TEKNILLINEN KORKEAKOULU
Teknillisen fysiikan
koulutusohjelma

ERIKOISTYÖ
Mat-1.125 Matematiikan erikoistyö
September 6, 2004

Data clustering with kernel methods derived from Kullback-Leibler information

Ilkka Kudjoi
58117T

Contents

1 Introduction

Assume that one has huge statistical data about some phenomenon like people's ratings for some service, statistics from nature, etc. Huge data is hard to handle and therefore it would be nice if it could be classified to acquire any generalisation for the data.

Computer science provides plethora of different classification methods for Multi-Layer Perceptron network and other constructions. In this study we shall measure the true information between the data instances by applying kernel methods which are derived from the two most popular information definitions, Kullback-Leibler and Fisher information.

By assuming that the data is created by a probabilistic model, we acquire expectation estimates for the kernel functions with Markov Chain Monte Carlo integration. By sampling we acquire a kernel matrix that indicates the distances between instances in the data. If the data is ordered due to our prior assumptions, a clear clusterisation of the matrix may be straight available. Otherwise the matrix will be reordered with symmetric row and column changes to reach the clusterisation.

From the ordered kernel matrix we are able to see which data instances belong to same clusters and which are distant from other instances. In other words, we acquire a classification for the data in form of a kernel matrix.

In this study we shall also infer whether additional noise in data or data sparsity affects the result considerably by testing the kernels with suitable artificial data sets.

2 Information measures

A sufficient statistic should contain all the information about parameters of a probabilistic model. Information is a measure of the evidence that a data set provides about a parameter in a parametric family. To make the idea more precise, two popular definitions of information is introduced in next sections, the Fisher Information [18, 8, 21] and the Kullback-Leibler information [11, 18, 21, 20, 22].

2.1 Fisher information

The Fisher information is designed to provide a measure of how much information a data set provides about a parameter.

Fisher information is defined with help of the *Fisher score*, gradient of the log-likelihood of the probability density function

$$U(x,\theta) \quad = \quad \nabla_\theta \log f(x|\theta) \tag{2.1}$$

In the definition it is required that the partial derivatives of $f(x|\theta)$ exist.

The Fisher score maps an example point x into a feature vector in the gradient space. This is called *Fisher score mapping. The Fisher information matrix* is defined then by the score by expected value of the outer product

$$I(\theta) \quad = \quad E[U(x,\theta)U(x,\theta)^T \mid \theta] \tag{2.2}$$

$$\quad = \quad \int_\Omega f(x|\theta)U(x,\theta)U(x,\theta)^T \mathrm{d}x \quad , \tag{2.3}$$

where we have to require that the score is squareintegrable.

The Fisher Kernel [8] is defined then by the equation

$$K(x_i,x_j) \quad = \quad U(x_i)^T I^{-1} U(x_j) \quad . \tag{2.4}$$

The Fisher information is designed to associate a distribution with a scalar value that tells how much information the distribution gives about the data. For example, the information that a sharp normal distribution gives is greater than the information given by a broad distribution.

For more complete introductions to the Fisher information, please refer to [18, 8, 21].

2.2 Kullback-Leibler information

Kullback-Leibler (KL) information or divergence [18, 21, 20, 22] is another definition for information between two distributions proposed by S. Kullback and R.A. Leibler in 1951 [11]. Fisher information is designed for one distribution but even then the Kullback-Leibler has a certain connection to Fisher information that will be discussed in Section 3.4.

Kullback-Leibler divergence is often recognised as a valid divergence measure between distributions, but unfortunately it is not straightforward to use the Kullback-Leibler divergence as a kernel,

because it is neither symmetric or it does not behave like inner product. Even so, the KL can be approximated with valid kernel functions which will discussed later.

S. Kullback and R.A. Leibler introduced the information in x for discrimination between two probability measures by

$$\log \frac{p(x)}{q(x)} \quad . \tag{2.5}$$

The mean information between $p(x)$ and $q(x)$ respect to $p(x)$ is defined then by[1]

$$\begin{aligned} KL(p \,||\, q) &= E_p \left[\log \frac{p(x)}{q(x)} \right] \\ &= \int_\Omega p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x \quad . \end{aligned} \tag{2.6}$$

The Kullback-Leibler divergence is the expected amount of information that a sample from p gives of the fact that the sample is **not** from distribution q.

If the probability distribution is discrete, equation (2.6) can be written as follows:

$$KL(p \,||\, q) = \sum_i^N p_i \log \frac{p_i}{q_i} \quad . \tag{2.7}$$

Note that KL is not symmetric.

From the definition (2.6) it follows that if $q(x)$ is zero wherever $p(x)$ is not, the information is infinite. In other words, this means that if there exists even a single point x such that $p(x) \neq 0$ and $q(x) = 0$, then the p cannot be from q. Note that this will not necessarily apply vice-versa. On the other hand, if p and q are equal, the amount of information that p is not from q is zero.

The Kullback-Leibler divergence has a clear connection to Shannon's entropy and coding theory [19]. Shannon's entropy for a set of probabilities is defined by

$$H(p) = E_{p(x)} \left(-\log p(x) \right) \tag{2.8}$$

$$= \int_\Omega -p(x) \log p(x) \mathrm{d}x \quad , \tag{2.9}$$

or by discrete version

$$H(p) = -\sum_i^n p_i \log p_i \quad . \tag{2.10}$$

In the definition $-\log p_i$ is the optimal coding length of sequence x_i, where p_i is probability of x_i. The Shannon's entropy gives expected coding length of a message when optimal coding is used. Smaller entropy allows better data compression ratio.

2.2.1 Capasitory discrimination

The capasitory discrimination is defined by

$$C(p, q) = KL(p \parallel m) + KL(q \parallel m) \quad , \tag{2.11}$$

where $m = \frac{1}{2}(p + q)$.

Capasitory discrimination is symmetrical and behaves like Kullback-Leibler for small distances. To prove the argument, consider two distributions, $p(\theta) = f(\theta)$ and $q(\theta) = f(\theta + \delta)$. If we assume f to be continuous, we may approximate the average distribution m by $m(\theta) = f(\theta + \frac{1}{2}\delta)$. Now we acquire by adapting the proof provided in Appendix C.3 that

$$C(p, q) \approx 2KL(p \parallel q) \quad , \tag{2.12}$$

when $p \approx q$.

3 Kernels and Kernel functions

3.1 Information kernels

Information kernels [21, 17, 9, 5] are functions in information space that aim at measuring the similarity between instances. The kernel value is greater if two samples of information are similar and in contrary it limits to zero if pieces are distant. The kernel may be seen as a distance measure, although it cannot be considered as a metric. The kernel is a measure of *similarity*.

Gärtner discusses the connection between kernels and inner product of in a Hilbert space [5]. Often there exist a feature transformation from the information space to the Hilbert space so that the kernel function may be defined by the inner dot product by mapping the information into the Hilbert space.

Kernel distance is a bit different from a metric. A general metric has following properties

[1] In the definition the convention $x \log x|_{x \to 0} = 0$ is used.

$$
\begin{aligned}
\forall\, x_0, x_1, x_2 &\in \Omega \ , \\
d(x_0, x_1) &\geq 0 \ , && \text{(3.1a)} \\
d(x_0, x_1) &= d(x_1, x_0) \ , && \text{(3.1b)} \\
d(x_0, x_2) &\leq d(x_0, x_1) + d(x_1, x_2) \ , && \text{(3.1c)} \\
d(x_0, x_0) &= 0 \ . && \text{(3.1d)}
\end{aligned}
$$

The kernel function should fulfil

$$
\begin{aligned}
\forall\, x_0, x_1 &\in \Omega \ , \\
K(x_0, x_1) &\geq 0 \ , && \text{(3.2a)} \\
K(x_0, x_1) &= K(x_1, x_0) \ , && \text{(3.2b)}
\end{aligned}
$$

and the features mentioned earlier in this section, i.e. the kernel value decreases when x_0 and x_1 become distant but the distance increases.

A metric can emerge a kernel function e.g. by

$$
K(x_0, x_1) = e^{-d(x_0, x_1)^2}, \tag{3.3}
$$

Often, but not generally, a kernel function may be converted to a metric by (3.3) or by

$$
d(x_0, x_1) = \sqrt{K(x_0, x_0) + K(x_1, x_1) - 2K(x_0, x_1)}. \tag{3.4}
$$

In the previous equation we need to assume that

$$
\begin{aligned}
&\forall\, x_0, x_1 \in \Omega \ , \\
&K(x_0, x_0) \geq K(x_0, x_1) \ .
\end{aligned}
$$

3.2 Kernel functions

In this study we compare different kernel functions and their relation to the Kullback-Leibler divergence.

3.2.1 Hellinger affinity kernel

In Section 3.4 we shall prove that Hellinger distance approximates true information (Kullback-Leibler divergence) between two distributions. Hellinger distance between two distributions is defined by

$$
\begin{aligned}
d^2(p, q) &= \tfrac{1}{2} \int_\Omega (\sqrt{p(x)} - \sqrt{q(x)})^2 \mathrm{d}x \\
&= \tfrac{1}{2} \int_\Omega (p(x) - 2\sqrt{p(x)q(x)} + q(x)) \mathrm{d}x \\
&= 1 - \int_\Omega \sqrt{p(x)q(x)} \mathrm{d}x \ .
\end{aligned} \tag{3.5}
$$

To derive a valid kernel function from kernel distance, we make use of the affinity term of the distance.

$$
k(p, q) = \int_\Omega \sqrt{p(x)q(x)} \mathrm{d}x \ , \tag{3.6}
$$

or its discrete version

$$
k(p, q) = \sum_i^n \sqrt{p_i q_i} \ . \tag{3.7}
$$

In some instances [9, 10] this kernel is also known as *Bhattacharyya Kernel*, because in statistics literature it is known as Bhattacharyya's measure of affinity between distributions.

3.2.2 Count kernel

Count kernel is much like Hellinger affinity kernel, but it hasn't such connection to valid information like Kullback-Leibler information as the Hellinger affinity has. The count kernel is defined by

$$
k(p, q) = \int_\Omega p(x)q(x) \mathrm{d}x \ , \tag{3.8}
$$

$$
k(p, q) = \sum_i^n p_i q_i \ . \tag{3.9}
$$

The count kernel is often used with sequence data, like in text analysis. In text processing, p_i is the count of symbol i in a sequence normalised by the length. That is why the kernel function is called *count kernel*. Koji Tsuda *et al* discuss the count kernel with modifications in [21] and Tony Jebara *et al* discuss the kernel as a *product kernel* in [10].

3.2.3 Triangular kernel

Triangular discrimination is introduced by Flemming Topsøe in [20]. Topsøe proves that triangular discrimination has a strong connection to Kullback-Leibler divergence, especially to the capasitory discrimination (2.11). The triangular discrimination is defined by

$$
\Delta(p, q) = \sum_i^n \frac{(p_i - q_i)^2}{p_i + q_i} \ . \tag{3.10}
$$

The discrimination is equal to capasitory discrimination (2.11) in the sense of metrics

$$\frac{1}{2}\Delta(p,q) \quad \leq \quad C(p,q) \quad \leq \quad \log(2)\,\Delta(p,q). \tag{3.11}$$

This is proved by Topsøe in [20].

Again, triangular discrimination is not valid kernel function as such. In this assignment, a valid kernel function is derived from the triangular discrimination by

$$k(p,q) \quad = \quad 1 - \widetilde{\Delta}(p,q), \tag{3.12}$$

where the target values of $\widetilde{\Delta}$ is normalised to interval [0,1].

3.2.4 Beta kernel

The name of the beta kernel is fully fictitious. The kernel is derived from the Multinomial distributions β in the co-occurrence model introduced in Section 4.2.

Actually beta kernel is only application of the Hellinger kernel (3.6). As in the ordinary Hellinger kernel we measure the distance between cluster probabilities $p(z|x)$ and $p(z|x')$, in this variation the distance will be measured between $p(Y,\theta|x)$ and $p(Y,\theta|x')$ by the Hellinger distance. In the distribution parameters z is the cluster, x is instance in the data, Y is the other data collection and θ denotes the model parameters in general. The actual distance sampling formulas will be discussed in MCMC-section 5.2.

3.3 Hellinger integral and distance

Hellinger integral [4, 13] is defined by

$$f(\alpha;x,y) \quad = \quad x^{\alpha}\,y^{1-\alpha} \quad , \tag{3.13}$$

$$H(\alpha;p,q) \quad = \quad \int_{\Omega} f(\alpha;p(x),q(x))\,\mathrm{d}x \quad , \quad \alpha \in (0,1) \quad . \tag{3.14}$$

Another function is defined by the Hellinger integral because it proves to be helpful,

$$L(\alpha;p,q) \quad = \quad \frac{1-H(\alpha;p,q)}{\alpha} \quad . \tag{3.15}$$

Hellinger distance can be defined by the previous equation by setting $\alpha = \frac{1}{2}$,

$$\begin{aligned} d^2(p,q) \quad &= \quad \tfrac{1}{2}\,L(\tfrac{1}{2}\,;p,q) \quad , \\ &= \quad 1 - H(\tfrac{1}{2};p,q) \quad , \\ &= \quad 1 - \int_{\Omega}\sqrt{p(x)q(x)}\mathrm{d}x \quad . \end{aligned} \tag{3.16}$$

Some interesting results have been derived with the Hellinger integral [4]. The function L behaves like Kullback-Leibler divergence when α limits to zero, because

$$\lim_{\alpha\to 0^+} f'_{\alpha}(\alpha;x,y) \quad = \quad -\,y\log\frac{y}{x} \quad . \tag{3.17}$$

The same written as a difference quotient

$$\begin{aligned} \lim_{\alpha\to 0^+} L(\alpha;p,q) \quad &= \quad \lim_{\alpha\to 0^+} \frac{1-H(\alpha;q,p)}{\alpha} \\ &= \quad KL(p \parallel q) \quad . \end{aligned} \tag{3.18}$$

Particular proofs for (3.17) and (3.18) are provided in Appendix C.1.

3.4 Connections between KL and Hellinger distance

Kullback-Leibler itself does not satisfy requirements of a kernel function, but there are several reasons that kernel distances derived from Kullback-Leibler divergence would be good information measures. From (3.17) and (3.18) we see that Hellinger integral and distance might be good approximations for Kullback-Leibler divergence. By writing (3.13) in exponential form we are able to see that it is convex[2] and monotonic function (which are features of an exponential function).

$$f(\alpha;x,y) = ye^{\alpha\log\frac{y}{x}} \quad . \tag{3.19}$$

Therefore we obtain $\forall\, 0 < \tilde{\alpha} < \alpha$

$$\begin{aligned} f'_{\alpha}(0;p,q) \quad &\leq \quad \frac{f(\widetilde{\alpha};x,y) - f(0^+;x,y)}{\widetilde{\alpha}} \quad , \\ &\leq \quad \frac{f(\alpha;x,y) - f(0^+;x,y)}{\alpha} \quad . \end{aligned} \tag{3.20}$$

From the inequality we see that L (3.15) is decreasing function and we acquire from (3.18) that

$$L(\alpha;p,q) \quad \leq \quad KL(p \parallel q) \quad \forall\, p,\, q. \tag{3.21}$$

[2] f is convex if $f(\frac{1}{2}(x_1+x_2)) \leq \frac{1}{2}(f(x_1)+f(x_2))\,\forall x_1, x_2$

By combining (3.21) and (3.16) we acquire

$$d^2(p,q) \quad \leq \quad \frac{1}{2} KL(p \parallel q) \quad . \tag{3.22}$$

The inequality means that Hellinger distance can be majored by Kullback-Leibler divergence. The inequality also proves that Kullback-Leibler is positive semidefinite. With this inequality and the definition (2.6) it follows that the divergence is zero only if the distributions are equal.

Unfortunately the distance and divergence cannot be considered as equivalent metrics, because

$$\nexists C > 0, \quad KL(p \parallel q) \quad \leq \quad C\, d^2(p,q) \quad \forall\, p,\, q. \tag{3.23}$$

This may be easily proved by setting q to zero at some point x that fulfils $p(x) \neq 0$. In this case, Kullback-Leibler information is infinite but Hellinger remains finite. Actually, Hellinger distance is always finite, and therefore equation (3.23) cannot hold.

In practise, this is not a problem, because the approximations are good enough. They are locally equivalent and the they behave nicely.

3.4.1 Small distances

The limiting behaviour of Hellinger distance and Kullback-Leibler divergence proves to be equal. I.e, when the distribution changes slightly, the change in the divergence may be approximated by the Hellinger distance. Both measures have the connection to Fisher information [18, 8, 21] with small distances. Proof for both equations are provided in Appendixes C.2 and C.3.

$$KL(f(x|\theta) \parallel f(x|\theta+\delta)) \quad = \quad \tfrac{1}{2} \sum_{i,j}^{N} I_{ij}(\theta)\delta_i\delta_j \ , \tag{3.24}$$

$$d^2(f(x|\theta), f(x|\theta+\delta)) \quad = \quad \tfrac{1}{8} \sum_{i,j}^{N} I_{ij}(\theta)\delta_i\delta_j \ ., \tag{3.25}$$

where θ is a parameter vector, n is a limiting scalar, δ is small parameter step and $I(\theta)$ is the Fisher information matrix.

On the basis of equations (3.22), (3.24) and (3.25) we propose that Hellinger distance can be used to approximate the Kullback-Leibler divergence, because they limit to the same expression fixed by a scalar.

4 Data and models

In this assignment we conducted a few test with different kernels and with artificial and natural data. We measured distances between users and documents in binary relevance data $\{user, document, boolean\ relevance\}$ and between instances in two-column co-occurrence data. To measure the distances, we must assume that the data is created through some probability model, which we shall introduce in next sections.

4.1 Binary relevance model

Experiments were originally started with movie ratings data that also being used in our research group for relevance prediction [15]. The data consists of 29,180 ratings from 134 users to 1282 movies. In this assignment the goal is to find out, in how many clusters the movies might divide, when the distances between movies are measured with kernel methods. The data model is specified as follows.

The model is a generative model where the data is considered *triplets* (u,d,r) where u,d and r are user, document and rating respectively. For the user collection, a vector of Multinomial parameters θ_U is drawn from Dirichlet($\theta_{\tilde{u}}$). The parameter vector θ_U contains a probability for each user to belong to user group $\tilde{u}$.

A user group $\tilde{u}$ is drawn from Multinomial(θ_U). Thereafter corresponding parameters $\beta_U(\tilde{u})$ may be chosen to be used when drawing the user u from Multinomial($\beta_U(\tilde{u})$).

The document distributions are drawn symmetrically. A vector of Multinomial parameters θ_D is drawn from Dirichlet($\alpha_{\tilde{d}}$). Like with users, the vector contains occurrence probabilities for document clusters $\tilde{d}$.

A document cluster $\tilde{d}$ is drawn from Multinomial(θ_D). As the document cluster is fixed, corresponding parameters $\beta_D(\tilde{d})$ may be taken and used to draw d from Multinomial($\beta_D(\tilde{d})$).

When the cluster pair $(\tilde{u}, \tilde{d})$ is drawn, a vector of Binomial parameters should be drawn from Dirichlet(α_R). Parameters $\theta_R(\tilde{u}, \tilde{d})$ indicate the

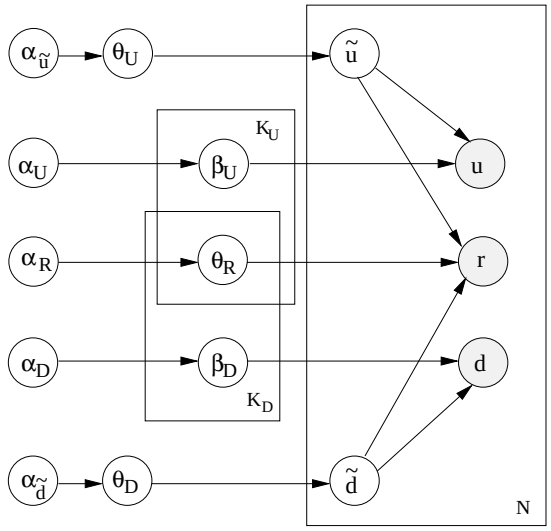

Figure 1: Binary relevance data model. The grey variables are the observables

probability of group $\tilde{u}$ to consider group $\tilde{d}$ relevant. For each cluster pair $(\tilde{u}, \tilde{d})$, a binary relevance r is drawn from Binomial$(\theta_R(\tilde{u}, \tilde{d}))$. Now the model has generated a triplet (u, d, r) with binary relevance r, which indicates whether the user likes the document or not. A figure of the model is provided in Figure 1.

4.2 Co-occurrence data model

Secondly we applied kernel methods to co-occurrence data. Intuitively, the kernel methods measure how often data points occur together. Here is the specification of the co-occurrence data model.

First, a vector of Multinomial parameters θ is drawn from Dirichlet(α). The dimension of θ is $N \times K$, where N is the count of data samples and K is the number of clusters in the model. The vector θ contains the cluster probabilities for each data sample.

When θ is drawn we may draw the co-occurrence cluster z from Multinomial(θ). As the cluster is fixed, x and y may be drawn from corresponding Multinomial distributions Multinomial(β_x) and Multinomial(β_y), where β_x and β_y are drawn from corresponding Dirichlet distributions Dirichlet(α_x) and Dirichlet(α_y).

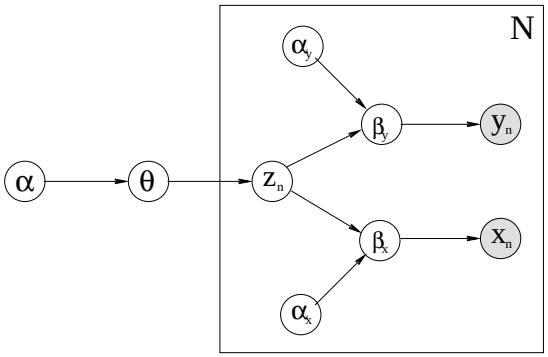

Figure 2: Co-occurrence data model

Figure of the co-occurrence data model is available in Figure 2.

5 MCMC-integration

In every kernel function that was introduced in previous sections we assumed that the distributions p and q are known. In practise, the distributions are not known explicitly and they must be approximated e.g. with MCMC-integration [14, 1, 12].

Markov Chain Monte Carlo (MCMC) integration is a method to estimate expected values from a distribution by drawing samples from it. The expected value of a function $a(\theta)$ with respect to $Q(\theta)$ is acquired trough integral over the probability space

$$E[a] \quad = \quad \int a(\theta) Q(\theta) \mathrm{d}\theta \quad . \qquad (5.1)$$

The integral may be calculated analytically or by using a quadrature [12], but unfortunately often it is impossible in practise. Other methods are also very expensive when the distribution is multidimensional. Therefore Markov Chain Monte Carlo integration is often unavoidable.

The integral can be approximated with the MCMC method by

$$E[a] \quad \approx \quad \frac{1}{N} \sum_{t=1}^{N} a(\theta^{(t)}) \quad . \qquad (5.2)$$

Here $\theta^{(1)}, \ldots, \theta^{(N)}$ are the samples from Q. The standard error of the integration formula [12], the expected value is the same as the standard error of

an average estimator, i.e.

$$\epsilon \quad = \quad \frac{\sigma}{\sqrt{N}} \quad . \tag{5.3}$$

Note that the error does not depend on the dimensionality of the distribution.

The main problem with MCMC-integration is generating samples from the distributions. The samples should be independent, but it is often impossible in practise. The sum (5.2) will nevertheless converge to the expected value if the dependence is thin. *A Markov Chain* [2, 1], having Q as its stationary distribution would be a good way to generate such samples [14].

The distributions in this assignment are cluster probabilities, i.e. p_i is the probability that certain observable belongs to cluster i. By sampling the cluster distributions and discriminating them with the kernel functions we get distances between different observables.

5.1 Unidentifiability

With almost all the kernel functions used in this assignment, we assign each data point to a cluster by sampling. After considering each data point, the occurrences are then normalised to get discrete probability distributions for each of the data points. The distribution shows the probabilities that the data point belongs to a certain class.

Unfortunately, in spite of the fact that the distribution may remain almost unchanged after each iteration, the model may sample data points to a different cluster. I.e, if data point collection d_a used to be in cluster A and d_b in cluster B, on the next iteration the clusters could be changed. This is a known difficulty in the analysis of MCMC results.

This means that the distances must be evaluated separately on each iteration because the samples may not be collected from more than one iteration. The distances may then be averaged after N iterations.

5.2 Sampling distributions

In Section 3.2 and in Section 4 we introduced kernel functions and data models respectively. In the kernels section the distances were evaluated between some unknown distributions p and q. In this subsection the actual distributions are introduced.

5.2.1 Binary relevance model sampling formula

In the binary relevance model we wish to evaluate the distances between distributions $p(\tilde{d}, \theta|d, D)$ and $p(\tilde{d}, \theta|d', D)$, where d:s are documents, D is the collection of all observations and θ is the model parameters, to acquire distance between d and d'. The Kullback-Leibler between the distributions is

$$\begin{aligned} & KL(p(\tilde{d}, \theta|d, D) \parallel p(\tilde{d}, \theta|d', D)) \\ = \quad & \sum_{\tilde{d}} \sum_{\theta} p(\tilde{d}, \theta|d, D) \log \frac{p(\tilde{d}, \theta|d, D)}{p(\tilde{d}, \theta|d', D)} \quad . \end{aligned} \tag{5.4}$$

The first sum is over all clusters and the second over all samples. By the formula of conditional probability we can write

$$p(\tilde{d}, \theta|d, D) \quad = \quad p(\tilde{d}|\theta, d, D) p(\theta|d, D) \quad ,$$

because the parameters θ hardly depends on a single data point d, it can reduced away

$$= \quad p(\tilde{d}|\theta, d, D) p(\theta|D) \quad .$$

Now the equation 5.4 can be written as

$$\begin{aligned} = \quad & \sum_{\tilde{d}} \sum_{\theta} p(\theta|D) p(\tilde{d}|\theta, d, D) \log \frac{p(\tilde{d}|\theta, d, D)}{p(\tilde{d}|\theta, d', D)} \\ = \quad & E_{p(\theta|D)} \left[p(\tilde{d}|\theta, d, D) \log \frac{p(\tilde{d}|\theta, d, D)}{p(\tilde{d}|\theta, d', D)} \right] \quad . \end{aligned} \tag{5.5}$$

Now we have a sampling formula for Kullback-Leibler divergence. The same derivation applies for the kernel functions.

By substituting d by u we get sampling formula for users.

5.2.2 Co-occurrence model sampling formula

For Count, Hellinger and Triangular kernel the derivation of the sampling formula is equal to the previous one.

$$KL(p(z, \theta|x, D) \parallel p(z, \theta|x', D))$$

$$= \sum_z \sum_\theta p(z,\theta|x,D) \log \frac{p(z,\theta|x,D)}{p(z,\theta|x',D)} \quad . \quad (5.6)$$

Again we apply conditional probability

$$p(z,\theta|x,D) \quad = \quad p(z|\theta,x,D)p(\theta|x,D) \quad ,$$

because the parameters θ have only a little dependence on a single data point x, it can reduced away

$$= \quad p(z|x,D)p(\theta|D) \quad .$$

Now the equation 5.6 can be written as

$$= \sum_z \sum_\theta p(\theta|D)p(z|\theta,x,D) \log \frac{p(z|\theta,x,D)}{p(z|\theta,x',D)}$$
$$= E_{p(\theta|D)} \left[p(z|\theta,x,D) \log \frac{p(z|\theta,x,D)}{p(z|\theta,x',D)} \right] \quad . \quad (5.7)$$

By substituting x by y we get sampling formula for instances y.

6 Experiments

6.1 Co-occurrence model and increasing noise

In this section we study the effect of random data in artificial data. A three-cluster artificial data of 100 instances of both data collections was created by making ten thousand doublets of both collections. One third of the data contained doublets of first third of the collections and second and the last third respectively.

Initially the data contains no noise at all. The amount of noise was added in intervals. We provide results with noiseless data and data containing 33 %, 75 %, 91 % and 95 % noise. The noise was added to the original data, and the data dimension is respectively 10000, 15000, 40000, 110000 and 200000 doublets.

This data was then measured by four different kernel methods, Count kernel, Hellinger kernel, Triangular kernel and Beta kernel. Visualisations of the kernels are available in Figure 3. Python module PyGist[3] was used for the visualisation.

[3] http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/python/pygist.html

6.1.1 Kernel visualisations

As we see from the figure, all the kernels perform pretty well as far as the amount of noise is below 75 %. It is characteristic for Hellinger distance that the distance between same instances is one. The count kernel does not have the same feature. Beta kernel behaves like Hellinger kernel and Triangular kernel is normalised to $[0, 1]$. Regardless of the greatest and smallest value Gist colours the greatest values with white and the smallest with black. Green is in the middle, blue are smaller and brown greater values.

The Count kernel stands out from the other kernels. It does not stand noise over 75 % and it behaves differently from the other kernels. It is also noticeable that behaviour of the three other kernels is pretty similar. This might be because the Count kernel is not a Kullback-Leibler approximation and therefore it does not reflect the true information divergences of the generative model of the data in contrast to Hellinger and Triangular distance.

The images include only kernel images of one data collection, because the dimension and the structure of the other collection is similar, the images would be alike.

6.1.2 Average distances

We also studied the average distances between instances in the kernel matrices. We calculated the average distance between instances in the same cluster and the average distance between instances in different clusters. This is easy as far the cluster structure is known. The results are in Table 1.

From the table we can easily see that the count kernel distances between instances in same cluster diminish when the noise grows. Hellinger distance preserves the distance between instances belonging to same clusters, but different clusters merge as the level of noise is increased. This is natural, because the random data contains doublets that belong to different clusters in the sense of the original data.

Triangular kernel behaves like Hellinger kernel, but partly because it is normalised, the minimum distance grows more moderately that with Hellinger. The image of the Beta kernel with high noise (Fig. 3) seems quite definite, but actually almost all distances are approximately one.

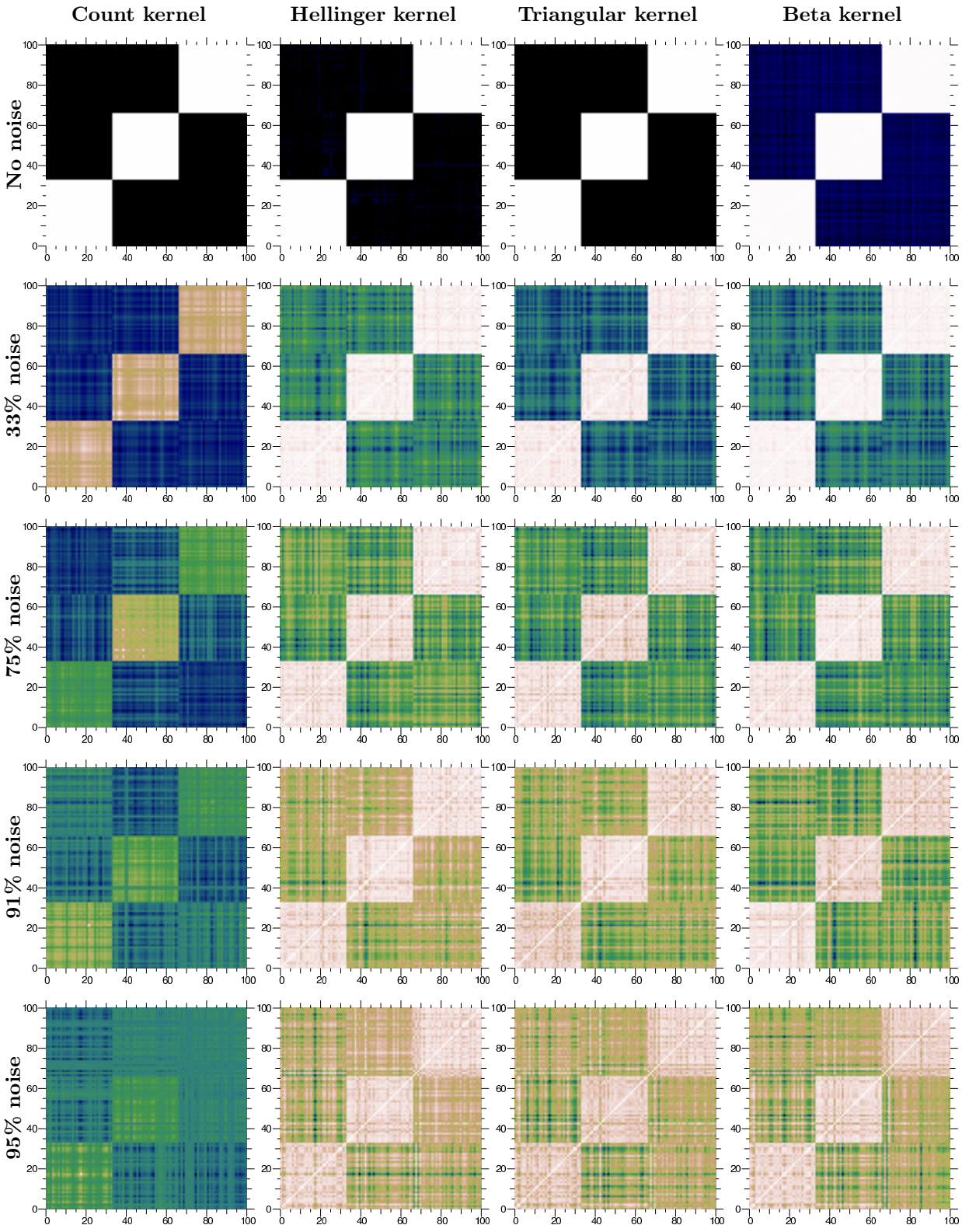

Figure 3: Visualisation's of kernel matrices with different noise levels and artificial data.

Table 1: Average distances between instances in same or different clusters

Same cluster				
Noise %	**Count**	**Hell.**	**Tri.**	**Beta**
0	1.002	1.002	1.000	1.000
33	0.774	0.988	0.976	0.997
75	0.482	0.987	0.952	1.000
91	0.392	0.991	0.940	1.001
95	0.371	0.990	0.919	1.001

Different cluster				
Noise %	**Count**	**Hell.**	**Tri.**	**Beta**
0	0.000	0.002	0.000	0.204
33	0.085	0.370	0.159	0.530
75	0.205	0.605	0.312	0.712
91	0.233	0.690	0.479	0.739
95	0.328	0.959	0.727	1.000

6.2 Co-occurrence model with sparse data

In addition, tests with sparse data were carried out. Figure 4 reflects the behaviour of the kernels when the data is sparse. Again, the co-occurrence data collection consists of one hundred instances each. The data was created like in the previous section, but this time four clusters were created for change.

Five different dimensions were tried. The sparsest data included only three hundred doublets, meaning that each instance co-occurred only three times in the data. Data files of 400, 500, 600 and 800 doublets were also used with 500 iterations.

From the kernel images (Fig 4) we may see that too few of occurrences of an instance in the data is not satisfactory. In this setup, the data should consist of six hundred doublets at least.

Amount of clusters affects the performance. If there was only three clusters in the data, less data would be needed.

6.3 Co-occurrence in forest cover type data

Tests with co-occurrence model and kernels were also carried out with natural data. We acquired forest cover type data from a data archive [6]. The data was contributed to the archive by Jock A. Blackard, Colorado State University[4].

The data includes 581012 instances of forest cover and soil type data. The description of the data is available in[5]. The data was parsed with Python into co-occurrence data of forty different forest soil types and seven different forest cover types.

Like with artificial data, the distances were measured with four different kernels. The cluster number was assumed to be three. Again, the kernels had to be ordered to reveal any structure in them, but in contrary to the movie rating kernels in the next section, the order of the instances is the same in all images. See the images in Figure 5.

As we can see from the images, the clusterisation is pretty good. From the cover types we notice that cover types "Ponderosa Pine", "Cottonwood/Willow" (fin. jättipoppeli/paju) and "Douglas-Fir" (fin. douglaskuusi) are similar. Also "Spruce/Fir" (fin. kuusi) and "Krummholz" (fin. vuorimänty) in addition to "Lodgepole Pine" and "Aspen" (fin. haapa) form similar pairs.

6.4 Movie ratings data and binary relevance -model

The second natural data test was conducted with movie ratings data. We tried to divide the movies per cluster with kernel functions. In this study we measured the distances between the cluster probability distributions $p(c|d)$, where c is the cluster and d document, i.e. movie with three different kernel functions, Count kernel, Hellinger kernel and Triangular kernel.

The kernel methods provided the kernel matrix also for the users, but results of user clustering were totally inadequate. In the model specification we must assume, how many clusters at most there are in the data. We assumed that there would be three clusters. If there were more, corresponding clusters would merge.

The movie distance kernel images may be seen in Figure 6. From the images we can see that the model is capable to find only two clusters in the

[4]The use of the data is unlimited with retention of copyright notice for Jock A. Blackard and Colorado State University.

[5]http://kdd.ics.uci.edu/databases/covertype/covertype.data.html

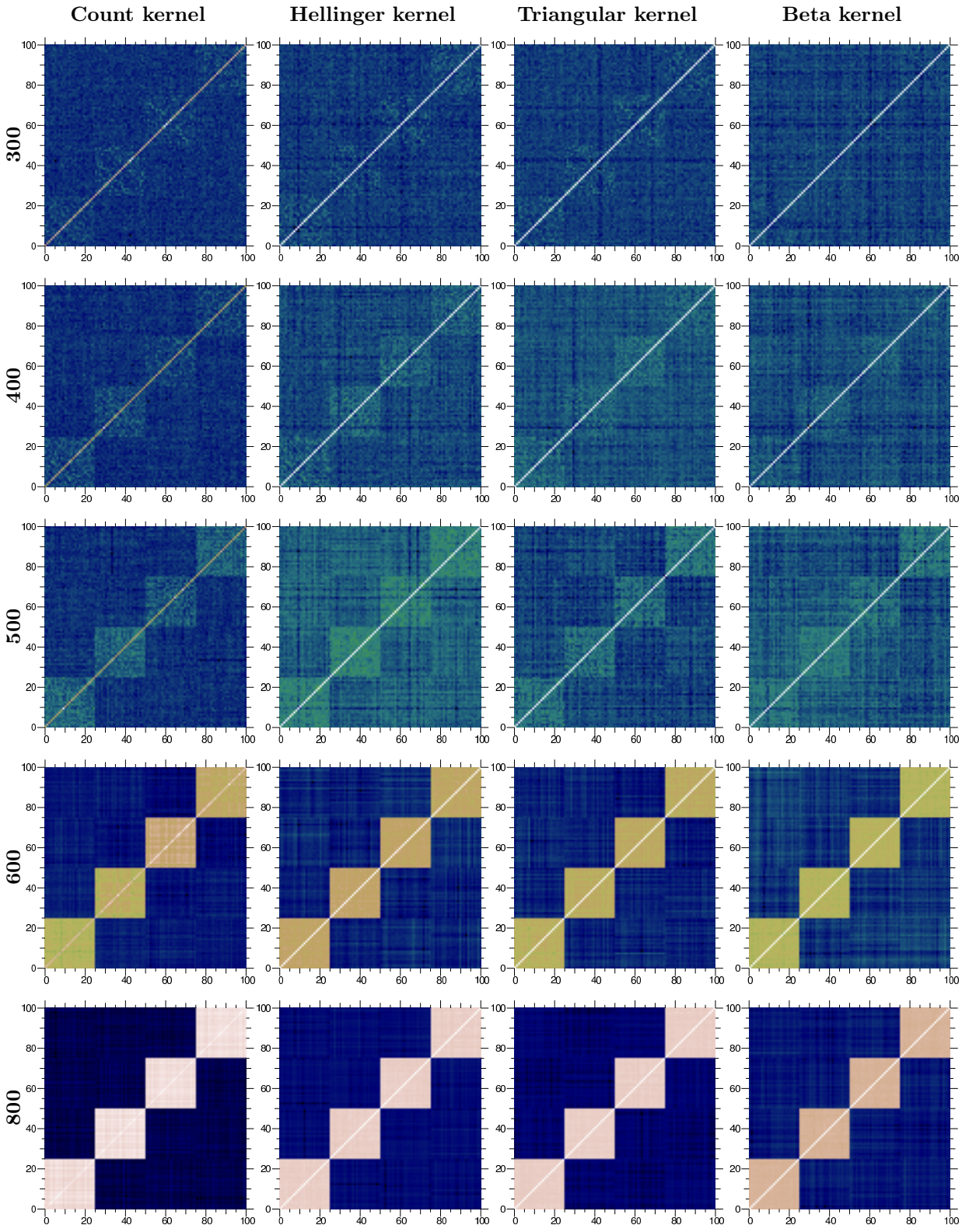

Figure 4: Visualisation of kernel matrices with sparse data.

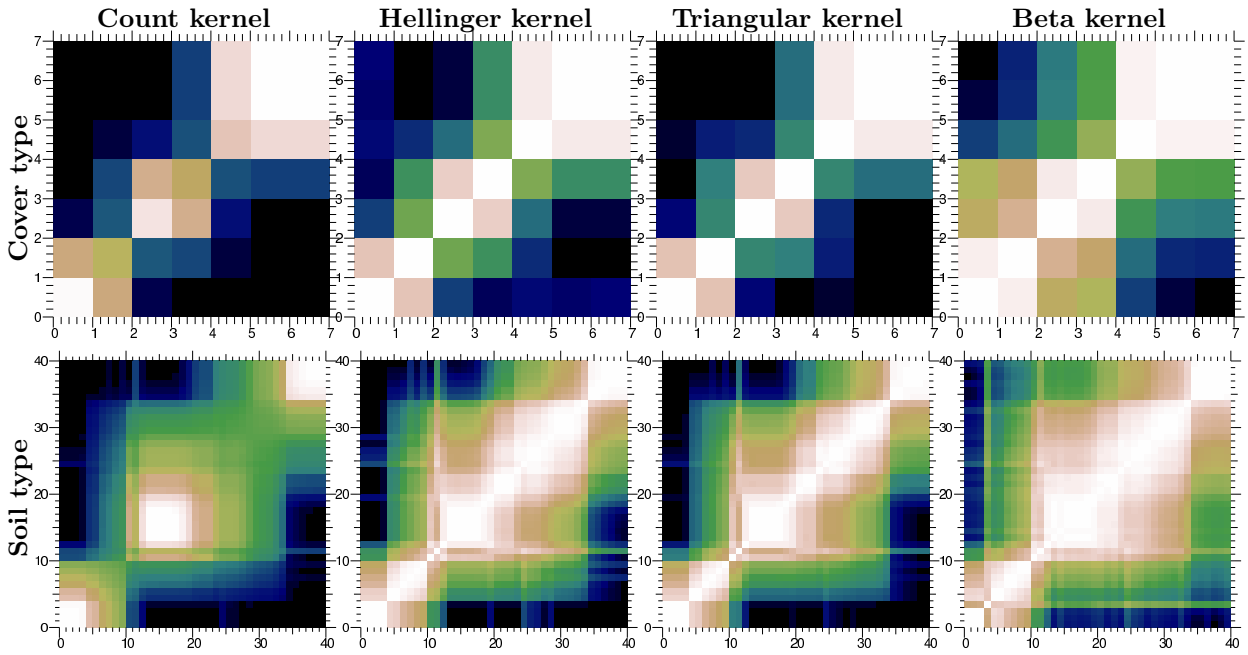

Figure 5: Forest cover type data kernels

movie collection. This might be because of the relevance is binary, instead of discriminating movies to different classes in sense of type, they might be divided into good and bad movies.

Because the original data is not ordered, the kernel matrix looks like a chessboard right after the kernel evaluation. Therefore the kernel matrix must be ordered with symmetric row and column changes. The procedure is explained in Appendix A. Note that the order of movies is not the same in all pictures.

7 Conclusions and Discussion

7.1 Approximating Kullback-Leibler information

Kullback-Leibler information was used in this work as a distance measure because it is used in many contexts as a divergence measure. As mentioned before, Kullback-Leibler itself cannot be used as kernel function as such. Nevertheless we assume that kernel functions derived from Kullback-Leibler information would perform well in our tests.

We chose symmetric kernel functions that fulfil inequalities and equalities compared to Kullback-Leibler. A kernel function fully equivalent to Kullback-Leibler does not exist, because KL is not limited in a way a kernel function should be.

Luckily, the results indicated that the approximations used in this assignment are promising. In addition, we noticed that the kernel function without connection to the Kullback-Leibler, the Count kernel, did not achieve the performance of the Hellinger and Triangular kernels.

The test with sparse data proved that the data model needs a reasonable amount of data to work properly.

7.2 Matrix ordering dilemma

If the original data being clustered is not ordered in means of clusterisation, the kernel matrix acquired will not show the cluster structure. Naturally, this is often the case with real world data. Therefore we should have a good ordering algorithm for the kernel matrix, so that we could reveal the cluster structure in it.

If the data collection dimension is any greater, the count of possible orderings is increased dramatically. N by N matrix may be ordered in $N!$ differ-

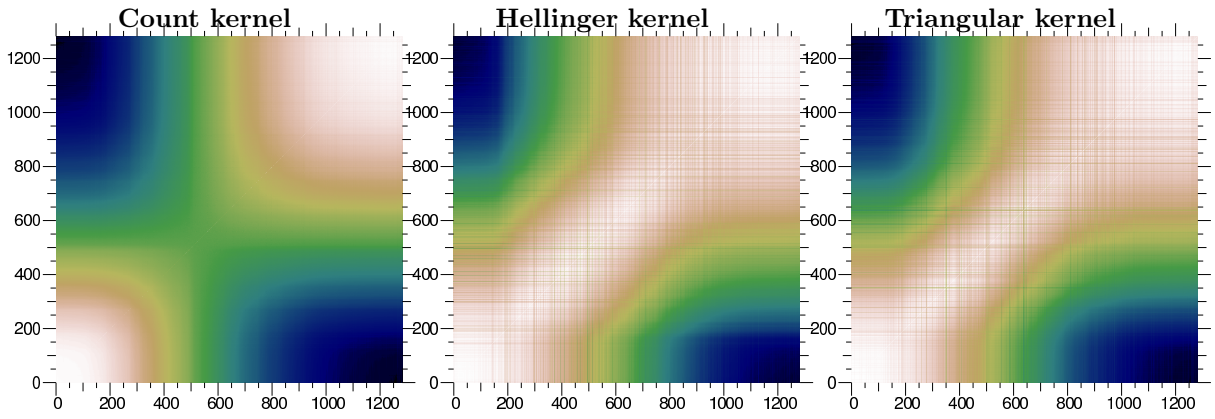

Figure 6: Visualisations of kernel matrices with movie ratings data

ent ways with symmetric row and column changes. Therefore we must use a greedy algorithm introduced in Appendix A that will not, in most cases, find the global minima of the defined energy function.

8 Acknowledgements

The experiment was mainly devised by the author with strong support of Kai Puolamäki and Esko Valkeila. I also want to thank Eerika Savia, Jarkko Salojärvi, Janne Sinkkonen and everyone else in the lab who helped me during my assignment.

When devising this work I was a summer student at the lab, and this work will be graded as my second special assignment in Helsinki University of Technology.

References

[1] Pierre Brémaud. *Markov Chains : Gibbs Fields, Monte Carlo Simulation, and Queues*. Number 31 in Texts in Applied Mathmetics. Springer, 1999.

[2] Richard Durrett. *Essentials of Stochastic Processes*. Springer Verlag, July 1999.

[3] Ronald F. Gariepy and William P. Ziemer. *Modern Real Analysis*. PWS Publishing Company, 1995.

[4] A. A. Guschin and E. Valkeila. Exponential statistical experiments: their properties and convergence results. *Statistics & Decisions*, (19):173–191, 2001.

[5] Thomas Gärtner. A survey of kernels for structured data. *ACM SIGKDD Explorations Newsletter*, 5:49–58, July 2003.

[6] S. Hettich and S. D. Bay. The uci kdd archive [http://kdd.ics.uci.edu], 1999. Irvine, CA: University of California, Department of Information and Computer Science.

[7] Antti Honkela. Nonlinear switching state-space models. Master's thesis, Helsinki University of Technology, May 2001.

[8] T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Proceedings of the 1998 conference on Advances in neural information processing systems II*, pages 487 – 493. MIT Press, 1999.

[9] Tony Jebara and Risi Kondor. Bhattacharyya and expected likelihood kernels. In B. Schölkopf and M. Warmuth, editors, *Proceedings of the Annual Conference on Computational Learning Theory and Kernel Workshop*, Lecture Notes in Computer Science. Springer, 2003.

[10] Tony Jebara, Risi Kondor, and Andrew Howard. Probability product kernels. *Journal of Machine Learning Research*, pages 819–844, July 2004.

[11] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22, 1951.

[12] Diego Kuonen. Numerical integration in s-plus or r: A survey. *Journal of Statistical Software*, 8(13), 2003.

[13] F. Liese and I. Vajda. *Convex Statistical Distances*. BSB B. G. Teubner, Leipzig, 1987.

[14] Radford M. Neal. *Bayesian Learning for Neural Networks*. Springer Verlag New York, 1996.

[15] K. Puolamäki, E. Savia, J. Sinkkonen, and S. Kaski. Two-way latent topic model for relevance prediction. Technical report, Helsinki University of Technology, 2004/2005. In preparation.

[16] Walter Rudin. *Principles of Mathematical Analysis*. McGraw-Hill Book Company, 1976.

[17] Saborou Saitoh. *Theory of reproducing kernels and its applications*. Longman Scientific & Technical, 1988.

[18] Mark J. Schervish. *Theory of Statistics*. Springer series in statistics. Springer, 1995.

[19] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–432, 623–656, July, October 1948.

[20] Flemming Topsøe. Some inequalities for information divergence and related measures of discrimination. *IEEE Transactions on Information Theory*, 46:1602–1609, 2000.

[21] Koji Tsuda, Taishin Kin, and Kiyoshi Asai. Marginalized kernels for biological sequences. *Bioinformatics*, 1(1):1–8, 2002.

[22] Harri Valpola. *Bayesian Ensemble Learning for Nonlinear Factor Analysis*. PhD thesis, Helsinki University of Technology, 2000.

A Ordering kernel matrix

When a kernel matrix is acquired through some kernel method, it might look all stirred because it is not ordered properly. By ordering the kernel matrix with symmetric row and column changes so that larger kernel values move near the diagonal and the lesser values vice versa, the kernel matrix may get divided in clusters. One way to order (later on *diagonalise*) the kernel matrix is to minimise properly selected energy function, like

$$E = \sum_i^n \sum_j^n (i-j)^\alpha K(x_i, x_j)^\beta \tag{A.1}$$

where $K(x_i, x_j)$ is the kernel distance between x_i and x_j, α and β are real constants (α should be a even number) and (i-j) is supposed to be a penalty weight that will pitch the large kernel values to move toward the diagonal where the penalty multiplier is smaller. Small values tend to go farther from the diagonal, where they stabilise the penalty multiplier.

The energy function will be minimised with symmetrical row and column-change operations. That is, if we change rows i and j the corresponding columns should also be changed. Because the energy function (A.1) is costly to evaluate, only energy changes will be evaluated.

Now consider K_0 the kernel matrix before the row and column operations and K_1 after. The energy change is then

$$\begin{aligned} \Delta E &= \sum_i^n \sum_j^n (i-j)^\alpha \big(K_1(x_i, x_j)^\beta - K_0(x_i, x_j)^\beta\big) \\ &= \sum_i^n \sum_j^n (i-j)^\alpha \Delta K(x_i, x_j)^\beta \quad . \end{aligned} \tag{A.2}$$

Now it should be noticed that matrix ΔK^β is nonzero only at those rows and columns that were changed. Because i:th row and i:th column are same even when multiplied with the penalty factor, we can take only the two changed rows into consideration. Lets note the row vectors with r_i and r_j. Moreover, we notice that $r_i^j = r_j^i = 0$ and $r_i^k = -r_j^k$ when $k \neq i, j$. Therefore (A.2) becomes

$$\begin{aligned} \frac{1}{2}\Delta E &= \sum_{k \neq i,j}^n (i-k)^\alpha r_i^k + \sum_{k \neq i,j}^n (j-k)^\alpha r_j^k \quad , \\ \frac{1}{2}\Delta E &= \sum_{k \neq i,j}^n (i-k)^\alpha r_i^k - \sum_{k \neq i,j}^n (j-k)^\alpha r_i^k \quad . \end{aligned}$$

Now if we choose $\alpha = 2$ and $\beta = 1$, we get

$$\begin{aligned} \frac{1}{2}\Delta E &= \sum_{k\neq i,j}^{n} [(i-k)^2 - (j-k)^2] r_i^k \ , \\ &= \sum_{k\neq i,j}^{n} (i-j)(i+j-2k) r_i^k \ . \end{aligned} \tag{A.3}$$

This is considerably less expensive to calculate than (A.2).

Unfortunately, this algorithm proves to be inaccurate if the matrix is difficult (if there exists no clear clusterisation or relative differences of the kernel matrix elements are small) partly because it is a greedy algorithm. In some cases this may be evaded by running the algorithm several times or by preprocessing the matrix by considering the initial matrix binary. In this binary approach kernel values above some selected value are considered one and below the value zero. By selecting the value properly a valid ordering of the kernel is acquired.

B Distributions

B.1 Multinomial distribution

Multinomial distribution is a discrete distribution for a set of random variables X_i.

$$p(X_1 = x_i, \ldots, X_n = x_n) = \frac{N!}{\prod_{i=1}^{n} x_i!} \prod_{i=1}^{n} \theta_i^{x_i} \ , \tag{B.1}$$

where x_i are positive integers and $\sum_i^n x_i = N$. The parameters θ_i are constants that satisfy $\sum_i^n \theta_i = 1$. The θ_i:s indicate the probability of x_i to occur in the sequence

$$P(X_i) = \theta_i \ .$$

The name of the Multinomial distribution comes from the multinomial series, because the probability of sequence x is given by corresponding coefficient of a multinomial series

$$(\theta_1 + \theta_2 + \cdots + \theta_n)^N \ .$$

B.2 Dirichlet distribution

Dirichlet distribution [7] is the conjugate prior of the parameters of the Multinomial distribution. The probability density of the Dirichlet distribution for variables θ with parameters α is defined by

$$p(\theta) = \text{Dirichlet}(\theta; \alpha) = \frac{1}{Z(\alpha)} \prod_{i=1}^{n} \theta_i^{w_i - 1}. \tag{B.2}$$

with conditions $\theta_i \geq 0 \ \forall i$, $\sum_i \theta_i = 1$ and $\alpha_i > 0 \ \forall i$. The alphas can be interpreted as "prior observation counts" for events governed by θ_i. The $Z(\alpha)$ is normalisation constant

$$Z(\alpha) = \frac{\prod_{i=1}^{n} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{n} \alpha_i)} \ . \tag{B.3}$$

For more details about Dirichlet distribution, please refer e.g. to [7].

B.3 Binomial distribution

Binomial distribution is a special case of Multinomial distribution. In this assignment the distribution is used to draw binary relevance values. The general definition of Binomial distribution is

$$p(n|N) = \binom{N}{n} p^n q^{N-n} \ . \tag{B.4}$$

In this case, N is one

$$\begin{aligned} p(0) &= p \ , \\ p(1) &= q = 1 - p \ . \end{aligned}$$

C Divergence limits

C.1 Hellinger integral inequality

By using (3.19) with monotonic convergence theorem [16, 3] we obtain

$$\begin{aligned} &\lim_{\alpha \to 0^+} \frac{1 - H(\alpha; q, p)}{\alpha} \\ = &\lim_{\alpha \to 0^+} \frac{1 - \int_\Omega q^\alpha(x)\, p^{1-\alpha}(x)\, \mathrm{d}x}{\alpha} \\ = &\lim_{\alpha \to 0^+} \frac{\int_\Omega p(x) \mathrm{d}x - \int_\Omega q^\alpha(x)\, p^{1-\alpha}(x)\, \mathrm{d}x)}{\alpha} \\ = &\lim_{\alpha \to 0^+} \int_\Omega \frac{q^0(x)\, p^{1-0}(x) - q^\alpha(x)\, p^{1-\alpha}(x)\, \mathrm{d}x}{\alpha} \end{aligned}$$

The theorem allows us to change the order of limit and integral.

$$\begin{aligned} &= \int_\Omega \lim_{\alpha\to 0^+} \frac{q^0(x)\,p^{1-0}(x) - q^\alpha(x)\,p^{1-\alpha}(x)\,\mathrm{d}x}{\alpha} \\ &= \int_\Omega -\frac{\mathrm{d}}{\mathrm{d}\alpha} p(x) e^{\alpha \log \frac{q(x)}{p(x)}}\Big|_{\alpha=0} \\ &= \int_\Omega p(x) \log \frac{p(x)}{q(x)} \quad = \quad KL(p \parallel q) \quad . \qquad \text{(C.1)} \end{aligned}$$

Here we assumed that the Kullback-Leibler divergence is finite, i.e.,

$$\{x \,|\, q(x) = 0\} \quad \subseteq \quad \{x \,|\, p(x) = 0\}.$$

If else, it would be

$$1 \quad > \quad \int_\Omega q^0(x)\,p^{1-0}(x)\mathrm{d}x \quad ,$$

and it would follow that the limiting value in the proof would be infinite as would be the Kullback-Leibler divergence. Therefore the assumption is not restrictive.

C.2 Small Hellinger distances

In this section we shall prove with Taylor series that Hellinger distance limits to Fisher information with small distances. The final term assures that the distribution density function is normalized. We also need to assume that $\theta \to f(x|\theta)$ is smooth enough.

$$d^2(f(x|\theta), f(x|\theta+\delta)) \quad = \quad \int \left(\sqrt{f(x|\theta)} - \sqrt{f(x|\theta+\delta)}\right)^2 \mathrm{d}x + \lambda(1 - \int f(x|\theta+\delta)\mathrm{d}x)$$

Clearly $d^2(f(x|\theta), f(x|\theta+\delta))|_{\delta=0} = 0$. The first order partial derivatives.

$$\begin{aligned}\frac{\partial d^2}{\partial \delta_i} &= \frac{1}{2}\int 2\left(\sqrt{f(x|\theta)} - \sqrt{f(x|\theta+\delta)}\right)\left(-\frac{1}{2\sqrt{f(x|\theta+\delta)}}\frac{\partial f(x|\theta+\delta)}{\partial \delta_i}\right) + \lambda\frac{\partial f(x|\theta+\delta)}{\partial \delta_i}\mathrm{d}x \\ &= \frac{1}{2}\int\left(1 - \sqrt{\frac{f(x|\theta)}{f(x|\theta+\delta)}} + \lambda\right)\frac{\partial f(x|\theta+\delta)}{\partial \delta_i}\mathrm{d}x \quad \to_{\delta\to 0} \quad 0 \ .\end{aligned}$$

We demand that λ is zero. Second order partial derivatives

$$\begin{aligned}\frac{\partial^2 d^2}{\partial\delta_i\partial\delta_j} &= \frac{1}{2}\int\frac{\partial^2 f(x|\theta+\delta)}{\partial\delta_i\partial\delta_j}\left(1 - \sqrt{\frac{f(x|\theta)}{f(x|\theta+\delta)}}\right) + \frac{1}{2}\left(\frac{f(x|\theta)}{f(x|\theta+\delta)}\right)^{\frac{3}{2}}\frac{1}{f(x|\theta)}\frac{\partial f(x|\theta+\delta)}{\partial\delta_i}\frac{\partial f(x|\theta+\delta)}{\partial\delta_j}\mathrm{d}x \\ &\to_{\delta\to 0}\frac{1}{4}\int\frac{1}{f(x|\theta)}\frac{\partial f(x|\theta)}{\partial\delta_i}\frac{\partial f(x|\theta)}{\partial\delta_j}\mathrm{d}x \\ &= \frac{1}{4}\int f(x|\theta)\frac{\partial \log f(x|\theta)}{\partial\delta_i}\frac{\partial \log f(x|\theta)}{\partial\delta_j}\mathrm{d}x \\ &= \frac{1}{4}E_{f(x|\theta)}\left[\frac{\partial \log f(x|\theta)}{\partial\delta_i}\frac{\partial \log f(x|\theta)}{\partial\delta_j}\right] \quad = \quad \frac{1}{4}I(\theta)_{ij} \ .\end{aligned}$$

The second order Taylor series is

$$d^2(f(x|\theta), f(x|\theta+\delta)) \quad = \quad \frac{1}{2}\delta_i\delta_j\frac{1}{4}\frac{\partial^2 d^2}{\partial\delta_i\partial\delta_j} + \mathcal{O}(\delta^3) \quad = \quad \frac{1}{8}\delta_i\delta_j I(\theta)_{ij} + \mathcal{O}(\delta^3) \ . \tag{C.2}$$

C.3 Kullback-Leibler on short distances

The Kullback-Leibler divergence [11, 21, 20, 22] can be approximated by Fisher information [18, 8, 21] when the difference between the distributions are small. Below is the proof with Taylor series. In order to derive the series, the partial derivatives should be evaluated. Therefore it is again assumed that $\theta \to f(x|\theta)$ is smooth enough.

First order partial derivatives of the Kullback-Leibler are given below. The final term assures that the distribution density function is normalized.

$$\begin{aligned}KL(f(x|\theta+\delta) \parallel f(x|\theta)) &= \int f(x|\theta+\delta)\log\frac{f(x|\theta+\delta)}{f(x|\theta)}\mathrm{d}x + \lambda(1 - \int f(x|\theta+\delta)\mathrm{d}x)\mathrm{d}x \\ \frac{\partial KL}{\partial\delta_i} &= \int\left[\frac{\partial f(x|\theta+\delta)}{\partial\delta_i}\log\frac{f(x|\theta+\delta)}{f(x|\theta)} + (1-\lambda)\frac{\partial f(x|\theta+\delta)}{\partial\delta_i}\right]\mathrm{d}x \\ \to_{\delta\to 0} \quad (1-\lambda)\int\frac{\partial f(x|\theta)}{\partial\theta_i}\mathrm{d}x &= (1-\lambda)E_{f(x|\theta)}\left[\frac{\partial \log f(x|\theta)}{\partial\theta_i}\right] \quad = \quad (1-\lambda)U_i \quad = \quad 0 \ .\end{aligned}$$

The first partial derivatives of Kullback-Leibler divergence must limit to zero when $\delta \to 0$, because $\delta = 0$ is a local minima of Kullback-Leibler divergence. Because the Fisher score U_i is usually not zero, λ must be one.

Second order partial derivatives of the Kullback-Leibler reads

$$\begin{aligned}
\frac{\partial^2 KL}{\partial\delta_i\partial\delta_j} &= \int\left[\frac{\partial^2 f(x|\theta)}{\partial\delta_i\partial\delta_j}\log\frac{f(x|\theta+\delta)}{f(x|\theta)} + \frac{1}{f(x|\theta+\delta)}\frac{\partial f(x|\theta+\delta)}{\partial\delta_i}\frac{\partial f(x|\theta+\delta)}{\partial\delta_j} + (1-\lambda)\frac{\partial^2 f(x|\theta)}{\partial\delta_i\partial\delta_j}\right]\mathrm{d}x \\
\to_{\lambda=1,\delta\to 0} &\int\frac{1}{f(x|\theta)}\frac{\partial f(x|\theta)}{\partial\delta_i}\frac{\partial f(x|\theta)}{\partial\delta_j}\mathrm{d}x \\
&= \int f(x|\theta)\frac{\partial\log f(x|\theta)}{\partial\delta_i}\frac{\partial\log f(x|\theta)}{\partial\delta_j}\mathrm{d}x \\
&= E_{f(x|\theta)}\left[\frac{\partial\log f(x|\theta)}{\partial\delta_i}\frac{\partial\log f(x|\theta)}{\partial\delta_j}\right] \quad = \quad I(\theta)_{ij} \quad .
\end{aligned}$$

Since the divergence is zero in $\delta = 0$, and the first order partial derivatives vanishes, the second order Taylor series of Kullback-Leibler is

$$KL(f(x|\theta) \parallel f(x|\theta+\delta)) \quad = \quad \frac{1}{2}\delta_i\delta_j\frac{\partial^2 KL}{\partial\delta_i\partial\delta_j} + \mathcal{O}(\delta^3) \quad = \quad \frac{1}{2}\delta_i\delta_j I(\theta)_{ij} + \mathcal{O}(\delta^3) \quad . \tag{C.3}$$