

USING A LAYERED ALGORITHM TO DETECT DRIVER COGNITIVE DISTRACTION

Yulan Liang¹ & John D. Lee²

¹Liberty Mutual Research Institute for Safety, Hopkinton, Massachusetts, USA

²University of Wisconsin, Madison, Wisconsin, USA

Email: yulan.liang@libertymutual.com

Summary: Detection of cognitive distraction presents an indispensable function for driver distraction mitigation systems. In this study, we developed a layered algorithm that integrated two data mining methods—Dynamic Bayesian Network (DBN) and supervised clustering method—to identify cognitive distraction from drivers' eye movements and driving performance measures. We used the data collected in a simulator study to compare the layered algorithm with the original DBN and found that the layered algorithm obtained comparable prediction performance as the original DBN. Meanwhile, the layered algorithm shortened training and prediction time and revealed rich information on the relationship between driver cognitive state and performance. This study demonstrates that data mining methods are suitable to identify human cognitive state from performance.

INTRODUCTION

A promising strategy to minimize the number of motor vehicle crashes caused by driver distraction is to develop adaptive distraction mitigation systems, which can provide assistance to reduce distraction based on the state of drivers and environment using automated sensors and other technology. For such a system, accurately identifying whether drivers are distracted is one of key functions.

Driver distraction diverts driver's attention away from the activities critical for safe driving toward a competing activity (Lee, Young, & Regan, 2008). Cognitive distraction is usually described as "mind-off-road" and represents shared central processing demand of driving and secondary tasks. Detecting cognitive distraction can be quite challenging because the symptoms of cognitive distraction are usually not readily apparent, hardly described by a simple linear relationship, and vary across individuals. Detecting cognitive distraction likely requires an integration of a large number of performance and physiological indicators (e.g., eye gaze measures) over relatively long period of time and needs to be personalized for different drivers (Liang, Reyes, & Lee, 2007). The challenge is how to integrate performance measures in a logical manner to comprehensively infer the driver's cognitive state. Data mining methods that can extract unknown patterns from a large volume of data present an innovative and promising approach.

In previous studies, two data mining methods—Support Vector Machines (SVMs) and Dynamic Bayesian Networks (DBNs)—successfully detected cognitive distraction from driver visual behavior and driving performance (Liang, Lee, & Reyes, 2007; Liang, Reyes, et al., 2007). SVMs, proposed by Vapnik (1995), are based on statistical learning theory and can be used for non-linear classification. This method is computationally efficient and minimizes prediction error to avoid over-fitting. Tested with the data collected in a simulator study, SVMs detected

cognitive distraction (i.e., when drivers were engaged in a cognitive secondary task) with an average accuracy of 81%, outperforming traditional logistic regression method. Nonetheless, SVMs do not consider time-dependent relationship between variables, and the resultant models do not present the relationships learned from data in an interpretable way.

Bayesian Networks (BNs) represent a probability-based approach that presents conditional dependencies between variables, and DBNs, one type of BNs, can model a time-series of events according to a Markov process. Compared with SVMs, DBNs are easy to interpret, can consider time-dependent relationship between cognitive state and performance, and obtain more accurate and sensitive models (Liang & Lee, 2008). However, DBNs are not computationally efficient, needing an average 20 minutes of processing time to train a model, compared to 15 seconds to train a SVM model with the same training data.

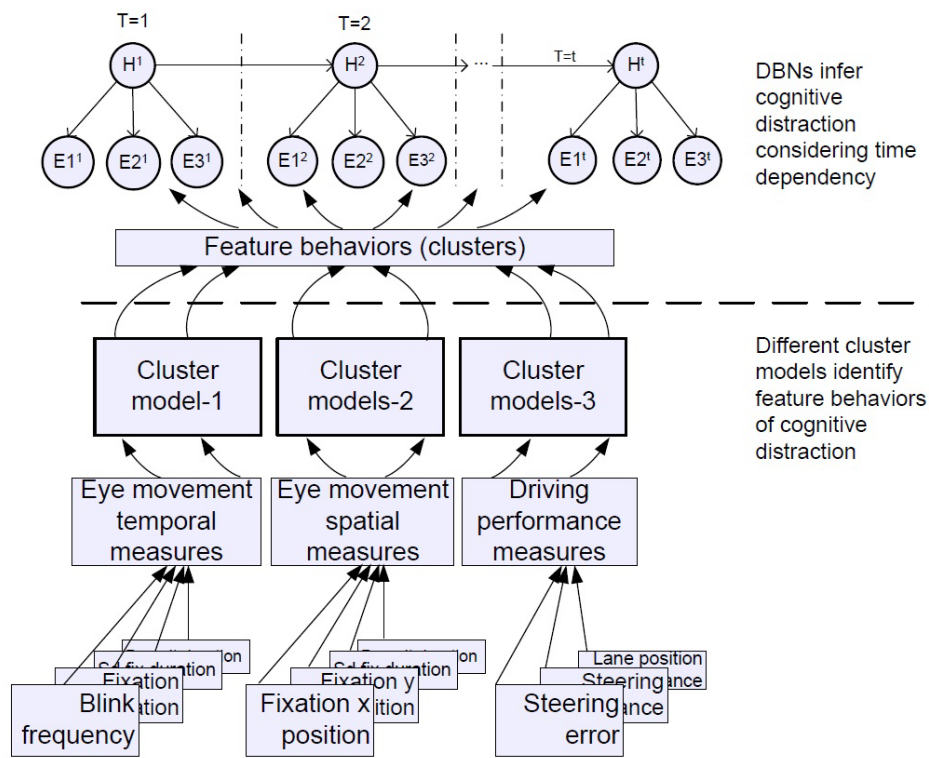


Figure 1. The structure of the layered algorithm. The curving, solid arrows indicate data flow. The straight, lined arrows in the DBN algorithm indicate associations between variables

To obtain accuracy, efficiency, and interpretability, we proposed a new approach that combines DBNs and a feature reduction method (e.g., clustering) in a hierarchical manner (Figure 1). At the lower layer, cluster models identify drivers' feature behaviors during cognitive distraction (i.e., clusters) based on a number of performance measures. At the higher-layer, a DBN model uses the labels of these feature behaviors as input values to recognize driver cognitive state. This algorithm reduces the number of input variables of DBNs and is expected to improve computational efficiency from the original DBN algorithm. At the same time, the layered algorithm preserves time dependency and ease of interpretation. In this study, we compared the layered algorithm with the original DBNs.

METHOD

Experimental data

All algorithms were trained and tested with the data collected in a simulator-based experiment. The experiment included six 15-minute drives: four distraction drives and two baseline drives. During each distraction drive, participants completed four separate interactions with an auditory stock ticker with a one-minute interval in between. The stock ticker task used simple auditory stimuli consisting of 3-letter stock names and 2-digit prices, but rendered high cognitive workload to drivers. It required participants continuously track the price changes of two different stocks and report the overall trend of the changes at the end. In the baseline drives, participants did not perform any secondary task. During all drives, participants were instructed to maintain vehicle position as close to the center of a straight lane as possible, to respond to the intermittent braking of a lead vehicle, and to report the appearance of bicyclists in the driving scene. Eye movement and driving performance data were collected at a rate of 60 Hz for nine participants (their average age was 45 years old with SD of 6.6) using a Seeing Machines faceLAB™ eye tracker and the DriveSafety™ driving simulator, respectively. Further details of the experiment and data reduction can be found in (Liang, Reyes, et al., 2007).

We defined “distraction” as occurrence in the distraction drives and “no distraction” as in the baseline drives, (Reyes & Lee, 2008). Each row, called instance, in the reduced data set included 19 continuous measures of visual and driving performance summarized over 30 seconds and corresponding driver cognitive state in that period. These 19 measures were divided into three groups based on their correlation and meanings —eye movement temporal measures, eye movement spatial measures, and driving performance measures (Table 1).

Table 1. Three groups of performance measures

Groups	Performance measures
Eye movement temporal measures	blink frequency, mean and standard deviation (SD) of fixation duration, pursuit duration, pursuit distance, pursuit speed, and percentage of the time spent on performing pursuit movements
Eye movement spatial measures	mean and SD of horizontal and vertical fixation location coordinates and direction
Driving performance measures	SD of steering wheel position, mean steering error, and SD of lane position

The layered algorithm and alternative algorithm

In the lower layer, three cluster models, each corresponding one group of measures, were built using a supervised clustering method (Figure 1). This method identifies clusters for a classified dataset so that the majority cases in one resultant cluster belonged to one class (Zeidat, Eick, & Zhao, 2006). Supervised clustering may discover some heterogeneous effects of cognitive distraction by identifying more than one feature behaviors for each cognitive state. At the higher layer, H^t and E_i^t ($i=1,2,3$) represented driver cognitive state and corresponding behaviors at a time step t , and the arrows represent the associations between cognitive state and behaviors (Figure 1). The across-time arrow occurred only between the cognitive states at two consecutive

time steps. The layered algorithm was compared with the original DBN with 19 performance measures as inputs.

Training and evaluation

For all three types of algorithms, we trained and tested individual models for each driver. First, we normalized performance measures by calculating z-scores. Then, we blocked the normalized data by two consecutive instances and assigned these blocks randomly into training and testing datasets. Both data sets contained multiple sequences of instances. Training data was composed of two thirds of the total data, and the remaining one third served as testing data. We trained the detection models with only the training data and used the testing data as “unseen” cases to evaluate the algorithms.

For the layered algorithm, the training procedure included building three cluster models at the lower layer and training the DBN model at the higher layer. The cluster models were trained using SRIDHCR algorithm (Zeidat, et al., 2006) programmed with Matlab R2006b. The final number of clusters for the cluster models ranged between two and six across different drivers. The DBN model in the layered algorithm and the original DBN algorithm were trained according to the protocol described in (Liang, Lee, et al., 2007) using the Matlab toolbox (Murphy, 2004) and accompanying structure learning package (LeRay, 2005).

We evaluated the algorithms in terms of prediction effectiveness and computational efficiency. Prediction performance measures included detection accuracy, hit rate, false alarm rate, and signal detection theory (SDT) measures (i.e., d' and response bias (Stanislaw & Todorov, 1999)). d' represents the ability of the model to detect driver distraction. The larger the d' value, the more effectively the model detects distraction. Response bias signifies how the model tend to under- or over-identify distraction. A value less than zero represents a tendency to overestimate driver distraction; and vice versa. Computational efficiency measures included CPU time to train and test the models. The computer used was a SONY VAIO laptop with Intel® Core™2 CPU (T5500 @ 1.66GHz) and 1GB of RAM. The Matlab software run on Microsoft Windows XP Service Pack 3, and no other applications running at the same time.

Table 2. The results of algorithm comparisons

	Layered algorithm (SD)	DBNs (SD)	Friedman's test χ^2_1 (p-value)
Accuracy (%)	88 (8)	88 (16)	0.11 (0.74)
d'	3.50 (1.81)	4.80 (2.54)	0.11 (0.74)
Response bias	1.82 (4.38)	-0.22 (4.47)	0.11 (0.74)
Hit rate	0.88 (0.08)	0.92 (0.14)	1.00 (0.32)
False alarm rate	0.14 (0.16)	0.16 (0.23)	0.00 (1.00)
Training CPU time (s)	13 (2.0)	1146 (131)	9.00 (0.003)
Testing CPU time (s)	0.95 (0.12)	5.91 (0.73)	9.00 (0.003)

RESULTS

We used Friedman's non-parametric tests to compare each evaluation measure across two types of algorithms. The layered, the original DBN algorithm achieved similar prediction effectiveness: all five measures were not statistically different between different algorithms

(Table 2). However, the training and testing time reduced greatly from the original DBN algorithm to the layered algorithms (Table 2). To train the layered algorithm for each driver required an average of 13 seconds, in contrast to 1146 seconds (19.1 minutes) required for the original DBNs. It suggests that the layered algorithm improves computational efficiency from the original DBNs and may be more practical to be used in the real-world distraction detection.

DISCUSSION

One advantage of data mining methods is they help to discover new knowledge by extracting hidden relationships in the data. Studying layered algorithms clarify aspects cognitive distraction revealed in traditional statistical analysis (e.g., ANOVA). This benefit can be seen in Figure 2.

A cluster model of eye movement temporal characteristics produced three clusters: Temp-1, Temp-2, and Temp-3. Temp-1 was primarily comprised of “no distraction” cases, and Temp-2 and Temp-3 “distraction” cases. Temp-1 had lower blink frequency compared to Temp-2 and Temp-3, indicating that drivers tend to blink faster when distracted. This may indicate diminished attention to visual control, which can increase involuntary eye movements and disrupt consolidation of visual information (Strayer, Drews, & Johnston, 2003).

The cluster model associated with eye movement spatial measures produced three clusters: Spat-1 comprised of “no distraction” cases, and Spat-2 and Spat-3 comprised of “distraction” cases. Drivers tended to look down during distraction, illustrated by larger vertical position of fixation (meanfy). It suggests that during distraction these drivers focused on the roadway near the vehicle, but not straight ahead. It could limit the drivers’ capability to foresee the driving situation. But this effect varied across individuals. Among nine drivers, three drivers tended to look down, and two drivers tended to look up during distraction, suggesting that driver eye-gaze patterns are somewhat idiosyncratic when visual scanning is disrupted by cognitive workload (Harbluk, Noy, Trbovich, & Eizenman, 2007; Victor, Harbluk, & Engström, 2005).

The cluster model built from driving performance measures produced three clusters: Driv-1 and Driv-2 featured “no-distraction” cases, and Driv-3 featured “distraction” cases. Although Driv-1 and Driv-3 shared similar steering-angle variance (std_steer), Driv-3 had larger steering error than Driv-1 (steer_error). It suggests that the driver steered more abruptly during cognitive distraction than no-distraction even when the steering angle changed in a similar range. This effect was found in six out of nine drivers. Meanwhile, the DBN model at the higher layer showed that clusters Driv-1, Driv-2, and Driv-3 occurred during “no distraction” (Driv-1: 0.45; Driv-2: 0.35; Driv-3: 0.20) while Driv-3 occurred dominantly over Driv-1 and Driv-2 during “distraction” (Driv-1: 0.14; Driv-2: 0.09; Driv-3: 0.77). It suggests that when drivers are not distracted, their driving performance varies substantially, but when they are distracted, their performance is more regular. It may reflect driver’s ability to employ many strategies to support satisfactory performance when demands are low, but relatively few strategies support satisfactory performance when demands are high (Goodrich, Stirling, & Frost, 1998). The across time transition between cognitive states was an identity matrix because the definition of distraction used in this study led to no natural transition between the cognitive states of drivers in training data.

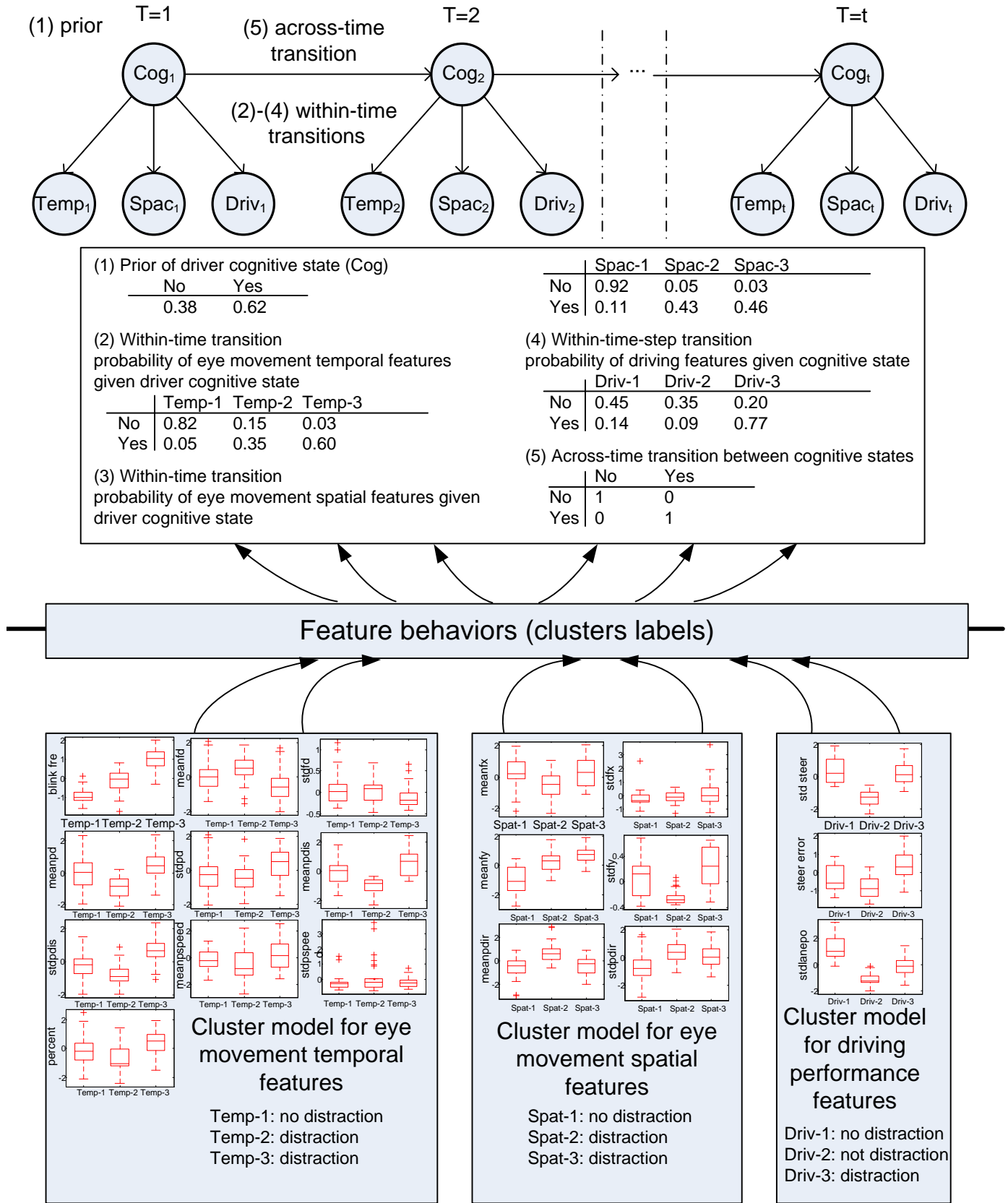


Figure 2. The example of trained layered algorithms

CONCLUSIONS

Based on the results, although the layered algorithm did not improve cognitive distraction detection, the layered algorithm significantly improved computational efficiency. The layered algorithm also provides useful insights concerning the effects of cognitive distraction on driver behavior, which have no equivalent in the SVM algorithm and other traditional statistical tests. This study demonstrated that data mining methods are suitable to identify human cognitive state from eye glance behavior and driving performance.

REFERENCES

- Goodrich, M. A., Stirling, W. C., & Frost, R. L. (1998). A theory of satisficing decisions and control. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 28(6), 763-779.
- Harbluk, J. L., Noy, Y. I., Trbovich, P. L., & Eizenman, M. (2007). An on-road assessment of cognitive distraction: Impacts on drivers' visual behavior and braking performance. *Accident Analysis & Prevention*, 39(2), 372-379.
- Lee, J. D., Young, K. L., & Regan, M. A. (2008). Defining driver distraction. In M. A. Regan, J. D. Lee & K. L. Young (Eds.), *Driver distraction: Theory, effects, and mitigation* (pp. 31-40). Boca Raton, FL: CRC Press, Taylor & Francis Group.
- LeRay, P. (2005). The BNT Structure Learning Package v.1.4 Retrieved May 20, 2005, from <http://banquiseasi.insa-rouen.fr/projects/bnt-slp/>
- Liang, Y., & Lee, J. D. (2008). Comparing Support Vector Machines (SVMs) and Bayesian Networks (BNs) in detecting driver cognitive distraction using eye movements. In R. Hammoud (Ed.), *Passive eye monitoring: Algorithms, applications and experiments* (pp. 285-300). Leipzig, Germany: Springer-Verlag Berlin Heidelberg.
- Liang, Y., Lee, J. D., & Reyes, M. L. (2007). Non-intrusive detection of driver cognitive distraction in real-time using Bayesian networks. *Transportation Research Record: Journal of the Transportation Research Board (TRR)*, 2018, 1-8.
- Liang, Y., Reyes, M. L., & Lee, J. D. (2007). Real-time detection of driver cognitive distraction using Support Vector Machines. *IEEE Transactions on Intelligent Transportation Systems*, 8(2), 340-350.
- Murphy, M. P. (2004). The Bayes Net Toolbox Retrieved June 9, 2004, from <http://bnt.sourceforge.net>
- Reyes, M. L., & Lee, J. D. (2008). Effects of cognitive load presence and duration on driver eye movements and event detection performance. *Transportation Research Part F: Traffic Psychology and Behaviour* 11(6), 391-402.
- Stanislaw, H., & Todorov, N. (1999). Calculation of signal detection theory measures. *Behavior Research Methods, Instruments, & Computers*, 31(1), 137-149.
- Strayer, D. L., Drews, F. A., & Johnston, W. A. (2003). Cell phone-induced failures of visual attention during simulated driving. *Journal of Experimental Psychology: Applied*, 9(1), 23-32.
- Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer.
- Victor, T. W., Harbluk, J. L., & Engström, J. A. (2005). Sensitivity of eye-movement measures to in-vehicle task difficulty. *Transportation Research Part F*, 8, 167-190.
- Supervised clustering: Algorithms and application (2006).