University of Victoria
Faculty of Computer Science
Spring 2021 CSC 503
Project Progress Report

Driving Cycle Segmentation, Clustering, and Prediction

Instructor
Dr. Nishant Mehta

Date:
Mar 15, 2021

Prepared by:

Yang Yang
Simbarashe Zuva
Haijia Zhu
Siyang Liu

Problem

Problem definition

Plug-in hybrid electric vehicle (PHEVs) is one the most promising solution to curb climate change and reduce air pollution. The fuel economy of PHEV highly depends on the energy management and power control strategy controller (EMS). Typically, optimal fuel consumption can be achieved by adopting an offline global optimization approach, such as dynamic programming. However, it is not feasible in real-life applications, as each driving data differs from time to time, and may consider completely different compare to typical driving cycles. Therefore, all existing PHEVs only have EMS with fixed control parameters, which may cause higher fuel consumption compared to a conventional vehicle due to bad matching of EMS and the various driving condition. A well-calibrated EMS can achieve the fuel consumption of PHEV closes to the global optimal fuel consumption only when it is operating at the designed condition.

One way to achieve that is to provide the vehicle onboard computer with the destination and traffic condition of the trip (driving data). And the onboard computer (or cloud computing if the internet is accessible) tunes the EMS control parameters and applies the corresponding parameters to the given driving data. Hence the vehicle can always operate at designed/optimal conditions.

However, providing the correct driving data may not be feasible, especially since the traffic condition is hard to generalize. Pattern recognition and machine learning technology can help solve such problems.

To do that, one should develop a system that can train a classifier and forecast the driving data accordingly. A simplified system diagram is presented in Figure 1. The light blue box at the top represents the offline global optimization process. This block could provide the optimal control parameter for any given driving cycle in the dataset by searching the optimal control parameter in the design space at given driving cycles using a global optimization algorithm. The predictor, marked in the orange box in the middle section in Figure 1, will forecast the driving cycle according to the current and past vehicle velocity once the vehicle starts operating. The predicted driving cycle can be used to find the matched pre-optimization EMS control parameter. As the vehicle proceeds, this process will repeat until the vehicle stops. Ideally, the accuracy of the prediction should increase as more features become available (first 60-second vehicle velocity data has fewer features than 300-second velocity data). The vehicle controller should evaluate the result at the end of the operation. If the system failed to predict the driving data, which indicate the system cannot find a driving cycle that matches that driving data (and this diving data may belong to a new cluster), the classifiers need to be re-trained, as shown in the orange box at the bottom in Figure 1.

In the initial document, two concepts, namely FSPM and TCPM, are proposed. However, the FSPM is removed from the plan due to two reasons:

1. This method only predicts the next few segments of the active cycle, the computed control parameters remain at a local optimum. For hybrid vehicle control and fuel consumption minimization, the combinations of local optimums may not lead to global optimal results since the problem may not be strictly convex.
2. Although the actual implementation proposed is unique, the basic concept of model predictive control has been well developed. A great amount of research has been conducted on such a predictive approach. With limited time for the project, no better results can potentially be obtained, thus, no significant research contributions can be made.

The research focus will then be shafted toward TCPM only. The outstanding problem is to identify the active mission. Since the FSPM is obsoleted, its potential training method – dynamic time wrapping is to be used in TCPM. Meanwhile, the originally proposed text presentation method will be used as well prior to the unsupervised typical cycle clustering. Both methods are used mainly because them can handle data with different dimension.

After the initial tests, it is observed that if similar driving data has not been "seen" by the training model, the system will classify the cycle based on the nearest Euclidean norm, and the classification outcomes are not ideal. Thus, an initial data processing is proposed to clear the input data for training. The detailed method will be discussed in the next section.

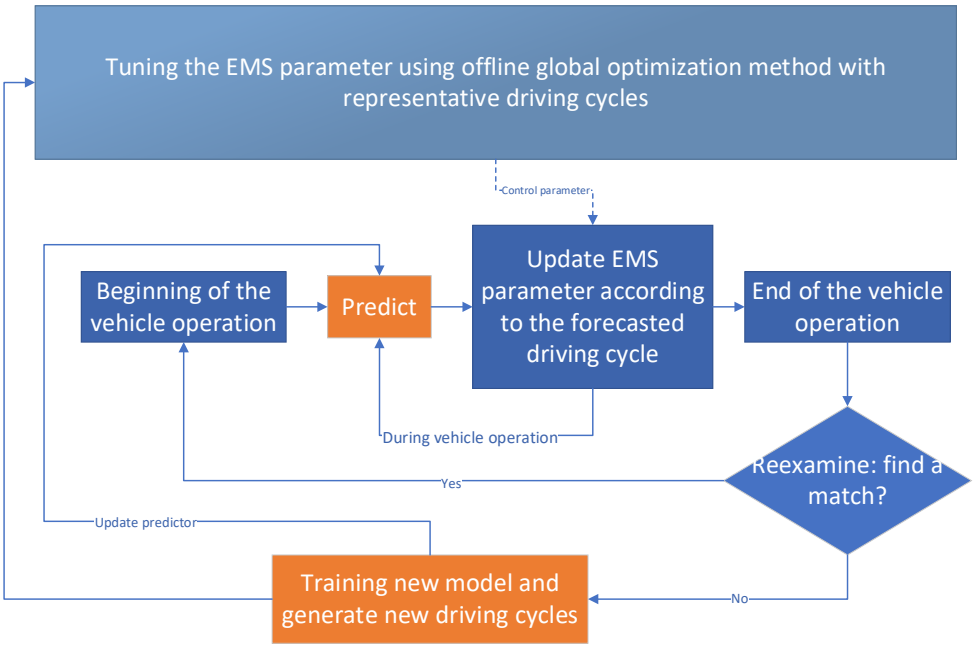


Figure 1 System Diagram

The objective of this project is to implement those modules, marked in the orange box in Figure 1, that can improve EMS and reduce the fuel consumption of PHEVs. The key of this adaptive-EMS is to develop a vehicle speed profile reconstruction mechanism and driving cycle/data pattern recognition. This adaptive-EMS can reduce fuel consumption by predicting the entire driving condition by feeding past and current vehicle velocity, and then adopting the pre-optimized control parameters.

Datasets

This project is to establish a driving cycle-based vehicle speed prediction method using machine learning techniques. Two different types of data will be used in the training processes: standard driving cycle and 2010-2012 California Household Travel Survey (CHTS) (provided by the Transportation Secure Data Center of the NREL). The mixed data sources ensure the data representation from both certified standards and the general public. The source and the difference of the two types of the data can be found in Appendix

Goals

As stated for hybrid vehicles, the control and energy management strategies are the deterministic factors for the system's performance. The control strategies are often governed by the vehicle speed; hence, predicting the vehicle speed for better control parameters has become an advanced method. However, currently, the main approaches are to predict the speed segments in the near future. These methods such as model predictive control ensure the local optimal, whereas the goal optimal is not guaranteed. The goal of this project is to accurately predict the vehicle's overall driving cycle according to the active driving information in order to apply the "optimal" control parameters.

Unlike the model predictive control, the prediction is made based on the knowledge of the current cycle, the proposed driving cycle prediction method requires prior knowledge of different cycles as training data.

After training with sufficient data, the model should be able to predict the active driving cycle based on the first few segments of the driving data by comparing it to the trained model.

The concept of the model is after implemented on the vehicle, the system will start to learn from the daily driving profiles and cluster the profiles into different groups. During the offline time, the optimal control through dynamic programming for each group will be developed and deployed to the controller. During the next mission, the active driving segments will be collected and compared to the clusters. The most similar typical cycle and control will be applied to the current mission. The system will closely monitor the similarity between the active cycle and representative cycle as well as the performance of the control in order to maintain or switch the control. If no similarity is recognized, the system will regard the active cycle as a new cycle. During the next offline period, the system will add the new cycle(s) to the known dataset to retraining the model and develop new control. The final goal to be achieved is an unsupervised learning system that can learn a person's driving habits and apply corresponding control to achieve goals such as minimizing fuel consumption.

To be more detailly discussed in the following sections, the main machine learning approaches have been modified from the original proposal; thus, the quantitative goal needs to be modified. There are two main prediction methods include Feature Segments Prediction Method (FSPM) and Typical Cycle Perdition Method (TCPM). FSPM method is removed from the process due to a few reasons illustrated below in the Plans and Progresses section. The measurable results for TCPM remain the same. The method targets the overall cycle and seeks for the global optimal, where a typical driving cycle can roughly represent a cluster of cycles. The goal of TCPM is that the typical cycle should be at least 85% accurate when comparing to the cycles to be represented. Overall, as the goal of these methods, the final fuel consumption needs to be reduced. At the first stage, a 5% overall fuel consumption reduction is planned.

Plans and Progress

Main Experiments Progress

The experiments are performed on standard driving cycles first to validate proposed concepts and algorithms. The main reason for using the standard cycles is that those cycles are widely used in dyno test for decades and is the representative [cite: 那个 nrel 写的讲 driving cycle 的论文] driving cycle for light-duty consumer vehicles. Meanwhile, performing experiments on small representative data set is time-efficient.

Feature extraction

The feature extraction is aimed at finding the similarity of each segment, which will be used to reconstruct the driving data later. Those segments come without any labels, thus clustering is needed to find segments that share similar features.

This process can be summarized as below:

First, all the driving data is divided into smaller segments. The duration of the segment is chosen to be 5 seconds by running a grid search on segment length and number of clusters.

Next, clustering is performed. At the time of writing this report, three different methods are tested including:

- K-means clustering
- Fuzzy C-means clustering
- Dynamic time wrapping

The number of clusters is crucial to find a good representative of the driving data. The Elbow method is used to determine the size of the cluster in this experiment. The detailed list of the methods and some finding can be found in XXX

Driving Data Text Representation, Clustering, and Prediction

As one of the approaches for driving cycle prediction, the text representation method uses the name of the 48 representative segments to indicate the segments. Thus, after representing the overall driving profile with the segments, a text string containing a sequence of names can be obtained. After all the training velocity profile is converted into strings, the bag-of-n-gram method is used to extract the frequency of appearance of n-grams (3 grams are considered in this experiment). The reason for converting the velocity profile to text strings is to eliminate the inherent time effect and enable the ability to handle driving data with different lengths. Meanwhile, by only measuring the appearance frequency, the algorithm can predict the active mission cycle using the collected information. The prediction accuracy may inevitably be low at the beginning of the mission; however, as cycle time increases, the prediction will converge. At the current testing stage, due to the initial data (raw California data and augmented data), several problems appearances including unconverted clustering (due to insufficient data similarity) and over-performing (due to data similarity). To resolve the problem, the data preprocessing aforementioned is required.

Action Items

Data preprocessing

The data from CHTS contains driving data with variance travel time and distance, which cannot be fed into the system directly for training. Thus, a data cleaning process is needed. Meanwhile, after a few initial tests, it is found that in order to cluster the text-represented speed profile, the training data needs to have sufficient similarity. Originally, the raw driving data is fed into the cluster. Since the raw data is from multiple drivers on different missions, no similar feature can be obtained, and the clustering process failed. Then, a set of profiles obtained by augmenting three different profiles are tested. However, since the data is, the n-grams from the text-represented profile are too close. The testing results achieve 100% accuracy and immediate correct classification by providing the first three segments of a new testing profile. The results are too good to be trusted. Therefore, a data mining process is proposed and under testing at the current stage. The aim is to filter the raw data to keep data that have similar travel distance, travel time, and general profile. This allows sufficient similarity between profiles to ensure successful unsupervised clustering processes; on the other hand, maintain a certain level of difference to ensure robustness.

X-mean Clustering

The number of clusters is crucial for any clustering analysis. The elbow method uses in this experiment. However, as we introduce more driving data into the test data set, the number of clusters should vary slightly. An automatic process should be developed to find the ideal number of clusters. One approach is to use x-means.

X-means is an extension of k-means, the algorithm can search the design space and found an ideal number of clusters. The algorithm starts with 2 clusters. Once the closeting process is finished, the split decision is evaluated by calculating the Bayesian Information Criterion, and then update the number of clusters. This process will repeat until the stop criterion is met.

The C/C++ code is generously provided by the author, after reaching out to him. However, the code is developed in late 1990 under the UNIX environment, some compiler is no longer available, and the instruction is out of date. At the time of writing this report, we are still working on having it work on Windows 10/Matlab environment.

Other time similarity measurement

For time series similarity measurement, only DTW is tested. Other methods, such as the hidden Markova model, or k-shape clustering may provide more insight and should be performed in the future.

Supervised Learning

As a parallel pool and a comparison set of the text representation method, the supervised training can be performed after the data preprocessing is done. Since the processed data will be clustered and "labelled" (using the cluster number), a supervised hypothesis can be trained using the DTW-based segmentation. By keeping the physical method of the segments (velocity profile), the cycle classification is expected to have a better result if the whole cycle data is given. However, the main difficulty is to predict the cycle with partial driving information.

Task Breakdowns

Since the work plan has been modified from the original proposal, a Gantt chart showing the new project plan and task breakdowns have been attached to the Appendix. Overall, the original timeline still applies. At the current stage, one full project design cycle has been performed. A few outstanding action items have been identified and will be resolved during the design iteration phase.

Initial Results

Data processing

After performing the process described in 0, similar driving data are grouped into the same cluster, as shown in Figure 4. The figure on the left presents two driving data in that cluster and four driving data on the right from another cluster. For the driving data from the same cluster, one can consider them travel on the same route under the same traffic condition, thus it is feasible to apply the same offline-optimized control parameter to this group.

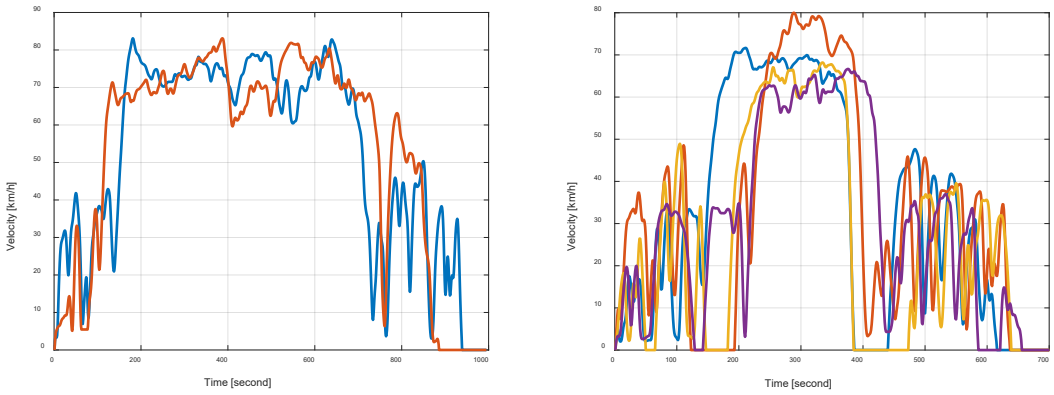


Figure 2 Similar Driving Data

Driving cycle reconstruction

Although the implicated features and combine of implicated feature and speed are evaluated, the result shows an inferior quality. Thus is not presented here but in Appendix C. The detailed list of tested methods is shown in Table 1, Appendix C.

The reconstructed driving cycle using the center of the cluster is shown in Figure 3, k-means, c-means, and DTW clustering shows good performance close to each other, where the blue line (UDDS) represents the original driving cycle. All three methods have a Euclidean distance of around 22. However, the DTW is chosen. Figure 4 and Figure 5 illustrate the difference of different method, for the k-means (and c-means), some cluster shows similarity, as shown in the left figure in Figure 4, but the right figure shows the algorithm groups symmetry driving data rather than similar data.

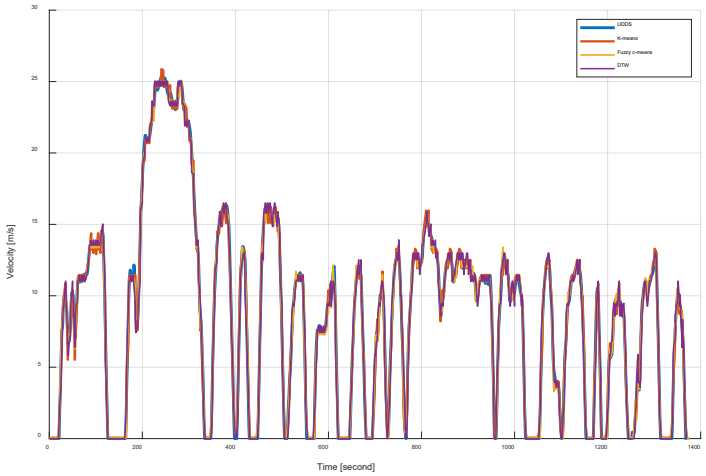


Figure 3 Reconstrued UDDS Driving Cycle

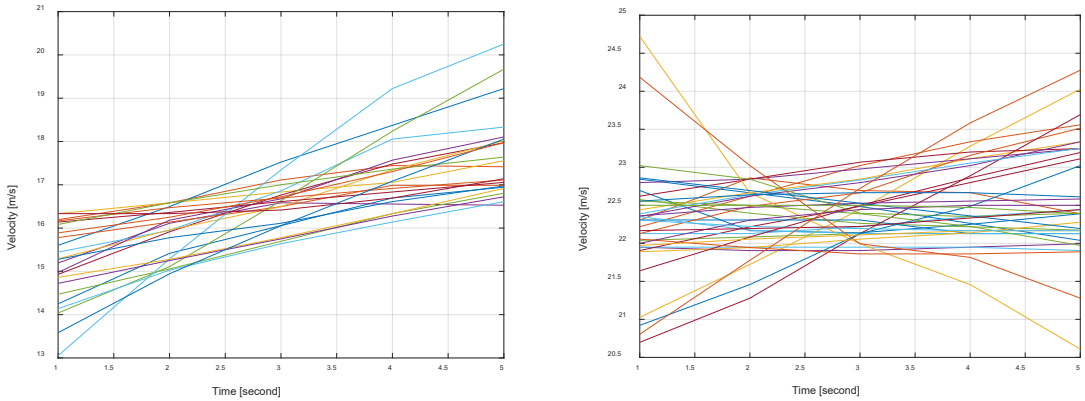


Figure 4 Similar pattern in a cluster using k-means clustering

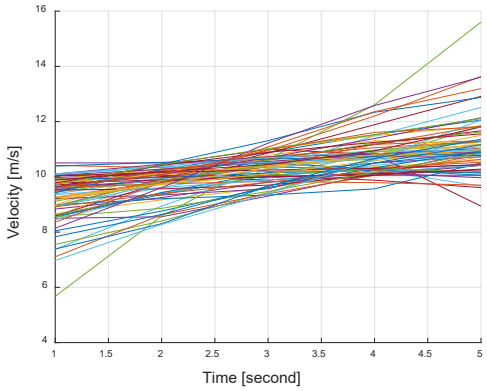


Figure 5 Similar pattern in a cluster using DTW clustering

The Eblow method is used to find the optimal cluster number. The eblow method result for DTW is presented in Figure 6 and 48 is chosen for this data set.

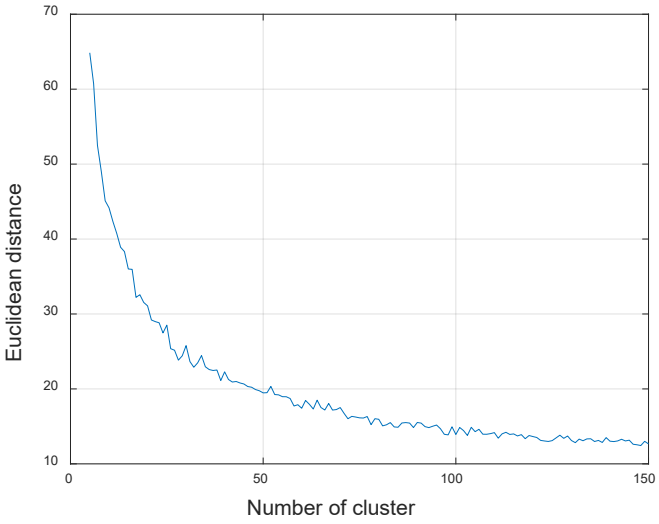
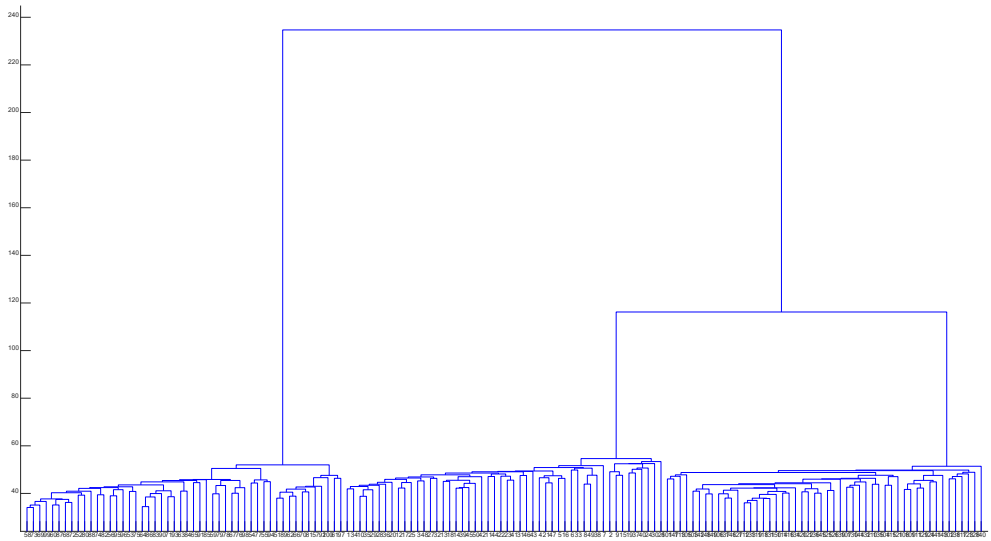


Figure 6 Elbow Method


240
220
200
180
160
140
120
100
80
60
40

Appendix A

Dataset

It is worth noting that all data to be used is the vehicles' speed profiles during the mission, the "types" mentioned at the beginning refers to the sources of the data. The first type is the "standard driving cycles" that are developed or well-recognized by governments, industries, and academia. Famous data includes Urban Dynamometer Driving Schedule (UDDS), China Light-Duty Vehicle Test Cycle (CLTC), Japan Fuel Economy Standards (JC08), etc. Such typical cycles are mainly obtained from the ADvanced VehIcle SimulatOR (ADVISOR) developed by US National Renewable Energy Laboratory (NREL). The second type is the house-hold vehicle driving data. Such data represents the driving habits of the general public. The specific dataset selected is the 2010-2012 California Household Travel Survey (CHTS) provided by the Transportation Secure Data Center of the NREL. The mixed data sources ensure the data representation from both certified standards and the general public.

Appendix B

For K-means and fuzzy c-means and fuzzy c-means, it is important to properly select variables and features for clustering. The vehicle speed is considered in this study, the possible useable features and their combination can be summarized as below:

- The speed at each time steps.
- Implicit representation, such as average speed, maximum velocity, minimum velocity, acceleration time ratio, deceleration time ratio, cruising time ratio, idling time, mean positive acceleration, mean negative acceleration, etc...
- Combination of both.

The speed is a direct indication of the driving condition. And is tested in the experiment.

The speed at each time steps seems like an ideal feature intuitively; however, most of the researchers in this field uses the implicit representation of the speed as features 【】 【】 【】. By using the implicit representation of the speed, the result of the clustering can easily find the physical representation of the cluster. Another benefit of using implicit representation is that the number of clusters can be significant reduced under certain conditions [cite: Optimal energy management strategy for a plug-in hybrid electric commercial vehicle based on velocity prediction]. Combining speed and implicit representation appears to be an ideal approach. However, the result obtained from the experiment suggested that this approach is the worst among the three methods.

When using implicit representation, dimension reduction is necessary. Intuitively, some features, such as mean velocity and travel distance, can be treated as linear under some conditions (e.g. when the segments have a time period of 1 second). Obtaining lower-dimensional data, while preserving as much of the data's variation as possible could potentially reduce the computational time of clustering, providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data. Principal component analysis (PCA) is performed in this experiment and the result shows that three features, acceleration time ratio, deceleration time ratio, and idle time, contain most of the information (75%).

Naturally, speed segments are sequences taken at successive equally spaced points in time. Dynamic time warping (DTW) is a robust method used to measure the similarity of time series. The experiment result is presented in XXXXXXXXXXXX

Dynamic time warping (DTW) is a robust method used to measure the similarity of time series and is developed in Matlab.

Table 1 List of Performed Clustering on driving data segments

Method	Input features
K-means	• Speed
C-means	
DTW	
K-means	• Average speed

C-means	• Maximum velocity • Minimum velocity • Acceleration time ratio • Deceleration time ratio • Cruising time ratio • Idling time • Mean positive acceleration • Mean negative acceleration
K-means	• Speed • Average speed • Maximum velocity • Minimum velocity • Acceleration time ratio • Deceleration time ratio • Cruising time ration • Idling time • Mean positive acceleration • Mean negative acceleration
C-means	

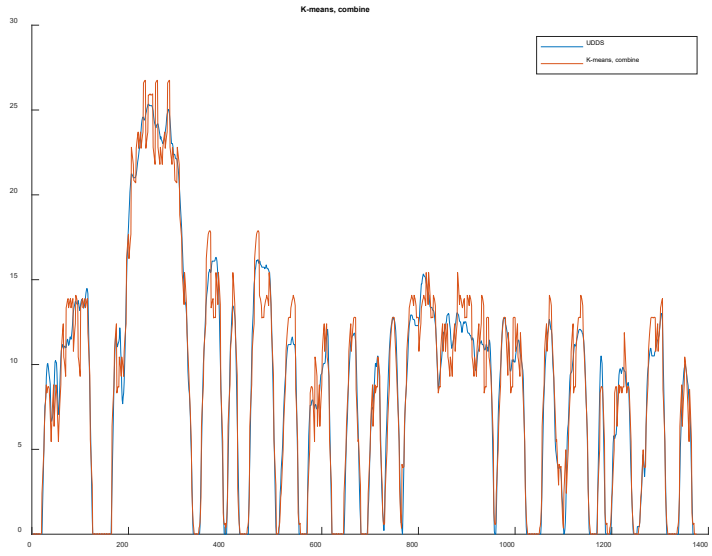


Figure 7 Reconstruct the dring cycle using a combination of speed and implicated features

Appendix C

	1 centers				
1 RS01	23.1877	23.1934	23.1625	23.1442	23.1292
2 RS02	4.7191	4.8535	4.9937	5.0944	5.2244
3 RS03	3.0629	3.5609	4.1335	4.7592	5.3830
4 RS04	11.1284	11.1837	11.2424	11.2626	11.2322
5 RS05	0.0143	0.0041	0.0038	0.0049	0.0185
6 RS06	19.9605	20.0367	20.0985	20.1221	20.0987
7 RS07	5.1208	6.2113	7.1638	8.0489	8.8693
8 RS08	12.4896	12.4655	12.4218	12.3625	12.2531
9 RS09	0.0879	0.1073	0.2573	0.8304	1.7771
10 RS10	26.2870	26.2923	26.3020	26.3413	26.3200
11 RS11	17.9752	18.0538	18.0669	18.0922	18.0605
12 RS12	7.5646	7.5867	7.6156	7.6340	7.6380
13 RS13	12.8612	12.9609	13.0453	13.0922	13.0973
14 RS14	6.5744	6.3860	6.1810	5.9069	5.5681
15 RS15	5.5922	4.4047	3.2108	1.9915	0.9970
16 RS16	3.8033	3.7100	3.6151	3.5180	3.4285
17 RS17	1.4456	2.1599	3.0247	3.8725	4.5019
18 RS18	16.0308	16.1036	16.1495	16.1602	16.1702
19 RS19	3.0721	2.0086	1.0817	0.4733	0.1991
20 RS20	4.7827	5.2263	5.6673	6.0794	6.4840
21 RS21	11.6729	11.8232	11.9301	12.0445	12.1573
22 RS22	14.3786	14.5012	14.5506	14.5812	14.5774
23 RS23	2.6798	2.6538	2.6272	2.4938	2.4488
24 RS24	10.3306	10.3550	10.2678	10.2059	10.0861
25 RS25	24.3724	24.3786	24.3752	24.3726	24.3884
26 RS26	31.7626	31.8048	31.7992	31.7967	31.7727
27 RS27	0.4072	0.7471	1.2966	2.0319	2.8801
28 RS28	8.5349	9.0082	9.5051	9.9494	10.3930
29 RS29	25.2743	25.2950	25.3238	25.3387	25.3118
30 RS30	15.3328	15.3469	15.3613	15.3699	15.3536
31 RS31	9.7057	10.1801	10.6168	11.0122	11.3090
32 RS32	21.9333	21.9356	21.9682	21.9714	21.9657
33 RS33	8.1148	7.0258	5.8474	4.5429	3.5095
34 RS34	9.2367	9.2200	9.1300	8.9948	8.8452
35 RS35	6.7753	6.8607	6.9168	6.9222	6.8716
36 RS36	4.2224	4.1502	4.0590	3.9860	4.0141
37 RS37	5.7372	5.5518	5.2761	4.9695	4.7095
38 RS38	9.9120	9.2541	8.3183	7.3523	6.3352
39 RS39	1.5741	0.8086	0.3246	0.1287	0.1069
40 RS40	28.1292	28.1188	28.1124	28.1117	28.0856
41 RS41	13.5584	13.6043	13.6581	13.6982	13.7269
42 RS42	21.0915	21.0742	21.0618	21.0574	21.0438
43 RS43	16.9852	17.0299	17.0657	17.0530	17.0255
44 RS44	11.7391	11.1992	10.5277	9.8234	9.1457
45 RS45	29.7827	29.7502	29.7828	29.7825	29.8456
46 RS46	35.1554	35.1957	35.2485	35.2779	35.2478
47 RS47	18.9988	19.0154	19.0414	19.0613	19.0307
48 RS48	7.1275	7.7842	8.3911	8.9419	9.4535

Appendix D

Two different approaches

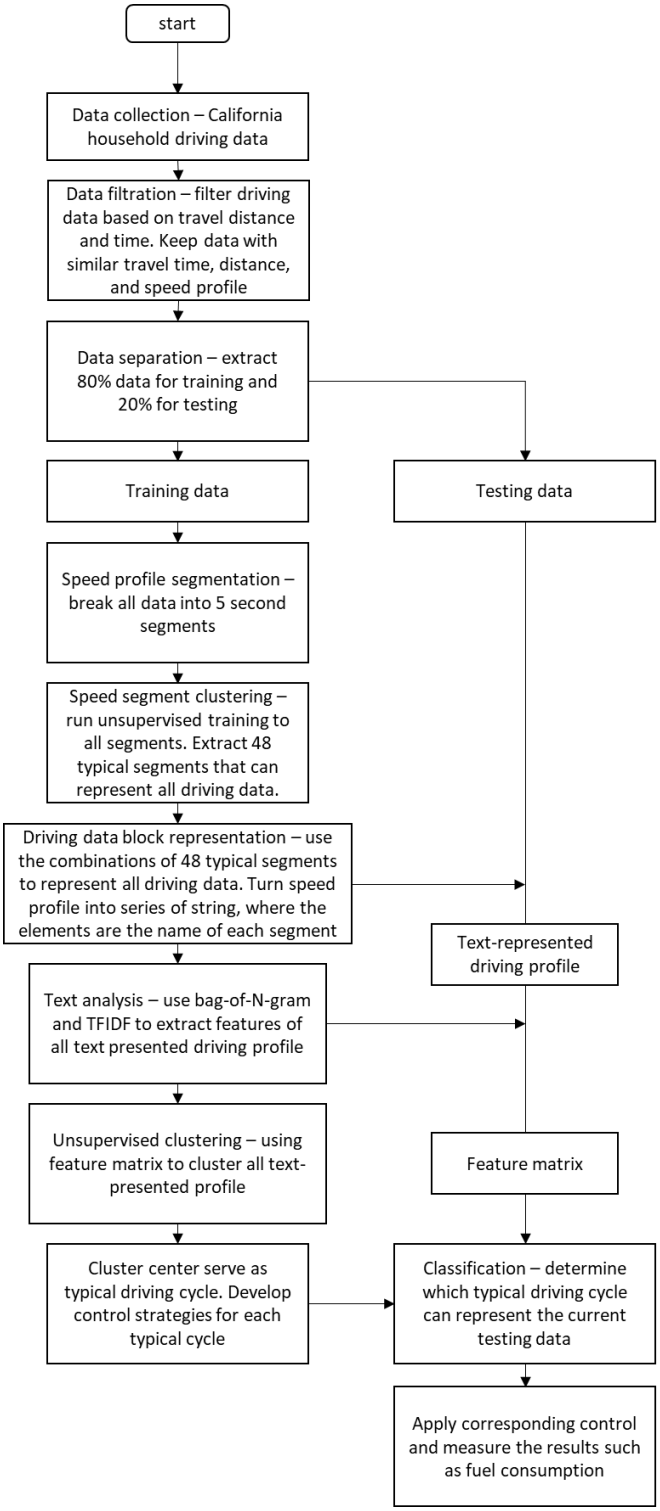


Figure 8 Unsupervised training approach.

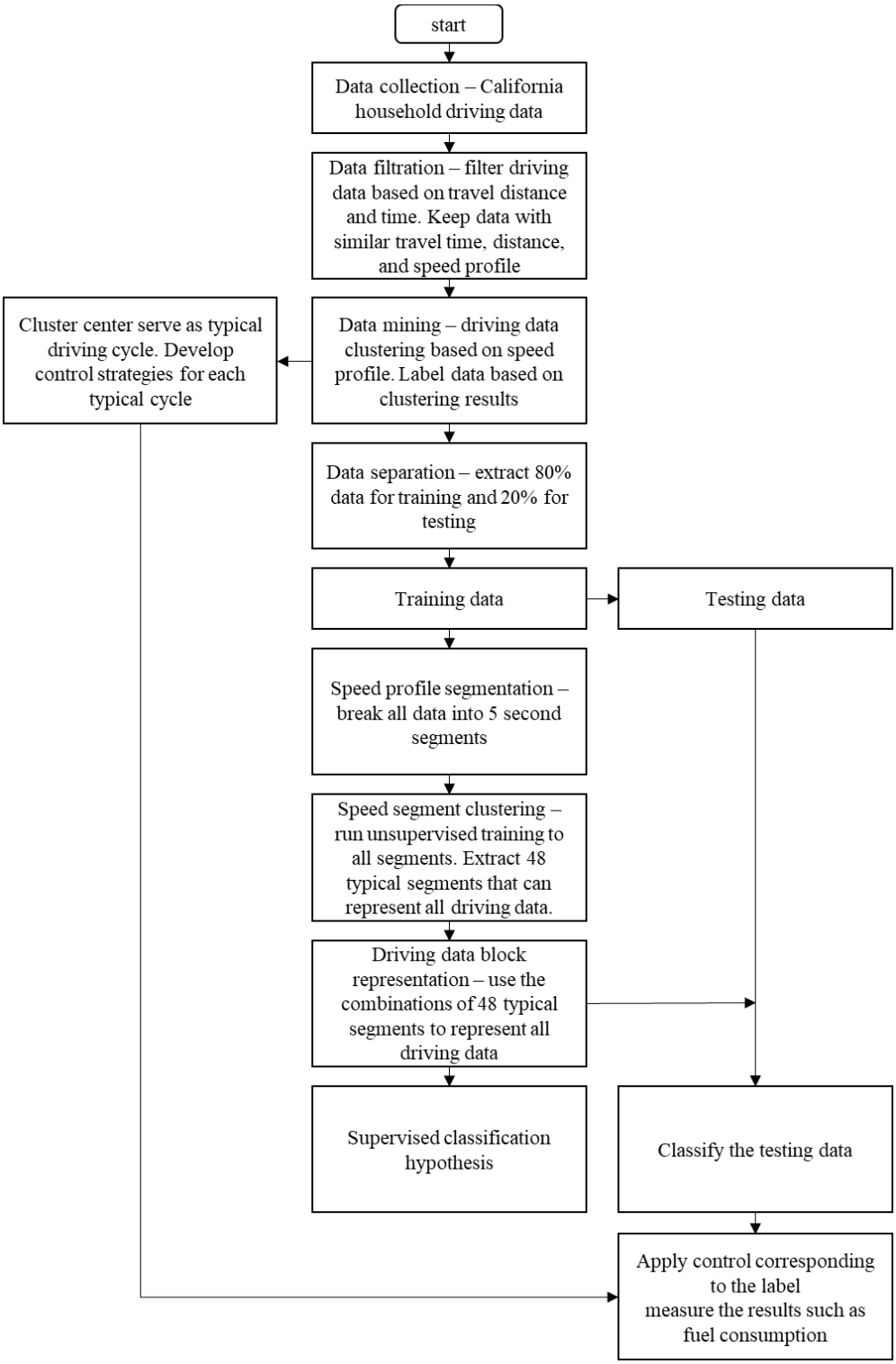


Figure 9 Supervised training approach.

Appendix E

Task Breakdowns

Driving Cycles Prediction

Company Name
Project Lead

Project Start Date: 2021-02-12
Scrolling Increment: 3

Milestone Description	Category	Assigned To	Progress	Start	No. Days
Data Collection	Milestone				
Data Collection	Goal	ALL: collect as many cycles as possible	100%	2021-02-15	5
Data Sorting	Low Risk	HZ: typical cycles SL: driving data SZ: typical cycles YY: driving data	85%	2021-02-19	2
Feature Extraction	Milestone				
Driving cycle selection	On Track	HZ, SL	95%	2021-02-21	1
Cycle break down	On Track	YY,SZ	95%	2021-02-22	1
Feature segments extraction	On Track	HZ,YY: K means SL,SZ: fuzzy C means HZ: DTW	95%	2021-02-23	2
TCPM Training	Milestone	ALL			
Cycle block representation	On Track	SL,HZ	95%	2021-02-25	1
Typical cycles extraction	Goal	YY,SL	95%	2021-02-26	2
Cycle based training	Goal	SZ,HZ	95%	2021-02-28	5
Control strategy development	Low Risk		50%	2021-03-04	5
Testing	On Track	SZ,YY	95%	2021-03-09	1
Design Iterations	Milestone	ALL			
Design Iterations	On Track		25%	2021-03-10	21
Documentation	Milestone	ALL			
Presentation	On Track		0%	2021-04-06	3
Report	On Track		2%	2021-02-15	57