Heavysnow Transformation

Guebin Choi and Hee-Seok Oh
Department of Statistics
Seoul National University
Seoul 08826, Korea

Draft: version of December 5, 2019

Abstract

This paper presents a new multiscale transforms, termed 'heavy-snow transform', which is motivated by observing snow accumulation. Heavy-snow transform provides multiscale visualization which can capture the shape of land (or data structure) with different resolution. Some measures based on heavy-snow transform are newly defined to describe dissimilarity between data and the importance of data. Furthermore, some statistical applications are studied.

Keywords: Graph signal; Similarity; Distance; Multiscale method.

1 Introduction

In this study we will propose a new method which can measure the distance between nodes in *graph signal*. In here, graph signal f is real valued function such that $f : \mathcal{V} \to \mathbb{R}$ where $\mathcal{V}$ is set of nodes (or vertices). We are interested in the distance or similarity(which is usually defined as the inverse of distance) in graph signal.

Let ν_i and ν_j be a specific nodes in $\mathcal{V}$ and $f(\nu_1)$ and $f(\nu_2)$ be a value at ν_i and ν_j. How to measure the similarity or disimilarity between $f(\nu_i)$ and $f(\nu_j)$? In other words, how can we define distance between $f(\nu_i)$ and $f(\nu_j)$?. The naive approach for measuring the distance between $f(\nu_i)$ and $f(\nu_j)$ is to use the Euclidean distance like: $\|f(\nu_i) - f(\nu_j)\|_2 := \sqrt{(f(\nu_i) - f(\nu_j))^2}$. Of course, someone can use other measures such as Gaussian kernel weighting function. Those measures do not have a problem when they measure the distance of points which resides Euclidean space. However, it is reasonable to think that the graph signal resides in not only Euclidean space but also graph space. (Note that $f(\nu) \in \mathbb{R}$ but $\nu \in \mathcal{V}$.) The problem is here. Distance between $f(\nu_i)$ and $f(\nu_j)$ depends on disimilarity in Euclidean domains but distance between ν_i and ν_j depends on disimilarity in graph domains, which is determined by *links* (or *edges*), i.e., $\mathcal{E}$. However the methods presented above have limitations in that they do not consider the $\mathcal{E}$ when measuring the distance between observations. In here, considering *links* between observation means that consider the structure of indices, i.e., $\mathcal{E}$. Let's move to Figure 1 to understand why you should consider the structure of index set in the graph signal.

Figure 1 is made from example introduced in Shuman (2012). There are three graph signal in Figure 1. The interesting fact is that those three graph have exactly same $f(\nu_i)$ for each $\nu_i \in \mathcal{V}$. The only difference between them is $\mathcal{E}$ which are represented in dotted lines. Due to $\mathcal{E}$ all connected nodes in (a) seems to have a similar values but (c) is not. In other words, (a) looks like a low-frequency signal, while (c) looks like a high-frequency signal. (b) feels halfway between (a) and (c). You can easily check this is true by comparing the spectral density of (a)-(c) represented in second row of Figure 1.

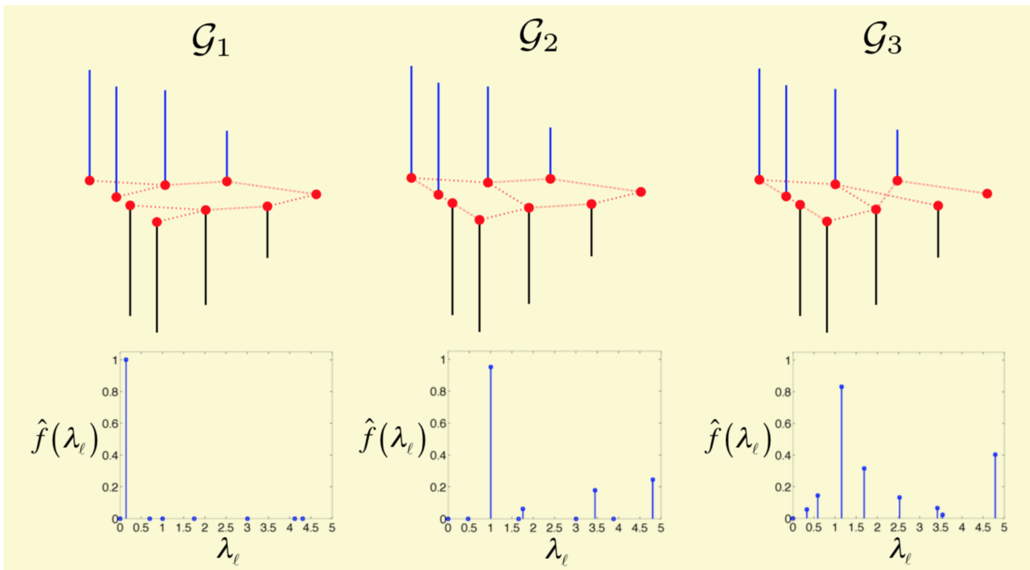

Figure 1: Example 1 in Shuman (2012)

The underlying structure of graph signal $\mathcal{E}$ affects distance between ν_i and ν_j, and it affects the similarity (or disimilarity) between $f(\nu_i)$ and $f(\nu_j)$ as discussed in Figure 1. Therefore, when analyzing graph signal, measuring the distance between ν_i and ν_j is no less important than measuring the distance between $f(\nu_i)$ and $f(\nu_j)$. In other words, the distance in the graph domain is just as important as the distance in the Euclidean domain.

Various methods have been developed to measure the differences between nodes. The most common method is the shortest path, which measures the shortest distance between two nodes. Klein and Randić (1993) suggest the resistance distance which can measure the distance between nodes in undirected graph. Lafon and Lee (2006) suggest the diffusion distance. However, those methods have limitations such that they are only interested in distance between ν_i and ν_j. That is, those methods could be suitable for graph $(\mathcal{V}, \mathcal{E})$ but not suitable for graph signal $f : \mathcal{V} \to \mathbb{R}$.

Thus, our goal is that: Develope new distance considering the distance in vertex domain and the distance in the Euclidean domain simultaneously. To do this, we propose a new distance that takes into account the distance between the vertex domain and the distance from the Euclidean domain at the same time through an elegant technique called *heavysnow transform*. We will formally define heavysnow transform in Section 2, instead, in this section, we will just breifly introduce the main idea and motivation of heavy transform. After that we will explain why our proposed distance is a reasonable measure which can properly mix the information of the graph and Euclidean domains.

The heavysnow transform is invented by observing snow accumulation over the land or ground. In this study, you can consider the ground as graph signal f and snow as some positive constant b we will stack on f. Let τ be the number of falling snow. Our propose distance, named *snow distance*, is defined

$$sdist(\nu_i, \nu_j; \tau) := \mathbb{E}\|h(\nu_i; \tau) - h(\nu_j; \tau)\|_2.$$

In here, $h(\nu_i; \tau)$ will be defined more strictly at Section 2. In this section, you can consider $h(\nu_i; \tau)$ as the updated value of $f(\nu_i)$ after τ. In other words you consider $h(\nu_i; \tau)$ as snowy ground. If we set $\tau = 2$, then $h(\nu_i; \tau)$ will be one of $f(\nu_i)$, $f(\nu_i) + b$ and $f(\nu_i) + 2b$.

As you can see, when $\tau = 0$, the snowdist between two nodes ν_i and ν_j becomes $\|f(\nu_i) - f(\nu_j)\|_2$, which is exactly same as Euclidean distance. However, as τ increases, the snow distance starts to mix the distance in the Euclidean domain and the distance in the graph domain. That is, as τ increases, the structure of $\mathcal{E}$ is additionally reflected in dissimilarity between $f(\nu_i)$ and $f(\nu_j)$. This is the key to considering both graph and Euclidean domains simultaneously.

How is this possible? This is due to the very unique feature of snow accumulation. For your easy and intuitive understanding, let's see Figure 2. In Figure 2-(a), (b), and (c), the curved lines which are located in the bottom of each figure represents ground or graph signal f. If you put a less viscous material like rain on f, it would look like (a). If we add a highly viscous material to f, it would look like (c). If material has medium viscosity like snow, it would look like (b). In other words, to have a shape like (b), the snow stacked on v_i with following characteristics:

(i) Snow can flow to adjacent areas.

(ii) Snow cannot flow to higher ground.

Note that (i) needs the information of $\mathcal{E}$, i.e., it needs information of graph domain since (i) is determinded by whether or not v_i and v_j are connected. On the other hand, (ii) relates to the values of $f(v_i)$ and $f(v_j)$ which is information of Euclidean domain. Therefore, the process of snow accumulation can be reproduced only with the information of both domains. As a result, the distance between snowy ground $h(\nu_i, \tau)$ and $h(\nu_j, \tau)$ is surprisingly mixed with information from both domains. (This is covered more closely in Section 2.)

If the viscosity is too high, the shape of the snow accumulations will always like (c), regardless of the structure of the index set $\mathcal{E}$. This is the result of considering only the values defined in the Euclidean domain and ignoring the graph domain information. Conversely, (a) tends to ignore information in the Euclidean domain too much. In (b), the information from the Euclidean domain and the vertex domain are well balanced.

The remaining of this paper is organized as follows. Section 2 defines heavy-snow transformation and proposes some statistics based on heavysnow transformation. Section 3 introduces some visualization techniques that can effectively show the results of heavysnow transform. Section 4 presents

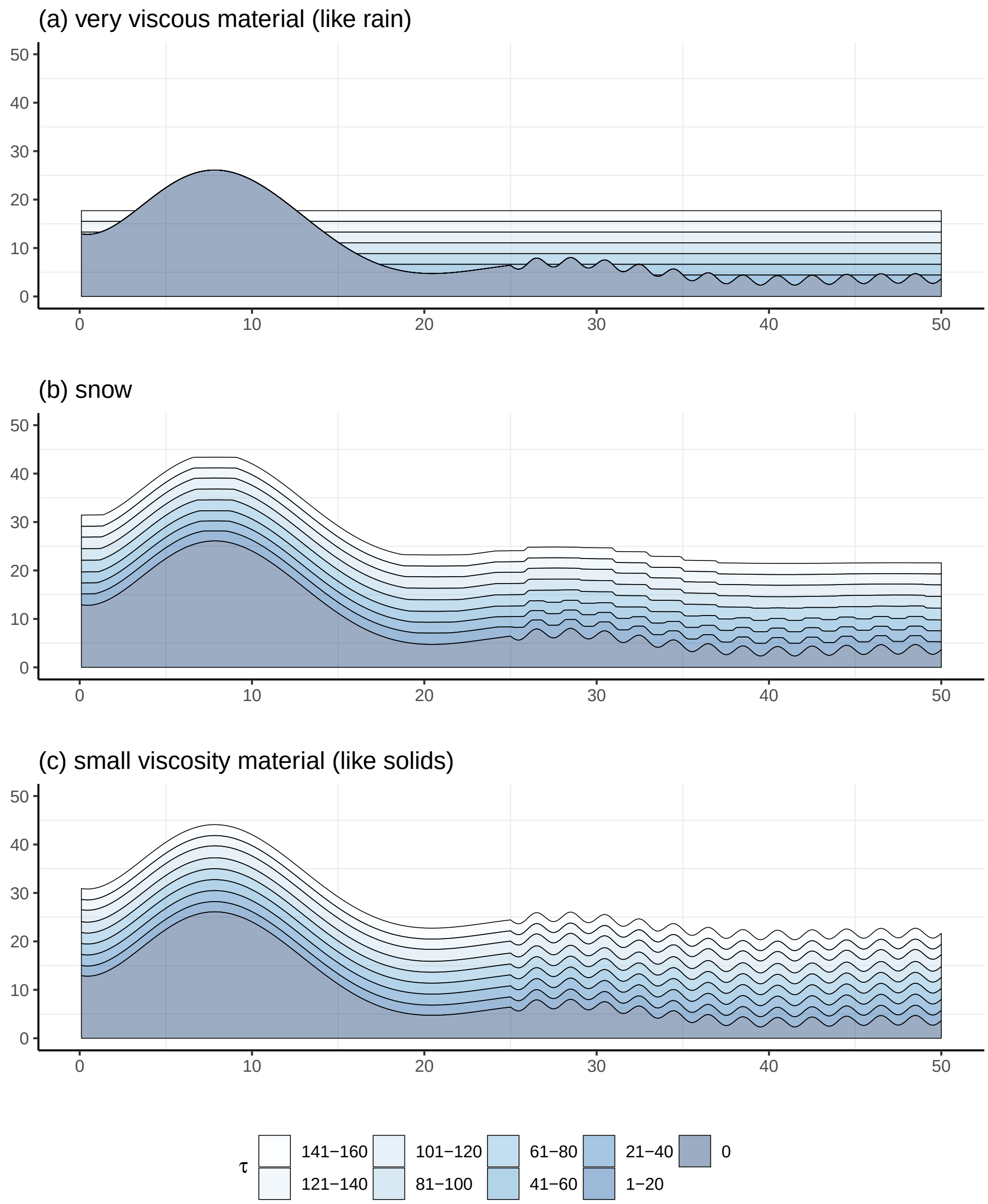

Figure 2: The underlying signal and accumulations with different viscosity. In here, τ represent that amount of stacked accumulations.

possible applications of heavysnow with various numerical experiments and real data analysis. Finally, concluding remarks are given in Section 5 where a possible use for smoothing is briefly discussed as a future research topic.

2 Heavy-Snow Transform

2.1 Definition of Heavy-Snow Transform

Before defining heavy-snow transform, we would like to explain data structure to which the transformation is applicable. Let $\mathcal{V}$ be set of nodes (or vertices) and $\mathcal{E}$ be set of links (or edges). In the context of heavy-snow transformation, the link means a way that snow can move in. Basically, you can consider $\mathcal{V} = \{\nu_1, \ldots, \nu_n\}$ as given index set such as $\mathcal{V} = \{1, 2, \ldots, n\}$. However, it is often reasonable to view node as realization of some random variable X. This can be illustrated by setting node as $v = X(\nu)$.

Let $X : \mathcal{V} \to \mathcal{M}$ be a random variable where $\mathcal{M}$ is isometrically embedded manifold in $\mathbb{R}^N$. Supposed that we observed $v_1, v_2, \ldots, v_n \ \sim \ i.i.d. \ F_X$ where F_X is cumulative distribution functions of X. The graph signal f is real valued function such that

$$f : \mathcal{M} \to \mathbb{R}.$$

Furthermore, we define a set of linked vertices with a particular vertex v_i as $\mathcal{N}_{v_i} = \{v_j : (\nu_i, \nu_j) \in \mathcal{E}\}$ where $\nu_i : X^{-1}(v_i)$. Note that $\forall M \in \mathcal{M} :\ V^{-1}(M) \in \sigma(X)$ where $\sigma(X)$ is smallest σ-field which makes X as random variable. Thus $X^{-1}(v_i)$ is always well defined.

From now on, we explain how to implement the phenomenon of snow accumulation. Let's come back to Figure 2 for easy understanding. Figure 2 (a), (b) and (c) show different viscous materials stacked on the same ground. For the case of Figure 2 (a), material flows to local minimum, which might be similar to the accumulation of rainwater. On the other hand, the material in Figure 2 (c) does not flow and just stacks up above ground. The most important property of the material in (a) is ‘continue to flow until blocked’. Note that small viscous materials do not accumulate. Those small viscous materials such as water accumulate only when they reach the local minimum. On the other hand, the most important property of highly viscous materials such as (c) is ‘stacking.’ They only accumulate and never flow.

The snow can be interpreted as having a medium viscosity between two materials of Figure 2 (a) and (c), as metioned in Section 1. In order to express the viscosity of the snow, we assume that snow has both feature of (a) and (c). In other words, snow continues to accumulate AND flow until blocked. For your easy understanding let’s assume some specific situation as follows:

1. (snow falls) Draw v_i from V with probability $\frac{1}{n}$. Suppose that snow falls at v_i as much as b.

2. (accumulation) Like (c) snow stacks on v_i. Thus, in this case, value of v_i, i.e., $f(v_i)$ will be updated by $f(v_i) + b$.

3. (flows/blocked) After that snow can move to one of $\mathcal{N}_{v_i}$ which is neighborhood of v_i. Of course, if $\mathcal{N}_{v_i} = \emptyset$, the snow will no longer flow. However, $\mathcal{N}_{v_i} \neq \emptyset$ does not necessarily mean that snow can flow (since snow cannot flow high). Thus, we should compare value of $f(v_i) + b$ and $f(v_j)$ to check flowbility of snow. In otherwords, we should check $\mathcal{U}_{v_i} := \mathcal{N}_{v_i} \cap \{v_j : f(v_j) \leq f(v_i) + b\}$ is emptyset or not.

    - (flow) In the case of $\mathcal{U}_{v_i} \neq \emptyset$, which is an unblocked situation, snow (which stays in v_i) can flow to v_j that is randomly sampled element from $\mathcal{U}_{v_i}$ with probability $\frac{w_{ij}}{\sum_{j \in \mathcal{U}_{v_i}} w_{ij}}$.
    - (blocked) If $\mathcal{U}_{v_i} = \emptyset$, which is a blocked situation, snow can’t flow anymore. In this cases, the next node is randomly selected from $V = \{v_1, \ldots, v_n\}$ with probability $\frac{1}{n}$.

4. In the cases of 'flows', go back to step 1 and repeat above procedure with assuming that snow falls at v_j (which is element of $\mathcal{U}_{v_i}$) as much as γb. In here $\gamma \in (0, 1)$ is parameter adjusting viscosity of snow. (If you choose very small γ, shape of snow accumulation will be (c) in Figure 2.) For 'blocked' situation, go back to step 1 with assuming that snow falls at v_j (which is randomly selected node in V) as much as b.

The following is the formal definition of heavy-snow transform.

Definition 1. Let $V := \{v_i : i = 1, 2, \ldots, n\}$ be the given index set or realization of X. For fixed i, let $\mathcal{N}_{v_i}$ be set of linked values with v_i and $f(v_i)$ be function valued mapped to v_i and $\boldsymbol{f}$ be $\boldsymbol{f} = \{f(v_i) : i = 1, \ldots, n\}$. Let b be the amount of falling snow and τ be the number of falling snows. Let $\{v^{(0)}, v^{(1)}, \ldots, v^{(\tau)}\}$ as *trace of snow* where $v^{(0)}$ is result of random selection from V with probability $\frac{1}{n}$ and $v^{(\tau)}$ is defined

$$v^{(\tau)} = \begin{cases} \text{sample from } \mathcal{U}_{v^{(\tau-1)}} \text{ with probability } \frac{w(v^{(\tau-1)}, v^{(\tau)})}{\sum_{j \in \mathcal{N}_{v^{(\tau-1)}}} w(v^{(\tau-1)}, v^{(\tau)})} & \mathcal{U}_{v_{(\tau-1)}} \neq \emptyset \\ \text{sample from } V \text{ with probability } \frac{1}{n} & \mathcal{U}_{v_{(\tau-1)}} = \emptyset \end{cases}$$

For any $\tau \in \{1, \ldots, n\}$, the τ-th *snowyground* of $\boldsymbol{f}$ is defined by

$$\boldsymbol{h}^{(\tau)} := \boldsymbol{h}^{(\tau-1)} + \boldsymbol{\xi}^{(\tau)}$$

where $\boldsymbol{h}^{(0)} = \boldsymbol{f}$ and $\boldsymbol{\xi}^{(\tau)} = \{\xi^{(\tau)}(v_i) : i = 1, \ldots, n\}$ is amount of stacked snow at τ whose elements are defined by

$$\xi^{(\tau)}(v_i) = \begin{cases} \gamma \xi^{(\tau)}(v_{(\tau-1)}) & \mathcal{U}_{v_{(\tau-1)}} \neq \emptyset \\ b & \mathcal{U}_{v_{(\tau-1)}} = \emptyset. \end{cases}$$

Define $\mathcal{H}(\boldsymbol{f}; \tau) := \{\boldsymbol{h}^{(0)}, \ldots, \boldsymbol{h}^{(\tau)}\}$ as *heavy-snow transform* of $\boldsymbol{f}$.

Let $\mathbf{S}$ be

$$\mathbf{S} := \begin{bmatrix} h^{(0)}(v_1) & h^{(1)}(v_1) & \dots & h^{(\tau)}(v_1) \\ h^{(0)}(v_2) & h^{(1)}(v_2) & \dots & h^{(\tau)}(v_1) \\ \dots & \dots & \dots & \dots \\ h^{(0)}(v_n) & h^{(1)}(v_n) & \dots & h^{(\tau)}(v_n) \end{bmatrix}$$

and call it *snow matrix* of $\boldsymbol{f}$.

2.2 Snowdistance and some properties

Definition 2. Let $\boldsymbol{f}$ be the graph signal on (V, E). Let $\mathcal{H}(\boldsymbol{f};\tau)$ be heavy-snow transform for $\boldsymbol{f}$. The *snow distance* between v_i and v_j is defined by

$$\rho(v_i, v_j) := \|\boldsymbol{h}_i - \boldsymbol{h}_j\|_2.$$

where $\boldsymbol{h}_i = \big(h^{(0)}(v_i), \dots, h^{(\tau)}(v_i)\big)$.

Thm 2.1. Let $(\mathcal{S}, \rho)$ be the metric space. Let $(\mathcal{V}, \psi)$.

Thm 2.2. continous

2.2.1 regular graph

Remark 2.1. Suppose $f(v_1), \dots, f(v_n)$ is $i.i.d$ random sample and (V, E, W) be the regular graph. Let $\mathcal{H}(\boldsymbol{f};\tau)$ be a heavysnow transform of $\boldsymbol{f}$. Since $f(v_i)$ is $i.i.d.$ random variable $\boldsymbol{h}_i$ can be thought as $i.i.d.$ random vector (need to prove, using 'regular graph') such that

$$\boldsymbol{h}_1, \dots, \boldsymbol{h}_n \sim F.$$

Define functional $T(F)$

$$T(F) := \int g(\boldsymbol{h}) dF.$$

Let F_n empirical distribution function of $\boldsymbol{h}_1, \dots, \boldsymbol{h}_n$, i.e.,

$$F_n(\boldsymbol{h} \le t) := \frac{1}{n} \sum_{i=1}^{n} 1(\boldsymbol{h}_i \le t).$$

From Glivenko-Cantelli lemma we have $F_n \to F$. Thus for sufficiently large n, we can say that F_n is neighborhood of F. Thus

$$\begin{aligned}T(F_n) &\approx T(F) + \int IF(\boldsymbol{h}, T, F) d(F_n - F)\\ &= T(F) + \int IF(\boldsymbol{h}, T, F) dF_n\end{aligned}$$

where $IF(\boldsymbol{h}; T, F, \tau) = \lim_{t \downarrow 0} \frac{T\left((1-t)F(\boldsymbol{h}) + t\delta(\boldsymbol{h})\right) - T(F(\boldsymbol{h}))}{t}$ is influence function of T. In here $\int IF(\boldsymbol{h}, T, F) dF = 0$ is used. Thus

$$\sqrt{n}\Big(T(F_n) - T(F)\Big) \to N(0, V(T, F))$$

where $V(T, F) = \int IF(\boldsymbol{h}, T, F)^2 dF$.

Remark 2.2. Let $d_{ij} = \|\boldsymbol{h}_i - \boldsymbol{h}_j\|_2$. Using above remark, we get $d_{ij} \sim N(T(F), V(T, F)/n)$ where $T(F) := \int d_{ij} dF$. Note that $T(F) \to 0$ as $\tau \to \infty$. Further, we can conclude that $V(T, F)/n \to 0$ as $\tau \to \infty$. Thus $\mathbf{D} \to \mathbf{0}$.

Remark 2.3. Define snesitivity curve as

$$SC_n(\boldsymbol{h}; \tau) = \frac{T\Big((1 - 1/n)F_{n-1}(\boldsymbol{h}) + 1/n\ \delta(\boldsymbol{h})\Big) - T(F_{n-1}(\boldsymbol{h}))}{1/n}.$$

This is sample version of $IF(\boldsymbol{h}, T, F)$.

3 Visulization

4 Numerical Experiments

4.1 Simulated Examples

In this section, we present some examples of heavy-snow transforms with one-dimensional data. For all examples, we set $\mathcal{N}_i = \{X_{i-1}, X_i, X_{i+1}\}$, $b = 0.02 \times \sqrt{V(\{X_i\})}$ and $p_{ij} = \frac{1}{3}$.

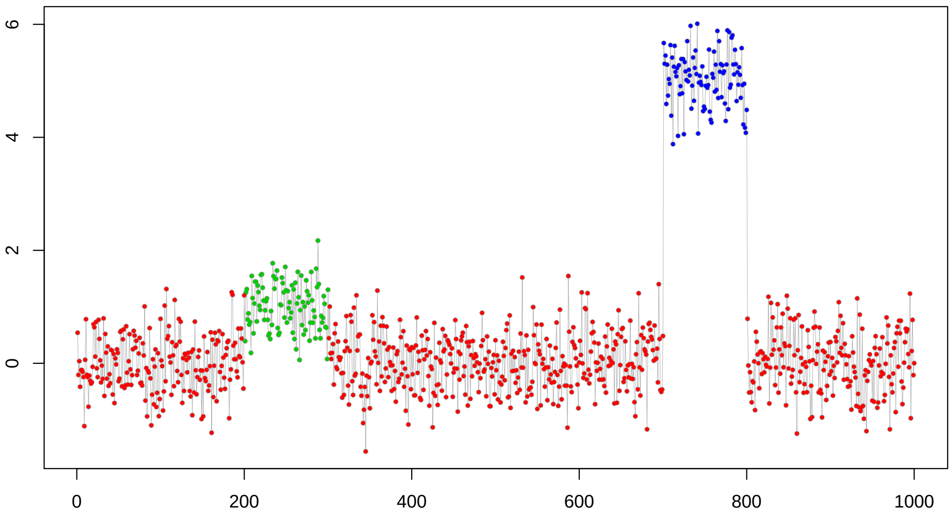

Figure 3: X_i in Example 4.1.

Example 4.1: Consider following Gaussian process $\{X_i\}$ generated by $N(\mu_i, 0.5)$, where

$$\mu_i = \begin{cases} 0, & 1 \le i \le 200 \\ 1, & 201 \le i \le 300 \\ 0, & 301 \le i \le 700 \\ 5, & 701 \le i \le 800 \\ 0, & 801 \le i \le 1000. \end{cases}$$

Figure 3 shows a realization of X_i, where observations with $\mu = 1$ are marked as green and observations with $\mu = 5$ are marked as blue.

The first row in Figure **??** is the result of the heavy-snow transform with $\mathcal{T} = 10,\ 200$ and 600, where the x-axis represents the index of the data, the y-axis represents τ, and the color represents the value of $Y_{i,\tau}$. As one can see, $\{Y_{i,\tau}\}_{i=1}^{1000}$ with small τ represents $\{X_i\}$ with high-resolution and $\{Y_{i,\tau}\}_{i=1}^{1000}$ with large τ represents $\{X_i\}$ with low-resolution. Thus, this map is clearly a multiscale representation.

The second row in Figure **??** shows principal component plot of $\{Y_{i,\tau}\}_{\tau=0}^{\mathcal{T}}$ with different $\mathcal{T}$. Note that blue and red/green ones are well separated in x-axis. Thus, it can be interpreted that the first principal component is thought of as an overall average of data. The interpretation of the second principal component could be more difficult than first one. It is related to the structure of linked data, that is, the land-shape. To understand the meaning of second principal component, suppose that snow falls as much as $\tau = 1$. Then value of $E(Y_{i,1} - X_i)$ is perfectly determined by considering structure between X_{i-1}, X_i and X_{i+1}. So, the values of $E(Y_{i,1} - X_i)$ are divided by the following types of land-shapes:

(i) local minimum: $X_i = \min(X_{i-1}, X_i, X_{i+1})$

(ii) local maximum: $X_i = \max(X_{i-1}, X_i, X_{i+1})$

(iii) flat: $X_{i-1} = X_i = X_{i+1}$

(iv) uphill: $X_{i-1} < X_i < X_{i+1}$

(v) downhill: $X_{i-1} > X_i > X_{i+1}$

(vi) flat-uphill: $X_{i-1} = X_i < X_{i+1}$ or $X_{i-1} > X_i = X_{i+1}$

(vii) flat-downhill: $X_{i-1} = X_i > X_{i+1}$ or $X_{i-1} < X_i = X_{i+1}$

Note that for all i, $P(X_{i-1} = X_i) = 0$ in this example. Thus, we could not consider (iii), (vi) and (vii). Note also that values of $E(Y_{i,1} - X_i)$ are the same in the cases (iv) and (v). Thus, the values of $E(Y_{i,1} - X_i)$, amount of snow accumulation, can be roughly divided into three groups according to the land-shapes and these are represented in three straight lines in the left panel of the second row.

The second row in Figure **??** also contains a multiscale concept. In $\mathcal{T} = 10$, data can be roughly divided into two groups, blue one and red/green ones. And each group can also be divided into three groups according to land-shape. If one increases the scale up to $\mathcal{T} = 200$, in other words, if you more widely consider the linked data, then groups of blue, green, and red dots are clearly distinguished, compared to $\mathcal{T} = 10$. Instead, three straight lines disappear, so we no longer know whether each observation is local maximum or minimum. In other words, we lose some local-scale information. Now, let's move the principal component plot with $\mathcal{T} = 600$, which shows the most global-scale analysis. We easily check that there are some points at which the value of the second principal component is very high. These points are located in the neighborhood of $i = 700$ or $i = 800$, which are the points where sudden mean changes occur. Note that detecting these points need more wide consideration about linked data. So these points are not separated from the main group in the local-scale analysis.

For a fixed δ, an importance of specific point X_k can be calculated by counting how many points are in $\{X_j :, d_{\mathcal{T}}(X_k, X_j) > \delta\}$. In this example, we choose δ as the median of $\{d_{\mathcal{T}}(X_i, X_j) : i, j \in \{1, \dots, 1000\}\}$. Results for calculation importance of data are shown in the third row. Importance is marked by various colors and sizes. Bigger ones are more important than small ones and purple ones are more important than red ones. The interesting thing is that importance of data changes over scales. In the local-scale analysis, observations which are located in $i \in (700, 800)$ are considered to be important. However, in the global-scale analysis, observations where sudden mean change occurs (such as $i = 700$ or $i = 800$) are considered to be more important.

Example 4.2: Consider a Gaussian process $\{X_i\}$ generated by $N(\mu_i, 0.5)$, where

$$\mu_i = \begin{cases} 0, & 1 \le i \le 512 \\ 5, & 513 \le i \le 1024. \end{cases}$$

Figure 4(a) shows a realization of X_i, where red, green, blue, light blue and purple colors represent $\{X_i\}_{i=1}^{472}$, $\{X_i\}_{i=473}^{502}$, $\{X_i\}_{i=503}^{522}$, $\{X_i\}_{i=523}^{552}$ and $\{X_i\}_{i=553}^{1024}$, respectively. As one can see, the data have a single change point in $i = 513$ and blue ones contain this sudden mean changes. Green and light blue ones are the neighborhood of the change point. Figure 4(b) shows principle component plot of $\{Y_{i,\tau}\}_{\tau=0}^{\mathcal{T}}$, where $\mathcal{T} = 120$. The red ones and purple ones are separated from each other. Thus, data with $\mu_i = 0$ and $\mu_i = 5$ are well separated in $\mathcal{T} = 120$. We would like to emphasize that the green, blue and light blue ones are also separated from their main groups because they are located around the change point. So which points are important in Figure 4(b)? One can easily check that red and purple area are denser than the areas where green, light blue and purple are located. So the importance of $\{X_i\}$ will be high in the neighborhood of the change point. Figure 4(c) shows $(i, imp(X_i, 120))$, the importance plot. Here, we choose δ as the median of $\{d_{\mathcal{T}=120}(X_i, X_j) : i, j \in \{1, \ldots, 1000\}\}$ as in Example 4.1. As expected, $imp(X_i, 120)$ has a peak in $i = 512$, where the sudden mean change occurs. Figure 4(d) shows another importance plot as in the third row of Figure 4.

Furthermore, the importance of X_i can be applicable to a smoothing problem that estimate μ_i from X_i. The following is the proposed smoothing procedure.

1. Calculate the importance of X_i, i.e., get the $imp(X_i, \mathcal{T})$.

2. For $b = 1, \ldots, B$, get $X_i^{(b)}$ such that

$$X_i^{(b)} = \begin{cases} X_i, & i \in M \\ \tilde{X}_i, & i \in M^c, \end{cases}$$

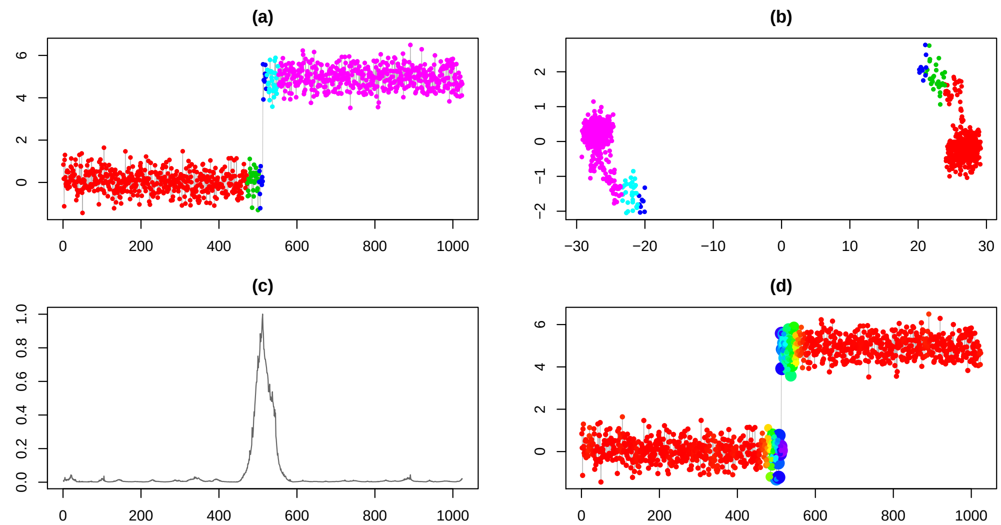

Figure 4: (a) X_i; (b) Principal component plot of $\{Y_{i,\tau}\}_{\tau=0}^{120}$; (c) Plot of $(i, imp(X_i, 120))$; (d) Plot of (i, X_i) where $imp(X_i, 120)$ are marked color and size.

where M is length-$N_0(< N)$ samples without replacement from a population of $\{1, 2, \ldots, N\}$ with weight

$$\mathbf{w} = (imp(X_i, \mathcal{T}), \ldots, imp(X_N, \mathcal{T}))$$

and $\tilde{X}_i$ is linearly interpolated value from $\{X_i\}_{i \in M}$.

3. Get $\hat{\mu}_i = \frac{1}{B}\sum_{b=1}^{B} X_i^{(b)}$

The key of the above procedure is that we do not use random sampling. Instead, we consider the importance of data as sampling weight. Thus, the important data is more likely to be sampled than other.

Figure 5 shows the smoothing results. Left-top panel shows the original data. The red line in the left-bottom denotes a wavelet fit by EbayesThresh (Johnstone and Silverman 2005). The green line in right-bottom represents a wavelet fit by SURE threshold (Donoho and Johnstone 1994). The result

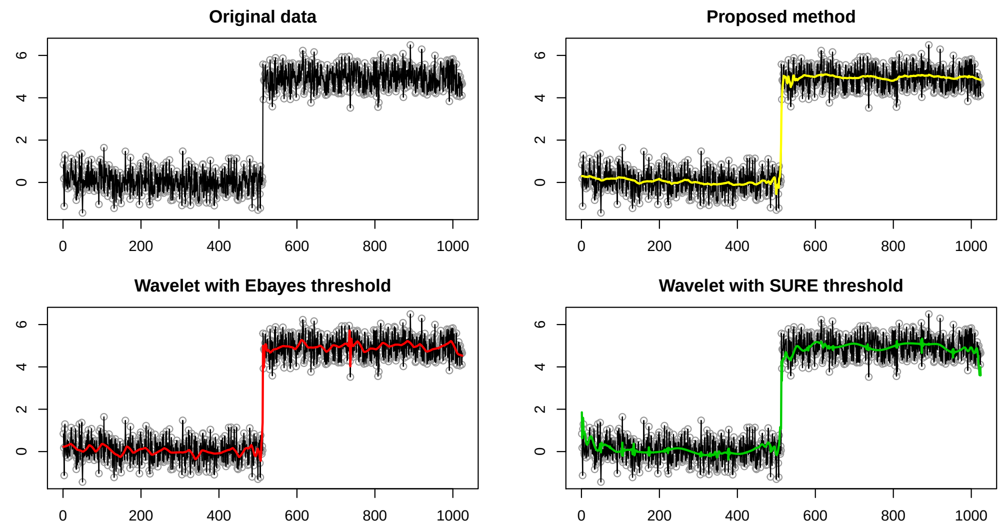

Figure 5: Smoothing results

by our proposed method is shown as the yellow line in the right-top panel. In this example, we choose $\mathcal{T}$ as 120 and $N_0 = 50$. As one can see, the performance of the proposed method is comparable to the wavelet fits and we want to emphasize that our method is not limited to the data structure.

4.2 Real Data Analysis

4.2.1 Avenger's

Data dercription

What movie is important?

Leskovec (2009) : 연구그룹들을 네트워크화해서 중심연구그룹(?)을 찾았음.

Decomposition

4.2.2 Les Misérables

Data dercription

Who is important in Les Misérables?

4.2.3 Earthquake

Data dercription

Smoothing

4.2.4 Distribution of Species of Animals

Data dercription

Smoothing

Where do the habitats overlap?

4.2.5 Seoul Metro data

Data dercription

In this example, we analyze subway passenger data of Seoul Metro Line 2 on March 7, 2014, which is represented in Figure 6. In this figure, x-axis denotes station, y-axis represents time, and color shows the number of people to get on Line 2. As one can see, each cell looks highly associated with vertical ones and horizontal ones; thus the data are spatially and temporally linked data. Another interesting point is that Line 2 is a circular line, so the next station of City Hall is Chungjeongno. Therefore, the data are arranged (linked) in a cylindrical shape.

So the exact data structure looks like Figure 7. Each point connected with temporally, and spatially. So this is a spatio-temporal data. If we fixed time then this is the circular data, and if we fixed the station then this data is time series.

Oneday analysis

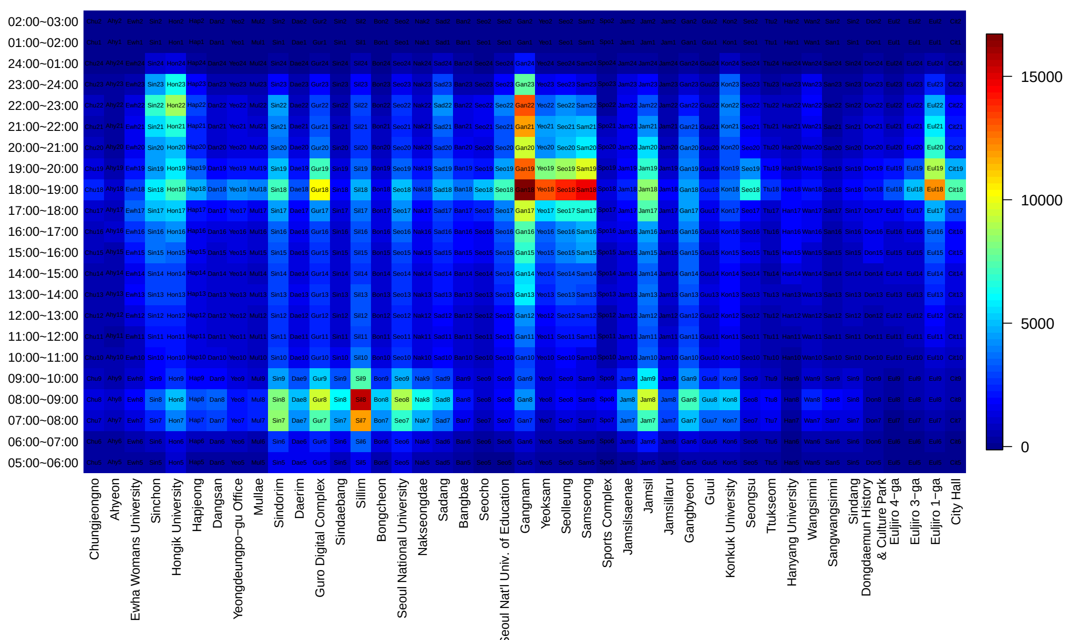

Figure 6: Subway passenger data of Seoul Metro Line 2 in March 7, 2014.

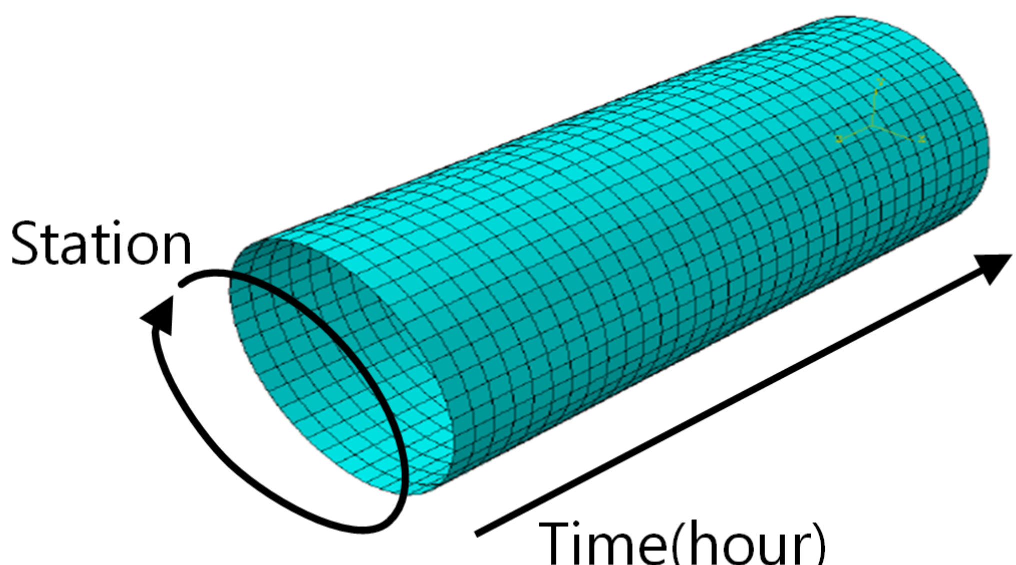

Figure 7: Data structure of subway passenger data with Seoul metro line 2 in March 7, 2014.

Figures 8 to 13 show the importance plot of subway passenger data with $\mathcal{T} = 10, 100, 200, 300, 400, 500$. In local scale analysis, the most important cell is Gangnam station at 6 pm. It looks like a very reasonable result because Gangnam station is considered the hottest place in Seoul at that time. In addition to Gangnam station, some stations or group of stations are also considered to be important. In the morning of rush hour time, the residence area such as Sindorim, Guro Digital Complex, Sillim, Seoul National University and Jamsil stations are considered to be important. In the evening of rush hour time, the office area such as Guro Digital Complex, Gangnam, Yeoksam, Seolleung, Samsung and Euljiro 1-ga are considered to be important as well.

When growing scale up to $\mathcal{T} = 300$, the importance of Euljiro 1-ga station surpass the Gangnam station, and in the case of growing scale more, the importance of Sillim station surpasses the Gangnam and Euljiro 1-ga stations. Why the importance of each cell changes over scale ? Look at the Figure 6 again. We easily check that vertical array from Gangnam station at 18:00 to Gangnam station at 24:00 have high values, and we also check that horizontal array from Gangnam station at 18:00 to Samsung station at 18:00 have high values, too. Thus, we expect that the value of Gangnam station at 18:00 will be high since it is cross point of those two arrays. But, the situation is something different in Sillim station at 7:00–9:00. The values of these two cells cannot be expected by any linked data because these are extremely high than other linked cells. In summary, the value of Gangnam station at 18:00 can be expected by linked data, but those of Sillim station at 7:00–9:00 cannot be expected by linked data. So if we examine data with global scale, that means if we more widely consider the linked data, then the value of Gangnam station at 18:00 is more predictable than Sillim stations at 7:00-9:00. Thus the value of Gangnam station is less important than Sillim station.

Type of area: commercial, residential, and office area.

When is the peak time for each region?

What is the most important day of the week in each region??

4.2.6 Messenger data

Data dercription.

Who is influential?

What is important Month?

Change of relationship.

Recommendation.

5 Concluding Remarks

Heavy-snow transform is a new multiscale visualization technique which is motivated by observing how snowfall accumulates. The concept of heavy-snow transform is shared with that of scale-space theory in computer vision. Heavy-snow transform is useful for evaluating some probabilistic structures of linked data whether a particular set of observations is similar to nearby other or not. We define the dissimilarity of the data and define the importance of the data based on it. We also introduced useful applications such as change-point detection and smoothing.

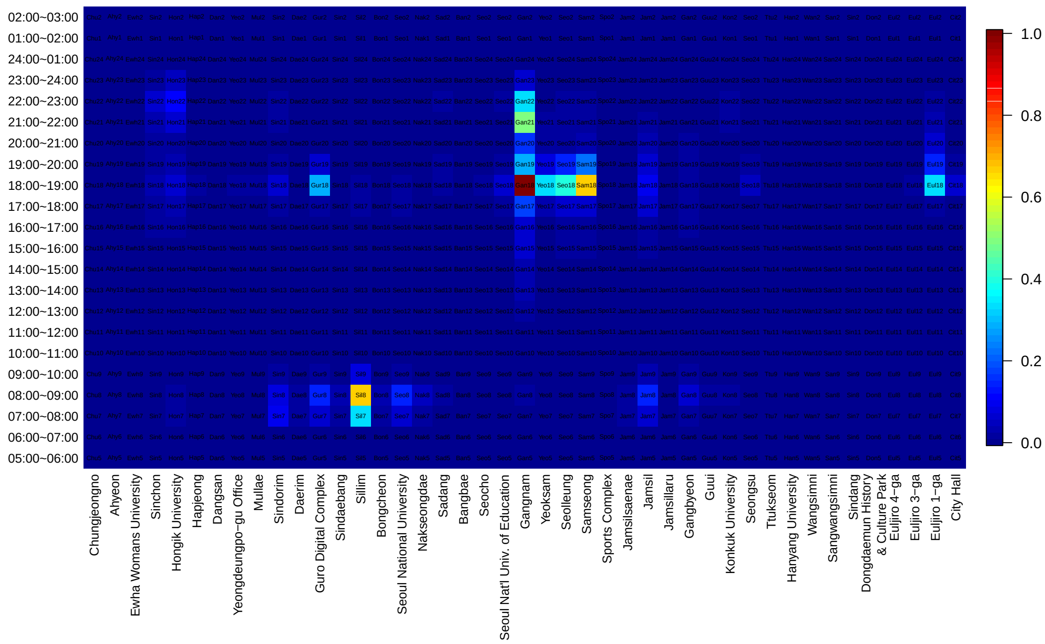

Figure 8: Importance plot of subway passenger data with $\mathcal{T} = 10$.

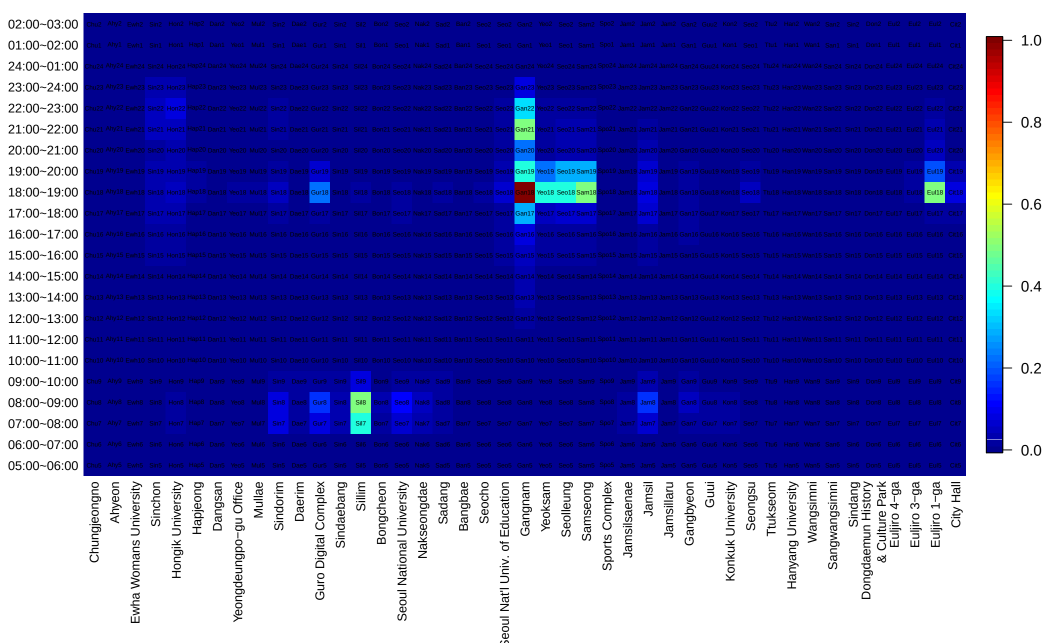

Figure 9: Importance plot of subway passenger data with $\mathcal{T} = 100$.

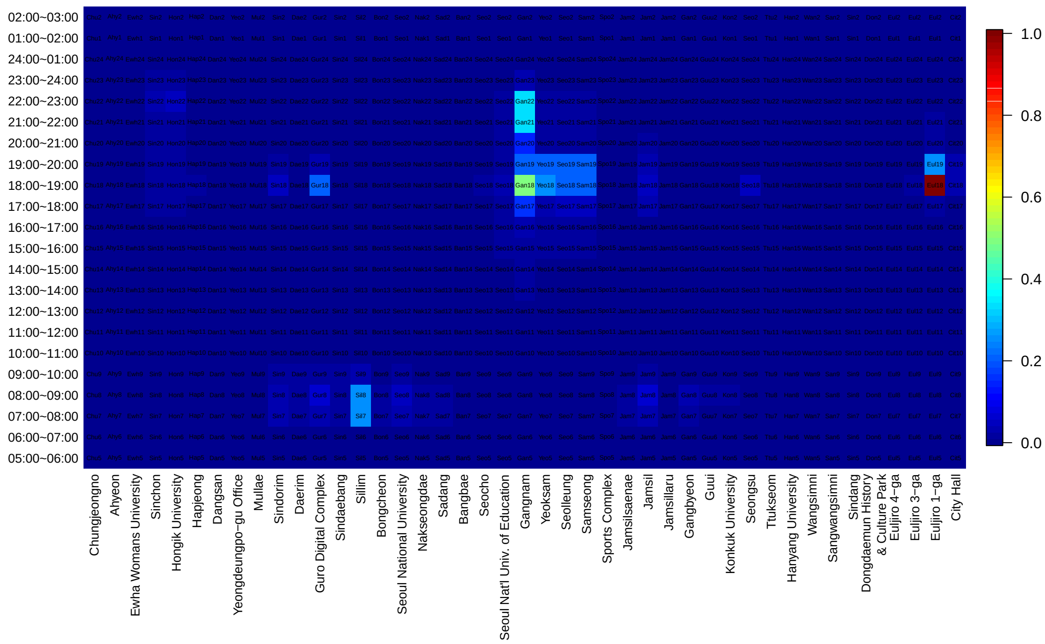

Figure 10: Importance plot of subway passenger data with $\mathcal{T} = 200$.

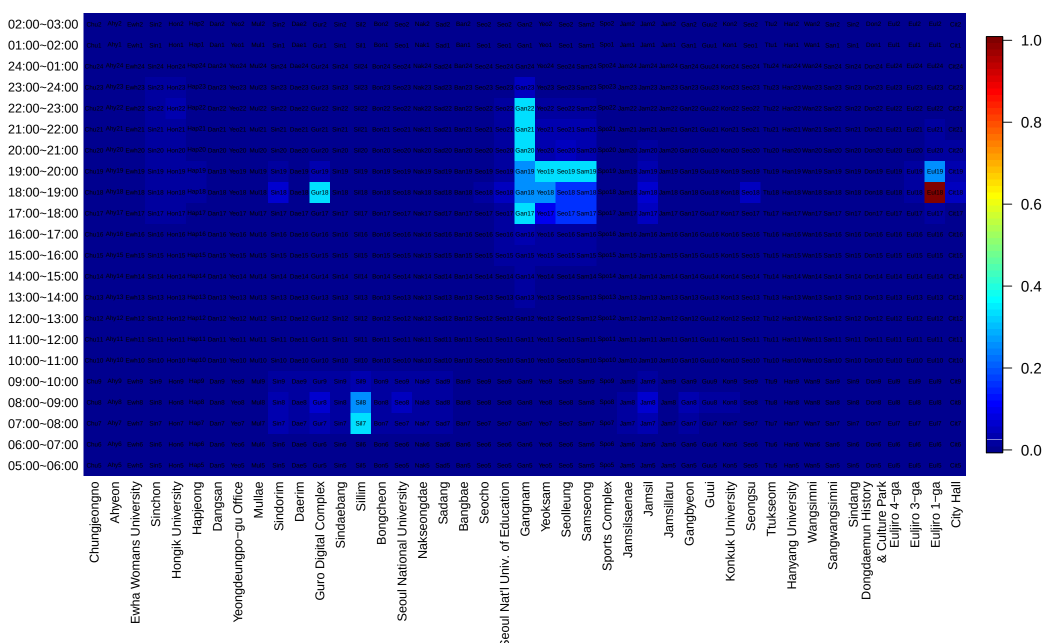

Figure 11: Importance plot of subway passenger data with $\mathcal{T} = 300$.

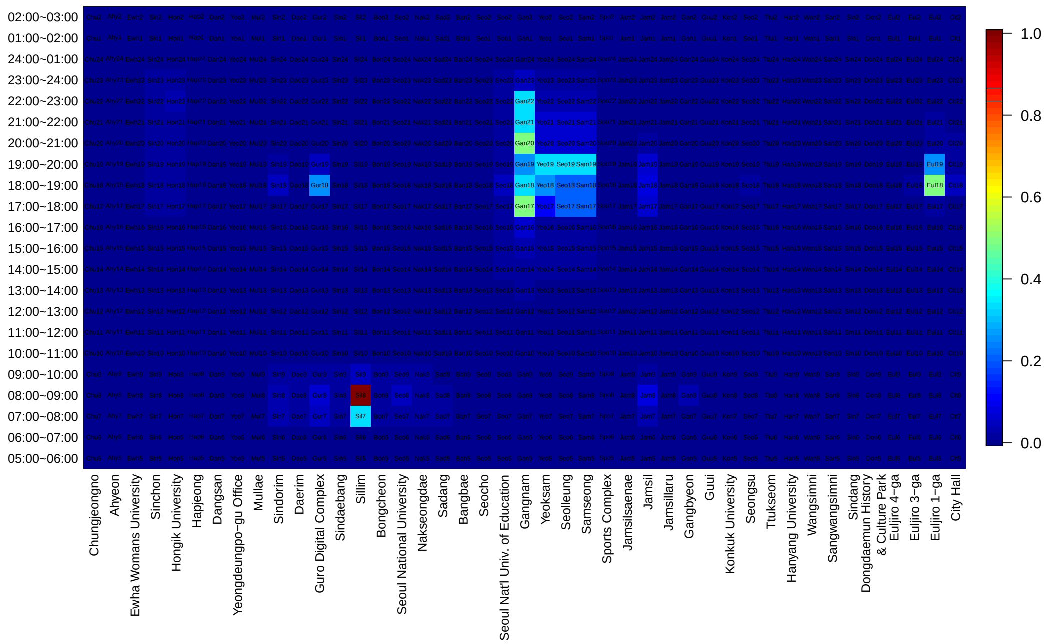

Figure 12: Importance plot of subway passenger data with $\mathcal{T} = 400$.

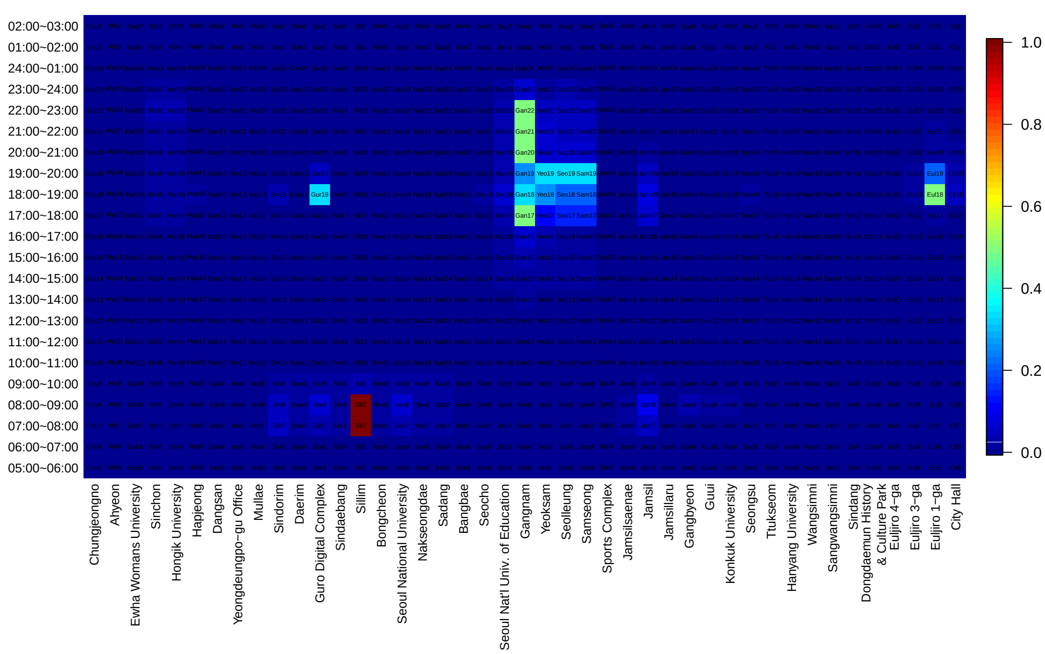

Figure 13: Importance plot of subway passenger data with $\mathcal{T} = 500$.

Appendix

Figures

Appendix

Smoothing

- 여기에서는 헴펠교재 P.316 의 내용을 참고하였다. 선형모델에서 모수 $\boldsymbol{\beta}$ 의 추정량 $\boldsymbol{\beta}_n$ 을 구하는 것은 아래를 푸는 것이고

$$\min_{\boldsymbol{\beta}_n} \sum_{i=1}^{n} \rho(y_i - \mathbf{x}_i \boldsymbol{\beta}_n) \tag{1}$$

이것은 다시 아래의 방정식의 해를 푸는것과 같다.

$$\sum_{i=1}^{n} \mathbf{x}_i^T \psi\big(y_i - \mathbf{x}_i \boldsymbol{\beta}_n\big) = 0 \tag{2}$$

이런 추정량 $\boldsymbol{\beta}_n$을 M-estimator라고 한다. 일반적인 회귀분석 모형에서는 $\psi(y) = y$ 가 되어 위의 식은

$$\sum_{i=1}^{n} \mathbf{x}_i^T \big(y_i - \mathbf{x}_i \boldsymbol{\beta}_n\big) = \sum_{i=1}^{n} \mathbf{x}_i^T y_i - \sum_{i=1}^{n} \mathbf{x}_i^T \mathbf{x}_i \boldsymbol{\beta}_n = \mathbf{X}'\mathbf{y} - \mathbf{X}'\mathbf{X}\boldsymbol{\beta}_n = 0 \tag{3}$$

와 같이 정리된다. 따라서 $\boldsymbol{\beta}_n = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$ 와 같이 정리된다.

- 반면 generalized M-estimator 는 아래를 최소화하는 해이다.

$$\sum_{i=1}^{n} \mathbf{x}_i \phi\big(\mathbf{x}_i, y_i - \mathbf{x}_i \boldsymbol{\beta}_n\big) = 0 \tag{4}$$

여기에서 ψ는 $1 \times p$ row-vector를 실수로 보내는 맵핑이다. 즉 $\psi : \mathbb{R} \times \mathbb{R}^p \to \mathbb{R}$ 이다. 만약에

$$\phi(\mathbf{x}, u) = \psi(u) \tag{5}$$

를 만족하면 이 경우 generalized M-estimator 는 classical M-estimator 와 같다.

- 이때 $\sum_{i=1}^{n} \mathbf{x}_i \phi\big(\mathbf{x}_i, y_i - \mathbf{x}_i \boldsymbol{\beta}_n\big) = 0$ 은 아래와 같이 쓸 수 있다.

$$\int \mathbf{x}\phi\big(\mathbf{x}, y - \mathbf{x}\boldsymbol{\beta}\big) dF_n = 0 \tag{6}$$

- WLS의 경우

$$\boldsymbol{\beta}_n = \big(\mathbf{X}'\mathbf{W}^2\mathbf{X}\big)^{-1}\mathbf{X}'\mathbf{W}\mathbf{y} \tag{7}$$

와 같이 된다. 이런 경우는 $\psi(y; \mathbf{w}_i\mathbf{X}) = \pi(\mathbf{w}_i\mathbf{X})$ 와 같이 선택되었다고 볼 수 있다.

Maronna and Yohai (1981)

- 일반적인 셋팅은 아래와 같다. (1.1)

$$y_i = \mathbf{x}_i\boldsymbol{\beta} + \epsilon_i \tag{8}$$

여기에서 ϵ_i 의 분산은 편의상 1 로 가정한다.

- 아래를 만족하는 $\boldsymbol{\beta}$ 를 구한다. (1.2)

$$\sum_{i=1}^{n} \mathbf{x}_i \phi\big(\mathbf{x}_i, y_i - \mathbf{x}_i \boldsymbol{\beta}_n\big) = 0 \tag{9}$$

이런 추정량 $\boldsymbol{\beta}_n$을 M-estimator라고 한다. 여기에서 ψ는 $p \times 1$ row-vector를 실수로 보내는 맵핑이다.

- 우리는 y_i 대신 $f(v_i)$ 를 대입하고 $\mathbf{x}_i$ 대신 $f(v_j),\ j \in 1, 2, \ldots, i-1, i+1, \ldots, n$ 를 대입한 셋팅을 생각하면 된다.

- 참고문헌: Maronna and Yohai (1981), Damien Garcia (2010)

- 적용자료: 지진자료 혹은 기상자료의 smoothing 을 수행하면 의미가 있어보인다. (시공간이 동시에 변화하는 자료들)

- τ 의 선택??

References

[1] Klein, D. J., and Randić, M. (1993). Resistance distance. Journal of mathematical chemistry, 12(1), 81-95.

[2] Shuman, D. I., Narang, S. K., Frossard, P., Ortega, A., and Vandergheynst, P. (2012). The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. arXiv preprint arXiv:1211.0053.

[3] Breunig, M. M., Kriegel, H. P., Ng, R. T., and Sander, J. (2000, May). LOF: identifying density-based local outliers. In ACM sigmod record (Vol. 29, No. 2, pp. 93-104). ACM.

[4] He, X., and Niyogi, P. (2004). Locality preserving projections. In Advances in neural information processing systems (pp. 153-160).

[5] Belkin, M., and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation, 15(6), 1373-1396.

[6] Ng, A. Y., Jordan, M. I., and Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In Advances in neural information processing systems (pp. 849-856).

[7] Tsang, I. W., and Kwok, J. T. (2007). Large-scale sparsified manifold regularization. In Advances in Neural Information Processing Systems (pp. 1401-1408).

[8] Lafon, S., and Lee, A. B. (2006). Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. IEEE transactions on pattern analysis and machine intelligence, 28(9), 1393-1403.

[9] Leskovec, J., Lang, K. J., Dasgupta, A., and Mahoney, M. W. (2009). Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. Internet Mathematics, 6(1), 29-123.