State of Australian Cities Conference 2013

An Open-Source Tool for Identifying Industrial Clusters in a Data-Poor Environment

Sophie Sturup, Jennifer Day, and Yiqun Chen
Faculty of Architecture, Building, and Planning, University of Melbourne,
ssturup@unimelb.edu.au
Faculty of Architecture, Building, and Planning, University of Melbourne,
jday@unimelb.edu.au; Faculty of Engineering, University of Melbourne,
yiqun.c@unimelb.edu.au

Acknowledgements: This project has been supported by the Australian National Data Service (ANDS) and the Australian Urban Research Infrastructure Network (AURIN) through the National Collaborative Research Infrastructure Strategy Program and the Education Investment Fund (EIF) Super Science Initiative.

Abstract: This paper describes an open-source software tool for identifying employment clusters, available through the Australian Urban Research Infrastructure Network. The purpose of developing this tool was to create a means to examine spatial employment clustering in metropolitan regions at finer spatial units than are currently supplied by the Australian Bureau of Statistics.

This project responds to a consensus among local policy makers, that Melbourne needs to adopt a multi-nodal metropolitan planning strategy in order to foster economic development and reduce commuting. For decades, metropolitan planning strategies that have sought to promote non-CBD centres in Melbourne. The tool further responds to a consensus among economic development planners that ABS data are insufficient to identify local urban clusters for analysis. We wish to understand whether spatial policies aimed at cluster development have actually resulted in employment clusters. This tool moves us toward examining those policies by providing a framework to identify whether and where local employment clusters have formed.

To build the tool, we have used the open-source Cran R spatial analysis packages. After the user specifies an industry of interest, the tool splits Census Destination Zones (DZNs) into smaller polygons based on land use data, and attributes Census Journey to Work (JTW) job destinations to each smaller polygon. Then, a modified Ward's algorithm clusters the small polygons using spatial and non-spatial attributes. The tool also allows the user to specify whether clustering should be at local or regional scale.

Introduction

This paper describes a workflow tools hosted by AURIN. The workflow works with a specially created data set to identify economic sector clusters in North West Melbourne (NWMMR), Australia. The tool and the data set were developed to demonstrate the value of data integration in a project funded by AURIN and ANDS.

The project responds to a consensus among Australian policy makers, that Melbourne needs a multi-nodal metropolitan form to foster economic development, and reduce commute burdens. For decades, Melbourne metropolitan planning strategies have sought to promote non-CBD activity centres. Initally these were essentially local shopping precincts (Melbourne and Metropolitan Board of Works 1954). But since 1981, Victoria has been trying to develop activity centres – local job, shopping, and recreational centres that serve the local population (McNabb et al 2001). Since the 2002 plan (Department of Infrastructure 2002), these activity centres gave aimed to reduce the need for commuters to travel to the city centre, and to supply firms with incentives to locale in a jobs cluster. In theory, there is a benefit to commuters and taxpayers, through reduced commutes and more local jobs, and economic agglomerative effects.

Melbourne lacks data suitable for analysis of economic spatial clusters at the urban level. The data available to assess economic clustering is limited to Census Journey-to-Work (JTW) data. Economic and firm-location, output and local GDP data is not publically available, and access is tightly controlled. The issues the lack of economic output data creates is beyond the scope of this paper, however even the data which is available is only available at relatively large scale, unsuitable to

analysis at local scale. Thus, we developed software that firstly could disaggregate widely-available Census data into spatial units appropriate for urban-level economic analysis, and secondly could locate clusters of employment types using the finer grained data. We developed two tools, the first tool splits the large spatial (Census JTW destination zones) units into much smaller spatial units, and attributes jobs to these smaller spatial units.

The second tool applies a clustering algorithm to any loaded spatial data to generate clusters for analysis. This clustering method is based on a modified Wards clustering algorithm, using an index resulting from Joshi's (2009, 2011) polygon dissimilarity function. The polygon dissimilarity function uses spatial (i.e., physical proximity) and non-spatial (e.g., job number similarity) attributes to calculate a value for each urban spatial unit. Users can select non-spatial attributes for inclusion in the polygon dissimilarity function according to what is available in their data set. When used with the data set generated by the first tool, attributes can be selected to represent value chains (meaningful groups of economic activity according to economic sector).

Together, the two tools provide an analytical process by which researchers can identify spatial clusters of industry in the NWMMR. In turn, these clusters can then be used to address issues of concern to urban policy making which researchers and analysts have struggled to address due to a lack of fine spatial data. The following paper describes both the tools and some detail of an example output.

ABOUT CLUSTERING

Our analysis is founded on a Ward's clustering framework, also known as a Spatial Hierarchical Clustering framework (see Carvalho *et al*, 2009). This framework constructs clusters based on the similarity of some feature of the spatial units to be clustered. The algorithm uses criterion that minimizes the total within-cluster variance once two polygons have been merged. To implement this method, we find the pair of clusters that leads to the minimum increase in total within-cluster variance. This increase is a weighted squared distance between cluster centres and is a single variable. At the initial step, all clusters are singletons (clusters containing a single point). To apply a recursive algorithm under this objective function, the initial distance between individual objects must be (proportional to) squared Euclidean distance.

The specific method applied is Ward's minimum variance method, which is a special case of the approach originally presented by Joe H. Ward, Jr. (1963). Ward suggests a general agglomerative hierarchical clustering procedure, where the criterion for choosing the pair of clusters to merge at each step is based on the optimal value of an objective function. This objective function could be "any function that reflects the investigator's purpose." In our case we have developed a function adapted from Joshi's Polygon Distance Function (Joshi 2009, 2011), which allows us to develop a univariate measure of difference that includes multivariate (spatial and non-spatial) data (see next section).

Almost all clustering procedures involve manipulations of univariate data. Even those processes that use multivariate data in clustering are actually univariate clustering procedures applied to objective functions based on multivariate data. For example, Srucca (2005) applies the K-means algorithm to a Getis-Ord spatial association statistic, to generate economic clusters in Italy. Feser (2005) combines a principle components analysis to distil the data into value-chain clusters, with Ward's algorithm, to arrive at spatial industry clusters of American counties. Entropy-based approaches, e.g., Jia and Jiang (2012), are more useful for micro-data such as intersections, or parcels.

One benefit of the Wards algorithm is that it allows the analyst to define a cluster based on polygon proximity, without need for a "seed" spatial unit to begin the clustering process. This means no pre-defined notion of where clusters are is needed to generate clusters. It allows a credible comparison of the actual location of such clusters in comparison with locations that government policies have pursued.

As a method of cluster analysis, hierarchical clustering seeks to build a hierarchy of clusters. Strategies for hierarchical clustering can be agglomerative or divisive. Agglomerative models take a bottom-up approach, wherein each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy. Divisive models take a top-down approach, where all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy. Agglomerative algorithms are more efficient to compute compared with divisive ones (Cimiano 2004).

TOOL DESCRIPTIONS

We have developed two tools to assist our research. The first generates a base map using the polygon splitting tool. The second is the clustering algorithm. The clustering algorithm is a 'workflow' within the AURIN portal and makes use of a polygon dissimilarity index. This section describes the geographic splitting tool, algorithm and polygon dissimilarity function.

Geographic Splitting

The geographic splitting tools parses traditional spatial units of analysis into finer spatial units that are suitable for sub-metropolitan spatial clustering analysis, and attributes jobs to these finer spatial units. This tool responds to a lack of availability of data at spatially disaggregated levels suitable for sub-metropolitan analysis. Studies examining local economic development in Australia have typically been forced to rely on Census Journey-to-Work (JTW) data which reports the number of jobs by industry type in spatial units called destination zones (DZNs). This has been problematic for three reasons: 1) DZNs can contain areas of land that are residential and without economic activity; 2) the DZNs are not constant from one Census to the next; and 3) DZNs are generally too large to be useful for sub-metropolitan clustering analysis. These problems affect analysis in the following ways.

Firstly, if an analyst is seeking to examine local economic clustering, the presence of large tracts of residential land can muddle the analysis. A DZN with a large land area may be largely residential, but contain an important and dense economic cluster on a small portion of its land area, e.g., the industrial area in Figure 1. If the density of jobs in that cluster is averaged over the entire DZN, the result could be a DZN with an overall low concentration of jobs. As a result of this low density, spatial clustering models may not register the economic activity in that DZN as being sufficient to be added to a cluster.

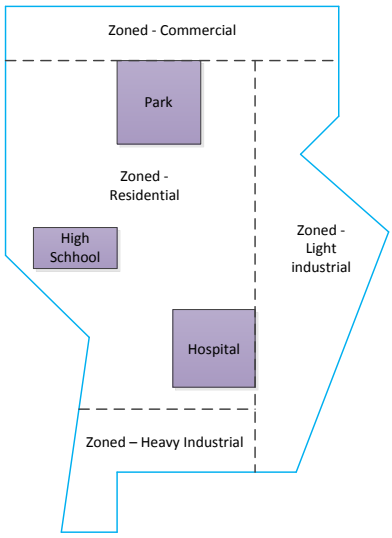
Second, in attempting to study economic change over time in a DZN, one must first consider whether the DZN boundaries have changed in the study period. If they have, then it is impossible to know whether economic changes which show up in models have resulted from boundary changes or actual economic change. Since DZN boundaries do frequently change between censuses, this makes temporal analysis problematic.

Third, the treatment of a DZN as a single spatial unit means that spatial clusters must consist of one or more entire DZNs. If DZNs are large, the clustered areas could become large and unrepresentative of the actual clusters that may exist within them. Imagine two small but dense manufacturing employment precincts sitting on either side of a shared border between two DZNs, which between them comprise 10 hectares. With our tool, it is possible to identify a ten-hectare cluster, rather than combining the far-larger DZNs into a much-larger cluster.

The polygon splitting tool parses DZN spatial units into smaller spatial units with the jobs of the DZN attributed to the appropriate sub-area of the DZN. These smaller polygons allow us to create more accurate descriptions of economic activity zones. With smaller geographies, changes in industrial characteristics observed over time can be more reliably attributed to changes in the economic activity under study rather than changes in Census and DoT spatial designations. Economic clusters can be identified at a scale meaningful for sub-metropolitan analysis.

We arrive at our smaller polygons by using Victorian government spatial land-use and land-use zoning data to parse each DZN into different types of precincts. We keep all precincts where economic activity is likely to occur, and delete the residential, recreational, and other-zoned sections where economic activity is generally sparse. Then, the JTW job numbers are ascribed to these parsed areas within a given DZN, by matching land uses with industry type. Figure 1 illustrates this process. In Figure 1, the DZN is outlined in blue. The area is parsed according to zoning types and known uses, e.g., hospitals, schools, and parks. Then, the 250 medical jobs are allocated to the hospital area of the zone, and the 34 retail jobs are allocated to the commercial area of the zone. We call these parsed polygons *middle polygons*.

Figure 1 Hypothetical Parsed DZN



Blue line = Destination Zone
Dotted line = Land use zoning – Vic Map Data
Purple areas = Government land – Vic Map Data
Areas checked for content – Valuer General's Data

Output will display:
Jobs by industry allocated to zones
EG: retail jobs = 34
Medical jobs = 250

We use six key datasets in creating the middle polygons. We start with the DZN spatial files, and then associate JTW data with the DZN spatial data to create an intermediate spatial dataset. The JTW data is available at four digit ANZSIC (Australian and New Zealand Standard Industrial Classification). Then, we prepare land-use zoning data: We combine the spatial Shapefile containing the allowable land uses with the DZN geography to split each DZN into middle sized polygons by land use zone. This creates a new base map of middle polygons. A rule set is then applied to allocate jobs for each DZN to the middle polygons. The rule set was developed by creating a table of allowed land use zones for each ANZSIC. To compile this listing, we scrutinized the Victorian 2006 land-use zoning statutes (Victoria Planning Provisions 2006, accessed 6 December 2012).

The attribution of job trips to each middle polygon is more complicated than Figure 1 suggests. Each DZN can produce many middle polygons. Although each middle polygon will contain only one zoning code, they can potentially contain many different types of jobs, and thus, many different industrial codes. Furthermore, there are many cases where a DZN produces multiple middle polygons with the same zoning. In these cases, we divide the jobs in the DZN among the middle polygons eligible to accept them. Lacking data for a more-precise attribution mechanism, we assign jobs to each of the middle polygons in proportion to the land area. We recognize that this process could unnecessarily disaggregate the jobs within a DZN, but it is a limitation of the data. Polygons with fewer than one job are discarded and do not appear in the final polygon set.

We also note a major limitation of our zone-code-to-ANZSIC mapping. According to Victorian zoning statutes, non-residential uses can be located in residential zones with special permissions. Our analysis ignores these special permissions, assigning jobs only to areas zoned appropriately to receive them. We do this because we lack data on the spatial locations of specially-permitted industrial activity that sits in residential zones.

Another problem that arises in the use of land-use zones to generate the middle polygons, is that land can be zoned for a use but not yet be in use. For instance, land in urban growth areas may be zoned for development, but no development may yet be on that land. Once we have this first iteration of middle polygons, we wish to verify that the land-use zones actually contain the designated uses. For this verification, we use property valuation data from the office of the Valuer General (VG) of Victoria.

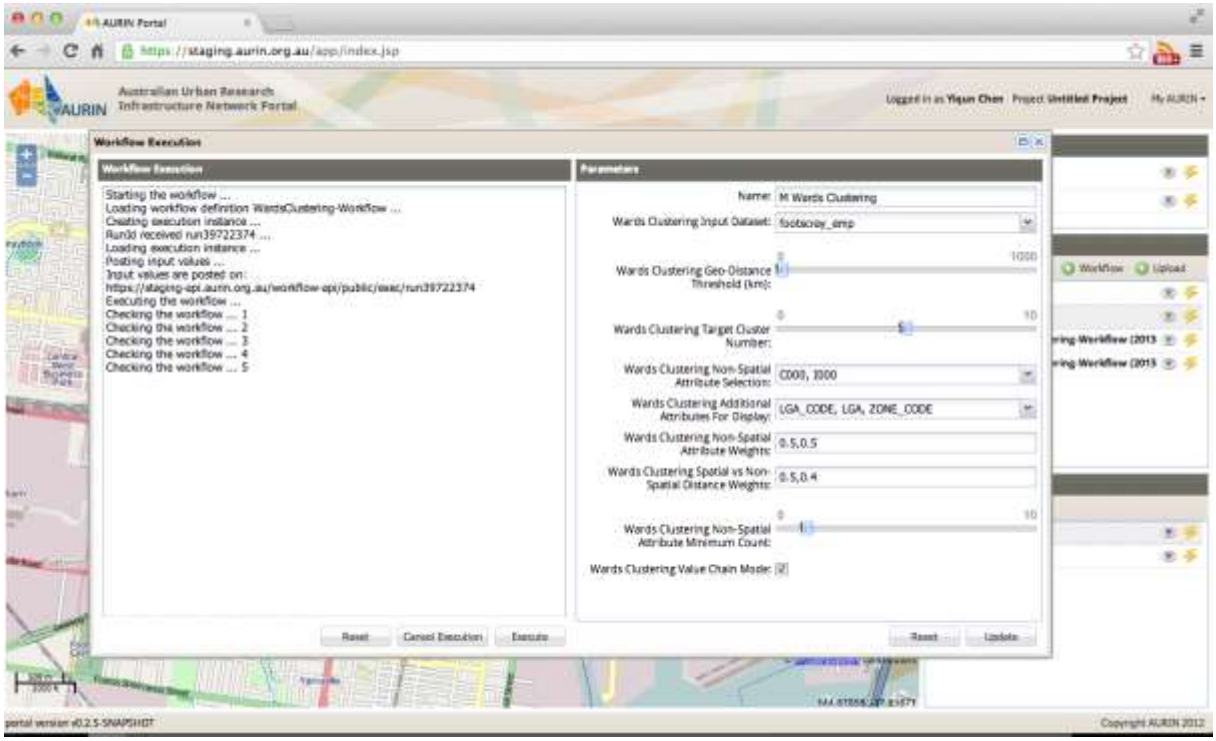
VG data contains information at a parcel level about the type of land use occurring on the parcel, e.g., residential, light industrial, etc. However the data is incomplete, it does not cover every parcel and therefore cannot be used to identify land uses across DZNs. Once the first iteration of middle polygons is defined, the tool overlays the VG data with the middle polygons. The tool then determines whether each middle polygon contains at least one small parcel from the VG data, with a land use matching that designated by the middle polygon base map. Middle polygons that do not have at least one parcel matching the land-use zoning are deleted from the dataset. This process reduced the number of middle polygons from 8,575 to 5,795 for the 2006 NWMMR data.

The output for this tool is new data set, involving polygons to which journey to work data is attached. This data set involves a larger number of polygons than the original data set provided by the ABS (DZNs). These new polygons are not necessarily contiguous, and they are generally much smaller in size than DZNs. The code for this process is available from https://github.com/yiqunc/AURIN_EmpClustering, the script is called Emp_BasemapBuilder.r.

Main Algorithm: MWards Clustering Workflow

The main algorithm is usable via a user interface on the AURIN portal, under Workflow/Employment Demonstrator/Mwards Clustering. The interface allows the user to upload any data set to run the clustering process on. We have used the data set generated by our first tool to demonstrate the features of the workflow. The workflow is built into the AURIN architecture as OMS3 components (Pettit et al 2013)

Figure 2: Screen shot AURIN Portal user interface for MWards Clustering



Geo-distance threshold

As shown in figure 2, the user interface allows users to select a clustering geo-distance threshold. When cluster centroid distances exceed this selected distance the algorithm will stop. This is a significant modification to Ward's algorithm. One problem with the standard Ward's clustering algorithm is that it does not have, within it, the capacity to incorporate spatial information. This is problematic because the algorithm could cluster polygons that are neither near each other nor spatially adjacent to each other. We address this problem with our MWards algorithm by introducing a distance threshold D_t to force the algorithm to cluster only polygons located within a user-defined proximity. A standard Ward's algorithm will continue to cluster polygons until all polygons have been formed into a single cluster (Ward 1963; Kaufman and Rousseeuw 1990; Carvalho, Albuquerque et al. 2009). Then, it is the responsibility of the analyst to apply an *ex post* decision process that chooses the right number of clusters for the particular data and setting to which the algorithm has been applied. Instead of allowing the algorithm to work until it has collapsed all spatial units into a single unit, our tool stops the clustering process when cluster centroids have become too far apart to form a cluster. This process has the advantage of allowing the user to identify clusters at various scales (neighborhood, metropolitan, regional, etc.).

Spatial Dissimilarity Function

While the Ward's process can only deal with one polygonal attribute at a time, this attribute can be a variable determined by an objective function with multiple variables. The tool uses a function based on a process devised by Joshi (2009) and Joshi (2011), which computes a Polygonal Dissimilarity Index (a value for each polygon in the database) using a Polygon Dissimilarity Function (PDF).

The user can select which non-spatial attributes in their data set they wish to include in their analysis, and choose weighting for them. Users can also select the weighting the PDF will apply between the spatial and non-spatial attributes. Equation (1) below describes the PDF distance as a function of spatial and non-spatial attributes. In equation (1), P_i and P_j are arbitrary polygons, and d_{ns} and d_s refer to the "distance" (Euclidean or otherwise computed) between these two polygons. The variables w_{ns} and w_s refer to the weighting given to the non-spatial and spatial attributes of a polygon. The distance between two polygons used by MWards ($D_{PDF}(P_i,P_j)$), is thus given by:

$$D_{PDF}(P_i,P_j) = w_{ns}d_{ns}(P_i,P_j) + w_s d_s(P_i,P_j) \quad (1)$$

where

$$w_{ns} + w_s = 1 \quad (2)$$

We note that our tool allows for multiple non-spatial attributes to contribute to the functions $d_s(P_i,P_j)$ and $d_{ns}(P_i,P_j)$. To compute the distance $d_{ns}(P_i,P_j)$ between non-spatial attributes, we use Euclidean distance calculated by combining the non-spatial attributes using a linear combination, with different weights potentially applied to different attributes. Equation 3 below describes a $d_{ns}(P_i,P_j)$ with multiple non-spatial attributes:

$$d_{ns}(P_i,P_j) = w_{ns1}d_{ns1} + w_{ns2}d_{ns2} + \ldots w_{ns}d_{ns} \quad (3)$$

where $w_{ns1} + w_{ns2} + \ldots w_{ns} = 1$, and where w_{ns1} through w_{ns} are user-defined weights applied to each non-spatial attribute. We note that the tool default settings set $w_{ns1} = w_{ns2} \ldots = w_{ns}$.

All of the non-spatial attributes must be normalized before the computation of the distance, $d_{ns}(P_i,P_j)$. This is required in order to allow attributes with different spatial units to be summed. Column normalization is performed by dividing all the values in the dataset by the largest value in the dataset (Han & Kamber, 2006).

The computation of spatial distances for the PDF is similar to the computation of the non-spatial distances. The computation of spatial distances is again done using Euclidean distances. However, the MWards tool presently only allows for one spatial distance to be considered in the PDF. The current calculation of this distance between intrinsic spatial attributes is computed using polygon centroid-to-centroid distances. We considered using a more complex formula for calculating the

distance but it became computationally difficult slowing the computer output to a point where it became unstable.

Value Chain Tool

The user can select whether to use a value chain mode or not. If selected, the algorithm will treat each non-spatial attribute separately, with a separate weighting applied as defined by the user. If not selected, the algorithm will sum the values of all non-spatial attributes prior to applying the selected weighting. This could be useful for a number of purposes; for instance, helping the user to identify industry sub-clusters, e.g., clusters of motor vehicle electrical parts suppliers within the larger motor vehicles manufacturing industry. The value-chain tool allows the user to establish industrial clusters that reflect real working relationships, rather than relying on clusters composed of single ANZIC codes. Construction of value-chain relationships is described in studies such as (Feser and Bergman 2000; Feser, Koo et al. 2001; Feser and Isserman 2005).

This option was developed because we discovered that some data sets (including the ANSZIC job data) appear to be hierarchical but are not, and we found that different clusters emerge when non-spatial attributes are summed at different points. For instance, the motor vehicle manufacturing industry is associated with ANZSIC 2310, 2311, 2312, 2313, and 2319. In the raw ABS data, ANZSIC 2310, although conceptually the parent of all ANZIC in the form 231X, is not actually the aggregation of the child ANZICs 2311, 2312, 2313, and 2319. Instead it captures some extra firms that failed to classify themselves into one of the four-digit child codes. We wanted the tool to be able to deal with this type of data.

Non-Spatial Attribute Minimum Count

Finally the user interface allows the user to set a minimum attribute count for the non-spatial attributes. This creates an additional test which is then run on the data that potentially reduces the number of middle polygons. This will allow the clustering processing to run faster where theory would suggest that polygons with very small attributes are not relevant. The user is able to specify the minimum of the non-spatial attribute below which the middle polygon will be excluded from the analysis. This means that if for example a polygon contains fewer jobs than this user-specified threshold, it will be deleted from the dataset. The default in the software is one. The number of polygons excluded by this user-specified minimum number depends on the non-spatial attributes selected.

Summary

In summary, MWards follows the following decision structure:

1) Let **C** be a database of N geographical units, defined as polygons. Each polygon *p*, contains information on industry-disaggregated journey to work (JTW) data.
2) Choose a geographic Euclidean distance threshold (D_t) to indicate a maximum tolerance such that two areas falling inside of this tolerance can be considered a cluster, and two areas falling outside of this tolerance cannot be combined into the same cluster.
3) Initiate **E**, an array with length of M, such that $1 \leq M \leq N*(N-1)/2$. **E** contains the following attributes: i) a pairwise listing of all polygon pairs p_i and p_j whose centroid-to-centroid Euclidean distance separation is less than D_t, ii) spatial distance between each p_i and p_j, and iii) attributive, i.e., non-spatial, distance between each p_i and p_j.
4) Row normalize the spatial distance and attributive distances in **E**.
5) Compute the polygon dissimilarity function (PDF) distance (d_{pdf}) for each polygon pair in **E** based on the two normalized distances.
6) Find the polygon pair p_{ij} that gives the smallest d_{pdf} within **E**.
7) Merge polygon p_i and p_j (both geometries and attributes) into a new one p_{new}
8) Remove all polygon pairs containing p_i and p_j from **E.** Also remove all of their attributes from **E**. Leave p_{new} in **E.**
9) Compute the Euclidean distance between p_{new} and the rest (N-2) of the polygons. Append those D_t satisfied p_{new} polygon pairs to **E**.
10) Use the resulting **E** as the array for Step 3. Repeat Steps 3 through 9 until no more polygons can be joined.

Full technical specifications and user inputs are detailed in a technical specification available from AURIN at the University of Melbourne

TOOL PERFORMANCE AND COMPARISON OF OUTPUTS

This section provides some examples of the outputs from the tool using the employment data we have prepared. We provide and compare outputs from various model specifications in order to understand how changing the parameter inputs changes the model output.

We use as our example clusters of motor vehicle manufacturing. After a review of the ANZIC structure we have built a hypothetical value chain of interest involving the ANZIC listed in Table 1. We note that the total number of jobs from the 4-digit categories 2311 through 2319 do not sum to the number of jobs in the 3-digit category 2310. This is because of the way the data are collected by ABS (See section 3.2.3). The difference noted in table 1, can be attributed to the VG property check process, which may exclude some areas that do have jobs in these categories in the Census data but are not found existing in VG property data (see section 2).

TABLE 1. Comparison of Algorithm Outputs with Census Job Counts, 2006

ANZSIC	ANZSIC Description	Number of Jobs, Algorithm	Number of Jobs, Census	Difference
2310	Motor Vehicle and Motor Vehicle Part Manufacturing	11,334.89	11428	93.11
2311	Motor Vehicle Manufacturing	6,856.98	6867	10.02
2312	Motor Vehicle Body and Trailer Manufacturing	1,055.00	1082	27.00
2313	Automotive Electrical Component Manufacturing	363.00	363	0.00
2319	Other Motor Vehicle Parts Manufacturing	2,917.91	2971	53.09
TOTAL		**11,192.89**	**11283**	**90.11**

We use these ANZSIC to examine the performance of the clustering tool under various user specifications. The variations we present here are related to:

1) Cluster scale – the distance threshold at which polygons can no longer be added to a cluster
2) Value chain or non-value chain options
3) Weighting of spatial and non-spatial attributes (must sum to 1)
4) Weighting of non-spatial sub-attributes (must sum to 1)

We reviewed the performance of the tool at various distance thresholds which pertain to different urban scales.

- 1km – walkable urban scale
- 5km – biking and transit scale
- 10km – urban auto scale
- 20km – regional scale

A summary of the major findings is as follows:

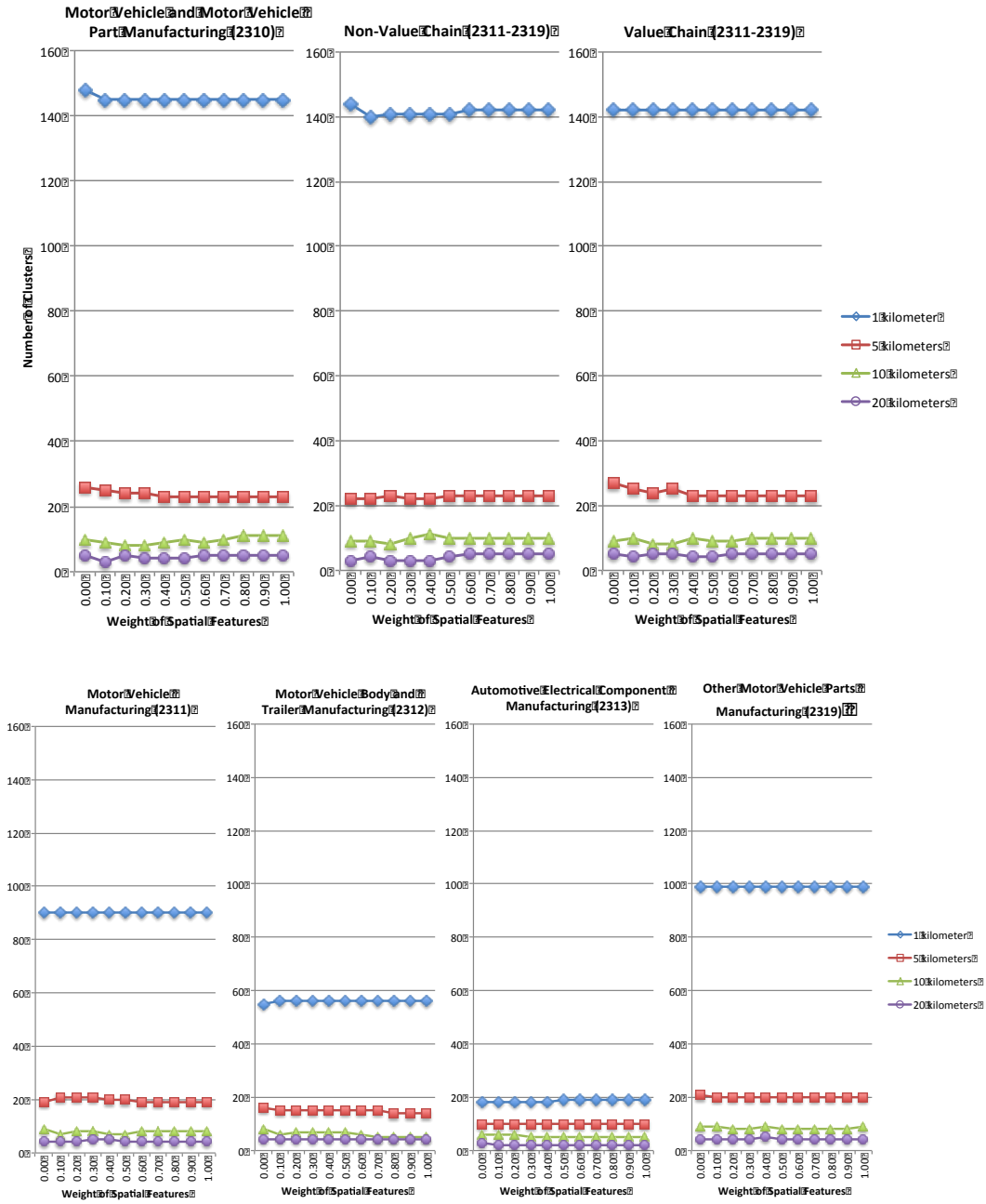
- As expected decreased distance thresholds lead to more smaller clusters being identified
- Cluster numbers are relatively stable across weighting combinations
- despite this stability, cluster configuration (which polygons are included in a cluster) is dependent on the relative weighting of spatial and non-spatial attributes

Cluster Scale

We calculated the number of clusters generated by the tool in the NWMMA under various specifications. Figure 3 shows the number of clusters generated for the 3-digit ANZSIC classification for motor vehicle and motor vehicle parts manufacturing. The value chain specification treats all of the 4-digit sub-classifications of ANZSIC 2310 as individual non-spatial attributes, while the non-value chain specification sums 2310, 2311, 2312, and 2319 to arrive at a single non-spatial attribute that the algorithm then uses. The 2310 values are different because not all jobs classify themselves in a 4-digit category.

These figures demonstrate that the most significant determinant to the number of clusters is the distance threshold. The weight of spatial and non-spatial parameters does not appear to have significantly affected the number of clusters. The number of clusters is clearly different across individual job codes. We found that there was little variation in the number of job clusters formed whether the value chain, or non-value chain mechanism was used.

Figure 3: Number of Clusters under Various Algorithm Specifications, 2006



Primacy of Spatial versus Non-Spatial Weights

This section explores the impacts on cluster configuration of variations in spatial weights and non-spatial weights. Recall that wn_s and w_s must sum to one. In this section, we choose two values from the extremes of the distribution of w_s, w_s=0.9 and w_s=0.1. This allows us to compare the cluster outputs when spatial weights are more and less important.

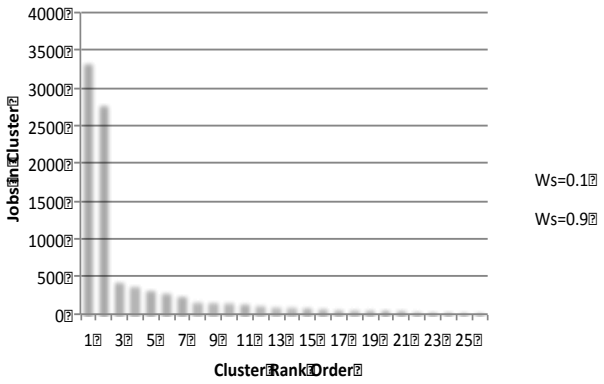
We examined the results in terms of the number of jobs in the polygons, and the spatial configuration to see whether different spatial weighting schemes results in some polygons being allocated to different clusters. Again, we use ANZSIC code 2310, motor vehicle and motor vehicle parts manufacturing.

We found that the distribution of jobs by cluster for the two different spatial weight combinations, at the one-, ten-, and twenty kilometre scales were largely the same (see for example Figure 4a). Interestingly, at the five kilometre distance threshold, the configuration where spatial weights are more important produces a significantly-different jobs distribution than the configuration where spatial weights are less important (Figure 4b).

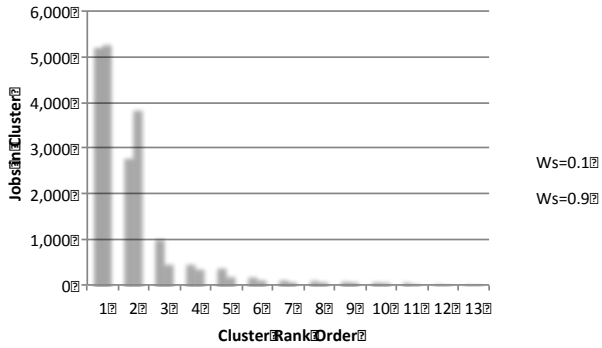
The reason for this behaviour is unclear and requires more testing. However, these findings highlight the importance of spatial and non-spatial weights. Analysts should carefully consider the importance of weighting schemes to their particular problem when using this tool.

Figure 4(a &b). Distribution of Jobs in Clusters by Spatial Weight, ANZSIC Code 2310, Motor Vehicle and Motor Vehicle Parts Manufacturing.

(a) Distance Threshold = 1km



(b) Distance Threshold = 5km



In addition to understanding the distribution of jobs within clusters in aggregate, the specification of spatial and non-spatial weights can also affect whether certain polygons are allocated to one cluster or another. Figure 5a shows the cluster configuration at a distance threshold of 20 kilometres and at w_s=0.1 (spatial weights are relatively less important), for ANZSIC code 2310. For simplicity, we will refer to the western cluster, in blue, and the eastern cluster, in red. As we know the number of jobs remain the stable no matter the configuration of spatial and non-spatial weights.

However, despite the stability of jobs numbers, changing the spatial weights causes some polygons to switch from the eastern to the western clusters. Notably, jobs at the southern end of the eastern cluster (in Yarra) are included in the eastern district when spatial weights are more important (w_s=0.9, Figure 5a). Then, they move to the western cluster when spatial weights are given less importance (w_s=0.1, Figure 5b). This simple example illustrates the importance of careful consideration of the spatial and non-spatial weights in analysis using this tool.

Figure 5(a) Spatial Weights Relatively More Important, Distance Threshold=20km, Ws=0.9

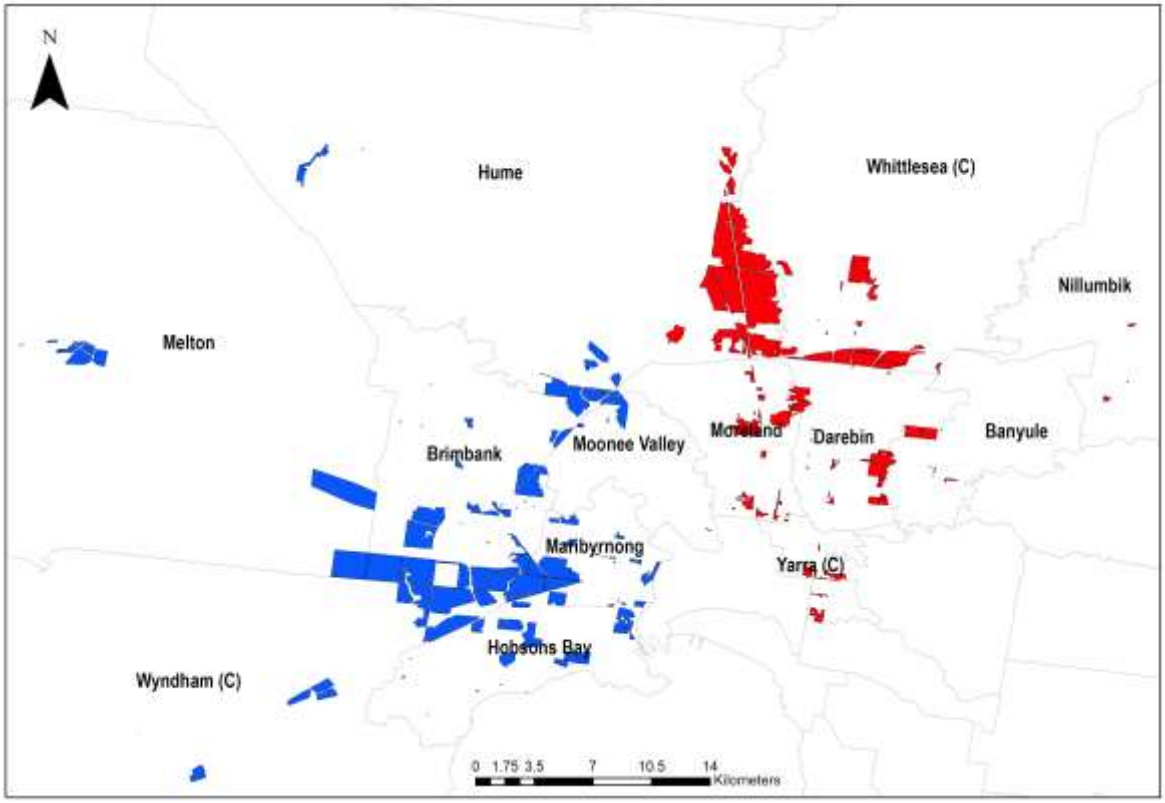
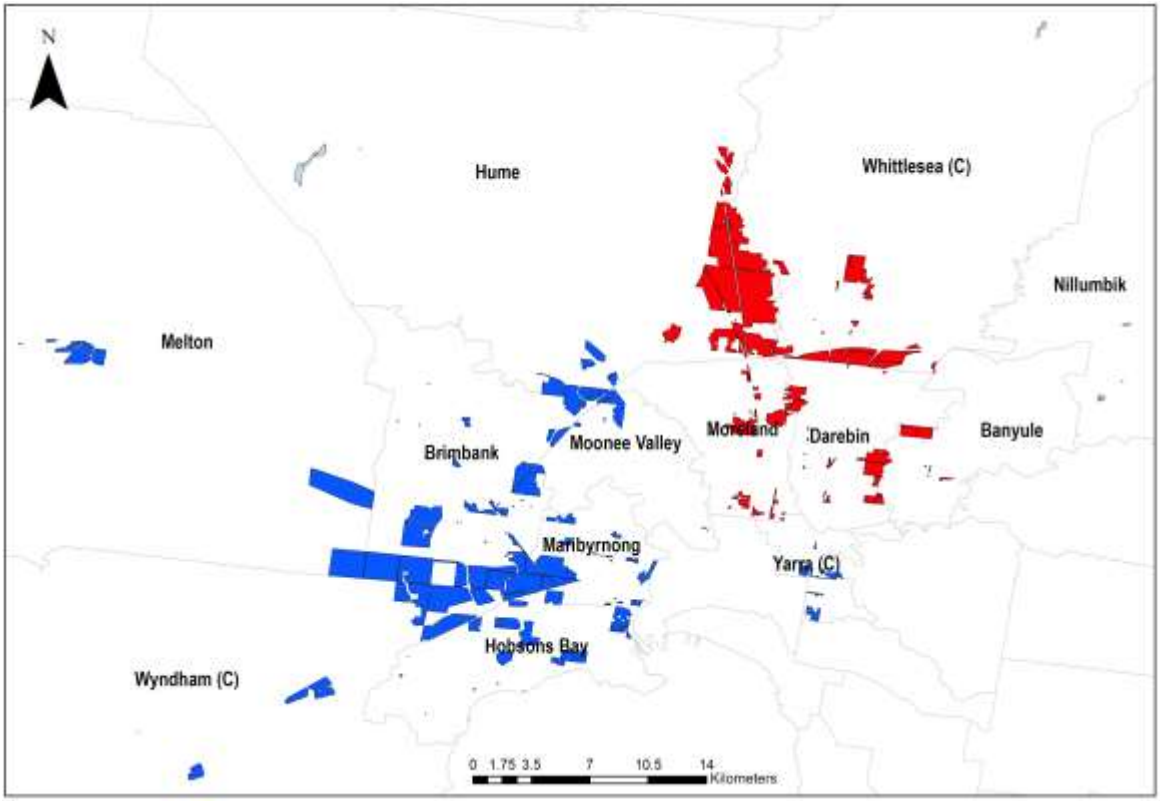


Figure 5(b) Spatial Weights Relatively Less Important, Distance Threshold=20km, Ws=0.1



Value Chain versus Non-Value Chain Specification

Finally, we consider the impacts of the value chain option on cluster configuration. The value chain input allows each element of a value chain to be considered separately as an input in the category of non-spatial characteristics. If a user selects the value chain option at the user interface, each element

of the value chain will be entered into the clustering algorithm as a non-spatial attribute, e.g., jobs in ANZSIC codes 2311, 2312, 2313, and 2319 will all be considered as individual non-spatial attributes of a polygon.

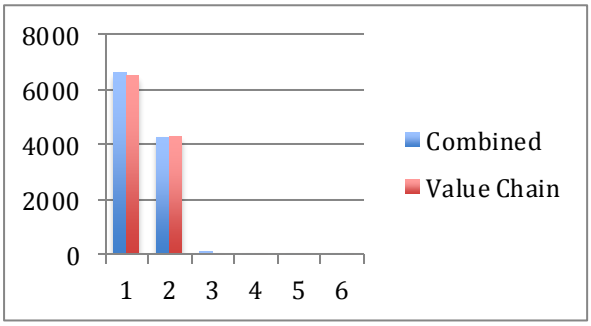
The value chain mode allows the analyst to consider the importance of sub-sectors within a main industry in cluster configuration. Thus, theoretically, the value chain clustering option will consider the job type (in this case, automobile manufacturing versus parts manufacturing) in the construction of the cluster.

As in the previous section, we considered the distribution of jobs in clusters at four distance thresholds (one, five, ten, and twenty kilometres), holding other parameter inputs constant. Small spatial weights of 0.1 were used here to increase the likelihood that non-spatial elements (value chains) would be relevant in cluster formation.

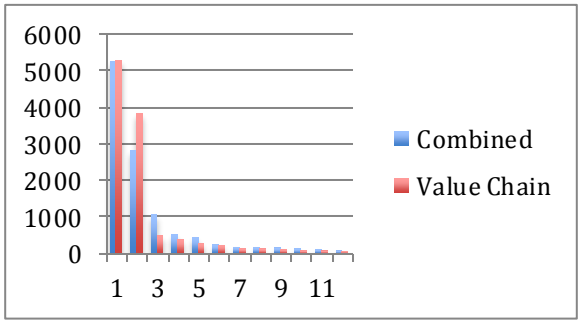
We found that particularly at the one-, ten-, and twenty kilometre distance thresholds, the selection of the value chain option does not drastically change the number of jobs allocated to various clusters (see of example figure 6a). At the five-kilometre distance threshold, however, the value-chain option produces a larger second cluster than the combined option (figure 6b). This is similar to the pattern shown in figure 4 above, for spatial weights, where the effects of changing input parameters are far more pronounced at the five-kilometre scale than at other scales.

Figure 6 Distribution of Jobs in Clusters by Value-Chain Input Type, ANZSIC Code 2310, Motor Vehicle and Motor Vehicle Parts Manufacturing,

(a) Distance = 10km, Spatial Weight = 0.1



(b) Distance = 5km, Spatial Weight = 0.1



Though the number of jobs do not change, there are some notable shifts in individual polygons' membership in clusters between the value chain and non-value chain configurations. Figure 7 shows a map of the clusters resulting when the algorithm is run on disaggregated data from ANZSIC codes 2311, 2312, 2313, and 2319 with a distance threshold of ten kilometres and non-spatial weight importance set at 0.1. Figure 8 shows the same configuration, but using the combined sum of codes 2311, 2312, 2313, and 2319.

Although the general cluster configurations do not change between Figures 7 and 8, there are a number of notable shifts by individual polygons. One very striking example is in the Wyndham cluster. In Figure 8, where the value chain option is not used, the Wyndham cluster is made up of three middle polygons. In Figure 7, another middle polygon joins the Wyndham cluster. The most interesting aspect of the addition of this fourth polygon is where it is located: squarely between polygons that remain in the Hobsons Bay, Maribyrnong, Moonee Valley cluster (which is teal in Figure 7 and red in Figure 8).

This discrepancy between the maps implies that the value chain tool is doing its job: to differentiate between places based on the configuration of their spatial and non-spatial attributes, particularly by considering non-spatial attributes differently in aggregated versus disaggregated form.

Figure 7 Disaggregated Value Chain Components (2311, 2312, 2313, 2319) Entered as Non-Spatial Weights, Distance Threshold=10km, Spatial Weight=0.1

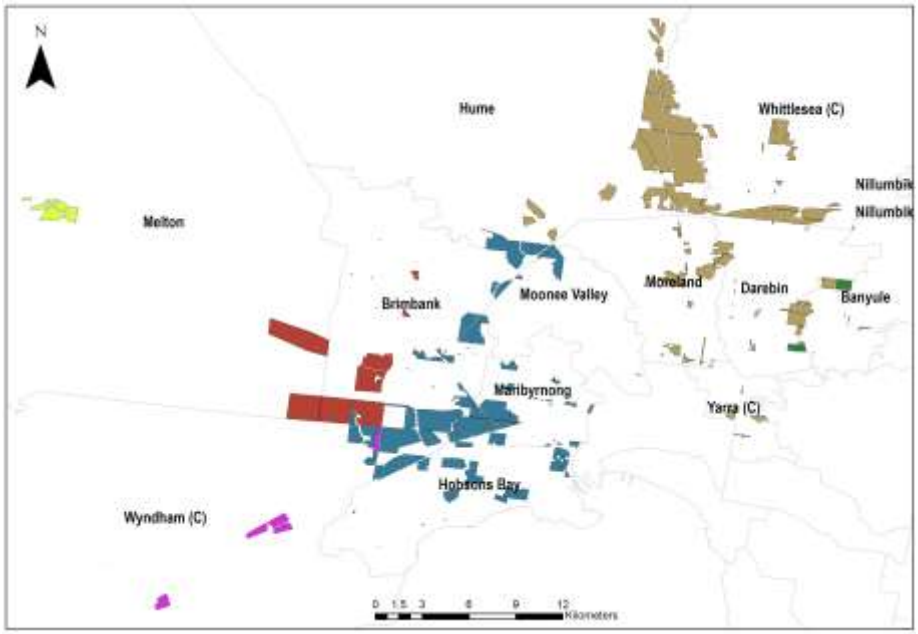
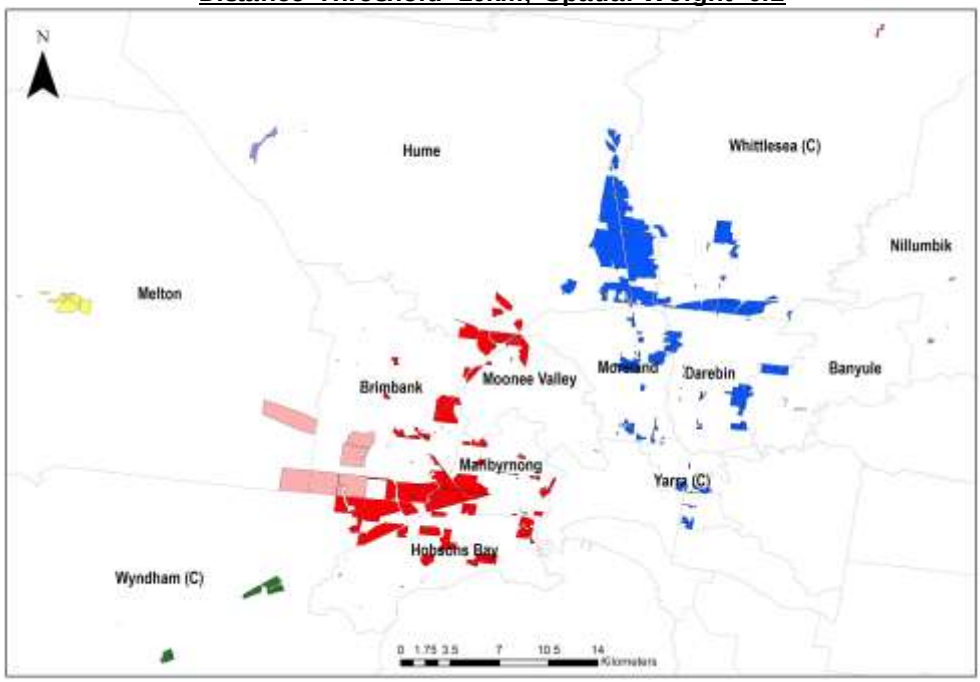


Figure 8 Combined ANZSIC Code 2310, Motor Vehicle and Motor Vehicle Parts Manufacturing, Distance Threshold=10km, Spatial Weight=0.1



5 Conclusions

This paper has described a tool, available on the AURIN Portal, that allows analysts to identify spatial clusters. The tool has various input parameters, and as we have illustrated, these parameters can change the resulting clusters.

Interestingly, the ability of input parameters to change the configuration of clusters seems to be most pronounced at certain spatial scales, namely; when the distance threshold is set to five kilometres. We are not entirely sure why this seems to be the case. We will need to design further research to uncover the reasons for this starting with discovering if the same phenomenon occurs across different years of data and across different ANZSIC code clusters.

We hope that this tool will be useful to analysts interested in economic and spatial analysis in Melbourne, and also to urban planners revising Melbourne's Metropolitan spatial plans. Our next step

is to use this tool to analyse the effects of Melbourne's sub-centre strategies implemented since 1981. In particular, we are interested in understanding whether industry clusters are occurring in the sub-region targeted by these policies, and if so, in which industries.

References

Carvalho, A. X. Y., P. H. M. Albuquerque, et al. (2009). "Spatial Hierarchical Clustering." Revista Brasileira de Biometria **27**(3): 411-442.

Cimiano, P., Hotho, A., and Staab, S. (2004). Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. Proceedings of ECAI'04, IOS Press.

Department of Infrastructure, Victoria (2002) *Melbourne 2030: Planning for Sustainable Growth*, Victorian Government, Melbourne

Feser, E. and A. Isserman (2005). Clusters and rural economies in economic and geographic space, Working Paper.

Feser, E. J. and E. M. Bergman (2000). "National Industry Cluster Templates: A Framework for Applied Regional Cluster Analysis." Regional Studies **34**(1): 1-19.

Feser, E. J., K. Koo, et al. (2001). "Incorporating Spatial Analysis in Applied Industry Cluster Studies." Economic Development Quarterly.

Hann, J., Kamber, M. (2011). Data Mining: Concepts and Techniques, 3rd ed. Morgan Kaufman Publishers.

Jia, T. and B. Jiang (2012). Scaling property of urban systems using an entropy-based hierarchical clustering method. Multidisciplinary Research on Geographical Information in Europe and Beyond. Proceedings of the AGILE'2012 International Conference on Geographic Information Science. J. Gensel, D. Josselin and D. Vandenbroucke. Avignon.

Joshi, D. (2011). Polygonal Spatial Clustering. Department of Computer Science, University of Nebraska - Lincoln. **PhD**.

Joshi, D., A. Samal, et al. (2009). A Dissimilarity Function for Clustering Geospatial Polygons. 17th International Conference on Advances in Geographic Information.

Kaufman, L. and P. J. Rousseeuw (1990). Finding Groups in Data: An Introduction to Cluster Analysis. New York, Wiley.

McNabb, P and Assoc. (2001) *Activity Centres Review: a study of policy and centres of activity in metropolitan Melbourne and Geelong (final report)*, University of Melbourne Research Team in association with Roy Morgan Research and Arup Transportation Planning, available at http://www.dpcd.vic.gov.au/__data/assets/pdf_file/0004/42808/Technical_Report_8_activity_centres_a.pdf, Last accessed 29 April 2013.

Melbourne Metropolitan Board of Works (1954) *Melbourne Metropolitan Planning Scheme 1954 Report,* Victorian Government, available at http://www.dpcd.vic.gov.au/planning/plansandpolicies/planningformelbourne/planninghistory/planning-scheme-1954, last accessed 29 April 2013

Srucca, L. (2005). Clustering multivariate spatial data based on local measures of spatial autocorrelation. An application to the labour market of Umbria. St. Louis, Ideas.

Ward, J. H. J. (1963). "Hierarchical Grouping to Optimize an Objective Function." Journal of the American Statistical Association **58**: 236–244.