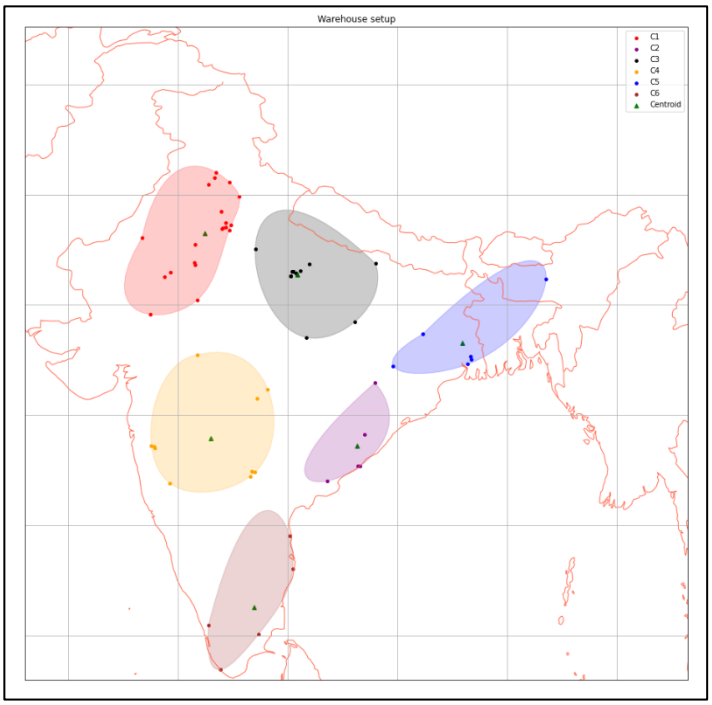
APPENDIX



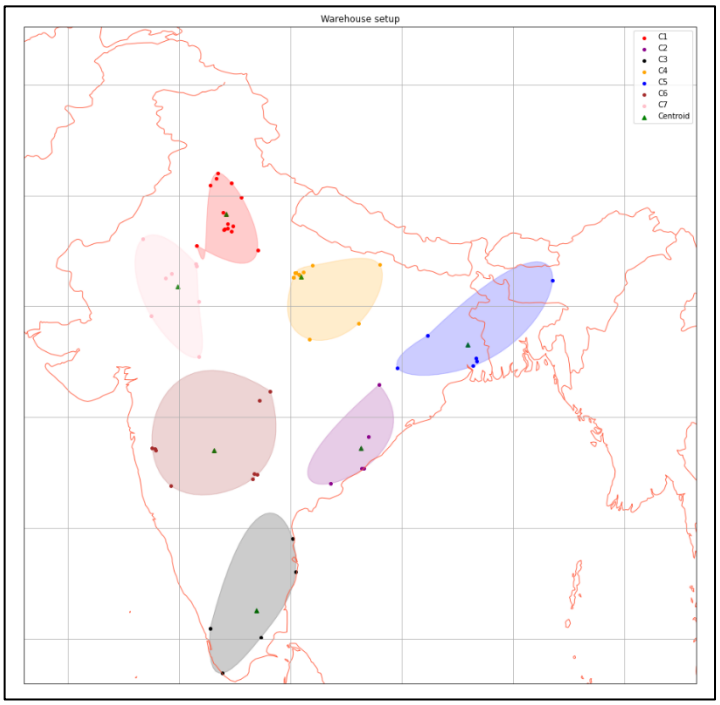
Number of Clusters = 3



Number of Clusters = 7



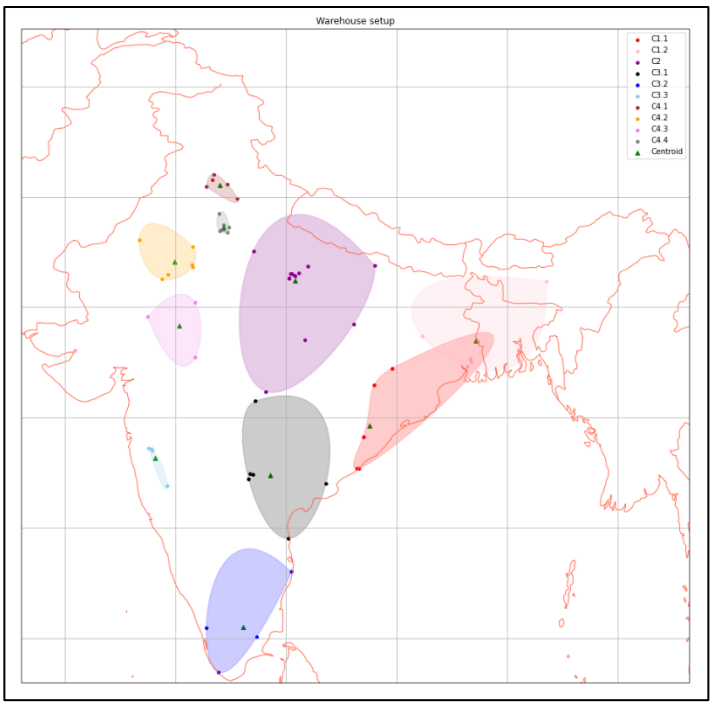
Number of Clusters = 6



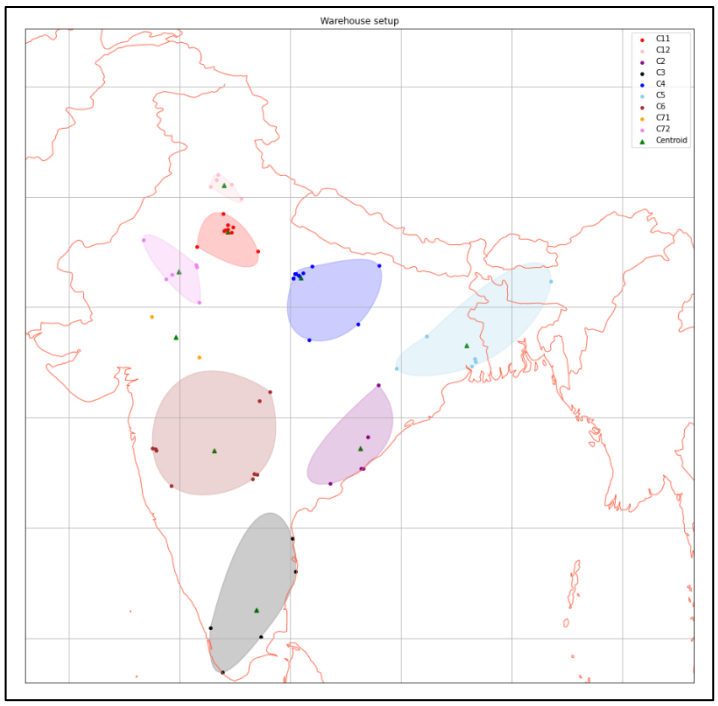
Number of Clusters = 7

In the above cases, the constraint of 100 kgs is not satisfied, and even it's not possible as we already calculated that the minimum number of warehouses required is 9. In any case, clusters in U.P. will not satisfy the constraint. So, it is a good idea to set up multiple warehouses in that cluster on its centroid.

Now, we will see two clusters analyses determined or derived from clusters of 4 and 7. Out of these two clusters, one will satisfy the criteria. As already seen in the above table, we can see that the total cost of 9 cluster distribution is more than that of 10 cluster distribution. We will look into which one to choose out of these two options.



Number of Clusters = 10



Number of Clusters = 9

Centroids	Latitude	Longitude
Cluster 1.1	*28.40857*	*77.15253*
Cluster 1.2	*30.54606*	*76.98288*
Cluster 2	*18.60683*	*83.1558*
Cluster 3	*11.31581*	*78.46255*
Cluster 4	*26.33564*	*80.48472*
Cluster 5	*23.26859*	*87.94301*
Cluster 6	*18.52351*	*76.56836*
Cluster 7.1	*23.64992*	*74.80795*
Cluster 7.2	*26.63094*	*74.95883*

Final allocation of warehouse

Latitude	**Longitude**	**Apparel**	**Books**	**Electronics**	**Grocery**	**Cluster**	**Warehouse no.**
29.23808	*76.97522*	*4.15*	*0.69*	*18.26*	*8.22*	*1.1*	*1*
28.71173	*77.16677*	*5.75*	*14.91*	*0.9*	*3.58*	*1.1*	*1*
28.61673	*77.42197*	*0.86*	*10.83*	*5.48*	*2.38*	*1.1*	*1*
28.53143	*77.15253*	*9.19*	*0.77*	*1.16*	*18.38*	*1.1*	*1*
28.47796	*77.0194*	*0.91*	*2.2*	*17.67*	*9.15*	*1.1*	*1*
28.44424	*77.00908*	*10.76*	*0.41*	*2.1*	*6.12*	*1.1*	*1*
28.37341	*77.32585*	*2.64*	*0.42*	*6.2*	*10.34*	*1.1*	*1*
27.73922	*75.78626*	*3.35*	*11.99*	*0.41*	*7.68*	*1.1*	*1*
27.5443	*78.51568*	*8.76*	*13.69*	*0.4*	*2.67*	*1.1*	*1*
30.45329	*76.39459*	*6.83*	*4.34*	*0.77*	*14.58*	*1.2*	*2*

31.02216	76.72433	6.74	0.21	14.4	1.14	1.2	2
30.75914	76.66274	6.48	4.1	0.3	18.21	1.2	2
30.5624	77.34411	3.51	19.53	0.43	8.64	1.2	2
29.93333	77.78865	0.89	8.9	3.22	18.97	1.2	2
21.47352	83.97705	12.28	0.45	4.98	9.06	2	3
17.6851	83.2035	5.22	0.16	16.51	2.92	2	3
17.01933	81.80065	0.53	8.29	2.32	18.89	2	3
17.7121	83.3121	0.48	18.61	7.6	3.54	2	3
19.1441	83.4857	0.6	19.4	9.08	3.6	2	3
10.48297	76.38469	1.21	8.26	0.57	12.96	3	4
14.52737	80.08754	4.16	18.14	0.27	9.23	3	4
13.0208	80.224	3.22	8.3	0.4	17.63	3	4
10.07873	78.67042	2.7	0.4	9.43	11.25	3	4
8.469167	76.94611	0.66	7.51	3.08	10.02	3	4
26.30172	80.1334	15.48	9.89	0.93	1.38	4	5
26.30172	80.13	3.04	9.54	0.53	15.58	4	5
26.87583	84.00567	13.59	7.41	4.04	0.54	4	5
26.84438	80.99371	1.88	9.36	0.75	12.28	4	5
26.53511	80.58228	8.26	3.56	0.59	19.48	4	5
26.52348	80.2346	2.1	0.59	5.25	14.33	4	5
26.51284	80.23489	16.18	9.99	0.11	2.2	4	5
26.51284	80.23489	0.34	16.41	1.12	6.46	4	5
26.51284	80.23489	17.72	8.04	0.77	3.18	4	5
26.51268	80.19654	10.99	0.25	2.07	7.07	4	5
26.51268	80.19655	8.36	3.76	0.51	18.68	4	6
26.51234	80.23197	14.06	8.61	1.51	0.14	4	6
26.50639	80.22811	0.65	8.47	2.56	16.97	4	6
26.50534	80.2262	9.62	0.56	2.73	18.06	4	6
26.5051	80.22758	3.13	9.44	0.26	17.33	4	6
26.50497	80.23439	1.71	18.83	6.74	0.62	4	6
26.50485	80.23493	0.85	1.78	17.41	7.65	4	6
26.50473	80.22675	3.08	12.41	0.48	6.55	4	6
26.50473	80.22675	9.77	1.07	0.63	14.27	4	7
26.50465	80.22665	0.2	7.15	1.57	17.25	4	7
26.50464	80.22658	0.41	8.11	4.96	16.72	4	7
26.50463	80.22668	1.27	17.46	5.75	0.62	4	7
26.50458	80.22616	2.09	9.6	0.89	19.05	4	7
26.50441	80.22658	2.26	9.3	0.2	14.24	4	7

26.50438	80.22662	4.92	5.15	0.2	10.41	4	7
26.50432	80.22654	18.41	0.45	2.93	7.42	4	8
26.43667	80.36278	2.76	10.65	0.8	6.42	4	8
26.42727	80.39772	1.58	5.37	0.93	12.82	4	8
26.30172	80.13363	4.49	6.61	0.29	13.31	4	8
26.30159	80.13346	0.8	8.93	2.3	14.82	4	8
24.21971	83.04339	1.74	0.4	5.26	12.04	4	8
23.5274	80.84029	0.67	7.9	18.02	3.03	4	8
26.16804	91.77818	6.95	19.12	0.81	2.92	5	9
23.68806	86.1595	6.42	17.15	0.33	4.58	5	9
22.65952	88.32818	0.55	13.67	3.2	8.43	5	9
22.5435	88.3797	0.29	17.67	9.75	4.62	5	9
22.311	88.2055	1.01	0.77	5.71	15.24	5	9
22.2414	84.807	16.36	2.19	0.71	7.23	5	9
21.18487	79.08888	0.71	19.56	1.95	6.2	6	10
20.75468	78.60686	7.39	0.61	2.68	17.09	6	10
18.62367	73.78216	7.74	2.49	0.78	15.73	6	10
18.57325	73.89909	15.6	0.11	5.46	3.27	6	10
18.5324	73.9441	10.48	0.66	4.88	6.71	6	10
17.47058	78.36697	0.16	7.72	4.95	16.49	6	10
17.41409	78.51293	4.42	0.8	15.09	8.05	6	10
17.23386	78.30173	0.96	2.45	10.84	8.95	6	10
16.92417	74.61254	0.43	16.34	3.8	6.17	6	10
24.56596	73.72895	1.69	18.2	0.61	9.88	7.1	11
22.73389	75.88694	9.88	0.15	2.66	18.96	7.1	11
28.04511	73.35319	6.24	0.5	3.74	19.61	7.2	12
26.928	75.748	0.39	3.9	6.09	17.43	7.2	12
26.83525	75.76238	1.02	6.78	0.73	13.47	7.2	12
26.48346	74.64461	10.5	0.24	1.05	8.03	7.2	12
26.28	74.38	8.53	0.59	3.56	17.77	7.2	12
25.2138	75.8648	3.23	18.68	0.61	9.43	7.2	12

I have used the cartopy module to show scatter plots in the geographical map. Also, I have used geographical distance instead of Euclidean distance to measure transportation cost as it seems to be the best measure. Also, Euclidean and geographical distance will be in proportion in this case.