barcelona|gse
graduate school of economics

Master Thesis

Topic:

Unsupervised learning in decision making

Author:	Domagoj Fizulic
	Felix Gutmann
Student number:	125604
	125584
Program:	M.S. Data Science
E-Mail:	domagoj.fizulic@barcelonagse.eu
	felix.gutmann@barcelonagse.eu

I Table of Contents

List of Figures

List of Tables

List of Algorithms

II List of mathematical symbols

Symbol	Meaning
μ	Mean of bandits
σ	Standard deviation of bandits
a	Action
$Q(a)$	Value function for action a
t	Discrete time step t
$R(a)$	Reward for action a
ϵ	Probability of exploration in epsilon greedy
α	Learning rate
η	Same parameter as α sometimes used in cognitive science
τ	Softmax or temperature parameter
θ	Inverse temperature parameter
X	Random variable
$H(X)$	Entropy of a discrete random variable X
$d(\cdot, \cdot)$	Distance Function
$S(\cdot, \cdot)$	Similarity Function
$K(\cdot, \cdot)$	Kernel function
$m(\cdot, \cdot)$	Matching function
$\Theta(\cdot)$	Heavy side step function
$\mathbb{R}_0^+$	Positive real numbers including zero
$\mathcal{X}$	Data set
$\mathbf{W}$	Weighted adjacency matrix
d_i	Degree of node i
$\mathbf{D}$	Diagonal matrix of degrees
$\mathbf{L}$	Graph laplacian

III List of abbreviations

Abbreviations	Description
IGT	Iowa gambling task
RL	Reinforcement learning
DTW	Dynamic time warp
EDR	Edit distance on real sequences
CH	Choices
BBC	Blockwise bad choices
ENT	Cumulative entropy
BENT	Blockwise entropy
CC	Concatenated
NMI	Normalized mutual information score
ARI	Adjusted rand index
VM	V-Measure score

1 Introduction and conceptual approach

Decision-making is a cognitive process of selecting an option from a set of possible alternatives based on certain criteria [Wang and Ruhe, 2007]. When analysing decision-making as a continuous process of interaction with the environment, learning becomes an important aspect. In general people show different individual learning behaviour. The learning procedure can be affected by different social, psychological and mental conditions. A common way to model this, is by estimating cognitive models and group people by their resulting parameters

In *supervised learning* data are predicted by training a classifier based on examples. The identity of observations in the training sample is known, which is used to connect patterns in the data with corresponding labels of observations. In contrast to that in *unsupervised learning* we don't know the ground truth. The objective is to discover natural clustering behaviour in the data itself and group objects into subsets, such that objects in those subsets are more closely related to each other [Murphy, 2012] and [Hastie et al., 2001]. A vast class of clustering algorithms with different approaches are proposed in the literature (e.g. hierarchical and optimization based clustering). This paper investigates, whether such unsupervised learning techniques can be used in the context of human decision making process to identify latent grouping. Approaching decision modeling conventionally requires a decent amount of explorative work and expertise. Unsupervised learning could serve as an easier tool to identify clusters of individuals. Thus, this paper operates in the intersection of machine learning and cognitive science. To our knowledge this particular setting has not been studied before.

We approach our research in the following way. We first set up a reinforcement learning based simulation framework to study theoretical boundaries of several clustering techniques and situation when they are applicable. Reinforcement learning is a widely used modeling framework within cognitive science. Furthermore, we able to control the clustering[1] and test our applied algorithms. Subsequently, we test our chosen methods on several real experiment data sets. Corresponding to our simulation framework we first apply clustering algorithms to data from a controlled n-armed bandit experiment.[2] A well known used approach to study human decision making process is the *Iowa gambling task*. Within this framework we analyse data sets from people with different criminal profiles and cocaine abusers.

Our objective is to use unsupervised learning methods to produce the same clustering created by the reinforcement learning or discover any other latent structures.

[1] Meaning we can choose the number of people in one cluster, because they should come from the same parameter settings.

[2] A detailed introduction is given in section 3.2

The report has the following structure. In section two we provide a short overview on related literature in the field. Section three is dedicated to the theoretical foundation of reinforcement learning and the related simulation. Furthermore, we describe our modeling approach in more detail and give an overview of applied algorithms, similarity and distance concepts. A mathematical formulation of the applied algorithms, similarity measures and cluster evaluation techniques can be found in the appendix.[3]
The rest of the paper addresses clustering our real experiment data. We keep the scope of this paper tight and thus some questions will remain open. We close this paper with a final summary of our results.

2 Relevant Literature

There exists a rich literature in cognitive science on identifying different behavioural groups. It has been shown that individuals with pre-frontal brain damage and decision-making defects continue to choose disadvantageously even after they learned the optimal strategy [Bechara et al., 1997].
A broad overview on other various results in the field can be found in [Steingroever et al., 2013]. Several studies identify specific drug-user groups, e.g. cocaine addicts [Stout et al., 2004], chronic cannabis users [Fridberg et al., 2010], heavy drinkers [Gullo and Stieger, 2011]. Furthermore, extensive set of research is focused around particular mental disabilities, e.g. Asperger's disorder [Johnson et al., 2006], psychopathic tendencies [Blair et al., 2001], bipolar disorder [Brambilla et al., 2012], schizophrenia [Martino et al., 2007] pathological gambling disorder [Cavedini et al., 2002], attention-deficit-hyperactivity disorder [Nirit Agaya, Eldad Yechiama, Ziv Carmelb, 2010]. Most popular reinforcement learning models for identifying behavioural differences between different disorders are Expectancy Valence model [Busemeyer and Stout, 2002] and Prospect Valence Learning model [Ahn et al., 2008].

3 Theoretical background of reinforcement learning and simulation experiments

This section is dedicated to a detailed outline of our analysis approach and the results of our simulation experiments. The type of data we want to analyse is gathered by observing peoples decisions over time. Hence, our data sets are in form of a $N \times M$ matrix, where N is the number of individuals and M is the number of trials in the

[3]Since our data are fairly small we will not discuss complexity of the algorithms.

experiment. The data for each individual can be seen as a categorical time series. In terms of modeling there are two challenges.
Some common algorithms relying for example on euclidean distance while others operating on similarities. Given the nature of our data we have to choose the right algorithm or similarity measure to cluster the data in a good way.
Furthermore, we demonstrate how to transform the categorical sequences and extract information about the behaviour (e.g. entropy, description see later on). Furthermore, transforming the data to real values makes them better suitable for most clustering algorithms.
Within this section we also give a broad overview of the applied algorithms etc. However, this section has a rather qualitative character. The appendix provides more formal background to applied algorithms, distance concepts and related clustering evaluation techniques.

3.1 Experiment design and problem formulation

Figure 1 depicts our simulation experiment design.[4] The objective is to obtain a set of sequential choices for a given parameter setting of an artificial agent. We first generate a set of rewards by sampling $n-$vectors from a normal distribution ("*multi arm bandits*"). The agent processes those rewards by sequentially choosing from those options. We repeat this procedure for several parameter settings and keep track of the resulting choices which will define our data set for the simulation.
We re-use those data for clustering.

[4]We implemented related coding for this project mainly in python. The code can be found on our github repository.

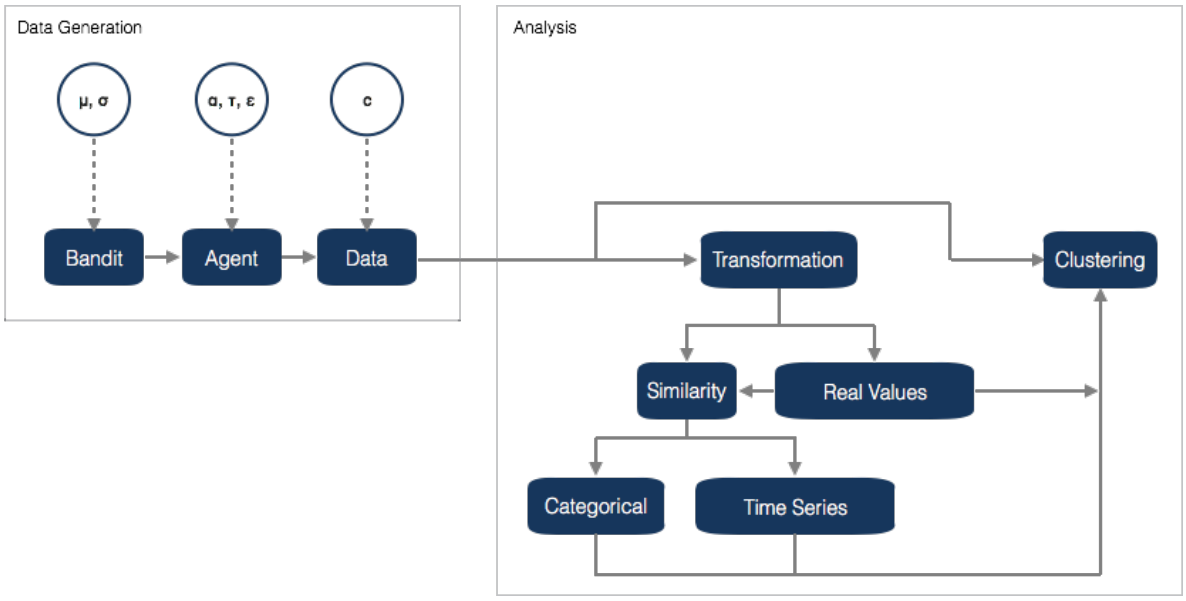


Figure 1: Flowchart experiment design

3.2 Reinforcement Learning background and multi arm bandits

Our simulation requires an artificial agent to produce decision data. The agent modeling is relying on *Reinforcement Learning* (RL). In the following we provide some necessary concepts from the field. The following definitions and intuitions are based on [Sutton and Barto, 2012]. RL is a branch of *machine learning* trying model the interaction of an artificial agent with its environment and the corresponding learning process.

In our particular setting the agent is confronted with the task of choosing sequentially from a set of N possible choices. The agent doesn't have any knowledge about the system a priori. Therefore, it has to learn the nature of the system by continuously interacting with its environment, while keeping track of the obtained information for each particular choice. Due to the lack of examples it has to explore different possible actions to identify the best action. Hence, it is useful to deviate from the current optimal strategy from time to time.

Each action in each step is associated with a given value based on the experience of the agent. This is modeled by defining a value function for each action a. The value function of an action a at time t is denoted as $Q(a)_t$. Hence the value function is defined as an average over the rewards for a given state. As mentioned it is necessary for the agent to explore its environment while simultaneously trying to optimize its utility. Thus, a crucial task of the agent is to balance exploration and exploitation of the environment. There are two basic approaches to model this trade-off; An *epsilon-*

greedy action selection method and a *softmax selection method*.
Within an epsilon greedy selection method the next action is chosen based on the highest current value function. However, to model exploration an random element is introduced to deviate from that greedy strategy with a certain probability (denoted by $\epsilon \in [0, 1]$). In general we can define the next action selected by the epsilon greedy strategy as:

$$a_{t+1} = \begin{cases} \text{random action} & \text{, with probability } \epsilon \\ \arg\max_i Q(i)_t & \text{, with probability } 1 - \epsilon \end{cases}$$

Hence, within the epsilon greedy method ϵ is controlling the balancing between randomization and greedy behaviour of the agent.
In the softmax action selection method each next action is sampled with a certain probability coming from a *Boltzmann Distribution)*. The probability for action a is computed by:

$$P(a)_{t+1} = \frac{e^{\frac{Q_t(a)}{\tau}}}{\sum_i^N e^{\frac{Q_t(i)}{\tau}}} \tag{1}$$

The softmax is essentially depending on the parameter τ. It has basically the same functioning as ϵ and therefore is controlling how deterministic or random the agent behaves. For increasing τ the numerator goes to one and the next action is picked uniformly. For low values in τ, actions with low value functions result in lower probabilities and hence in a greedy strategy. The parameter is also sometimes called *temperature*.[5].
After selecting an action the agent has to update its believes about it. The update rule is for the value function for an action is defined as:

$$Q(a)_{t+1} = Q(a)_t + \alpha \left[R(a)_t - Q(a)_t \right], \tag{2}$$

where α is the non negative *learning rate* and $R(a)_t$ is the reward of action a at time t. The learning rate is defining how much the reward of the current action is changing the value function of that action. A challenge might be how to initialize the value function.

[5] Another variation is to use the inverse of τ and denote it by θ (see, e.g. [Stojic et al., 2015])

However, for convenience we set them to zero for all options. Those are the basic necessary ingredients to model artificial decision process. In the following we elaborate on further data processing.

3.3 Data handling, unsupervised learning methods and similarities

As mentioned we find two main challenges concerning the data. First, our data have a categorical nature. Furthermore, the learning process and corresponding behavioural changes also impose changing dependence over time.
A well studied approach clustering such data types are *hidden markov models*. A decent study on that can for example be found in [Pamminger, 2007], [Pamminger and Fruhwirth-Schnatter, 2009] and [Pamminger and Fruehwirth-Schnattery, 2010]. However, we are not considering them.
The different applied algorithms operate on distinct distance or similarity concepts. Given our data we have to think carefully think about distance and similarity measures to respect the nature of our data and algorithms.[6] Furthermore, besides considering only raw choices we might try to re-express our data to discriminate them in different ways. One way is to apply *Shannon's Entropy* introduced by [Shannon, 1948]. It is computed by using the empirical probability for given set of choices. For a discrete random variable X with probability p the entropy is defined as [MacKay, 2005]:

$$H(X) := -\sum_{i=1}^{N} p_i \log_2 p_i \tag{3}$$

Entropy in general gives measure on how random the outcome of a random variable is. In context of this paper it serves as an indicator of the randomness in peoples behaviour. If an individual acts deterministically his or her choice will result in low entropy values, while high entropy values indicate random behaviour.[7] Hence, entropy also gives an impression on the exploration and exploitation. Within the entropy framework we consider two types of entropies. For each time step we compute the entropy using choices done so far. We call this *cumulative entropy*. Furthermore, we might want to observe more sharply how individuals adapt their behaviour over time. Considering an experiment with 100 trials we then compute the entropy for different blocks of choices (e.g.

[6] A formal definition of the distance and similarity can be found in the appendix
[7] From an applied perspective once we get values for X such that $p_i = 0$, we set $H(X) = 0$ for convenience [Bishop, 2006]

ten choices). We call this *blockwise entropy*. Mapping choices to an entropy data set aims on discriminating individuals by their level of random behaviour.
A second approach is based on the experimental setting. We know that within framework there are a set of choices, which are disadvantageous for the participant. Following e.g. [Yechiam et al., 2008] or [Ahn et al., 2008], we then compute block wise the ratio of disadvantageous choices. We call this *blockwise disadvantageous choice*. Within that framework we also consider a cumulative version of this ratio, which we call *cumulative disadvantageous choice*

Within our analysis we consider a broad selection of several clustering techniques and similarity concepts. Figure 2 shows an overview of the algorithms and their corresponding distance/similarity requirements. A technical description for all of them is provided in the appendix.

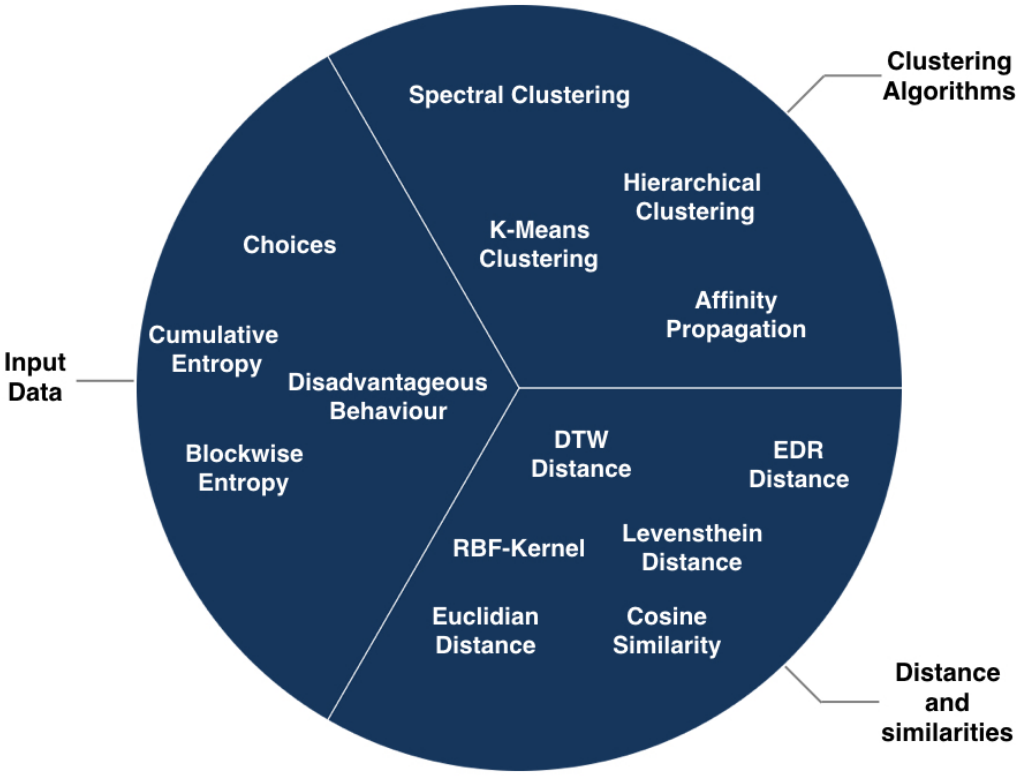


Figure 2: Input data, algorithms and proximity measures

3.4 Simulation setup and results

We observed that the clustering results of our simulation are fluctuating. Hence, to report meaningful results, the following numbers are an average over 20 simulations. We also report the corresponding standard deviations to give an impression on the sensitivity of the clustering results.

Our implementation is quite flexible. We can control both all parameters for each individual agent and their reward sets. Furthermore, we can control the number of trials for each agent and produce customized, multiple asymmetric cluster sizes. We also can set prior values for each agents' value functions.
However, within our experiment we kept the settings more simple. First we fix means and standard deviations for the bandits. Then fix a value for α and run the simulations for different differences in τ. We only consider two different clusters in each settings.
This procedure is repeated for fixed values of τ and varying α. Finally, we again repeat this procedure within a different setting for the bandit means.
This task confronted us with an enormous grid search over different parameters. Note that computing distances is very costly and we to run our simulations multiple times to get an impression of the variability. To keep computation time reasonable, we let the agents perform 100 trials and set the number of agents to 10 for each parameter setting. We ended up using solely the softmax function. The main reason for that decision is that we estimate those parameters later on for the experimental data.

The following table 1 shows a snippet of our simulation results. As described in the last section we have a broad range of similarity measures and algorithms. Therefore, we selected results in such a way, that they give an impression when clustering is working well and such that they indicate the lower bound before clustering works almost perfectly. Naturally, we also found a lot of settings where clustering was not very successful. A detailed overview on numbers are given by tables 7 to 14 in the appendix.[8] In general we find that our algorithms pick up the generated clusters when the differences in tau is sufficiently large. We tested this for different fixed values of $\alpha = \{0.1, 0.5, 1\}$. In general, we can say that if the difference in τ is larger than one, perfect clustering is possible and stable..[9] However, in some settings it also possible earlier.[10]
We repeated this procedure by fixing different values for τ and increase differences in α.

[8] To give an intuition about the numbers one can informally note that numbers below 0.200 - 0.300 do not show any really good clustering patterns.
[9] Standard deviation is equal to one
[10] See for example 10

We are not able to achieve good clustering within that setting. In the summary table we listed an example, where we fixed the difference in α to 0.9 and still results are not satisfying. Only if we fix the difference to an extreme difference of 0.99 we are able to cluster within that setting.
Given the results we can conclude that we are able to discriminate individuals when their behaviour is significantly different with respect to randomness.
Concerning the data type the blockwise entropy seems to be overall the most successful one. In terms of algorithms we see that spectral clustering is in general outperforming the other algorithms. Furthermore, the clustering results based on choices is constantly worse than the other data types.

	Method	Similarity	Choices	Ratio disad. Choices	Entropy	Entropy Block	Concat	Normal. MI	Adj. Rand index	V-Measure
Mu = {0,2,4}										
α = {0.1}	Spectral	RBF				x		0.642 (0.200)	0.653 (0.206)	0.641 (0.200)
τ = {0.1, 0.7}	Spectral	DTW			x			0.580 (0.209)	0.611 (0.205)	0.580 (0.201)
	K-Means	Euclidean					x	0.382 (0.181)	0.309 (0.237)	0.381 (0.180)
	Spectral	Levensthein	x					0.350 (0.114)	0.241 (0.163)	0.347 (0.116)
	Spectral	Euclidean		x				0.306 (0.155)	0.198 (0.179)	0.303 (0.156)
Average								**0.367 (0.150)**	**0.307 (0.195)**	**0.364 (0.151)**
α = {0.5}	Spectral	DTW				x		0.804 (0.212)	0.804 (0.235)	0.803 (0.213)
τ = {0.1, 1.0}	Spectral	EDR			x			0.652 (0.178)	0.663 (0.203)	0.652 (0.178)
	Ward	Euclidean				x		0.621 (0.208)	0.596 (0.249)	0.620 (0.209)
	Spectral	Overlap	x					0.599 (0.355)	0.571 (0.406)	0.597 (0.357)
	K-Means	Euclidean				x		0.562 (0.170)	0.525 (0.220)	0.562 (0.170)
	Spectral	Levensthein	x					0.368 (0.133)	0.270 (0.169)	0.366 (0.134)
	K-Means	Euclidean	x					0.275 (0.144)	0.171 (0.147)	0.272 (0.145)
	K-Means	Euclidean					x	0.399 (0.194)	0.338 (0.224)	0.398 (0.194)
	Spectral	Euclidean		x				0.314 (0.109)	0.197 (0.135)	0.311 (0.110)
Average								**0.408 (0.186)**	**0.351 (0.217)**	**0.406 (0.186)**
α = {1.0}	Spectral	DTW				x		0.792 (0.174)	0.808 (0.170)	0.792 (0.174)
τ = {0.1, 1.0}	Spectral	EDR		x				0.280 (0.198)	0.201 (0.215)	0.278 (0.198)
	Spectral	Levensthein	x					0.276 (0.101)	0.154 (0.113)	0.272 (0.111)
	Spectral	EDR			x			0.236 (0.196)	0.171 (0.184)	0.235 (0.195)
	K-Means	Euclidean					x	0.099 (0.105)	0.052 (0.097)	0.098 (0.104)
Average								**0.266 (0.203)**	**0.204 (0.225)**	**0.263 (0.204)**
Mu = {0,1,2}										
α = {0.1}	Spectral	DTW				x		0.824 (0.166)	0.837 (0.170)	0.824 (0.166)
τ = {0.1, 0.5}	Spectral	DTW			x			0.636 (0.213)	0.661 (0.205)	0.636 (0.213)
	K-Means	Euclidean					x	0.532 (0.182)	0.543 (0.206)	0.531 (0.182)
	Spectral	Levensthein	x					0.411 (0.125)	0.333 (0.160)	0.410 (0.127)
	Spectral	Euclidean		x				0.242 (0.163)	0.132 (0.187)	0.238 (0.165)
Average								**0.394 (0.210)**	**0.338 (0.257)**	**0.391 (0.212)**
α = {0.5}	Spectral	DTW				x		0.769 (0.179)	0.783 (0.185)	0.769 (0.179)
τ = {0.1, 0.5}	Spectral	EDR		x				0.335 (0.172)	0.252 (0.191)	0.333 (0.174)
	Spectral	Levensthein	x					0.324 (0.116)	0.213 (0.128)	0.322 (0.117)
	Spectral	EDR			x			0.221 (0.182)	0.149 (0.161)	0.221 (0.181)
	K-Means	Euclidean					x	0.137 (0.137)	0.110 (0.152)	0.136 (0.137)
Average								**0.248 (0.202)**	**0.196 (0.212)**	**0.246 (0.203)**
α = {1.0}	Spectral	DTW				x		0.618 (0.192)	0.627 (0.201)	0.617 (0.192)
τ = {0.1, 0.7}	Spectral	Levensthein	x					0.252 (0.137)	0.163 (0.147)	0.249 (0.137)
	Spectral	RBF			x			0.215 (0.158)	0.207 (0.188)	0.215 (0.157)
	Spectral	DTW		x				0.205 (0.218)	0.192 (0.247)	0.205 (0.218)
	K-Means	Euclidean					x	0.185 (0.205)	0.165 (0.225)	0.185 (0.205)
Average								**0.248 (0.183)**	**0.220 (0.213)**	**0.246 (0.184)**
Mu = {0,2,4}										
τ = {1.0}	Spectral	DTW				x		0.300 (0.232)	0.273 (0.254)	0.299 (0.232)
α = {0.1,0.9}	Spectral	Cosine		x				0.211 (0.158)	0.159 (0.160)	0.210 (0.158)
	Spectral	Cosine			x			0.191 (0.117)	0.137 (0.127)	0.189 (0.118)
	Spectral	Cosine	x					0.164 (0.110)	0.115 (0.109)	0.162 (0.109)
	K-Means	Euclidean					x	0.138 (0.153)	0.091 (0.176)	0.137 (0.153)
Average								**0.145 (0.067)**	**0.101 (0.075)**	**0.143 (0.067)**

Table 1: Selected simulation results

4 Data Analysis

Our simulation results suggested that we can cluster the decision behaviour under some given constraints. In the following we apply our methods to three different real experiment datasets. The first one is within a multi arm bandit framework, while the remaining are based on data from the Iowa Gambling Task.

4.1 Multi-arm bandit experiment data

The first data set is related to [Stojic et al., 2015]. The data was gathered in a 20-arm bandit online experiment, in which users were compensated with small amount of money in exchange. Four different distributional settings were given to different people. In total the data sets consists of 429 participants divided in 199 female and 229 male participants.[11] The average age 33.04 with standard deviation 11.75. Furthermore, the participants overall have a stronger higher education background. 261 participants have college degree, 39 a graduate degree and PhD respectively. 127 have a high school degree and 2 declined to answer.

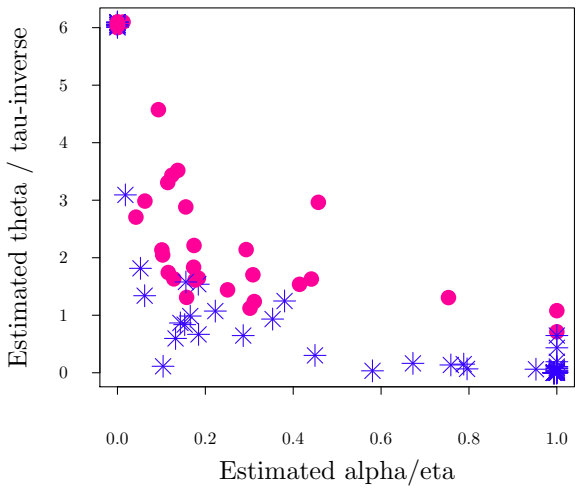
We try to discover clustering in the data. Within our simulation we set parameters, which classify individual subjects. Using the experimental data we come from the other way. The reinforcement learning model is quite closely related to the expectancy valence model from cognitive science. Referring to equation (1) and equation (2) we can try to recover the parameters by optimising those function based on the observed choices.
In the following we analyzed two out of four sub experiments They differ by the level of noise of each arm and therefore the amount of reward distribution overlap.
We maximised the likelihood of the observed choice sequences in order to estimate α and τ (or θ respectively) for each individual in the dataset. This is a simplified version of the expectancy valence model as we do not estimate *attention to gains* parameter. In our setting, the individuals are described in a two-dimensional space, as shown in plots below.
We continue by running our unsupervised learning models on the choice sequences and its by-products: cumulative entropy and blockwise entropy. Initially we hoped to achieve the same two-dimensional clusters given by the EV model, but this is not something we observed. However, some unsupervised models were able to detect linear and non-linear relations between τ inverse and α that were related to different behaviours

[11] One did not wish to answer

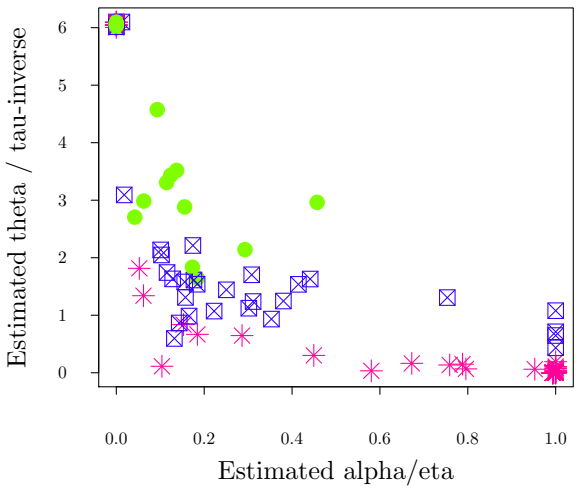
observed in the experiment.

Top two graphs visualize estimates α and τ inverse of individuals in the high-noise experiment. The colours and shapes represent cluster labels produced by our unsupervised models.[12] The top left graph shows one of the best two-cluster models. One could quite successfully draw an almost horizontal linear separator. This coincides with our simulation results where we observe that τ is more significant parameter of behaviour. Furthermore, one could fit a low order polynomial separator the would fit the labels almost perfectly. The top right graph follows the same argument as one could quite successfully fit two parallel almost horizontal separators. Same as in the previous graph, better clustering would be achieved by separators taking the convex form and being stacked in a non-crossing arrangement. Notice that in all four graphs the clustering fails as alpha estimator approaches zero. This was also observed in our simulation when setting alpha close to zero. Two bottom graphs represent data from the low noise experiment. We can see the average τ inverse is higher which represents less randomization by the individuals. This is not surprising as due to lower noise, individuals can more easily differentiate gains between different choices and hence need to explore less. We may deduce that in the low-noise scenario alpha plays a stronger role as it is not possible to fit a satisfying horizontal linear separator. Following the same approach as in the first two graphs, the clustering can be improved by using convex separators.

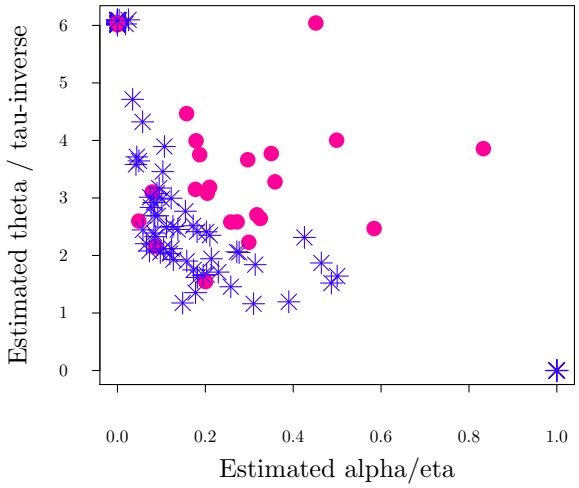
[12] We produce two and three clusters



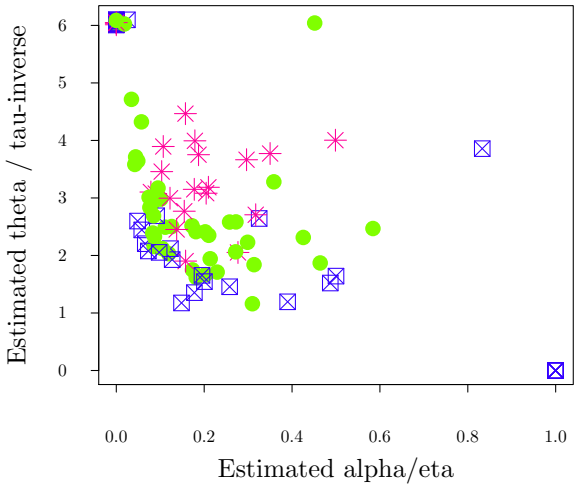
(**a**) Spectral - RBF - blockwise entropy



(**b**) Spectral - DTW - blockwise entropy



(**c**) Ward clustering - blockwise entropy

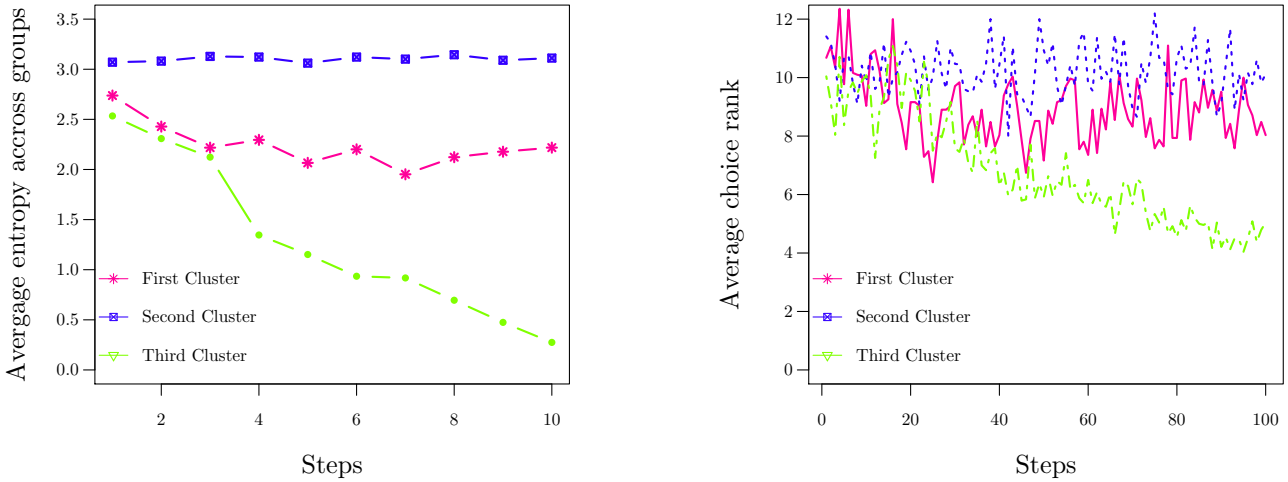


(**d**) Spectral - cosine - blockwise entropy

Figure 3: Clustering on choices vs. model parameter estimation (top sub figures: experiment 1, high noise, bottom: experiment 1, low noise)

The following two graphs represent blockwise entropy and the average choice reward ranking for the three clusters also pictured in the graph b) above. Graph a) below shows

mean blockwise entropy for each of the three clusters. We can observe the following: individuals assigned to all three clusters on average have similar levels of entropy in the block of first 10 draws; individuals in the first cluster maintain the same level of random behaviour throughout all 100 draws while the second cluster slightly reduces the randomization of choices but and then also maintains it at that rate; only the members of the third cluster constantly decrease exploration until the end of the game. Graph b) below shows the average choice rank per cluster, lower the rank higher the reward it brings to individuals. We can observe ideal behaviour by the third cluster as they manage to simultaneously reduce exploration level and still continue to improve average choice rank. It is interesting to see that the first and second cluster do not manage to improve the choice ranking as the experiment progresses.



(**a**) Blockwise entropy by cluster

(**b**) Average choice rank by cluster

Figure 4: Blockwise entropy and average choice rank based on clustering results for experiment 1

4.2 IGT data

As mentioned in the introduction the Iowa Gambling Task is a common tool to monitor human decision making process. Within the test participants try to maximize rewards by choosing sequentially cards from four different decks with different reward structures. There are existing several slightly different versions of test. In general the setting is such

that two decks have distributions with negative expectations, while two have positive. However, each deck has its own variance and a set of profits. [Steingroever et al., 2015] provides a publicly available data collection for several sources also giving an overview of some different notions of the test.

4.2.1 Prison data

We were provided with experimental data from [Yechiam et al., 2008] and we try to apply our methods to cluster different groups in the data. The participants had to perform a modified version of the Iowa gambling task, where reward structure of the decks are changing over time.
Our data cover 96 individuals with different criminal profile. Within this data we don't have a control group. Given that participants performed a different version of the test we could add a control data from publicly available data sets (see [Steingroever et al., 2015]).
Table **??** gives a broad summary of some demographics of the participants. The samples for each groups are not balanced.

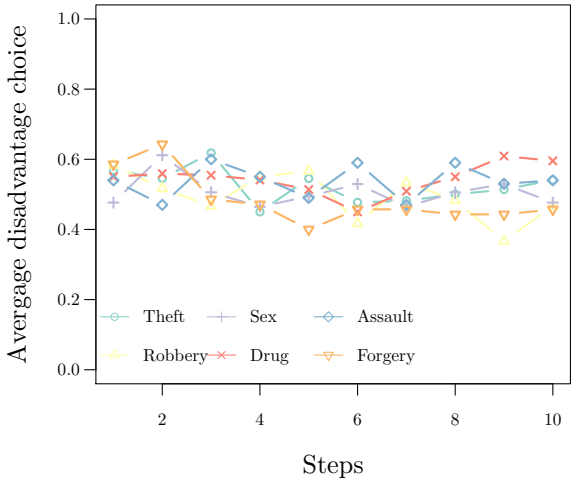
ID	Criminal profile	Count	Age	TABE Score	Education	Beta IQ
1	Theft/Burglary	22	25.36 (7.03)	11.09 (1.29)	7.38 (3.34)	92.91 (14.37)
2	Robbery	6	24.17 (9.83)	11.00 (0.63)	9.22 (3.30)	96.50 (7.58)
3	Sex	17	33.41 (13.59)	10.97 (1.47)	9.15 (2.98)	99.65 (11.74)
4	Drug	22	30.91 (10.11)	11.64 (1.85)	9.06 (2.70)	100.36 (12.92)
5	OWI	4	38.75 (7.27)	10.88 (1.93)	7.12 (1.17)	94.25 (10.40)
6	Assault	10	27.20 (8.77)	12.30 (2.41)	7.62 (2.28)	94.50 (11.29)
7	Escape/ Failure To Appear	4	2.008 (5.60)	11.00 (1.35)	7.78 (3.21)	96.50 (14.18)
8	Vandalism	1	18.00 (NA)	11.00 (NA)	9.40 (NA)	90.00 (NA)
9	Forgery	7	34.57 (13.14)	10.93 (5.15)	9.83 (3.82)	100.71 (11.01)
10	Probabiton	1	38.00 (NA)	12.00 (NA)	6.30 (NA)	92.00 (NA)
11	Other	2	35.00 (9.90)	11.50 (0.00)	9.20 (4.67)	95.00 (5.66)

Table 2: Summary prison data (means with standard deviation in parenthesis)

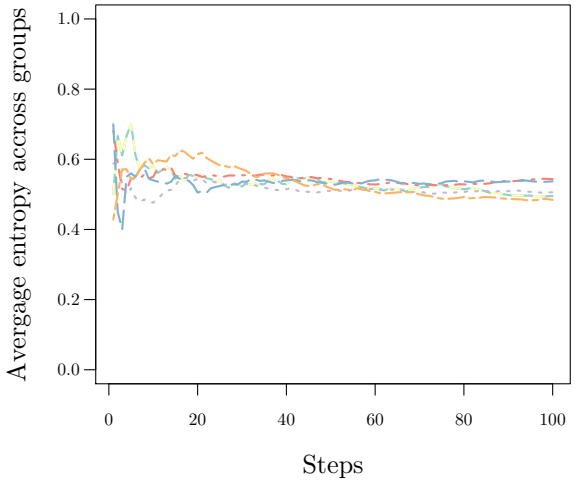
In figure 5 we display some of the behavioural data. Given the number of participants we averaged them by criminal profile to see if we can find differences in across groups. We excluded "OWI", "escape", "vandalism", "probation" and "others" from this figures since the number of observations is fairly small. This also gives a more clear insight in the data.

Figure 5 (a) show the blockwise ratio of disadvantageous choices and (b) the development of cumulative disadvantageous choices. A clear separation can not be observed from those plots. From figure (b) we can at most see a slight decreasing trend for forgery, sex robbery and theft, while drug and stay slightly higher.[13]
Figure 5 (d) shows the average cumulative entropy averaged across groups. We observe a random of exploring behaviour independent from group affiliation. Notably, people with assault/murder profile are show constantly the highest value. We can see the same situation from the blockwise entropy from figure (c).

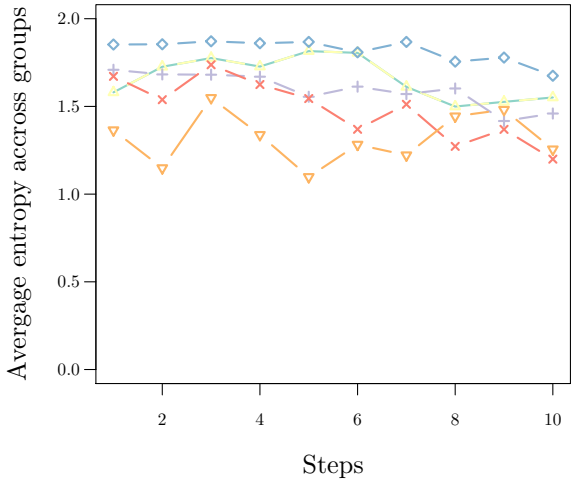
[13]Interesting in that context might be that [**?**] found a constant disadvantageous behaviour for cocaine abusers (see next section)



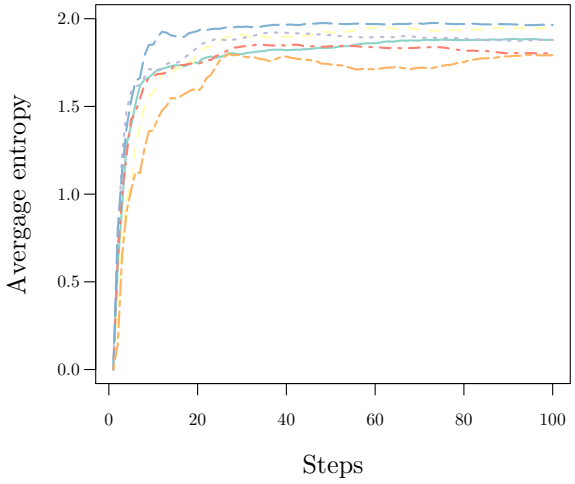
(**a**) Block wise picks from disadvantageous deck



(**b**) Cumulative picks from disadvantageous deck



(**c**) Block wise entropy



(**d**) Cumulative entropy

Figure 5: Disadvantageous behaviour and entropy averaged by criminal profile

[Yechiam et al., 2008] found three clusters by using the attention to recent outcomes(ARO) and attention to gains(AG) parameters from the expectancy valance model. The most distinct group were the robbery convicts with the only negative attention to gains mean at -0.36 and the highest attention to recent outcomes mean 0.57. The second cluster is made of assault and murder convicts with ARO of 0.26 and AG of 0.1. The third

cluster is formed of all the remaining prisoner groups with ARO means between 0 and -0.1 and AG between 0.1 and 0.2.

Based on those findings and what we observed in figure 5 we decided to cluster convicted assault-murder and robbery individuals against the other criminal groups. The following table depicts selected resulting clustering performance.[14] We find the most accurate clustering among all pairwise experiments between assault/murders and forgery, both on entropy and cumulative bad choices, which is suggested by the descriptive findings from figure 5. The listed algorithms performed remarkably better than the average performance. Furthermore, we managed to find some weaker clustering results between robbery. In contrast to [Yechiam et al., 2008] we manged to isolate rather assault/murder criminals from the rest than robbery.

Groups	Method	Similarity	C	CBC	BBC	E	EB	CC	NMI	ARI	VM
2 vs. 9	Spectral	EDR				x			0.382 (0.000)	0.235 (0.000)	0.382 (0.000)
	Spectral	EDR			x				0.232 (0.000)	0.226 (0.000)	0.232 (0.000)
	Spectral	Cosine					x		0.197 (0.000)	0.008 (0.000)	0.191 (0.000)
	Spectral	RBF		x					0.134 (0.000)	0.075 (0.000)	0.134 (0.000)
	K-Means	Euclidean						x	0.116 (0.000)	-	0.105 (0.000)
	K-Means	Euclidean	x						0.111 (0.022)	-	0.100 (0.020)
Average									**0.096 (0.074)**	**-**	**0.093 (0.073)**
6 vs. 9	Ward	Euclidean					x		0.561 (0.000)	0.560 (0,000)	0.560 (0,000)
	Spectral	DTW				x			0.435 (0.000)	0.387 (0.000)	0.432 (0.000)
	Spectral	DTW		x					0.435 (0.000)	0.387 (0.000)	0.432 (0.000)
	Spectral	EDR			x				0.333 (0.000)	0.381 (0.000)	0.333 (0.000)
	Spectral	Overlap	x						0.111 (0.000)	0.118 (0.000)	0.111 (0.000)
	K-Means	Euclidean						x	0.106 (0.066)	0.033 (0.054)	0.094 (0.059)
Average									**0.249 (0.167)**	**0.202 (0.189)**	**0.244 (0.169)**

Table 3: Selected results for prison data clustering

4.2.2 Cocaine Abusers data

Finally we study data from several cocaine abusers. There are 12 addict individuals performing the IGT. The control group consist out of 14 participants. Candidates among the drug abusers were selected as active users with additional drug abusing past, but without any known additional mental illness [Stout et al., 2004]. Table 4 gives a summary of demographic profile.

Table 4: Demographic summary of cocaine abusers (means with standard deviations in parenthesis)

Demographic indicator	Drug abusers	Control Group
Share of men	79%	100%
Age	36.90 (10.30)	30.00 (6.10)
Estimated IQ	93.70 (10.30)	105.00 (7.62)

[14] The remaining results can be found in table **??** and **??** in the appendix

Within the gambling task framework they found that cocaine abusers choose persistently cards from disadvantageous decks. The effect is still present after controlling for the low IQ score.
We cluster cocaine abusers against the control group. The following table 5 depicts results of the clustering. Again we averaged over 20 simulations to report overall clustering performance. However as depicted the data set is fairly small and results are quite stable.
Again our best clustering is achieved using block wise entropy, besides the listed K-Means algorithm, spectral clustering based on both cosine similarity and with rbf kernel achieved the same results. However, the results for this data set are rather low and we could not find good clustering between the control group and cocaine abusers. This also is indicated by the average over all applied methods.

Method	**Similarity**	**C**[0]	**CBC**[1]	**BBC**[2]	**E**[3]	**EB**[4]	**CC**[5]	**NMI**[6]	**ARI**[6]	**VM**[7]
K-Means [*]	Euclidean					x		0.270 (0.000)	0.262 (0.000)	0.270 (0.000)
Ward	Euclidean		x					0.270 (0.000)	0.262 (0.000)	0.270 (0.000)
K-Means	Euclidean						x	0.209 (0.030)	0.171 (0.043)	0.208 (0.030)
Spectral	Levenstein	x						0.178 (0.000)	0.181 (0.000)	0.178 (0.000)
Spectral	Cosine			x				0.171 (0.000)	0.117 (0.000)	0.171 (0.000)
Spectral	DTW				x			0.042 (0,000)	0.014 (0.000)	0.042 (0.000)
Average[†]								**0.139 (0.002)**	**0.104 (0.003)**	**0.138 (0.002)**

Notes

[0] Clustering based on choices participant did
[1] Clustering based on cumulative disadvantageous choice of participants
[2] Clustering based on block wise disadvantageous choice. Block size = 10
[3] Clustering based on cumulative entropy
[4] Clustering based on block wise entropy. Block size = 10
[5] Clustering based on entropy and choices concatenated for each participant
[6] Normalised mutual infrormation score (description see appendix C)
[7] Adjusted rand index (description see appendix C)
[8] V-Measure (description see appendix C)
[*] Spectral Clustering on block wise entropy with RBF kernel and cosine similarity produced the same results
[†] Average over all algorithms including the ones displayed in the table

Table 5: Clustering results for cocaine abusers vs. control group

5 Conclusion and possible extensions

In this paper we wanted to evaluate whether unsupervised learning methods can be used to cluster individuals based on their behaviour and produce similar results to the convenient cognitive models.
We first set up a reinforcement framework and studied clustering performance. We found that we can cluster behaviour if the difference between greedy and explorative behaviour is sufficiently large. We were not able to cluster well over different learning

rates for fixed exploring behaviours.
In the following we tried to cluster several data sets. Clustering on the multiarm bandit experiment data set showed that we found different clustering structure as produced by parameter estimation for the data. Furthermore, we applied unsupervised learning to two different IGT based experiment data. For the data with participants with different criminal profiles, we found that assault murders were clustering the best vs. forgery group. In general we can't find as significant clustering as the underlying study by [Yechiam et al., 2008]. Finally we tried to cluster decision data from cocaine abusers vs. a normal control group. We only found weak clustering between those two groups. Our analysis so far showed that people are not perfectly separate themselves according to their true group affiliation. In general we assume that healthy participants and those we assumed to have decision making deficits show significantly different behaviour. However, we observe that their behaviour seems to be quite similar given data produced in this type of experiment. Furthermore , in most of the applied unsupervised techniques we as analysts have to set the number of clusters we assume to be in the data (so in our case two: for control group and for patients with habits). We applied another algorithm called affinity propagation, which identifies the number of clusters itself (a closer description can be found in[Brusco and Köhn, 2008]). In general we find that the algorithm is assigning many more clusters to our data than the number we assume to be there. This suggests that there may be more natural clusters in the data instead of simply splitting by healthy and ill labels. This can be addressed by Bayesian nonparametric models. [Gershman and Blei, 2012]
We can conclude that unsupervised learning might have some opportunities. However given our experimental results and the insights from the previous paragraph we can conclude that our data are to overlapping and our similarity measures are not able to identify the different groups.

6 Acknowledgments

First of all we want to thank our supervisor Hrvoje Stojic for his valuable input and support. Furthermore, we want to thank him for providing us with the data from his experiment. Furthermore, we want to thank the authors from [Stout et al., 2004] and [Yechiam et al., 2008] for providing us with the experimental data.

List of Literature

[Ahn et al., 2008] Ahn, W.-K., Busemeyer, J. R., Wagenmakers, E.-J., and Stout, J. C. (2008). Comparison of decision learning models using the generalization criterion method. *Cognitive Science*, 32(8):1376–1402.

[Bechara et al., 1997] Bechara, A., Damasio, H., Tranel, D., and Damasio, A. R. (1997). Deciding Advantageously Before Knowing the Advantageous Strategy. *Science*, 275:1293–1295.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[Blair et al., 2001] Blair, R. J., Colledge, E., and Mitchell, D. G. V. (2001). Somatic markers and response reversal: is there orbitofrontal cortex dysfunction in boys with psychopathic tendencies? *Journal of Abnormal Child Psychology*, 29(6):499–511.

[Boriah et al., 2008] Boriah, S., Chandola, V., and Kumar, V. (2008). Similarity measures for categorical data: A comparative evaluation. *SIAM*.

[Brambilla et al., 2012] Brambilla, P., Perlini, C., Bellani, M., Tomelleri, L., Ferro, a., Cerruti, S., Marinelli, V., Rambaldelli, G., Christodoulou, T., Jogia, J., Dima, D., Tansella, M., Balestrieri, M., and Frangou, S. (2012). Increased salience of gains versus decreased associative learning differentiate bipolar disorder from schizophrenia during incentive decision making. *Psychological Medicine*, pages 1–10.

[Brusco and Köhn, 2008] Brusco, M. J. and Köhn, H.-F. (2008). Clustering by passing messages between data points. *Science (New York, N.Y.)*, 319(5864):726; author reply 726.

[Busemeyer and Stout, 2002] Busemeyer, J. R. and Stout, J. C. (2002). L8 A contribution of cognitive decision models to clinical assessment: decomposing performance on the Bechara gambling task. *Psychological assessment*, 14(3):253–262.

[Cavedini et al., 2002] Cavedini, P., Riboldi, G., Keller, R., D'Annucci, A., and Bellodi, L. (2002). Frontal lobe dysfunction in pathological gambling patients. *Biological Psychiatry*, 51(4):334–341.

[Everitt et al., 2009] Everitt, B. S., Landau, S., and Leese, M. (2009). *Cluster Analysis*. Wiley Publishing, 4th edition.

[Fratev et al., 1979] Fratev, F., Polansky, O. E., Mehlhorn, A., and Monev, V. (1979). Application of distance and similarity measures. The comparison of molecular electronic structures in arbitrary electronic states. *Journal of Molecular Structure*, 56(C):245–253.

[Fridberg et al., 2010] Fridberg, D. J., Queller, S., Ahn, W. Y., Kim, W., Bishara, A. J., Busemeyer, J. R., Porrino, L., and Stout, J. C. (2010). Cognitive mechanisms underlying risky decision-making in chronic cannabis users. *Journal of Mathematical Psychology*, 54(1):28–38.

[Gabadinho et al., 2009] Gabadinho, A., Ritschard, G., Studer, M., and Nicolas, S. M. (2009). SUMMARIZING SETS OF CATEGORICAL SEQUENCES Selecting and visualizing representative sequences. *Methods*, pages 6–8.

[Gershman and Blei, 2012] Gershman, S. J. and Blei, D. M. (2012). A tutorial on Bayesian nonparametric models. *Journal of Mathematical Psychology*, 56(1):1–12.

[Gullo and Stieger, 2011] Gullo, M. J. and Stieger, A. A. (2011). Anticipatory stress restores decision-making deficits in heavy drinkers by increasing sensitivity to losses. *Drug and Alcohol Dependence*, 117(2-3):204–210.

[Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA.

[Hubert and Arabie, 1985] Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218.

[Johnson et al., 2006] Johnson, S. a., Yechiam, E., Murphy, R. R., Queller, S., and Stout, J. C. (2006). Motivational processes and autonomic responsivity in Asperger's disorder: evidence from the Iowa Gambling Task. *Journal of the International Neuropsychological Society*, 12:668–676.

[Luxburg, 2007] Luxburg, U. V. (2007). A Tutorial on Spectral Clustering. (March):1–32.

[MacKay, 2005] MacKay, D. J. C. (2005). *Information Theory, Inference, and Learning Algorithms David J.C. MacKay*, volume 100.

[Martino et al., 2007] Martino, D. J., Bucay, D., Butman, J. T., and Allegri, R. F. (2007). Neuropsychological frontal impairments and negative symptoms in schizophrenia. *Psychiatry Research*, 152(2-3):121–128.

[Morzy et al.,] Morzy, T., Wojciechowski, M., and Zakrzewicz, M. Clustering sequences of categorical values.

[Murphy, 2012] Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press.

[Nirit Agaya, Eldad Yechiama, Ziv Carmelb, 2010] Nirit Agaya, Eldad Yechiama, Ziv Carmelb, Y. L. (2010). Non-specific effects of Methylphenidate (Ritalin) on cognitive ability and decision- making of ADHD and healthy adults. (972).

[Pamminger, 2007] Pamminger, C. (2007). Bayesian Clustering of Categorical Time Series Using Finite Mixtures of Markov Chain Models Bayesian Clustering of Categorical Time Series Using Finite Mixtures of Markov Chain Models.

[Pamminger and Fruehwirth-Schnattery, 2010] Pamminger, C. and Fruehwirth-Schnattery, S. (2010). Model-based clustering of categorical time series. *Bayesian Analysis*, 5(2):345–368.

[Pamminger and Fruhwirth-Schnatter, 2009] Pamminger, C. and Fruhwirth-Schnatter, S. (2009). Bayesian Clustering of Categorical Time Series Using Finite Mixtures of Markov Chain Models. 43(0907).

[Ren et al., 2011] Ren, J., Cao, S., and Hu, C. (2011). A Hierarchical Clustering Algorithm Based on Dynamic Programming for Categorical Sequences. 5:1575–1581.

[Richter et al.,] Richter, C., Luboschik, M., and Martin, R. Sequencing of Categorical Time Series.

[Rosenberg and Hirschberg, 2007] Rosenberg, A. and Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 1(June):410–420.

[Serra and Arcos, 2014] Serra, J. and Arcos, J. L. (2014). An empirical evaluation of similarity measures for time series classification. *Knowledge-Based Systems*, 67:305–314.

[Shannon, 1948] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(July 1928):379–423.

[Shirali and Vasudeva, 2005] Shirali, S. and Vasudeva, H. L. (2005). *Metric Spaces*. Springer.

[Steingroever et al., 2015] Steingroever, H., Fridberg, D. J., Horstmann, A., Kjome, K. L., Kumari, V., Lane, S. D., Maia, T. V., Mcclelland, J. L., Pachur, T., Premkumar, P., Stout, J. C., and Wetzels, R. (2015). Data from 617 Healthy Participants Performing the Iowa Gambling Task : A "Many Labs" Collaboration. *Journal of Open Psychology Data*.

[Steingroever et al., 2013] Steingroever, H., Wetzels, R., and Wagenmakers, E.-J. (2013). Absolute Performance of Reinforcement-Learning Models for the Iowa Gambling Task. *Decision*.

[Stojic et al., 2015] Stojic, H., Analytis, P. P., and Speekenbrink, M. (2015). Human behavior in contextual multi-armed bandit problems. *Proceedings of the 37th Annual Conference of the Cognitive Science Society*, 1:1–6.

[Stout et al., 2004] Stout, J. C., Busemeyer, J. R., Lin, A., Grant, S. J., and Bonson, K. R. (2004). Cognitive modeling analysis of decision-making processes in cocaine abusers. *Psychonomic Bulletin & Review*, 11(4):742–747.

[Sutton and Barto, 2012] Sutton, R. and Barto, A. (2012). Reinforcement Learning : An Introduction.

[Vinh et al., 2010] Vinh, N. X., Epps, J., and Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research*, 11:2837–2854.

[Wang et al., 2013] Wang, X., Mueen, A., Ding, H., Trajcevski, G., Scheuermann, P., and Keogh, E. (2013). Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery*, 26(2):275–309.

[Wang and Ruhe, 2007] Wang, Y. and Ruhe, G. (2007). The cognitive process of decision making. *International Journal of Cognitive Informatics and Natural Intelligence*, 1(2):73–85.

[Yechiam et al., 2008] Yechiam, E., Kanz, J. E., Bechara, A., Stout, J. C., Busemeyer, J. R., Altmeier, E. M., and Paulsen, J. S. (2008). Neurocognitive deficits related to poor decision making in people behind bars. *Psychonomic Bulletin & Review*, 15(1):44–51.

Appendix

A Metrics and Similarities

This part of the appendix formally defines metrics and similarities and dissimilarities (proximity) used in this paper. We first define some basic general concepts followed by a description of the applied distance and similarity concepts.

A.1 Distances vs. Similarities

Let $\mathcal{X}$ be a dataset and let $\boldsymbol{x_i}, \boldsymbol{x_j}$ be two datapoints, such that $\boldsymbol{x_i}, \boldsymbol{x_j} \in \mathcal{X}$.

A distance function assign for pairs a points a non negative real number as distance. $d : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}_0^+$. Furthermore, the following points have to be fulfilled [Shirali and Vasudeva, 2005].

1. $d(\boldsymbol{x_i}, \boldsymbol{x_j}) \geq 0$
2. $d(\boldsymbol{x_i}, \boldsymbol{x_i}) \geq 0$
3. $d(\boldsymbol{x_i}, \boldsymbol{x_j}) = d(\boldsymbol{x_j}, \boldsymbol{x_i})$
4. $d(\boldsymbol{x_i}, \boldsymbol{x_j}) \leq d(\boldsymbol{x_i}, \boldsymbol{x_j}) + d(\boldsymbol{x_j}, \boldsymbol{x_i})$

A distance can be seen as a measure for dissimilarity of two points [Everitt et al., 2009]. Besides distance some algorithms operate on a *similarity* matrix. Formally a similarity is a function $S : \mathcal{X} \times \mathcal{X} \mapsto [0, 1]$. Also for similarity we can define the following properties [Fratev et al., 1979]:

1. $0 \leq S(\boldsymbol{x_i}, \boldsymbol{x_j}) \leq 1, \text{for } i \neq j$
2. $S(\boldsymbol{x_i}, \boldsymbol{x_i}) = 1$
3. $S(\boldsymbol{x_i}, \boldsymbol{x_j}) = S(\boldsymbol{x_j}, \boldsymbol{x_i})$

Once we have computed distance or a similarity we can compute for two data points we can use this information to transform it to a similarity or the distance vice versa [Boriah et al., 2008, page 4]:

$$S(\boldsymbol{x_i}, \boldsymbol{x_j}) = \frac{1}{1 + d(\boldsymbol{x_i}, \boldsymbol{x_j})} \quad \Leftrightarrow \quad d(\boldsymbol{x_i}, \boldsymbol{x_j}) = \frac{1}{S(\boldsymbol{x_i}, \boldsymbol{x_j})} - 1 \tag{4}$$

A.2 General Similarity measures

The default similarity measures used by many machine learning libraries (e.g. python sci-kit) is the Gaussian kernel of RBF-Kernel. This kernel function can be also be seen as a similarity. For two point is is defined as [Murphy, 2012, page 480]

$$K_{RBF}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \exp\left(-\frac{||\mathbf{x} - \mathbf{y}||}{2\sigma^2}\right) \tag{5}$$

Another common similarity measure is cosine similarity. It is expressing the angle between two vectors and is defined as [ibidem, page 480]:

$$S_{cos}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \frac{\mathbf{x}^T\mathbf{x}}{||\mathbf{x}||||\mathbf{y}||} \tag{6}$$

A.3 Similarity measures for categorical data

Referring to equation 9 we define a simple measure for the categorical aspect of the data. Note that [SOURCE] defines the simplest measure for categorical data. However, since the probability that two people behave exactly the same in our context is arguably zero we modify this concept slightly. So we relax that and define the overlap similarity just as the count of overlapping instances. This serves as a benchmark similarity for categorical data.

$$d_O(\boldsymbol{x}, \boldsymbol{y}) := \sum_{i=1}^{N} \mathbb{1}_{(x_i = y_i)} \tag{7}$$

Furthermore, Levenstein distance (or edit distance) is a basic way to measure similarity between categorical sequences [Richter et al., , page 1] or [Gabadinho et al., 2009, page 2], however some authors stating that it might perform poorly in their task [Ren et al., 2011, page 3] or a poor measure at all [Morzy et al., , page 5]

It is relying on solving a dynamic programming problem. Define $D_{0,j} = j$ and $D_{i,0} = i$:

$$D_{\mathbf{x},\mathbf{y}}(i,j) = \min \begin{cases} D_{i-1,j} + 1 \\ D_{i,j-1} + 1 \\ D_{i-1,j-1} + \mathbb{1}_{(x_i = y_i)} \end{cases} \tag{8}$$

A.4 Distance measures for time series data

We use three different similarity measures for time series. E.g. [Wang et al., 2013] or [Serra and Arcos, 2014] provide an overview and empirical evaluation on common similarity measures for time series. The following definitions are taken from the latter. Empirical research suggest that simple euclidean distance for time series performs quite well and is hard to beat. Hence, the first distance measure for time series is simply euclidean distance between two time series. They might be converted to similarity based on equation 4. Let $\boldsymbol{x}, \boldsymbol{y}$ be two time series over N-periods. Then the L_2 distance between two time series is define as:

$$d_E(\boldsymbol{x}, \boldsymbol{y}) := \sqrt{\left(\sum_{i=1}^{N} (x_i - y_i)^2 \right)} \tag{9}$$

We also considered the edit distance on real sequences (EDR). It is basically an real valued version to the Levensthein distance. It also relies on solving a dynamic programing problem. For $i = 1, \ldots, M$ and $j = 1, \ldots, N$ we have to compute.[15]

$$D_{i,j} = \begin{cases} D_{i-1,j-1} & \text{,if } m(x_i, y_j) = 1 \\ 1 + \min(D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}) & \text{, if } m(x_i, y_j) = 0 \end{cases} \tag{10}$$

where, $m(\cdot, \cdots)$ is the matching function. For $x_i and y_j$ it is defined as:

$$m(xi, yj) = \Theta(\epsilon - f(x_i, y_j)) \tag{11}$$

where ϵ is scalar, such that $\mathbb{R}_0^+$. $\Theta(\cdot)$ denotes the Heaviside step function and is defined as $\Theta(z) = 1$ if $z \geq 0$.

Finally we considered dynamic time warp (DTW), which is probably one of the most successful distance measures for time series. It is also relies on dynamic programming. A description can be found in [Wang et al., 2013].

[15] The time series can have different length. However, in our case $N = M$

B Algorithms

B.1 K-Means Clustering

The following algorithm 1 describes the K-means clustering algorithm. The basic idea is that one assigns randomly centroids in the space and compute for each point the distance to The pseudo code for the algorithm comes from [Murphy, 2012]

input : $\boldsymbol{m}_k$

repeat

- Assign each data point to its closest cluster center: $z_i = \arg\min_k ||\boldsymbol{x}_i - \boldsymbol{\mu}_k||_2^2$;
- Update each cluster center by computing the mean of all points assigned to it: $\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{i:z_i=k} \boldsymbol{x_i}$;

until *converged*;

Algorithm 1: K-Means clustering

B.2 Hierarchical Clustering - Agglomerative

Algorithm 2 shows the pseudo code for agglomerative clustering. Ward is a notion of that algorithm. The pseudo code can be found in [Murphy, 2012]

input : initialize clusters as singletons:
output: Importance values for each node v

for $i \leftarrow 1$ ***to*** n **do**
  $C_i \leftarrow \{i\}$;
end
Initialize set of clusters for merging:
$S \leftarrow \{1, \ldots, n\}$;
repeat
  Pick 2 most similar clusters to merge:;
  $(j, k) \leftarrow \arg\min_{j,k \in S} d_{j,k}$;
  Create new cluster $C_\ell \leftarrow C_j \cup C_k$ Mark j and k as unavailable: $S \leftarrow S \setminus \{j, k\}$;
  if $C_l \neq \{1, \ldots, n\}$ **then**
    Mark ℓ as available: $S \leftarrow S \cup \{\ell\}$;
  end
  for $i \in S$ **do**
    Update dissimilarity matrix $d(i, \ell)$;
  end
  no more clusters are available for merging;
until *convergence*;

Algorithm 2: Agglomerative Clustering

B.3 Spectral Clustering

For the spectral clustering algorithm we formally introduce some graph notation. If not stated otherwise the following derivation follows [Luxburg, 2007]. In the following we consider a weighted and simple undirected graph.

$$G = \{V, E\} \tag{12}$$
$$V = \{v_1, \ldots, v_n\} \tag{13}$$
$$E = \{e_1, \ldots, e_n\} \tag{14}$$

Furthermore let the graph has a weighted and symetric ($|V| \times |V|$) adjacency matrix, such that:

$$\boldsymbol{W} = \begin{cases} w_{i,j} & \text{,if } v_i v_j \in E \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

The *degree* of a node is defined as the sum of edge weights of connected nodes. Formally we denote the degree of node i as:

$$d_i := \sum_{j=1}^{n} w_{ij} = \sum_{i=1}^{n} w_{ij} \tag{16}$$

Using the last expression we define matrix $\boldsymbol{D}$ as the diagonal matrix of the degress

$$\boldsymbol{D} := diag(\boldsymbol{d}) \tag{17}$$

The algorithm works on the *Laplacian* matrix defined by:

$$\boldsymbol{L} := \boldsymbol{D} - \boldsymbol{W} \tag{18}$$

Former versions of the algorithm are applied on the graph laplcian. However, there were proposed newer versions using the so called *normalized laplacian*. Since also the python version is using this package we will focus on this version of the algorithm. Following that the normalized graph laplacian is defined as:

$$\boldsymbol{L}_{norm} := \boldsymbol{D}^{1/2} \boldsymbol{L} \boldsymbol{D}^{1/2} = \boldsymbol{I} - \boldsymbol{D}^{1/2} \boldsymbol{W} \boldsymbol{D}^{1/2} \tag{19}$$

IncMargin1em

input : Similarity matrix $S \in \mathbb{R}^{n \times n}$ and number of clusters k
output: Clusters $A_1, \ldots, A_k$ with $A_i = \{j | y_i \in C_i\}$

1. Construct a similarity graph. Let **W** be its weighted adjacency matrix
2. Compute the normalized Laplacian **L**.
3. Compute the first k eigenvectors $u_1, \ldots, u_k$ of the generalized eigenproblem **Lu** $= \lambda$**Du**.
4. Let $U \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $u_1, \ldots, u_k$ as columns.
5. **for** $i = 1, \ldots, n$, let y_i be the vector corresponding to the $i - th$ row of U
6. Cluster the points $(y_i)_{i=1,\ldots,n}$ in $\mathbb{R}^k$ with the K-Means algorithm into clusters $C_1, \ldots, C_k$

Algorithm 3: Spectral clustering

C Clustering Evaluation

In this section we introduce some clustering metrics. Evaluating clustering and other unsupervised methods performance requires a more complex approach since the true labels are unknown. The algorithm assigns each point to a cluster. Despite we generated the "true" clusters they might not be comparable. A simple example we might consider the following situation. Let y denote the labels of data the data and y' the corresponding prediction such that $y, y' \in \{0, 1\}$. In a small example let our data points be like $y = (1, 1, 0, 0)$ and the corresponding prediction $y' = (0, 0, 1, 1)$. Obviously the clustering worked perfectly, however comparing "labels" would produce an accuracy of zero.

There a several clustering metrics, which respect such a situation. We consider several information based metrics. Most of the measures use some sort of entropy. The following concepts can be found in [Rosenberg and Hirschberg, 2007] and [Vinh et al., 2010]. Additional reading is [Hubert and Arabie, 1985]. First we might introduce the contingency table.

	V_1	V_2	$\ldots$	V_c	Σ
U_1	$n_{1,1}$	$n_{1,2}$	$\ldots$		a_1
U_2	$n_{2,1}$	$\ddots$			a_1
$\vdots$	$n_{1,1}$	$\ldots$			a_1
U_R	$n_{1,1}$	$\ldots$			a_1
	b_1	b_2	$\ldots$	b_c	N

Table 6: Contigency Table

N_{11}**:** Number of pairs that are in the same cluster in both v and u
N_{00}**:** Number of pairs that are in different clusters in both v and u
N_{01}**:** Number of pairs that are in the same cluster in u but different in v

N_{10}**:** Number of pairs that are in the same cluster in v but different in u

N_{11} and N_{00} can be interpreted as agreements and N_{01} and N_{10} as disagreements between v and u. We the define the Rand index as:

$$RI(u,v) = \frac{N_{00} + N_{11}}{\binom{N}{2}} = \frac{N_{00} + N_{11}}{N_{00} + N_{11} + N_{01} + N_{10}} \tag{20}$$

Rand index can take values between 0 and 1, where 1 indicates perfect clustering. One of the drawbacks of the Rand index is that the expected value of two random allocations does not take a constant value. Adjusted Rand index proposed by proposed by [Hubert and Arabie, 1985] is also upper bounded by 1, can take negative values and has the expected value of zero.

$$\begin{aligned} ARI(u,v) &= \frac{\text{index} - \text{expected index}}{\text{maximum index} - \text{expected index}} \\ &= \frac{2\left(N_{00}N_{11} - N_{01}N_{10}\right)}{\left(N_{00} + N_{01}\right)\left(N_{01} + N_{11}\right) + \left(N_{00} + N_{10}\right)\left(N_{10} + N_{11}\right)} \end{aligned}$$

Mutual information score measures the similarity between two random variables. It estimates the amount of information known about one random variable after observing the other, it determines the joint distribution. Therefore, mutual information is also closely related to the entropy of a random variable. Normalized mutual info score is a normalization of MI in order to scale the score values between zero and one. The information score is:

$$I(u,v) = \sum_{i=1}^{R}\sum_{j=1}^{C} \frac{n_{i,j}}{N} \log \frac{n_{i,j}/N}{a_i b_j / N} \tag{21}$$

$$\tag{22}$$

The normalization can be done in different ways. This is one based on max:

$$NMI_{max}(u,v) = \frac{I(u,v)}{\max\left(H(u), H(v)\right)} \tag{23}$$

A clustering result satisfies homogeneity if each cluster contains observations that belong to a single class. A result also satisfies completeness if all observations that belong to the same class have been assigned to the same cluster. V measure score expresses the harmonic mean between homogeneity and completeness measures. The homogeneity is defined as:

$$h = \begin{cases} 1 & \text{,if } H(u,v) = 0 \\ 1 - \frac{H(u|v)}{H(u)} & \text{otherwise} \end{cases} \tag{24}$$

The completeness is defined as:

$$c = \begin{cases} 1 & \text{,if } H(v,u) = 0 \\ 1 - \frac{H(v|u)}{H(v)} & \text{otherwise} \end{cases} \tag{25}$$

Using both we can define the V-Measure:

$$V_\beta = \frac{(1+\beta)hc}{\beta h + c} \tag{26}$$

D Simulation Data

Tau	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1, 0.3}	Spectral	DTW				x		0.136 (0.134)	0.094 (0.146)	0.135 (0.134)
	Spectral	EDR			x			0.134 (0.135)	0.097 (0.130)	0.134 (0.135)
	Spectral	Levensthein	x					0.093 (0.134)	0.050 (0.128)	0.092 (0.134)
	Spectral	Cosine		x				0.113 (0.134)	0.082 (0.150)	0.113 (0.134)
	K-Means	Euclidean					x	0.093 (0.135)	0.050 (0.130)	0.092 (0.135)
{0.1, 0.5}	Spectral	RBF				x		0.331 (0.185)	0.328 (0.208)	0.331 (0.186)
	Spectral	RBF			x			0.322 (0.197)	0.301 (0.221)	0.321 (0.197)
	Spectral	EDR			x			0.315 (0.189)	0.328 (0.206)	0.315 (0.189)
	K-Means	Euclidean			x			0.313 (0.187)	0.261 (0.213)	0.312 (0.187)
	Spectral	EDR		x				0.226 (0.127)	0.131 (0.113)	0.223 (0.127)
	Spectral	Levensthein	x					0.219 (0.144)	0.126 (0.121)	0.217 (0.144)
	K-Means	Euclidean					x	0.208 (0.143)	0.124 (0.122)	0.207 (0.143)
{0.1, 0.7}	Spectral	RBF				x		0.642 (0.200)	0.653 (0.206)	0.641 (0.200)
	Spectral	DTW			x			0.580 (0.209)	0.611 (0.205)	0.580 (0.201)
	K-Means	Euclidean			x			0.533 (0.248)	0.539 (0.275)	0.533 (0.249)
	K-Means	Euclidean					x	0.382 (0.181)	0.309 (0.237)	0.381 (0.180)
	Spectral	Levensthein	x					0.350 (0.114)	0.241 (0.163)	0.347 (0.116)
	Spectral	Euclidean		x				0.306 (0.155)	0.198 (0.179)	0.303 (0.156)
{0.1, 1}	Spectral	DTW				x		0.938 (0.131)	0.942 (0.126)	0.938 (0.131)
	Spectral	Euclidean				x		0.811 (0.136)	0.833 (0.129)	0.811 (0.136)
	Ward	Euclidean				x		0.797 (0.205)	0.796 (0.218)	0.797 (0.205)
	Spectral	DTW			x			0.765 (0.197)	0.791 (0.184)	0.765 (0.197)
	K-Means	Euclidean			x			0.757 (0.180)	0.774 (0.185)	0.757 (0.180)
	K-Means	Euclidean					x	0.796 (0.218)	0.687 (0.265)	0.696 (0.219)
	Spectral	Overlap	x					0.600 (0.299)	0.571 (0.353)	0.598 (0.302)
	Spectral	Euclidean		x				0.335 (0.091)	0.220 (0.138)	0.332 (0.093)

Table 7: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 2, 4\}$, $\sigma = \{1, 1, 1\}$, $\alpha = 0.1$)

Tau	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1, 0.3}	Spectral	DTW				x		0.136 (0.134)	0.094 (0.146)	0.135 (0.134)
	Spectral	EDR			x			0.134 (0.135)	0.097 (0.130)	0.134 (0.135)
	Spectral	Levensthein	x					0.093 (0.134)	0.050 (0.128)	0.092 (0.134)
	Spectral	Cosine		x				0.113 (0.134)	0.082 (0.150)	0.113 (0.134)
	K-Means	Euclidean					x	0.093 (0.135)	0.050 (0.130)	0.092 (0.135)
{0.1, 0.5}	Spectral	RBF				x		0.331 (0.185)	0.328 (0.208)	0.331 (0.186)
	Spectral	RBF			x			0.322 (0.197)	0.301 (0.221)	0.321 (0.197)
	Spectral	EDR			x			0.315 (0.189)	0.328 (0.206)	0.315 (0.189)
	K-Means	Euclidean			x			0.313 (0.187)	0.261 (0.213)	0.312 (0.187)
	Spectral	EDR		x				0.226 (0.127)	0.131 (0.113)	0.223 (0.127)
	Spectral	Levensthein	x					0.219 (0.144)	0.126 (0.121)	0.217 (0.144)
	K-Means	Euclidean					x	0.208 (0.143)	0.124 (0.122)	0.207 (0.143)
{0.1, 0.7}	Spectral	RBF				x		0.642 (0.200)	0.653 (0.206)	0.641 (0.200)
	Spectral	DTW			x			0.580 (0.209)	0.611 (0.205)	0.580 (0.201)
	K-Means	Euclidean			x			0.533 (0.248)	0.539 (0.275)	0.533 (0.249)
	K-Means	Euclidean					x	0.382 (0.181)	0.309 (0.237)	0.381 (0.180)
	Spectral	Levensthein	x					0.350 (0.114)	0.241 (0.163)	0.347 (0.116)
	Spectral	Euclidean		x				0.306 (0.155)	0.198 (0.179)	0.303 (0.156)
{0.1, 1}	Spectral	DTW				x		0.938 (0.131)	0.942 (0.126)	0.938 (0.131)
	Spectral	Euclidean				x		0.811 (0.136)	0.833 (0.129)	0.811 (0.136)
	Ward	Euclidean				x		0.797 (0.205)	0.796 (0.218)	0.797 (0.205)
	Spectral	DTW			x			0.765 (0.197)	0.791 (0.184)	0.765 (0.197)
	K-Means	Euclidean			x			0.757 (0.180)	0.774 (0.185)	0.757 (0.180)
	K-Means	Euclidean					x	0.796 (0.218)	0.687 (0.265)	0.696 (0.219)
	Spectral	Overlap	x					0.600 (0.299)	0.571 (0.353)	0.598 (0.302)
	Spectral	Euclidean		x				0.335 (0.091)	0.220 (0.138)	0.332 (0.093)

Table 8: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 2, 4\}$, $\sigma = \{1, 1, 1\}$, $\alpha = 0.5$)

Tau	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1, 0.3}	Spectral	Overlap	x					0.103 (0.019)	0.002 (0.007)	0.087 (0.020)
	Spectral	EDR		x				0.096 (0.061)	0.016 (0.049)	0.088 (0.059)
	Spectral	EDR				x		0.068 (0.080)	0.008 (0.085)	0.063 (0.078)
	Spectral	Cosine			x			0.043 (0.067)	-	0.043 (0.067)
	K-Means	Euclidean					x	0.036 (0.041)	-	0.036 (0.041)
{0.1, 0.5}	Average	Euclidean		x				0.133 (0.115)	0.040 (0.077)	0.127 (0.115)
	K-Means	Euclidean	x					0.112 (0.124)	0.035 (0.085)	0.110 (0.122)
	Spectral	Euclidean				x		0.103 (0.096)	0.060 (0.092)	0.103 (0.095)
	Spectral	Cosine			x			0.080 (0.088)	0.038 (0.092)	0.080 (0.088)
	K-Means	Euclidean					x	0.079 (0.097)	0.034 (0.088)	0.079 (0.096)
{0.1, 0.7}	Spectral	DTW				x		0.361 (0.200)	0.352 (0.211)	0.361 (0.200)
	Spectral	EDR				x		0.223 (0.083)	0.087 (0.080)	0.217 (0.086)
	Ward	Euclidean				x		0.195 (0.143)	0.103 (0.135)	0.192 (0.143)
	K-Means	Euclidean		x				0.167 (0.123)	0.064 (0.114)	0.161 (0.123)
	K-Means	Euclidean	x					0.157 (0.127)	0.060 (0.116)	0.151 (0.127)
	Spectral	DTW			x			0.105 (0.140)	0.087 (0.179)	0.105 (0.140)
	K-Means	Euclidean					x	0.102 (0.183)	0.063 (0.201)	0.101 (0.183)
{0.1, 1.0}	Spectral	DTW				x		0.792 (0.174)	0.808 (0.170)	0.792 (0.174)
	Spectral	RBF				x		0.692 (0.164)	0.692 (0.184)	0.692 (0.164)
	K-Means	Euclidean				x		0.606 (0.188)	0.578 (0.233)	0.606 (0.189)
	Spectral	EDR		x				0.280 (0.198)	0.201 (0.215)	0.278 (0.198)
	Spectral	Levensthein	x					0.276 (0.101)	0.154 (0.113)	0.272 (0.111)
	Spectral	EDR			x			0.236 (0.196)	0.171 (0.184)	0.235 (0.195)

Table 9: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 2, 4\}$, $\sigma = \{1, 1, 1\}$, $\alpha = 1$)

Tau	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1, 0.3}	Spectral	RBF				x		0.290 (0.173)	0.279 (0.205)	0.279 (0.173)
	Spectral	EDR			x			0.284 (0.253)	0.279 (0.274)	0.284 (0.253)
	Spectral	Levensthein	x					0.275 (0.147)	0.195 (0.156)	0.272 (0.148)
	K-Means	Euclidean					x	0.260 (0.126)	0.200 (0.150)	0.258 (0.127)
	Spectral	Euclidean		x				0.215 (0.144)	0.126 (0.159)	0.211 (0.144)
{0.1, 0.5}	Spectral	DTW				x		0.824 (0.166)	0.837 (0.170)	0.824 (0.166)
	Spectral	DTW			x			0.636 (0.213)	0.661 (0.205)	0.636 (0.213)
	K-Means	Euclidean					x	0.532 (0.182)	0.543 (0.206)	0.531 (0.182)
	Spectral	Levensthein	x					0.411 (0.125)	0.333 (0.160)	0.410 (0.127)
	Spectral	Euclidean		x				0.242 (0.163)	0.132 (0.187)	0.238 (0.165)
{0.1, 0.7}	Spectral	DTW				x		1.00 (0.000)	1.00 (0.000)	1.00 (0.000)
	Spectral	DTW			x			0.726 (0.228)	0.750 (0.214)	0.726 (0.228)
	K-Means	Euclidean					x	0.721 (0.211)	0.741 (0.207)	0.721 (0.211)
	Spectral	Levensthein	x					0.333 (0.147)	0.222 (0.195)	0.329 (0.150)

Table 10: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 1, 2\}$, $\sigma = \{1, 1, 1\}$, $\alpha = 0.1$)

Tau	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1, 0.3}	Average	Euclidean		x				0.101 (0.051)	0.005 (0.023)	0.088 (0.050)
	Average	Euclidean				x		0.093 (0.060)	0.006 (0.024)	0.084 (0.057)
	Spectral	Overlap	x					0.091 (0.073)	0.007 (0.029)	0.084 (0.070)
	Spectral	EDR			x			0.071 (0.077)	0.025 (0.072)	0.070 (0.076)
	K-Means	Euclidean					x	0.057 (0.064)	0.014 (0.069)	0.057 (0.064)
{0.1, 0.5}	Spectral	RBF				x		0.256 (0.190)	0.221 (0.195)	0.255 (0.190)
	Spectral	Overlap	x					0.092 (0.079)	0.008 (0.041)	0.087 (0.075)
	Ward	Euclidean		x				0.092 (0.085)	0.007 (0.040)	0.088 (0.082)
	K-Means	Euclidean			x			0.080 (0.100)	0.043 (0.122)	0.079 (0.100)
	K-Means	Euclidean					x	0.062 (0.093)	0.026 (0.116)	0.062 (0.093)
{0.1, 0.7}	Spectral	DTW				x		0.618 (0.192)	0.627 (0.201)	0.617 (0.192)
	Spectral	Levensthein	x					0.252 (0.137)	0.163 (0.147)	0.249 (0.137)
	Spectral	RBF			x			0.215 (0.158)	0.207 (0.188)	0.215 (0.157)
	Spectral	DTW		x				0.205 (0.218)	0.192 (0.247)	0.205 (0.218)
	K-Means	Euclidean					x	0.185 (0.205)	0.165 (0.225)	0.185 (0.205)
{0.1, 1.0}	Spectral	RBF				x		0.952 (0.098)	0.960 (0.082)	0.952 (0.098)
	Spectral	DTW		x				0.489 (0.269)	0.491 (0.290)	0.488 (0.269)
	K-Means	Euclidean					x	0.453 (0.214)	0.427 (0.252)	0.452 (0.215)
	Spectral	DTW			x			0.406 (0.203)	0.403 (0.226)	0.406 (0.203)

Table 11: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 1, 2\}$, $\sigma = \{1, 1, 1\}$, $\alpha = 1$)

Tau	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1,0.3}	Spectral	Cosine			x			0.088 (0.116)	0.038 (0.105)	0.088 (0.115)
	Spectral	Cosine	x					0.087 (0.099)	0.019 (0.075)	0.084 (0.097)
	Average	Euclidean				x		0.083 (0.107)	0.026 (0.095)	0.080 (0.106)
	Spectral	Cosine		x				0.039 (0.055)	0.000 (0.068)	0.039 (0.055)
	K-Means	Euclidean					x	0.037 (0.057)	-0	0.037 (0.056)
{0.1,0.5}	Spectral	EDR				x		0.088 (0.087)	0.027 (0.078)	0.083 (0.086)
	Spectral	Overlap	x					0.066 (0.051)	0.026 (0.062)	0.065 (0.050)
	Spectral	Cosine			x			0.047 (0.064)	0.002 (0.067)	0.047 (0.063)
	Spectral	DTW		x				0.030 (0.045)	-0	0.030 (0.045)
	K-Means	Euclidean					x	0.029 (0.045)	-0	0.029 (0.045)
{0.1,0.7}	Average	Euclidean				x		0.087 (0.038)	0.007 (0.039)	0.076 (0.034)
	Spectral	Cosine	x					0.081 (0.076)	0.013 (0.044)	0.078 (0.072)
	K-Means	Euclidean			x			0.070 (0.074)	0.014 (0.061)	0.068 (0.072)
	Spectral	EDR		x				0.048 (0.054)	0.003 (0.063)	0.047(0.053)
	K-Means	Euclidean					x	0.041 (0.071)	-0	0.041 (0.071)
{0.1,0.9}	Spectral	Cosine	x					0.180 (0.110)	0.069 (0.081)	0.175 (0.109)
	Spectral	EDR		x				0.161 (0.125)	0.097 (0.129)	0.159 (0.124)
	K-Means	Euclidean			x			0.142 (0.127)	0.049 (0.097)	0.139 (0.126)
	Average	Euclidean				x		0.115 (0.034)	0.008 (0.016)	0.100 (0.037)
	K-Means	Euclidean					x	0.058 (0.108)	0.016 (0.115)	0.058 (0.108)
{0.5, 0.9}	Spectral	Cosine	x					0.169 (0.125)	0.075 (0.097)	0.164 (0.126)
	Spectral	EDR			x			0.158 (0.184)	0.111 (0.188)	0.157 (0.184)
	Spectral	EDR		x				0.150 (0.138)	0.103 (0.149)	0.148 (0.137)
	Average	Euclidean				x		0.104 (0.040)	0.003 (0.014)	0.091 (0.038)

Table 12: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 2, 4\}$, $\sigma = \{1, 1, 1\}$, $\tau = 0.1$)

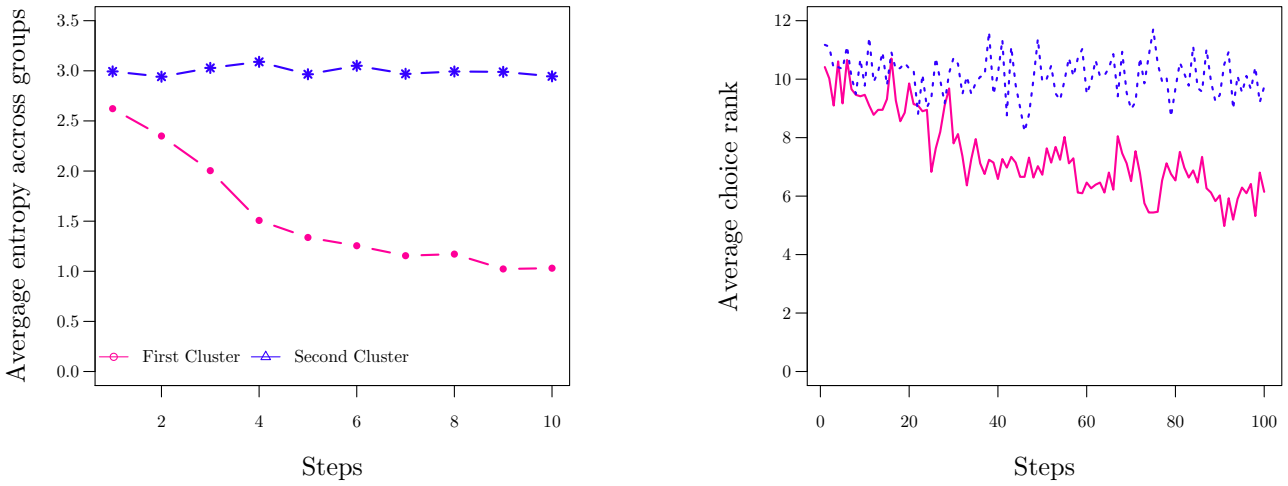
Alpha	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1,0.3}	Spectral	Cosine				x		0.131 (0.114)	0.100 (0.133)	0.131 (0.113)
	K-Means	Euclidean	x					0.130 (0.199)	0.075 (0.212)	0.127 (0.199)
	Spectral	Cosine		x				0.119 (0.104)	0.068 (0.094)	0.118 (0.103)
	K-Means	Euclidean					x	0.113 (0.156)	0.089 (0.181)	0.113 (0.156)
	Spectral	RBF			x			0.101 (0.140)	0.073 (0.162)	0.101 (0.140)
{0.1,0.5}	Spectral	Cosine			x			0.199 (0.131)	0.141 (0.129)	0.198 (0.131)
	Spectral	Cosine		x				0.191 (0.157)	0.128 (0.148)	0.190 (0.157)
	Spectral	Euclidean				x		0.168 (0.183)	0.134 (0.186)	0.168 (0.182)
	K-Means	Euclidean					x	0.122 (0.121)	0.085 (0.122)	0.121 (0.121)
	Spectral	Cosine	x					0.063 (0.081)	0.033 (0.109)	0.063 (0.081)
{0.1,0.7}	Spectral	Cosine		x				0.277 (0.229)	0.222 (0.240)	0.276 (0.229)
	Spectral	Cosine				x		0.278 (0.272)	0.183 (0.254)	0.271 (0.183)
	K-Means	Euclidean			x			0.227 (0.178)	0.186 (0.176)	0.227 (0.177)
	Spectral	Cosine	x					0.190 (0.114)	0.152 (0.116)	0.189 (0.114)
	K-Means	Euclidean					x	0.178 (0.150)	0.147 (0.147)	0.177 (0.149)
{0.1,0.9}	Spectral	DTW				x		0.300 (0.232)	0.273 (0.254)	0.299 (0.232)
	Spectral	Cosine		x				0.211 (0.158)	0.159 (0.160)	0.210 (0.158)
	Spectral	Cosine			x			0.191 (0.117)	0.137 (0.127)	0.189 (0.118)
	Spectral	Cosine	x					0.164 (0.110)	0.115 (0.109)	0.162 (0.109)
	K-Means	Euclidean					x	0.138 (0.153	0.091 (0.176)	0.137 (0.153)

Table 13: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 2, 4\}$, $\sigma = \{1, 1, 1\}$, $\tau = 0.5$)

Alpha	Method	Similarity	C	BBC	E	EB	CC	NMI	ARI	VM
{0.1, 0.3}	Spectral	Overlap	x					0.103 (0.019)	0.002 (0.007)	0.087 (0.020)
	Spectral	EDR		x				0.096 (0.061)	0.016 (0.049)	0.088 (0.059)
	Spectral	EDR				x		0.068 (0.080)	0.008 (0.085)	0.063 (0.078)
	Spectral	Cosine			x			0.043 (0.067)	-0	0.043 (0.067)
	K-Means	Euclidean					x	0.036 (0.041)	-	0.036 (0.041)
{0.1, 0.5}	Average	Euclidean		x				0.133 (0.115)	0.040 (0.077)	0.127 (0.115)
	K-Means	Euclidean	x					0.112 (0.124)	0.035 (0.085)	0.110 (0.122)
	Spectral	Euclidean				x		0.103 (0.096)	0.060 (0.092)	0.103 (0.095)
	Spectral	Cosine			x			0.080 (0.088)	0.038 (0.092)	0.080 (0.088)
	K-Means	Euclidean					x	0.079 (0.097)	0.034 (0.088)	0.079 (0.096)
{0.1, 0.7}	Spectral	DTW				x		0.361 (0.200)	0.352 (0.211)	0.361 (0.200)
	Spectral	EDR				x		0.223 (0.083)	0.087 (0.080)	0.217 (0.086)
	Ward	Euclidean				x		0.195 (0.143)	0.103 (0.135)	0.192 (0.143)
	K-Means	Euclidean		x				0.167 (0.123)	0.064 (0.114)	0.161 (0.123)
	K-Means	Euclidean	x					0.157 (0.127)	0.060 (0.116)	0.151 (0.127)
	Spectral	DTW			x			0.105 (0.140)	0.087 (0.179)	0.105 (0.140)
	K-Means	Euclidean					x	0.102 (0.183)	0.063 (0.201)	0.101 (0.183)
{0.1, 1.0}	Spectral	DTW				x		0.792 (0.174)	0.808 (0.170)	0.792 (0.174)
	Spectral	RBF				x		0.692 (0.164)	0.692 (0.184)	0.692 (0.164)
	K-Means	Euclidean				x		0.606 (0.188)	0.578 (0.233)	0.606 (0.189)
	Spectral	EDR		x				0.280 (0.198)	0.201 (0.215)	0.278 (0.198)
	Spectral	Levenstein	x					0.276 (0.101)	0.154 (0.113)	0.272 (0.111)
	Spectral	EDR			x			0.236 (0.196)	0.171 (0.184)	0.235 (0.195)
	K-Means	Euclidean					x	0.099 (0.105)	0.052 (0.097)	0.098 (0.104)

Table 14: Simulation results (Setting: Rounds = 20, Size = 20, $\mu = \{0, 2, 4\}$, $\sigma = \{1, 1, 1\}$, $\tau = 1.0$)

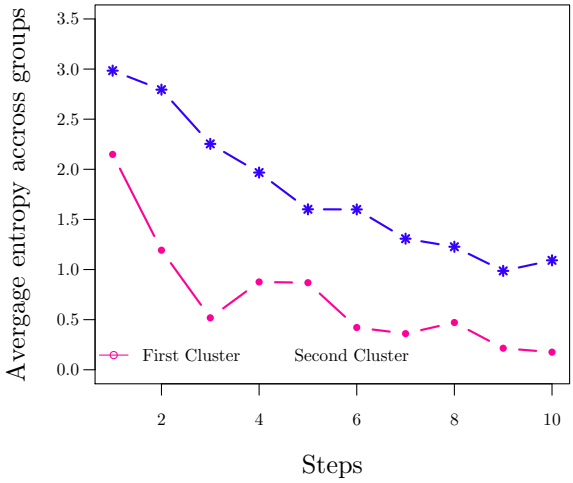
E Multi arm bandit data



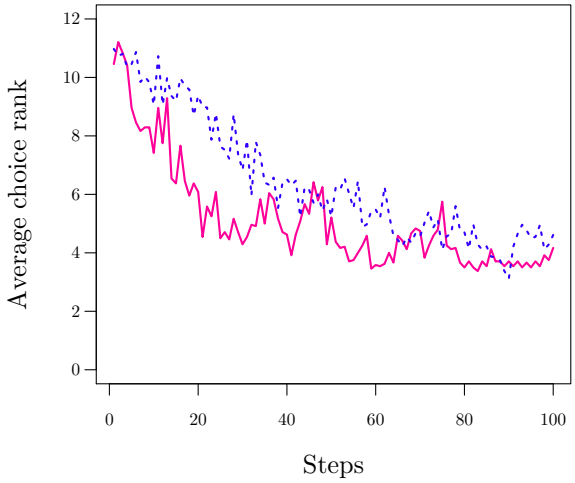
(**a**) Blockwise entropy by cluster

(**b**) Average choice rank by cluster

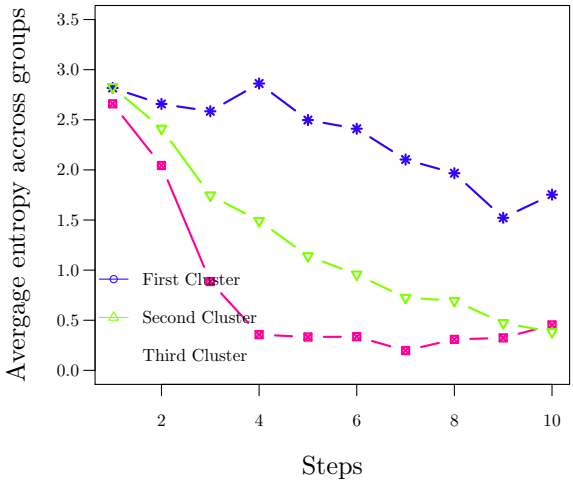
Figure 6: Blockwise entropy and average choice rank based on clustering results for experiment 1



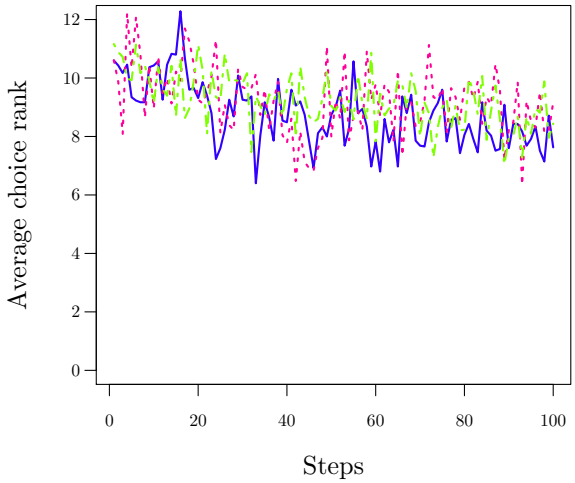
(**a**) Blockwise entropy by cluster



(**b**) Average choice rank by cluster



(**a**) Blockwise entropy by cluster



(**b**) Average choice rank by cluster

Figure 7: Blockwise entropy and average choice rank based on clustering results for experiment 2

F Prison data

Groups	Method	Similarity	C	CBC	BBC	E	EB	CC	NMI	ARI	VM
6vs1	Spectral	EDR				x			0.237 (0.000)	0.020 (0.000)	0.237 (0.000)
	K-Means						x		0.167 (0.000)	-	0.166 (0.000)
	Spectral	Euclidean		x					0.057 (0.000)	0.036 (0.000)	0.057 (0.000)
	Spectral	EDR			x				0.037 (0.000)	0.034 (0.000)	0.037 (0.000)
	K-Means	Euclidean						x	0.023 (0.028)	-	0.023 (0.028)
	K-Means		x						0.020 (0.022)	-	0.020 (0.023)
	Average								0.068 (0.068)	-	0.067 (0.068)
6vs2	K-Means			x					0.289 (0.052)	0.217 (0.063)	0.278 (0.055)
	K-Means					x			0.165 (0.000)	0.083 (0.000)	0.145 (0.000)
	Spectral	Cosine					x		0.165 (0.000)	0.083 (0.000)	0.145 (0.000)
	Ward	Euclidean			x				0.072 (0.000)	0.083 (0.000)	0.071 (0.000)
	K-Means		x						0.041 (0.022)	-	0.041 (0.022)
	K-Means	Euclidean						x	0.028 (0.023)	-	0.028 (0.023)
	Average								0.116 (0.088)	0.058 (0.081)	0.110 (0.083)
6vs3	Spectral	warp				x			0.205 (0.000)	-	0.203 (0.000)
	K-Means						x		0.174 (0.000)	-	0.171 (0.000)
	Spectral	warp		x					0.116 (0.000)	-	0.110 (0.000)
	Ward	Euclidean			x				0.116 (0.000)	-	0.110 (0.000)
	Lavenstein		x						0.116 (0.000)	-	0.110 (0.000)
	K-Means	Euclidean						x	0.110 (0.023)	-	0.105 (0.022)
	Average								0.089 (0.056)	-	0.086 (0.055)
6vs4	Spectral	EDR				x			0.360 (0.000)	0.227 (0.000)	0.342 (0.000)
	Spectral	Euclidean					x		0.246 (0.000)	0.229 (0.000)	0.246 (0,000)
	Ward	Euclidean		x					0.105 (0.000)	-	0.102 (0.000)
	Complete	Euclidean			x				0.848 (0.000)	-	0.080 (0.000)
	K-Means	Euclidean	x						0.049 (0.000)	-	0.049 (0.000)
	K-Means	Euclidean						x	0.080 (0.050)	-	0.079 (0.048)
	Average								0.089 (0.088)	-	0.088 (0.088)
6vs9	Ward	Euclidean					x		0.561 (0.000)	0.560 (0,000)	0.560 (0,000)
	Spectral	DTW				x			0.435 (0.000)	0.387 (0.000)	0.432 (0.000)
	Spectral	DTW		x					0.435 (0.000)	0.387 (0.000)	0.432 (0.000)
	Spectral	EDR			x				0.333 (0.000)	0.381 (0.000)	0.333 (0.000)
	Spectral	Overlap	x						0.111 (0.000)	0.118 (0.000)	0.111 (0.000)
	K-Means	Euclidean						x	0.106 (0.066)	0.033 (0.054)	0.094 (0.059)
	Average								0.249 (0.167)	0.202 (0.189)	0.244 (0.169)

Table 15: Results for prison data for group assault vs. rest

Groups	Method	Similarity	C	CBC	BBC	E	EB	CC	NMI	ARI	VM
2vs4	Spectral	eucsim/rbf		x					0.213 (0.000)	0.007 (0.000)	0.211 (0.000)
	Levenstein		x						0.213 (0.000)	0.007 (0.000)	0.211 (0.000)
	Spectral	EDR				x			0.132 (0.000)	0.104 (0.000)	0.131 (0.000)
	Spectral	EDR			x				0.054 (0.000)	-	0.054 (0.000)
	Spectral	eblock					x		0.018 (0.000)	-	0.017 (0.000)
	K-Means	Euclidean						x	0.005 (0.005)	-	0.005 (0.005)
									0.042 (0.063)	-	0.041 (0.063)
2vs6	K-Means			x					0.3 (0.048)	0.23 (0.062)	0.291 (0.510)
	K-Means					x			0.165 (0.000)	0.083 (0.000)	0.145 (0.000)
	Spectral	Cosine					x		0.165 (0.000)	0.083 (0.000)	0.145 (0.000)
	Ward	Euclidean			x				0.072 (0.000)	0.083 (0.000)	0.071 (0.000)
	K-Means	Euclidean						x	0.027 (0.020)	-	0.027 (0.020)
	K-Means		x						0.023 (0.021)	-	0.023 (0.021)
									0.112 (0.090)	0.057 (0.082)	0.106 (0.085)
2vs9	Spectral	EDR				x			0.382 (0.000)	0.235 (0.000)	0.382 (0.000)
	Spectral	EDR			x				0.232 (0.000)	0.226 (0.000)	0.232 (0.000)
	Spectral	Cosine					x		0.197 (0.000)	0.008 (0.000)	0.191 (0.000)
	Spectral	eucsim/rbf		x					0.134 (0.000)	0.075 (0.000)	0.134 (0.000)
	K-Means	Euclidean						x	0.116 (0.000)	-	0.105 (0.000)
	K-Means		x						0.111 (0.022)	-	0.100 (0.020)
									0.096 (0.074)	-	0.093 (0.073)

Table 16: Results for prison data for group robbery vs. rest