After outlier removal

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **mean** | **std** | **min** | **25%** | **Median** | **75%** | **max** | **N** |
| **Distance (m)** | 6634 | 9927 | 50 | 840 | 3147 | 8092 | 99810 | 177938 |
| **Duration (sec)** | 1537 | 1285 | 60 | 616 | 1204 | 2081 | 10799 | 177938 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Wolf *et al.* (2001) | Distance; duration; land-use matching | Georgia, USA 3 days  ﻿March 23 through April 28, 2000 | 13 (156 trips) | 93% |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Probabilistic Methods: Multinomial Logit Model | | | | |
| Oliveria *et al.* (2014) | Duration; Mode; Land use; Personal locations proximity | Georgia, USA 2011 | 1,354 (﻿10,512 trips) | 70% (95-97% for work and school) |

Extra:

Respondents:

B & M: 1104

Oliv 1,354

Xiao 321

Mont 156

Kim 793

Yaz 6845

Zhu 10,372

Geographically weighted regression between weather (temperature and precipitation; 3.2.2) and mode/purpose in given areas. A model will be built for each individual mode and purpose. This form of regression makes use of the queen’s case contiguity matrix to account for the spatial lag. It is hoped that this will give an idea of sensitivity of transport in certain regions to weather i.e. Downtown will likely not be as effected by rain.

### 3.4.3 Spatio-Temporal Investigation of Model Inputs

- directionality throughout day

- space-time calendar (after Arribas-Bel & Tranos, 2017) for given regions of Montreal

- space-time interaction tests in pysal

- space-time evenness grid

- cross-correlation and Coefficient of determination between given regions/DAs of Montreal

“We aim to create conditions for ‘effective’ space-time forecasting” (Yue & Yeh, 2008).

ST-KDE:

See Wei *et al.* (2018) (Also uses grid-ified data)

To-Add:

* [On visualisation of big geodata] We should carefully generalize, e.g., emphasize the important while removing the unimportant, (Li *et al.*, 2016)
* Reason for: Insight into those spatial and temporal trends can improve the performance of Intelligent Transportation Systems (ITS).(Tayyab *et al.*, 2014)

Sentences:

The casual links have been harder to prove in the literature

Owing to the…

To Add:

* This has affected probabilistic models over machine learning models (Oliveria et al., 2015)

Arguably, there is a need for more encompassing approach to the modelling inputs has allowed more generic model creation that can applied to a range of different purposes (as we do not know which purposes rely on which i.e. POI may be important for tourist purpose but not for work purposes).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Gong *et al.* (2018) | POI; Socio-demographics; Temporal Features; Trip Duration; Weather | ﻿Hakodate City, Japan 2012–2013 | (﻿9981 trips) | n/a |

Spatial-Temporal:

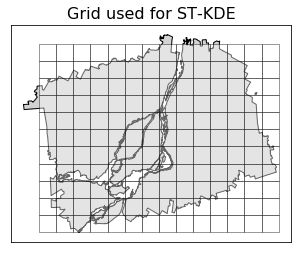
ST-K

Space-time Ripley’s K and spatial-temporal randomness -> end to start

[One STKDE Ripley’s K]

Space-time KDE end to start

ST-KDE



**Figure 4.X**

[Some sort of plot for KDE]

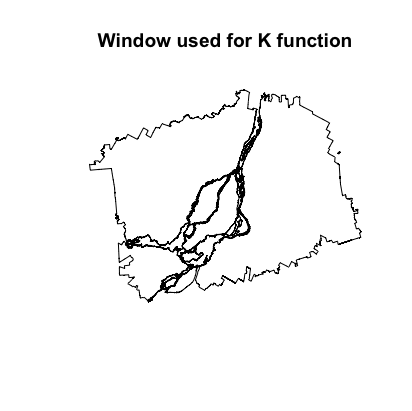
Join the ST-KDE back to the data -> if there a trip at a particular time and space (a grid region) can be flagged as in the given cluster (0,1)

11 columns of STKDE cluster

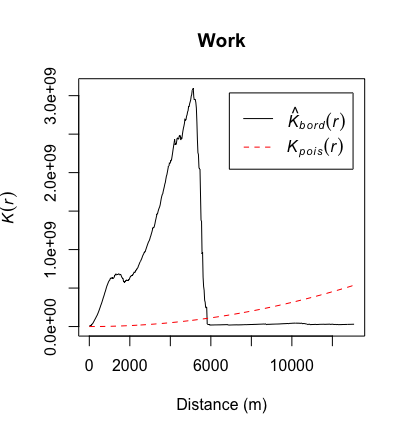
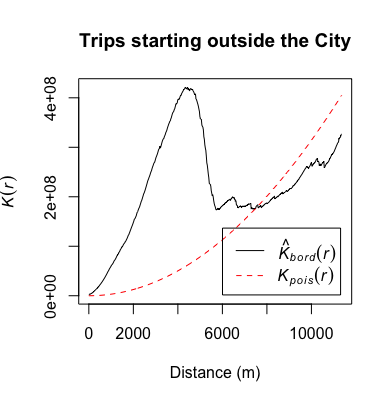
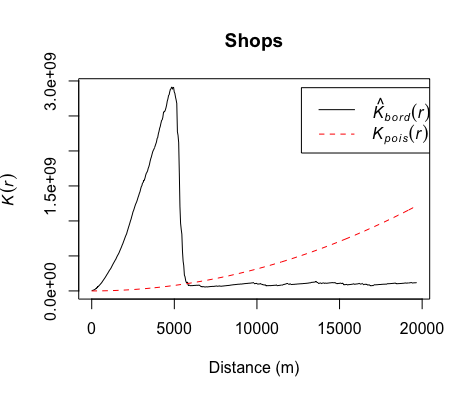
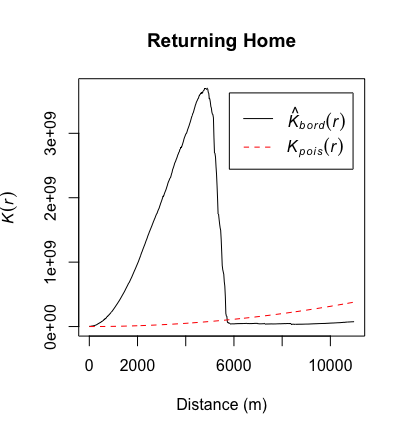
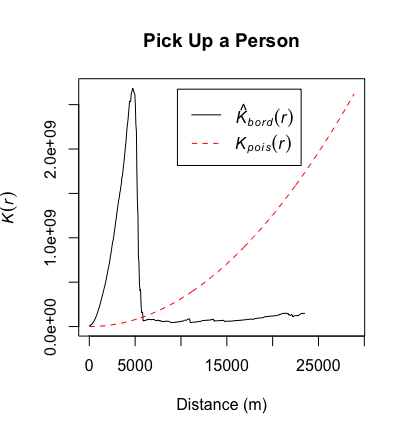
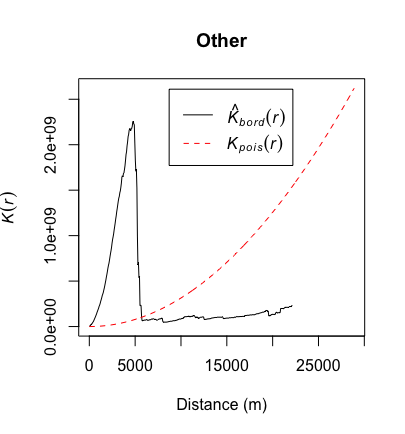
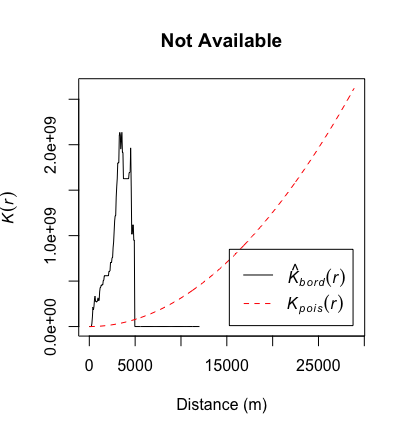
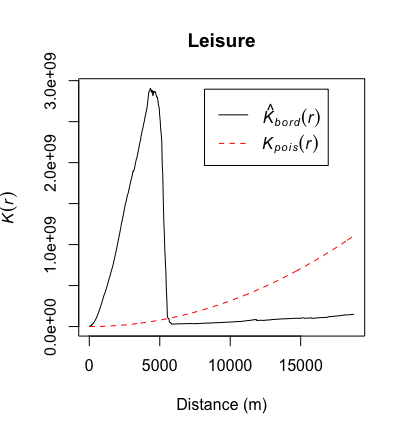
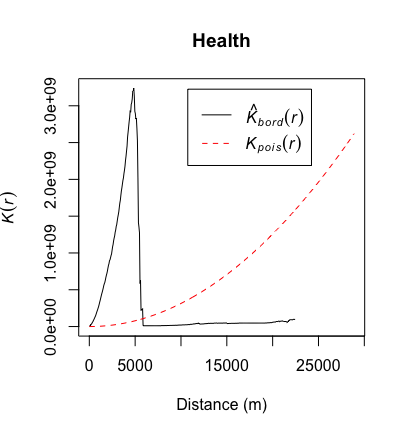
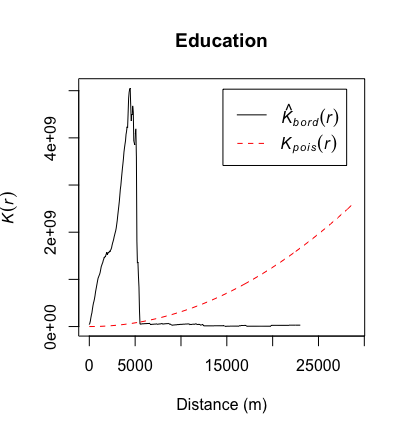
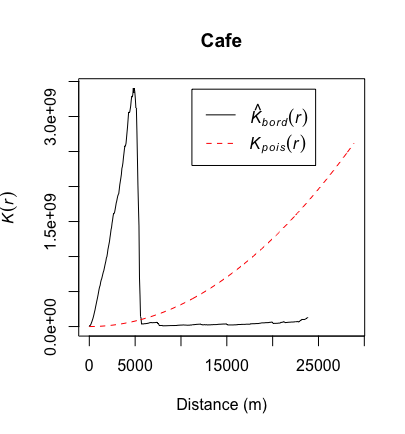
For a given trip, dissemination areas within the intersection are given a value of 1 (red) and all other areas are given a value of 0 (grey). For the spatial analysis in this report, data has been grouped using *geopandas groupby* function into individual modes and purposes of travel (see **Table 3.1** in section 3.2.1 for types) and then visualisations have been produced from each showing the total intersection count in each dissemination area across the entire study period (see X.X).

**Figure 3.7** Example of the spatial join between a route and the underlying dissemination areas (route in **blue**; overlapping dissemination areas in **red**)

Ripley’s K on start and end -> spatial randomness



**Figure 4.X** Window used for Ripley’s K function



“Using this window, a points pattern object is created from the Treasure Hunt location coordinates and estimations for Ripley’s K is calculated. This estimation of K is plotted in Figure 5 against a Poisson distribution (which indicates Complete Spatial Randomness of Treasure Hunt locations). As shown, the trend in estimated K is far higher than the Poisson distribtion implying there is clustering and spatial dependence of Treasure Hunt locations across London.”

We assess spatial dependency of trips from each type of trip purpose using Ripley’s K-function calculated using the R’s *spatstat* library. Specifically, the Ripley’s K can be used as a measure of spatial distribution of start and end points of the trips indicating whether there is Complete Spatial Randomness within the data (Dixon, 2002). These

# Appendices

## Appendix 1 Notification not to apply for Ethical Approval

## Appendix 2 Python Script used for retrieval of Climatological data:

Code is available online at https://github.com/Thomasjkeel/MSc\_Dissertation under ‘. (This program consists of 500 lines written in Python computer language).  
*Description:* contains the code to create the figures and carry out the statistical analysis of this IGS.

# Appendix 3 Mean Direction and Distance Calculations

is calculated for each pair of points in each trip, the mean direction is calculated from the collection of these points using:

Calculations for mean direction in degrees:

*V\_east = magnitudes \* np.mean(np.sin(from\_dir \* np.pi/180))*

*V\_north = magnitudes \* np.mean(np.cos(from\_dir \* np.pi/180))*

*then..*

*mean\_dir = np.arctan2(V\_east, V\_north) \* 180/np.pi*

*mean\_dir = (360 + mean\_dir) % 360*

*mean\_dir = np.mean(mean\_dir)*

*Calculations for mean distance (magnitude) in degrees:*

*C = (1. / len(from\_dir)) \* (np.sum(np.cos(from\_dir \* np.pi/180)))*

*S = (1. / len(from\_dir)) \* (np.sum(np.sin(from\_dir \* np.pi/180)))*

*then…*

*resultant\_magnitude = (C\*\*2 + S\*\*2)\*(1./2.)*