Classification of Wheel Configurations using Geophones signals

Internship Report

Yaswanthkumar GOTHIREDDY

yaswanthkumar.gothireddy@rennes-sb.com

MSc Data and Business Analytics
Rennes School of Business, Rennes
France

Supervisors:
Supervisor 1, Supervisor 2
Barthélémy MORVAN , Jean-michel SIMONIN
LAMES, IFSTTAR, Université Gustave Eiffel
Nantes,France

September 2, 2020

Acknowledgements

Working at LAMES, IFSTTAR, Université Gustave Eiffel is interesting. Over the last six months, the support and the encouragement I got from my supervisors and my colleagues was exceptional.
I sincerely thank lab director Jean-michel SIMONIN and my internship supervisor Barthélémy MORVAN for accepting me to the Internship and lending their constant support and guidance over the entire period.
I have to thank Denis Lievre for his IT support and help to relocate the IT setup from my house back to the lab.
Finally, I have to thank Ruth Balshah, Cecile Auffray from Rennes School of Business and Gaelle Lebot from Université Gustave Eiffel for their administrative support.

Abstract

The objectives of the internship are

1. To do a bibliographic study to understand the applicability of Machine learning and Deep learning in pavement engineering.
2. Extract Features from the available data
3. Build accurate Machine learning and Deep learning models to predict wheel configurations utilizing data acquired by very few Geophones.

Contents

Acknowledgements i

Abstract ii

1 Introduction 1

1.1 Description of the test site 1

1.1.1 Fatigue Machine 1

1.1.2 The pavement structure 1

1.1.3 Description of the instrumentation 3

1.2 Description of the experiment 6

2 Literature Review 8

3 Data 10

3.1 Examples of Geophone signals 12

3.2 Features Extracted from the Geophone signals 18

4 Methodology 22

4.1 Phase 1: Design and data ingestion 22

4.1.1 Data Preparation 22

4.1.2 Exploratory data analysis 24

4.1.3 Feature Extraction 34

4.2 Phase 2: Proof of concept 37

4.2.1 Modelling algorithms 37

4.2.2 Model validation 37

5 Discussion 38

5.1 Machine learning 38

5.1.1 Machine Learning Algorithms used in this internship . . . 38

5.2 Deep Learning 43
5.2.1 Convolutional Neural Networks(CNN) 44
5.2.2 An Example CNN architecture from the built models . . . 44

6 Results **50**
6.1 Convolutional neural network model results 50
6.1.1 Validation curves 51
6.2 Feature based model results 52
6.2.1 With all variables: 52
6.2.2 After removing load variables: 52
6.2.3 Group wise models 55
6.3 Cross-validation and Optimization of the best model 61
6.3.1 Gridsearch CV 61
6.3.2 Scikit-Optimize 62
6.3.3 Optuna 64

7 Conclusion **65**

8 Limitations and Future Work **66**

Bibliography **67**

Chapter 1

Introduction

1.1 Description of the test site

1.1.1 Fatigue Machine

The fatigue merry-go-round is a major piece of equipment for IFSTTAR in Nantes. It is a traffic simulator allowing full-scale studies of pavements under accelerated heavy traffic. In its category, it offers unique performance notably by its size, but also by the speed of the loads which can reach 100km/hr at the end of the arm for 65 kN. The armory consists of a central turret and four arms at the end of which rolling loads can be attached, reproducing the axle configurations of common heavy goods vehicles. In the middle of the arm, an intermediate bearing rolling on a concrete ring helps stabilize the arms and limits dynamic overloads. The undercarriages running at the end of the arms have an original suspension system with low stiffness allowing the control of the applied loads. The loads can move laterally during the rotation of the ride to simulate the transverse sweep of road traffic.

1.1.2 The pavement structure

The track is located on ring B of the roundabout circled in red in 1.2. It consists of a half ring corresponding to the blue arrow. The arrow gives the direction of rotation of the carousel.

The experimental pavement is an existing thick bituminous structure . It consisted of 2.5cm of BBTM + 7cm ofBBSG + 34cm of GB3 + 40cm of GNT + 30cm of blocking in 80/150. For the experimental part relating to the project, 9cm of materials were milled. A new 7cm thick wearing course has been put in place.. It has been milled to 1/4 by 9cm deep for a new wearing course 7cm thick and for 1/4 by 6.5cm deep for a new working layer 4.5cm thick

Figure 1.1: Fatigue Machine at IFSTTAR, Nantes

Source: LAMES, IFSTTAR

Figure 1.2: Aerial View of the carrousel

Source: Google Earth

1.1.3 Description of the instrumentation

- **Load cells:**
  - Sterela supplied and installed 2 latest generation piezo quartz kistler type sensors (type G) and 2 polymer sensors from Meas-spec.
  - Fareco supplied and installed 2 piezo-ceramic sensors from Thermocax.
  - IFSTTAR supplied 2 Kistler piezo quartz type F sensors and 2 piezo-ceramic sensors from Thermocax. These sensors were installed by installers from the above companies.

  1.1 summarizes all the characteristics of the sensors. Companies have set up systems for the acquisition,automatic processing and transmission of their data. IFSTTAR has set up an acquisition system for all of these sensors described in more detail later in this report

- **Geophones and Temperature probes:**
  - Temperature probes were set up at different depths 2 pitches of the ring. These probes record the air temperature on the runway surface and at the following depths every 15 minutes : 2.5, 3.5, 4.5, 7, 7.5, 13, 25, 34 cm.To record the behavior of the track, it was instrumented with 2 sensors anchored 6m deep at the start and end of the section.
  - To complete the control of the mechanical condition of the roadway without disturbing its operation, geophones were placed on the surface of the milled layer before the installation of the wearing course and on the surface of the roadway near the anchored sensors and weighing.

1.3 shows the distribution of these sensors with:

- In red the geophone at the interface under the wearing course
- In blue the geophones on the surface of the road
- In magenta the 2 anchored sensors
- In cian (on the right of the figure), 2 geophones located 1m from the sensor anchored at a depth of 0.40m and 0.20m.

No.	Provider	Technology	Maker	Type	Serial Number	Sensitivity [pC/N]	Length [m]
1	IFSTTAR	Quartz	Kistler	9195F421	1964006	-1.752	1.75
2	IFSTTAR	Quartz	Kistler	9195F311	1904454	-1.761	1.50
3	FARECO	Céramique	Thermocax	VB FW CI1/3.6m/RG 58GT/45m (AS FF 801 REV8)	16663/01/15	0.97*	3.60
4	FARECO	Céramique	Thermocax	VB FW CI1/3.6m/RG 58GT/45m (AS FF 801 REV8)	16663/01/13	0.97*	3.60
5	STERELA	Quartz	Kistler	9195GC41	4635566	?	1.75
6	STERELA	Polymère	Meas-Spec	Roadtrax BL	JBL - 145068	53	2.00
7	STERELA	Quartz	Kistler	9195GC31	4470198	?	1.50
8	STERELA	Polymère	Meas-Spec	Roadtrax BL	JBL - 141675	53	2.00
9	IFSTTAR - C - 01	Céramique	Thermocax	VB FW CI1/3.2m/RG 58GT/50m (FF AS 801 REV8)	16662/13/21	0.97*	3.60
10	IFSTTAR - C - 02	Céramique	Thermocax	VB FW CI1/3.2m/RG 58GT/50m (FF AS 801 REV8)	16662/13/25	0.97*	3.60

Table 1.1: Characteristics of the load cells installed on the Pavement test bench

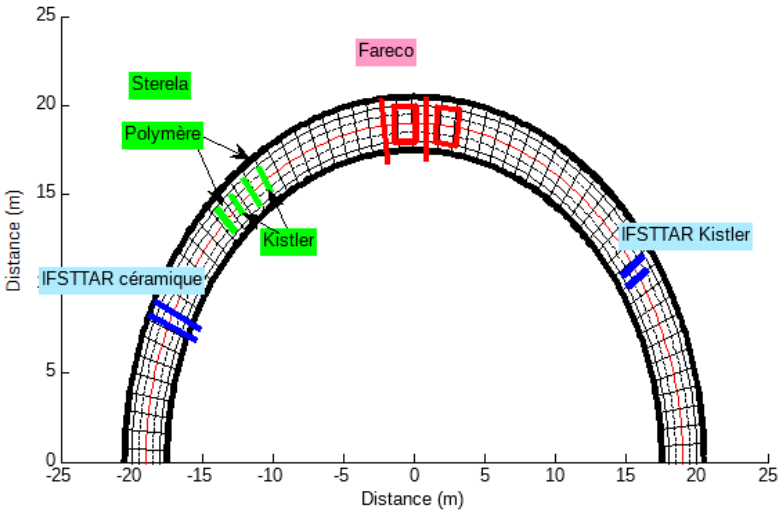

Figure 1.3: Positioning of the load cells on the CSA overload ring

Source: LAMES, IFSTTAR

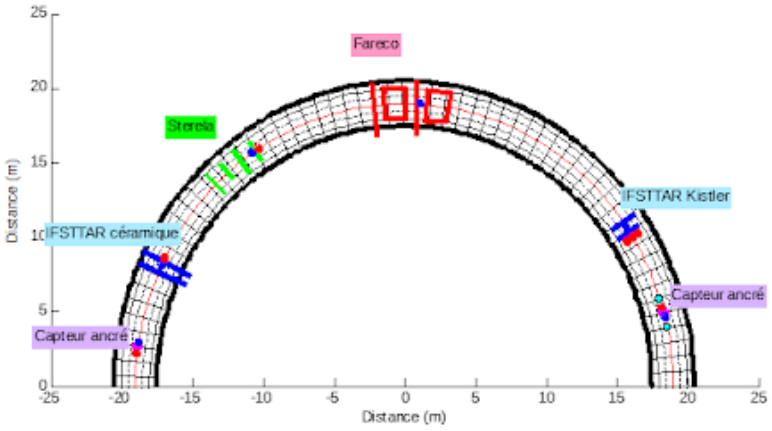

Figure 1.4: Positioning of the sensors on the track

Source: LAMES, IFSTTAR

1.2 Description of the experiment

After the development of the whole of the acquisition system, experimentation was conducted in 8 sequences.

1. **Sequence 1:** 4 identical arms, switching to 11 positions at 3 speeds (4.5, 7 and 10 rpm). This sequence was performed twice on June 23 afternoon (between 2 p.m. and 4 p.m.) and on June 26 morning between 7:25 a.m. and 11:44 a.m. During this second half day, the tests at 4.5 rpm were renewed at the end of the morning (10.57-11.54am) due to a malfunction of a charge amplifier

2. **Sequence 2a:** Variants in loads (45 kN and 55kN) and tire pressure (8.5 and 7 bars) ; The pressures are checked early test. A check at the end of the test shows an increase in this pressure in the tire. switching to 11 positions at 3 speeds (4.5, 7 and 10 rpm). This sequence was performed 2 times June 30 afternoon (between 13h42et 5:13 p.m.) and the 1 st July morning between 7:07 ET 10:00.

3. **Sequence 2b:** Variants in loads (45 kN and 55kN) and tire pressure (8.5 and 9 bars) ; The pressures are checked at the start of the test. A check at the end of the test shows an increase in this pressure in the tire. Change to 11 positions with 3 speeds (4.5, 7 and 10 rpm). This sequence was performed twice on July 2 in the afternoon (between 1:48 p.m. and 4:09 p.m.) and on July 4 in the morning between 7:09 a.m. and 9:35 a.m.

4. **Sequence 3:** Riding variants (Single wheel, twinning, tandem and tridem) ; Change to 11 positions at 2 speeds (4.5and 7 rpm). This sequence was performed twice on July 24 afternoon (between 2:01 p.m. and 3:59 p.m.) and on July 25 morning between 7:16 a.m. and 9:02 a.m.

5. **Sequence 4:** Test at 1 rpm with a plate placed above the weighing sensors so as not to press on it. These tests were carried out on July 7 between 10:02 a.m. and 11:28 a.m. with the exception of the test on the second ceramic sensor of the IFSTTAR carried out in the afternoon at 1:50 p.m. for safety reasons.

6. **Sequence 5:** Various tests for verification by comparing 2 similar situations but by reversing various components.

7. **Sequence 6:** Tests of speed increase or stopping of the merry-go-round to identify any natural vibration mode of the merry-go-round from the accelerometric signals.

8. **Sequence 7:** Test with an FWD load with 4 load levels, 3 repetitions by placing the FWD on the sensor, then 30cm further at a radius of 19m.

Figure 1.5: Different wheel configurations used in the experiment

Source: LAMES, IFSTTAR

9. **Sequence 8:** This sequence is similar to sequence 3 performed on November 6.

Chapter 2

Literature Review

The recent advances in using Artificial Intelligence in pavement engineering research have shown us the different approaches and techniques in predicting pavement defects, pavement management, Pavement distress forecasting, and structural evaluation of pavement systems etc. The Applications of AI in Pavement engineering research can be classified into two categories, Machine learning and Neural Networks.

Machine learning :

Pedro et al. used random forests for time series prediction of IRI for 5 years and 10 years using previous IRI measurements, structural, climatic and traffic data [1]. Shreedhar et al. used support vector machines, tried to detect debondings with Ground penetrating radar using trimmed A-scans as input and prediction of 1/0 for debonding presence or absence [2]. Osman et al. used Support Vector Machines, tried to predict airfield pavement responses(temperature, curling and bending strains) using the data collected by strain and temperature sensors on the JF Kennedy airport pavement [3].

Neural Networks:

The previous research in Pavement engineering before the evolution of Deep Learning mostly concentrated on approximating complex functions without generalization or considering the model complexity. Stephen et al. first used a computer vision based neural network model comprising Multi-layer perceptrons of 3 layers to develop an automated pavement evaluation system. Later, with due advances in Neural network research [4]. Attoh-Okine used an MLP-BP model to develop a pavement roughness progression model using several influencing factors such as pavement structural deformation, incremental traffic loadings, extent of cracking and thickness of surface layer, incremental variation of rut depth, surface defects such as patching and potholes, and environmental and other non traffic related variables such as road age etc. And, due evolution, several vari-

ations of the Neural networks such as Hybrid genetic adaptive neural network training(GNNT) for condition prediction of five different types of pavements, NN-knowledge based expert systems for determining respective maintenance and action strategies, Hankel- transforms as a forward model and Neural networks as an inverse model for the prediction of layer moduli from the FWD test data and surface wave measurements, an expert system based on Wavelet transform and radon neural network for classification of pavement distress from images have been published [5].

The recent advances in application of Artificial Intelligence also considered generalization and model complexity. Ostadi & Shelley developed a framework for evaluating ML models for pavement engineering research taking into account not only prediction accuracies but also generalization and model complexity [6]. Eisenbach et al. presented the GAPs data set(a pavement distress detection data set) with state of the art regularization techniques for generalization for road condition acquisition and assessment[7].

Utilizing costly sensors such as Weight In Motion to characterize road traffic do not only burdens the road managers, but also servicing them requires time consuming procedure which in turn stresses the road traffic. On the other hand, Geophones are less costly and have long been used to study deflection. In this research, we used minimum number of Geophones to accurately predict wheel configurations of the load on the Fatigue merry-go-around considering generalization and model complexity.

Chapter 3

Data

The acquisition of all the sensors is carried out using 3 acquisition systems. The first is on board the carousel. It includes 13 measurement channels: the 8 on-board accelerometers and the 5 detection signals (called top tour and top arm). The other 2 systems are stationary in a trailer at the edge of the runway. One acquires 17 channels: the 10 weighing sensors, the 2 anchored sensors, and the 5 detection signals. The other records 16 channels: the 15 geophones and the turret detection signal.

The experiment carried out in 1 month made it possible to collect 367 different measurement configurations, some 116 GiB of data. The various configurations envisaged were carried out, the only restriction being the temperatures which remained relatively homogeneous over the period allowing only to make observations over 2 temperature ranges.

The Data acquired is stored in the Matlab files. Each sequence with different configurations saved into different files. For example, Sequence1 with all the wheel configurations simple, with the same tyre pressure and force, with different displacement positions saved into different files. In total, In sequence1, we have 93 matlab files.

In each matlab file, we have 46 signals

No.	Measurement	Comments
1	Synchronisation signal C5	
2	Acceleration on arm 1 low level	
3	Acceleration on arm 2high level	
4	Acceleration on arm lower level	
5	Acceleration on arm 1 high level	
6	Acceleration on arm 3 lower level	
7	Acceleration on arm 4high level	
8	Acceleration on arm 4 lower level	
9	Synchronisation signal C1	
10	Synchronisation signal C2	
11	Synchronisation signal C3	
12	Synchronisation signal C4	
13	Acceleration on arm 1 high level	
14	Synchronisation signal C5	
15	Synchronisation signal C1	
16	Synchronisation signal C2	
17	Synchronisation signal C3	
18	Synchronisation signal C4	
19	Piezoquartz sensor from IFSTTAR 1	
20	Piezoquartz sensor from IFSTTAR 2	
21	Céramique sensor from IFSTTAR 1	
22	Céramique sensor from IFSTTAR 2	
23	Piezoquartz sensor for private company 1	
24	Piezoquartz sensor for private company 2	
25	polylmère sensor for private company 1	
26	polylmère sensor for private company 2	
27	Ceramic sensor for private company 1	Not available during sequence 1 Noisy signal during sequence 2 (and 3 ?)
28	Ceramic sensor for private company 2	Not available during sequence 1 Noisy signal during sequence 2 (and 3 ?)
29	Deflexion 1	
30	Deflexion 2	
31	Synchronisation signal5 C5	
32	Geophone 1	
33	Geophone 2	
34	Geophone 3	

35	Geophone 4	
36	Geophone 5	
37	Geophone 6	
38	Geophone 7	
39	Geophone 8	
40	Geophone 9	
41	Geophone 10	
42	Geophone 11	
43	Geophone 12	
44	Geophone 13	
45	Geophone 14	
46	Geophone 15	

Table 3.1: Signals in order of acquisition

The Sensors we used in this research are Geophones.

Besides the acquired data, experiment specific data is saved into an Excel file.

Excel file comprises four sheets

- **Sheet1:** Describing the 46 sensor specific data i.e sensitivity, where the sensor is situated on the test bed i.e., zone of each sensor, for ex: top, bottom(for the accelerometers), zones depicting in the fig, for each company etc., acquisition system i.e., wifi, fixed etc.,unit of each sensor output. I.e KN for WIM sensors, m/s for Geophones etc., and the number order in each acquisition system.
- **Sheet2:**Experiment specific data for each Matlab file, I.e., experiment name, time at which experiment was conducted, sequence number, temperature, speed, each arm torque, tyre pressure, displacement position, etc.
- **Sheet3:** Temperature extracted by the Acquisition system at a depths of 3.5cm.
- **sheet4:** Synthesis of sheet 3.

3.1 Examples of Geophone signals

The acquired data has been treated with a low pass filter of 500Hz to remove the acquisition sampling from the data. After that, the repeated signals has

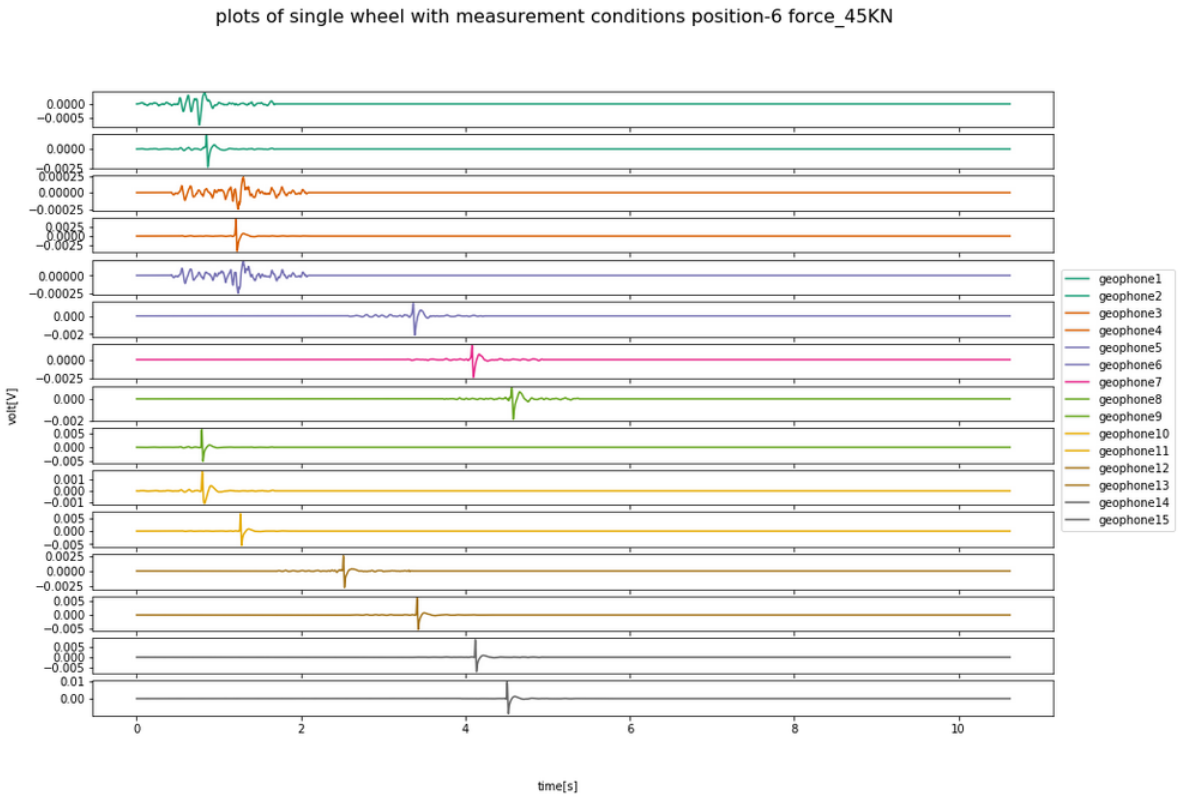

Figure 3.1: plots of Geophones signals from file 306

been masked. Treated data has been saved into Matlab files with file names representing the experiment number.

Three files [file 290, 295, 306] are considered initially to understand the geophone outputs. The measurement conditions of these files are different.

- file 290 measurement conditions are speed 4.5 rpm, transversal position 11
- file 295 measurement conditions are speed 4.5 rpm, transversal position 6
- file 306 measurement conditions are speed 7 rpm, transversal position 6.

several individual plots and overlapping plots have been plotted.

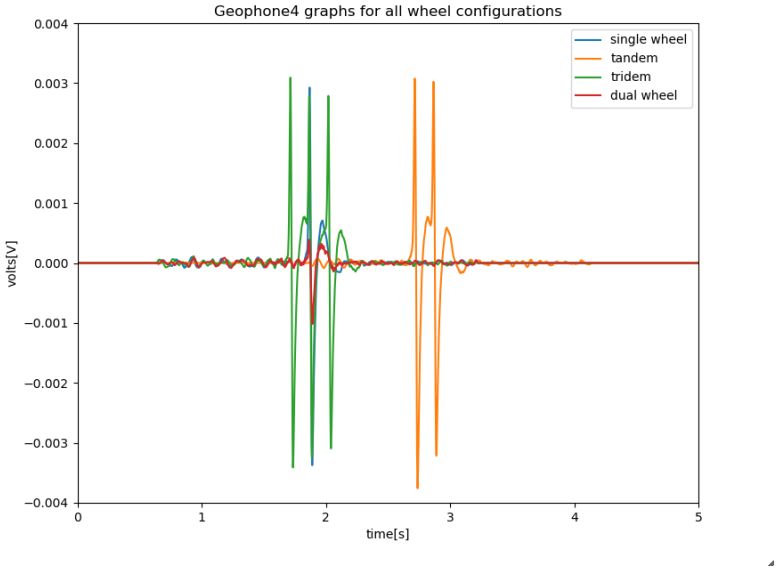

Figure 3.2: plots of all the wheel configurations from file 295 and Geophone 4

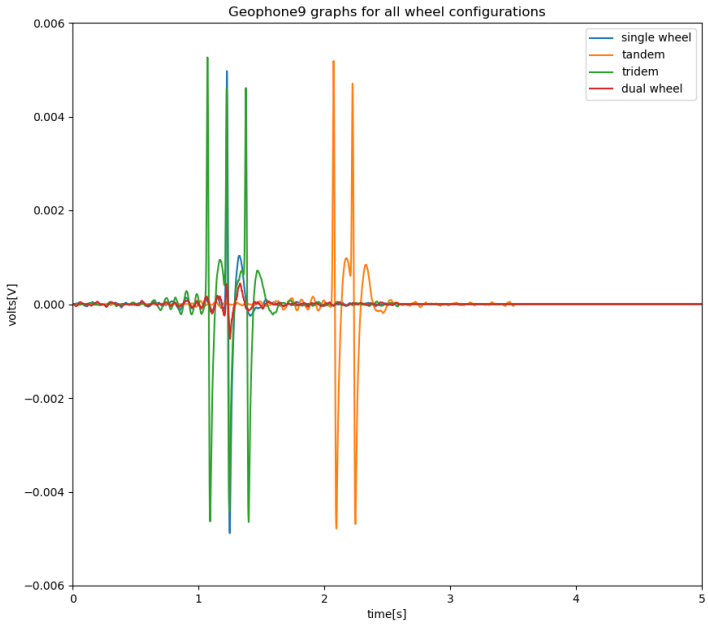

Figure 3.3: plots of all the wheel configurations from file 295 and Geophone 9

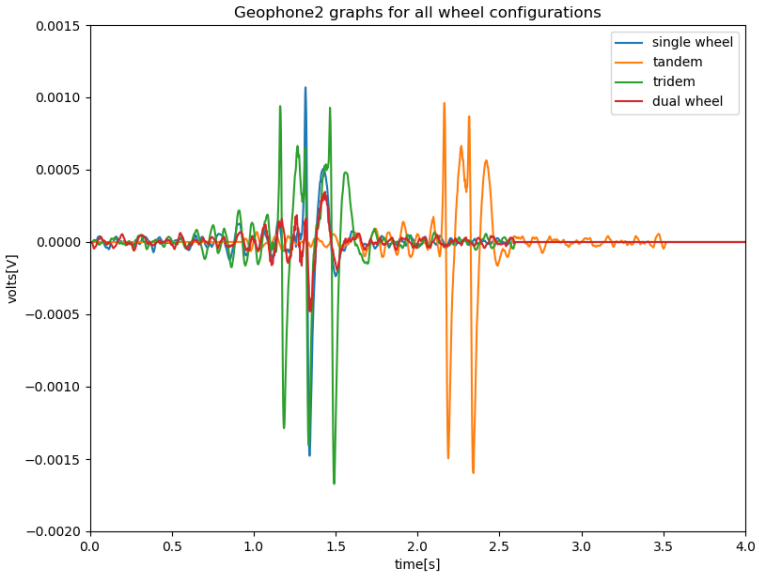

Figure 3.4: plots of all the wheel configurations from file 295 and Geophone 2

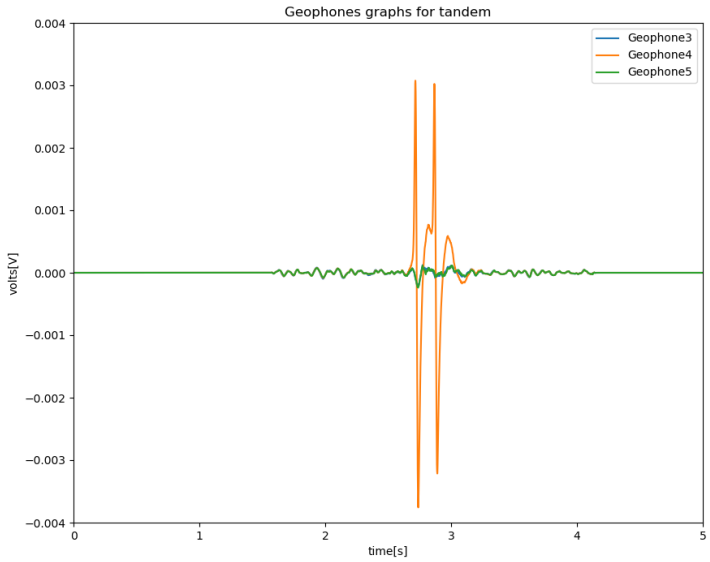

Figure 3.5: plots of tandem wheel configuration from file:295 and Geophones 3, 4, 5.

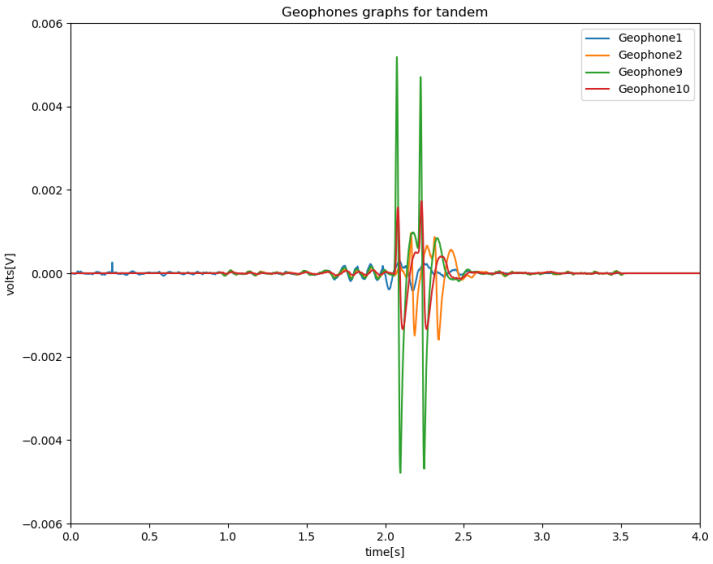

Figure 3.6: plots of tandem wheel configuration from file:295 and Geophones 1, 2, 9, 10

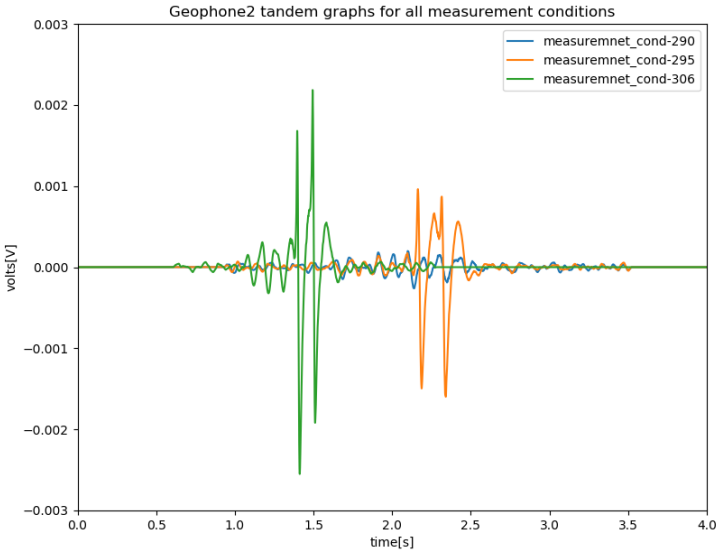

Figure 3.7: Overlapping Geophone 2 plots of the three files(file:290,295,306) with tandem wheel configuration

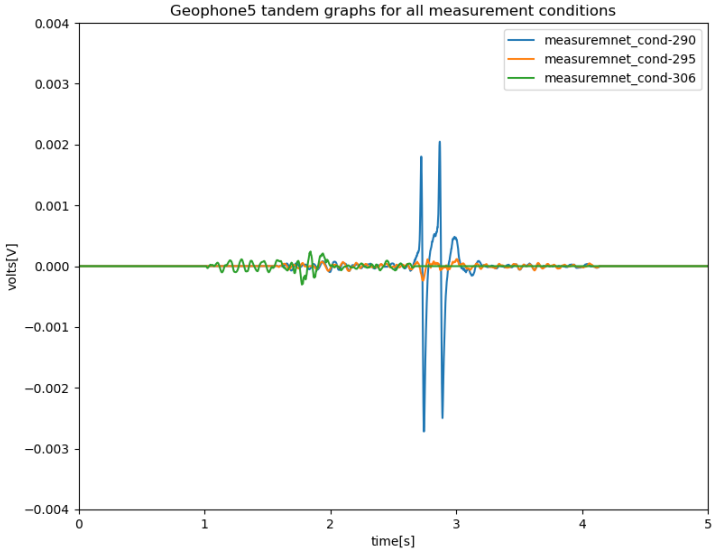

Figure 3.8: Overlapping Geophone 5 plots of the three files(file 290, 295, 306) with tandem wheel configuration

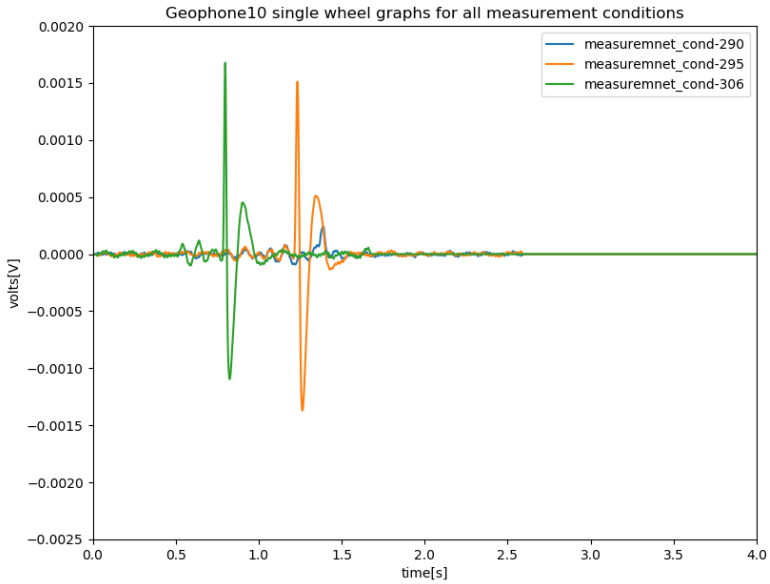

Figure 3.9: Overlapping Geophone 10 plots of the three files(file 290, 295, 306) with tandem wheel configuration

From the plots 3.1 to 3.9, the signals are of different lengths for different speeds. The outputs of the geophones vary even the transversal positions of the load remain same i.e., Geophones are placed at different transversal positions. Even at the same depths the outputs vary due to the position of the geophones.

3.2 Features Extracted from the Geophone signals

Different features have been extracted to build machine learning models.

The features have been classified in to

- Statistical features: maximum - mean, maximum negative peak, maximum positive peak, mean - minimum, signal maximum,signal mean, signal minimum, signal standard deviation
- Frequency features: fft amplitude, fft frequency, psd amplitude
- Obtained features: deflection, rms, total negative peaks, total positive peaks,peaks 10 mean, peaks 3 mean, peaks 2 mean, K1, K2, K3, Time differences of 15 Geophones.
- Configuration features: Charge (kN), Date/heure, Déplacement, Nom de projet, Numéro fichier, Pression (bar), Séquence, Température, Type roue, Vitesse (tr/min), depth of geophone, geophone no

Statistical features:

- **maximum - mean**: Difference between the geophone signal maximum and the signal mean
- **maximum negative peak**: Geophone signal maximum negative peak calculated with threshold
- **maximum positive peak**: Geophone signal maximum positive peak calculated with threshold
- **mean - minimum**: Difference between the geophone signal mean and geophone signal minimum
- **signal maximum**: Geophone signal maximum
- **signal mean**: Geophone signal mean
- **signal minimum**: Geophone signal minimum
- **signal standard deviation**: Geophone signal standard deviation

Frequency features:

- **fft amplitude**: maximum amplitude in fft of the geophone signal
- **fft frequency**: frequency at the maximum amplitude in the fft of the geophone signal
- **psd amplitude**: maximum amplitude in the psd of the geophone signal

Obtained features:

- **deflection**: deflection is the integration along the given axis using the composite trapezoidal rule.
- **rms**: Root mean square value of the signal
- **total negative peaks**: total negative peaks in the signal below a threshold
- **total positive peaks**: total positive peaks in the signal above a threshold

Threshold calculation:

File:295 and Single wheel Configuration						
	Threshold = 0.00005		Threshold = 0.0001		Threshold = 0.0002	
geophone number	total positive peaks	total negative peaks	total positive peaks	total negative peaks	total positive peaks	total negative peaks
1	10	8	7	6	2	1
2	7	7	4	4	2	2
3	8	8	2	1	0	0
4	8	8	2	2	2	1
5	9	9	1	1	0	0
6	9	9	2	2	2	1
7	11	10	2	2	2	1
8	7	8	4	3	3	2
9	7	8	6	4	2	2
10	4	4	2	2	2	1
11	9	7	2	2	2	2
12	15	13	6	5	2	1
13	8	6	4	2	2	2
14	7	6	2	3	2	2
15	11	10	2	4	2	3

Table 3.2: the number peaks with the change in the threshold value for single wheel configuration

File:295 and Dual wheel Configuration						
geophone number	Threshold = 0.00005		Threshold = 0.0001		Threshold = 0.0002	
	total positive peaks	total negative peaks	total positive peaks	total negative peaks	total positive peaks	total negative peaks
1	11	8	8	5	2	2
2	9	9	5	5	1	1
3	8	9	1	2	0	1
4	8	9	1	1	0	0
5	8	9	1	1	0	0
6	9	8	3	3	1	1
7	14	11	2	4	1	1
8	7	8	4	4	2	1
9	10	10	4	4	2	3
10	4	3	1	1	1	1
11	8	8	3	1	2	1
12	15	16	3	4	1	1
13	7	8	2	2	2	1
14	11	10	3	6	2	1
15	8	8	4	4	3	2

Table 3.3: the number peaks with the change in the threshold value for dual wheel configuration

By looking at the tables 3.2 and 3.3, we picked threshold value 0.0001V. threshold 0.00005 is very small and includes noisy peaks and threshold 0.0002 is very large and excludes the important peaks.

- **peaks 10 mean**: Top 10 positive peaks mean extracted using threshold value.
- **peaks 3 mean**: Top 3 positive peaks mean extracted using threshold value.
- **peaks 2 mean**: Top 2 positive peaks mean extracted using threshold value.
- **K1**: maximum peak extracted using threshold/ peaks 10 mean
- **K2**: peaks 3 mean / peaks 10 mean
- **K3**: peaks 2 mean / peaks 10 mean
- **Time differences**: Time differences between each geophone signal maximum to the other geophone signal maximums.There are 15 geophones so, we have 15 time variables ('Time 1(s)','Time 2(s)', 'Time 3(s)', 'Time 4(s)','Time 5(s)', 'Time 6(s)', 'Time 7(s)', 'Time 8(s)', 'Time 9(s)' ,'Time 10(s)', 'Time 11(s)', 'Time 12(s)', 'Time 13(s)','Time 14(s)', 'Time 15(s)). 3.10 shows the box plot of time differences.

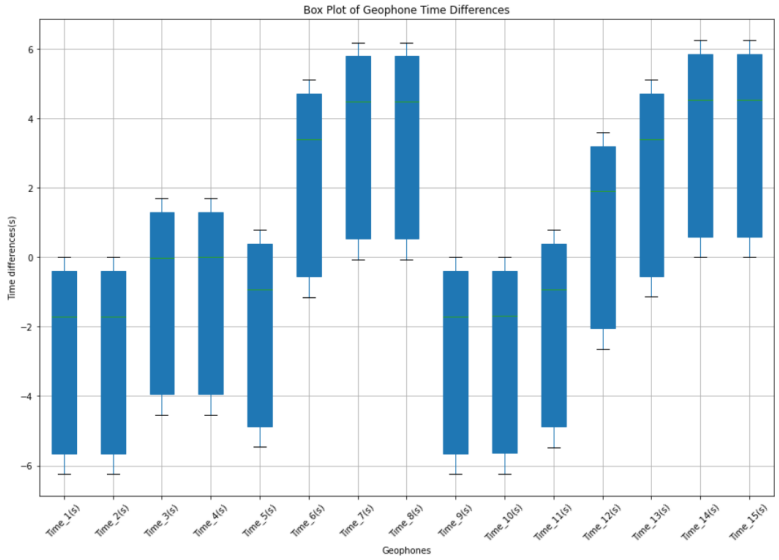

Figure 3.10: Time difference box plot for the file:435 in sequence 7

Configuration Features:

- **Charge(KN)**: load on each wheel
- **Date/heure**: experiment date and hour.
- **Déplacement**: lateral position of the wheel. There are 11 lateral positions on the test bed.
- **Nom de project**: Name of the project
- **Numéro fichier**: experiment number
- **Pression (bar)**: Tyre pressure
- **Séquence**: Sequence number in the experiment
- **Température**: External temperature
- **Type roue**: Wheel configuration
- **Vitesse (tr/min)**: speed of the test-bed
- **depth of geophone**: depth of Geophone situated on the test-bed.
- **geophone no**: Geophone number

Chapter 4

Methodology

In this Internship, We adapted the phased approach followed by [8].

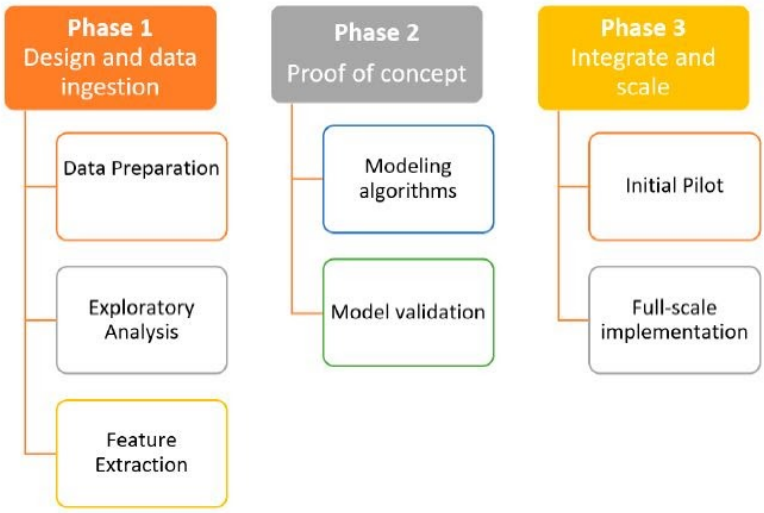

Figure 4.1: Three phase approach to develop advanced predictive model-adapted from [8]

Phase 1 and Phase 2 from 4.1 are implemented Where as, phase 3 is out of the context of this internship.

4.1 Phase 1: Design and data ingestion

4.1.1 Data Preparation

The data is in two formats i.e., matlab files and txt files. matlab files are treated files which are used for feature extraction. txt files are the raw data files.

Each txt file is named by its file number, geophone number and type of the wheel for example: "0290_G06Jum.txt".

- 0290 - file number
- G06 - geophone number
- Jum - wheel configuration

The data in each file is for a single geophone output for a particular wheel configuration.

We grouped txt files dataset into 5 Groups

- Group 1: Geophone 1 and Geophone 2
- Group 2: Geophone 3, Geophone 4, Geophone 5, Geophone 6, Geophone 7, Geophone 8, Geophone10
- Group 3: Geophone 9, Geophone 11, Geophone 12, Geophone 13, Geophone 14, Geophone 15
- Group 4: Geophone 3, Geophone 4, Geophone 5, Geophone 11
- Group 5: Geophone 1, Geophone 2, Geophone 9, Geophone 10

txt files are used to construct Neural Network models. The txt files are the raw data files of shape 8001*10. Each column is the raw data output of 1 second length from the geophone. The raw data is first checked for 'NaN' values and are dropped if found. 'NaN' Values are appended in each file to adjust to the different lengths of the raw data and produce the same shape matrices.

As we are working on sequence 3 files, the output labels are extracted from the file names. Each file is named by its file number, geophone number and type of the wheel for example: "0290_G06Jum.txt" The output labels are extracted from the files names and used for creating a output matrix. The output matrix length must match the number of input signals to the neural net. Each file has 10 repeated signals. we augmented the output matrix accordingly.

we constructed a large matrix of all the signals from a group for input and corresponding label matrix for output.

Matlab files are used to construct feature based models. Each Matlab file consists of 15 geophone signals for a wheel configuration. As shown in 3.1. Each Geophone signal consists of single output of the processed data and the masked values the rest of the time. The lengths of each signal varies according to the speed of the test bed. We manually extracted individual signals according to the speeds.

And then, we extracted as specified in 3.2, the statistical, frequency, and obtained features from the data.

An Excel file named "Essais CSA-Surcharge.xls" containing the configuration features was provided. Also, an accurate temperature file was provided. We merged the configuration features and temperature data with the extracted features from the matlab file and saved into a csv file.

4.1.2 Exploratory data analysis

Exploratory data analysis is used to summarise the data, understand different statistical features of the data in order to model the problem. One significant reason to do Exploratory data analysis is to find the outliers in the dataset. Outliers in our problem are either induced by data acquisition system, or by the sensor itself.

Group wise summary statistics for Group1

Statistical features of Group 1 sequence 3								
	Signal maximum	signal minimum	signal std	signal mean	mean-minimum	maximum-mean	maximum positive peak	maximum negative peak
count	352	352	352	352	352	352	352	352
mean	0.000357	-0.000465	0.000080	7.390416e-07	0.000465	0.000356	0.000334	0.000459
std	0.000236	0.000376	0.000046	1.385933e-06	0.000376	0.000236	0.000246	0.000380
min	0.000114	-0.002550	0.000024	-1.021041e-06	0.000078	0.000113	0.000000	0.000000
25%	0.000218	-0.000602	0.000048	-9.025741e-08	0.000200	0.000216	0.000201	0.000191
50%	0.000293	-0.000337	0.000067	1.126716e-07	0.000338	0.000293	0.000274	0.000335
75%	0.000416	-0.000200	0.000098	1.660974e-06	0.000603	0.000416	0.000382	0.000602
max	0.002184	-0.000075	0.000304	7.194593e-06	0.002550	0.002184	0.002184	0.002550

Table 4.1: sequence 3 Group1 summary statistics of statistical features

By looking at these tables 4.1, 4.2, and plots 4.2, 4.3, we understand that the ranges are different for some variables.

From box plots 4.5 to 4.6, there are a lot of outliers in our data.

Obtained features of Group 1 sequence 3								
	10 max peaks mean	rms	total neg-ative peaks	total positive peaks	deflection	3 max peaks mean	k1	k2
count	352	352	352	352	352	352	352	352
mean	0.000175	0.000030	4.232955	4.562500	0.015284	0.000312	2.119659	1.728723
std	0.000091	0.000007	1.928123	2.226891	0.028672	0.000195	0.524293	0.254965
min	0.000074	0.000019	0.000000	0.000000	-0.021352	0.000109	1.312788	1.190698
25%	0.000111	0.000024	3.000000	3.000000	-0.001861	0.000185	1.816540	1.555484
50%	0.000146	0.000030	4.000000	5.000000	0.002314	0.000252	2.057903	1.720642
75%	0.000212	0.000035	6.000000	6.000000	0.034478	0.000388	2.334620	1.870123
max	0.000821	0.000050	10.000000	10.000000	0.149138	0.001657	5.831355	2.748577

Table 4.2: sequence 3 Group1 summary statistics of obtained features

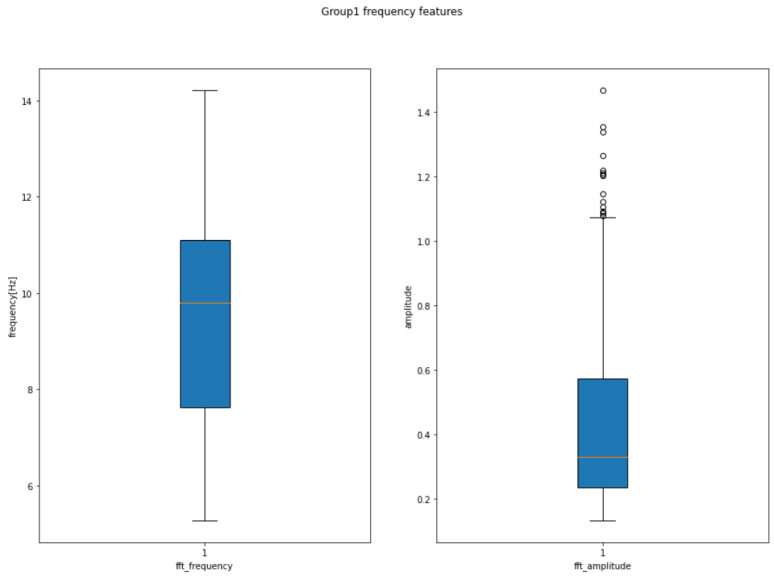

Figure 4.2: Frequency features of Group1

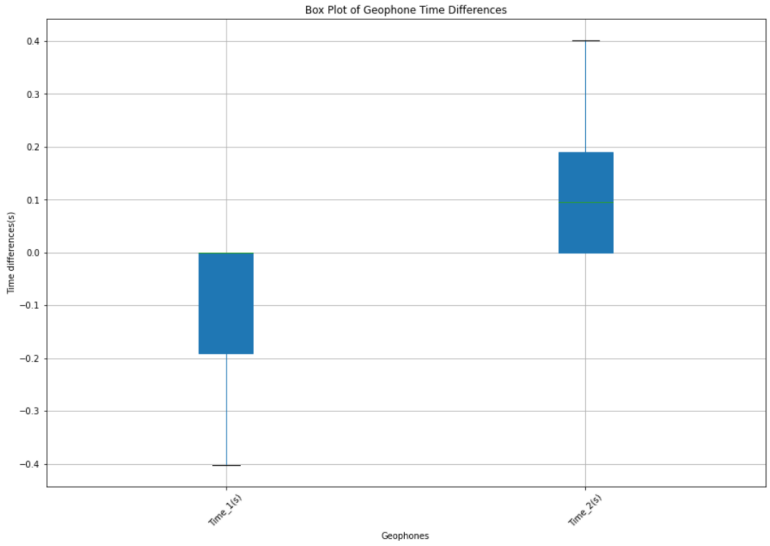

Figure 4.3: Time differences boxplot of Group1 times

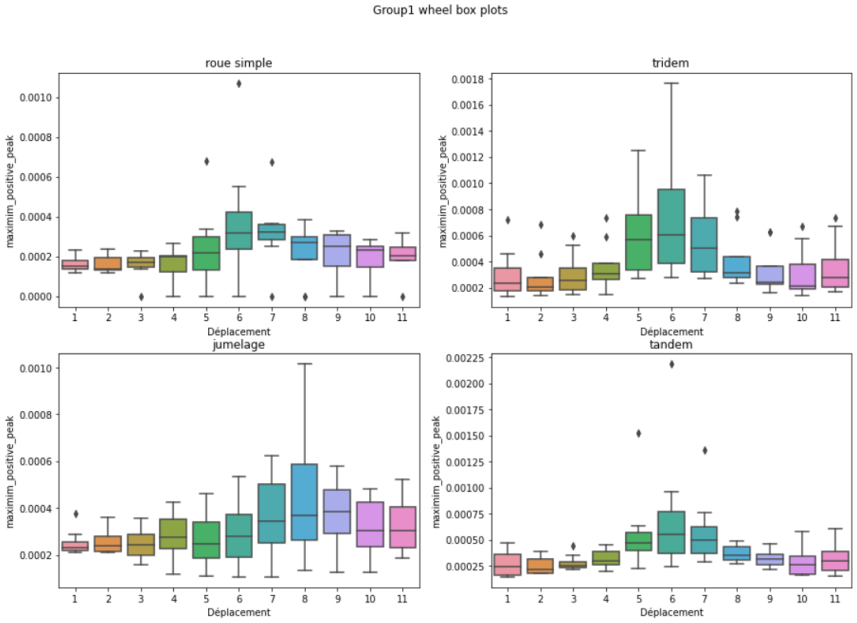

Figure 4.4: Group1 Boxplot of position vs maximum positive peak

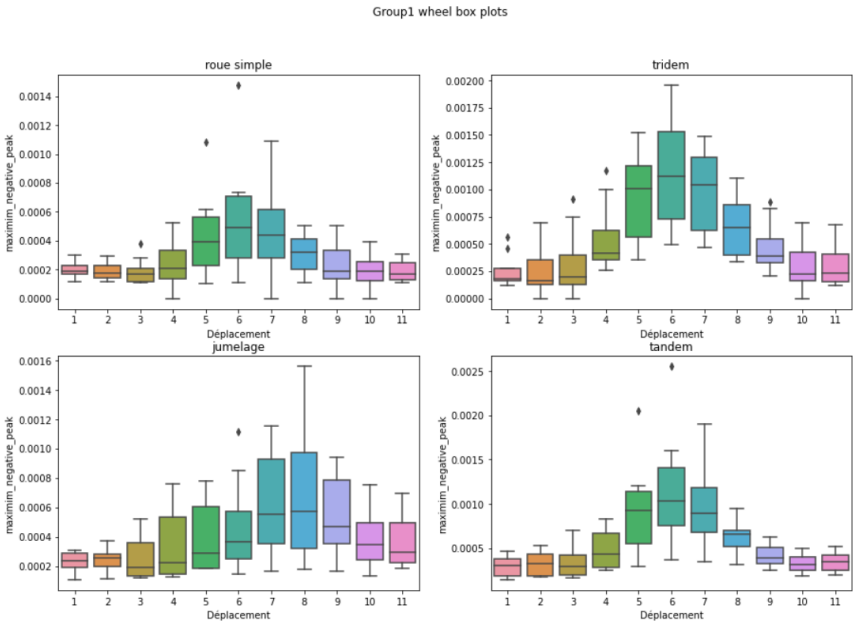

Figure 4.5: Group1 Boxplot of position vs maximum negative peak

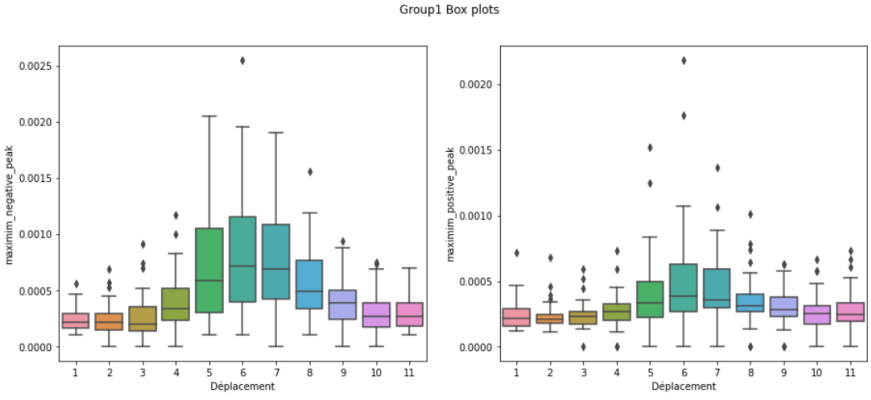

Figure 4.6: Group1 boxplots of position vs maximum positive peaks and maximum negative peaks

Group wise summary statistics for Group5

Statistical features of Group 5 sequence 3								
	Signal maximum	signal minimum	signal std	signal mean	mean-minimum	maximum-mean	maximum positive peak	maximum negative peak
count	704	704	704	704	704	704	704	704
mean	0.000429	-0.000516	0.000074	7.390416e-07	0.000516	0.000429	0.000397	0.000487
std	0.000915	0.000941	0.000119	3.703119e-07	0.000941	0.000915	0.000928	0.000955
min	0.000044	-0.006706	0.000008	-2.665598e-06	0.000036	0.000045	0.000000	0.000000
25%	0.000080	-0.000393	0.000017	-1.355209e-07	0.000081	0.000080	0.000000	0.000000
50%	0.000125	-0.000147	0.000034	6.468616e-08	0.000148	0.000125	0.000124	0.000147
75%	0.000264	-0.000082	0.000062	2.165915e-07	0.000393	0.000264	0.000264	0.000393
max	0.007366	-0.000036	0.000881	8.319107e-07	0.006706	0.007365	0.007366	0.006706

Table 4.3: sequence 3 Group5 summary statistics of statistical features

Obtained features of Group 5 sequence 3									
	rms	deflection	10 max peaks mean	3 max peaks mean	2 max peaks mean	k1	k2	total positive peaks	total negative peaks
count	704	704	704	704	704	704	704	704	704
mean	0.000030	0.000420	0.000202	0.000477	0.000584	2.714708	1.879149	1.603693	1.34659
std	0.000007	0.007660	0.000258	0.000802	0.000978	1.352666	0.531411	1.721106	1.32243
min	0.000019	-0.055127	0.000055	0.000064	0.000066	1.184990	1.125537	0.000000	0.00000
25%	0.000024	-0.002802	0.000083	0.000129	0.000143	1.820664	1.473398	0.000000	0.00000
50%	0.000030	0.001334	0.000115	0.000188	0.000216	2.306167	1.693167	1.000000	1.00000
75%	0.000035	0.004473	0.000183	0.000393	0.000472	3.066218	2.210857	3.000000	2.00000
max	0.000050	0.017195	0.002234	0.007030	0.007142	8.317134	3.191243	8.000000	5.00000

Table 4.4: sequence 3 Group5 summary statistics of obtained features

By looking at these tables 4.3, 4.4, and plots 4.7, 4.8 , we understand that the ranges are different for some variables.

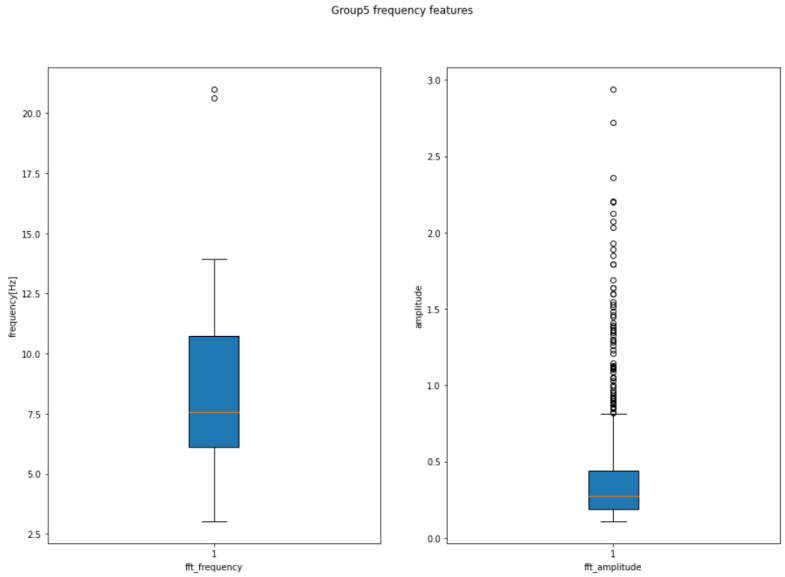

Figure 4.7: Frequency features of Group5

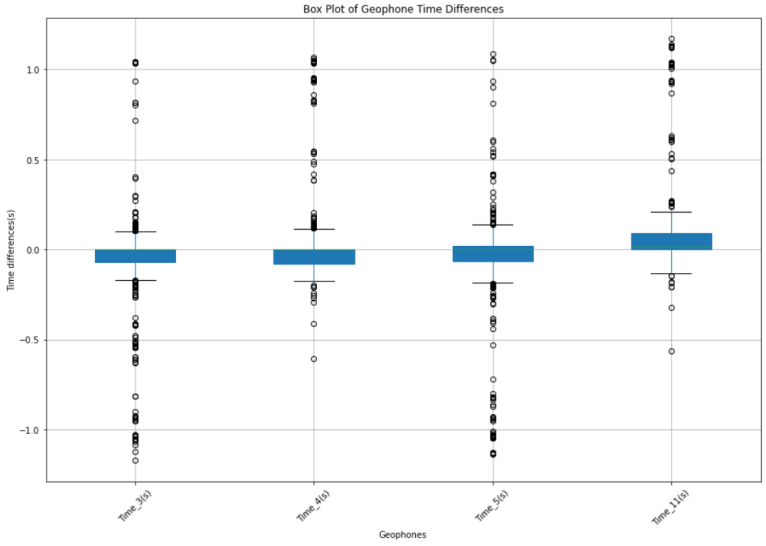

Figure 4.8: Time differences boxplot of Group5 times

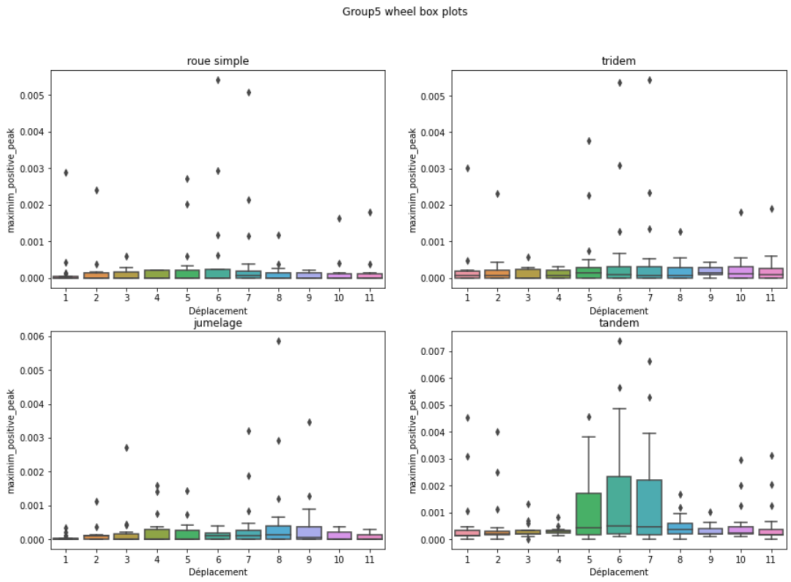

Figure 4.9: Group5 Boxplot of position vs maximum positive peak

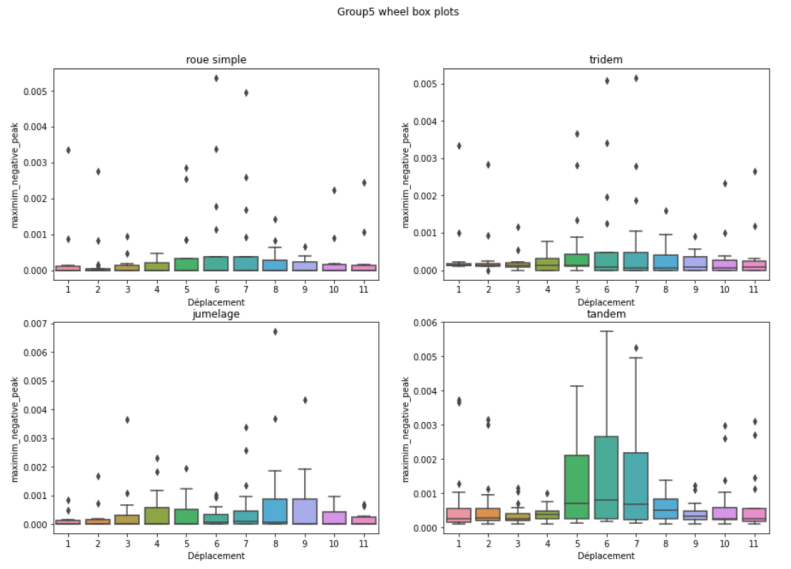

Figure 4.10: Group5 Boxplot of position vs maximum negative peak

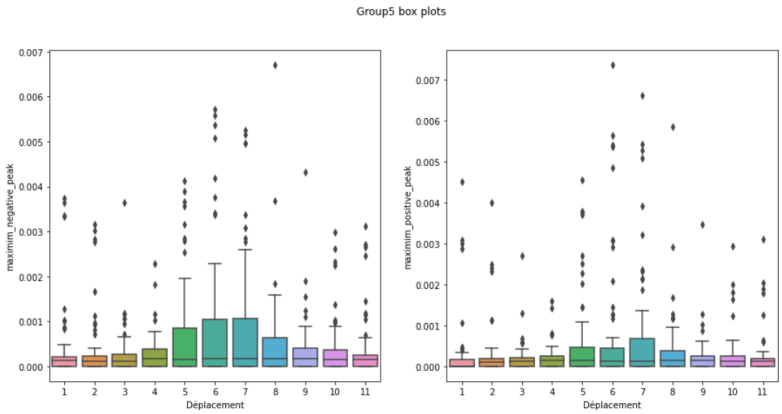

Figure 4.11: Group5 boxplots of position vs maximum positive peaks and maximum negative peaks

From box plots 4.9 to 4.11, there are a lot of outliers in our data.

This procedure has been repeated for other groups.

Outlier Analysis

To understand outliers, we plotted some of the geophone signals that are outliers along with the geophone signals that are obtained using same experiment configuration.

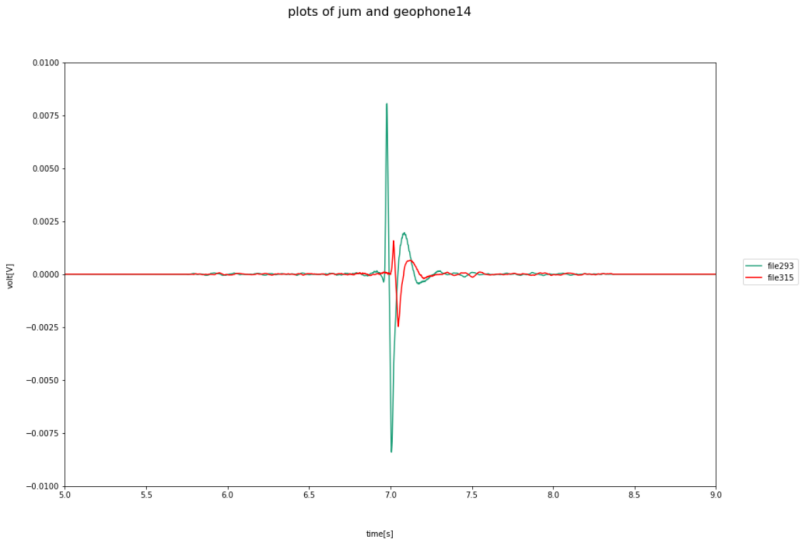

Figure 4.12: Group5 files 293, 315 geophone 14 signals [outlier in green]

From plots 4.12, 4.13, 4.14, 4.15, 4.16, we understand that the outliers are

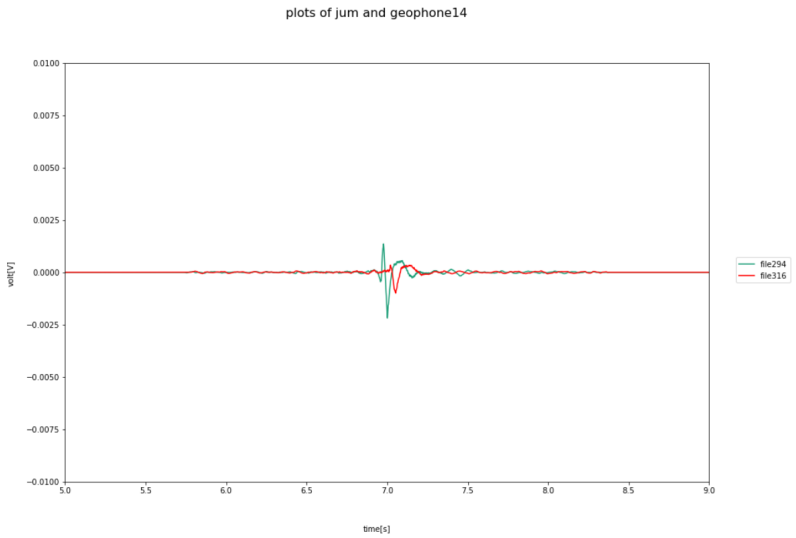

Figure 4.13: Group5 files 294, 316 dual wheel geophone 14 signals [outlier in green]

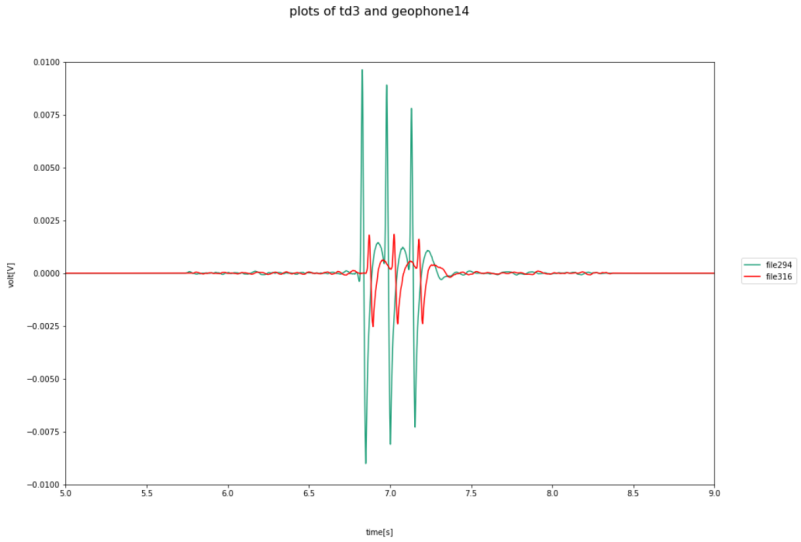

Figure 4.14: Group5 files 294, 316 tridem geophone 14 signals [outlier in green]

actually the good signals obtained. We cannot straight away drop this data from the entire data set. By dropping this outlier data, model may miss real signals from classifying the wheels correctly.

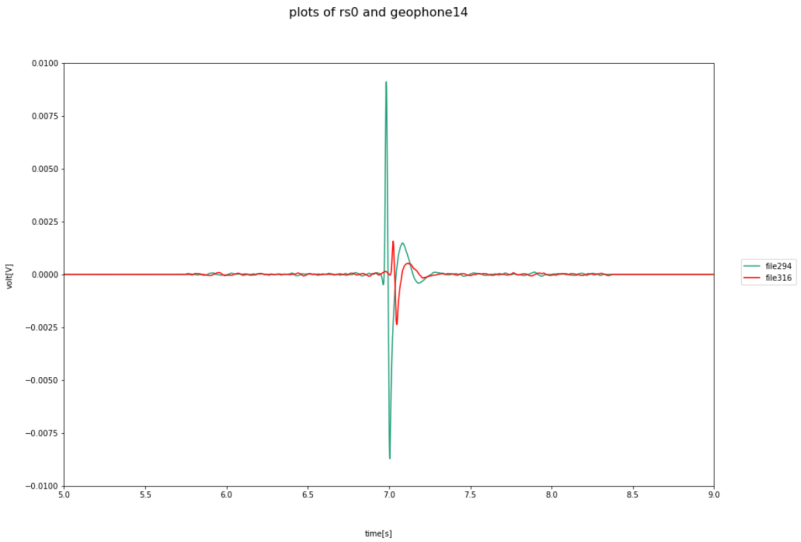

Figure 4.15: Group5 files 294, 316 single wheel geophone 14 signals [outlier in green]

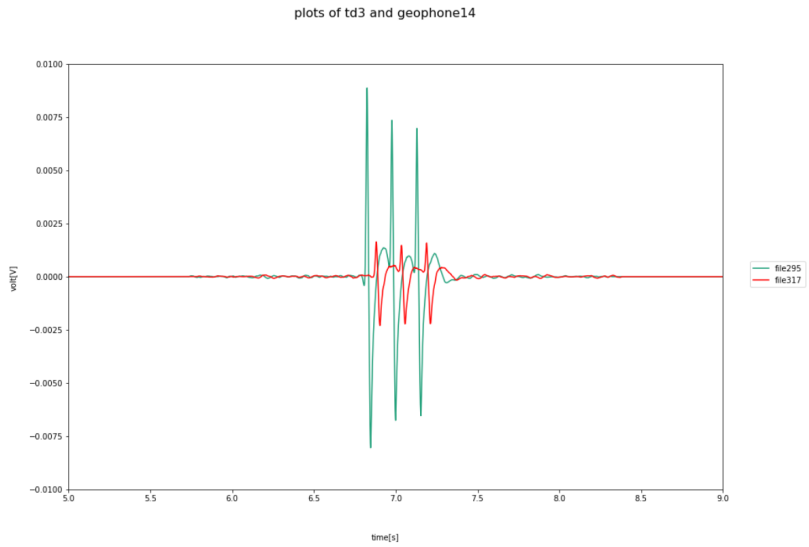

Figure 4.16: Group5 files 295, 317 tridem geophone 14 signals [outlier in green]

Correlation plot

Correlation plots are used to understand the dependence of the variables. To reduce the computational complexity of the models, we use correlation plots to remove highly correlated variables.

"Signal minimum" is highly correlated in 4.17 with "maximum positive peak",

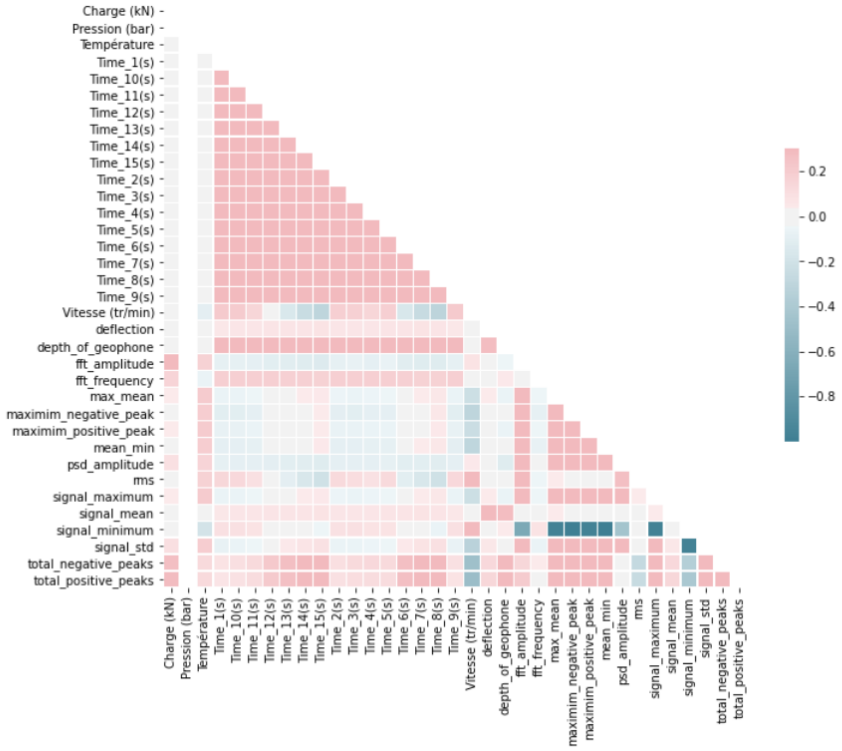

Figure 4.17: Correlation between all the variables in sequence 3

"maximum negative peak", "maximum - mean", "mean-minimum". We removed "signal minimum" henceforth.

4.1.3 Feature Extraction

Feature extraction is the name for methods that select and /or combine variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set.

Statistical and Obtained features in 3.2 are extracted using python packages Numpy and Scipy.

The number of features that we can extract from a signal are innumerable. The approach we followed here is to first extract both time domain[statistical features, rms etc.], frequency domain[fft amplitude, fft frequency] and the subject domain features[deflection, total positive peaks, total negative peaks] and build the machine learning models. If the model performance is not satisfactory, extract extra features.

Subject domain features:

Deflection: The integral of geophone output signal corresponds to the deflection of the pavement.This thought has led to calculate integral using trapezoidal rule. we used scipy's trapz method to calculate the integral.

Total positive peaks and Total Negative peaks: From the Geophone signals, the number of distinct peaks corresponds to the wheel configuration. This thought has led to extracting these peaks. Scipy has "find_peaks" method to calculate number of peaks. By carefully choosing threshold value and peak prominence, we can extract useful information. In the plot 4.18, we used a prominence of 0.00005 to extract as many peaks. The same values are used to extract positive peaks in plot 4.19.

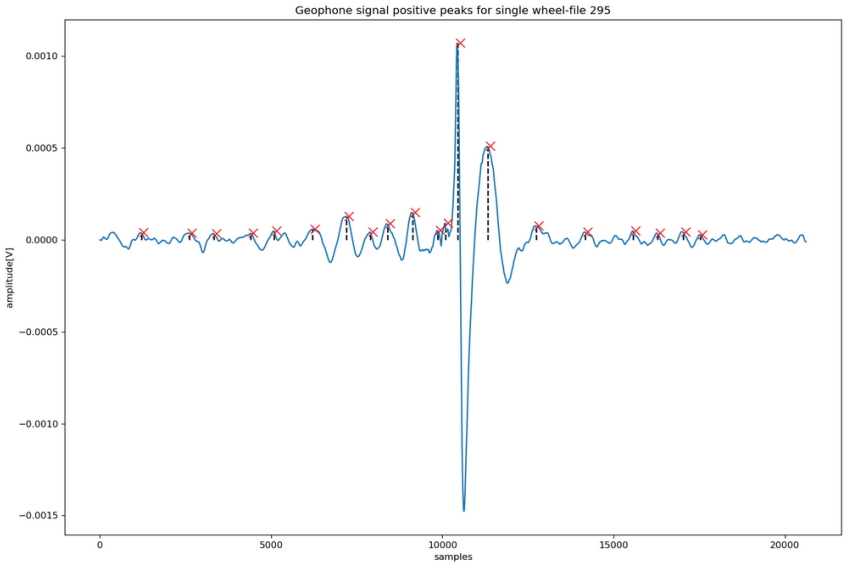

Figure 4.18: example plot of positive peaks extraction using prominence of 0.00005 for single wheel configuration

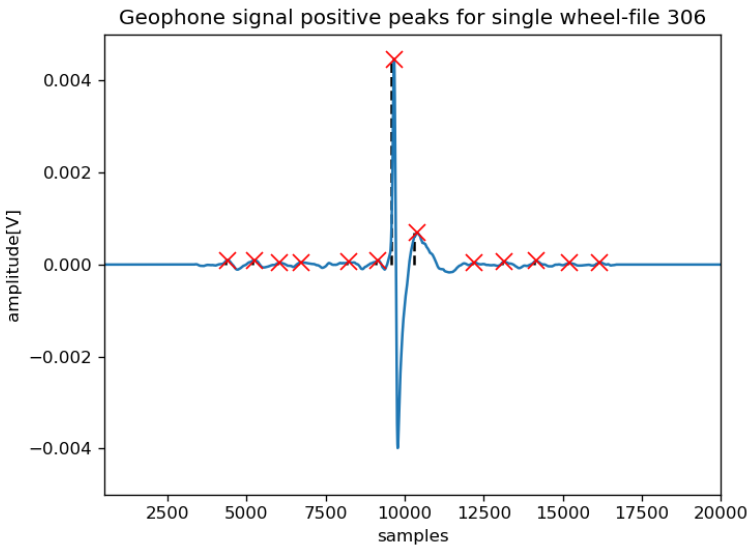

Figure 4.19: example plot of positive peaks extraction using prominence of 0.00005 for single wheel configuration

Time differences: Time differences vary with the speeds of the loads. As the geophones are placed at different locations on the test-bed, the signals acquired are at different times (4.20) as the load moves on the geophones. This thought has led to extract time based features.

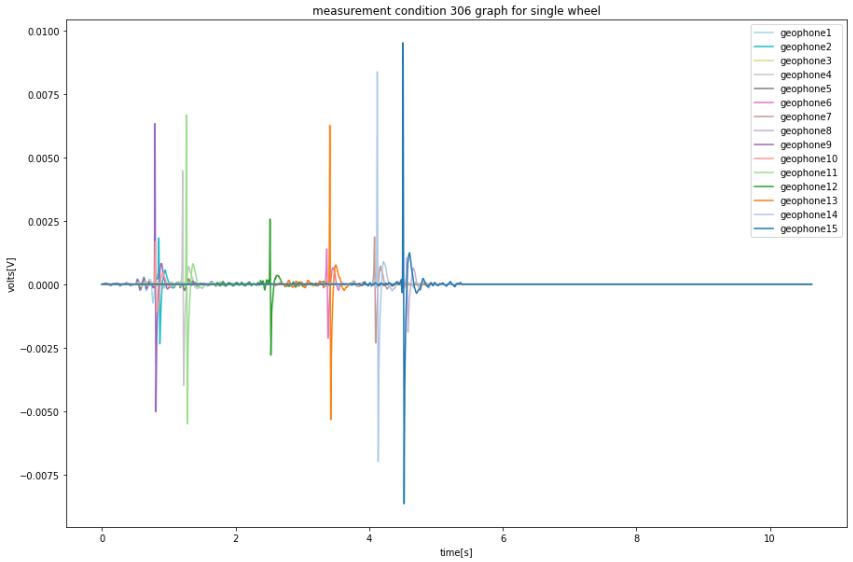

Figure 4.20: file 306 plots of all the geophone signals

The significance of these features have been studied by using feature impor-

tances in the Machine learning models.

4.2 Phase 2: Proof of concept

4.2.1 Modelling algorithms

We used both Deep learning and Machine Learning for model building. At first, we built Convolutional Neural Network models using raw data. Multiple models have been built on each group to get the model with best accuracy.

Second, we build Machine learning models on extracted features. we used two phased approach on ML models. First, models have been built on both categorical and numerical features. Second, models have been built on one hot encoded categorical features and numerical features with standardization.

Cross validation has been used to generalize well on unseen data. External Optimization packages such as Scikit-optimize and Optuna has been used to get better accuracies.

Finally, we compared the Deep learning and Machine learning models.

4.2.2 Model validation

We tested our best model using sequence 8 data, while sequence 3 and sequence7 have been used to build the model. While cross validating the model, we used all the three sequences data.

Chapter 5

Discussion

In this chapter, we discuss about the machine learning and deep learning algorithms, and tools used for building these ML models.

5.1 Machine learning

Machine learning is the study of algorithms that improve automatically with experience. Machine learning algorithms build a mathematical model based on sample data also called as "training" data, in order to infer upon without being explicitly to do so.

Machine learning algorithms are broadly classified into three categories based on the feedback available to the learning system.

- Supervised learning: The algorithm is provided with both inputs and outputs.And, the goal is to learn to map inputs to outputs.
  Algorithms: linear regression, logistic regression, support vector machines, naive bayes, decision trees, ensemble methods, etc.
- Unsupervised learning: No outputs to map to inputs. The algorithm tries to find patterns in the input data.
  Algorithms: K-means clustering, auto encoders, self-organizing maps, expectation maximization algorithm, etc.
- Reinforcement learning: The algorithm interacts with the dynamic environment where it must perform certain task(for example: driving a car). As the algorithm navigates through the problem space, it is provided with the feedback which is analogous to reward which it maximizes.
  algorithms: Q-learning, Monte Carlo, Deep Q Network, etc.

5.1.1 Machine Learning Algorithms used in this internship

- Decision trees

- Random forest algorithm
- Gradient Boosting classifier
- Ada boosting Algorithm
- Support vector machines with linear and Radial basis function kernels
- Multi layer perceptron algorithm
- K-Nearest neighbors algorithm

Decision trees:

Decision trees trains by recursive partitioning of the feature space. It will stop training when all the nodes are pure(homogeneous leaves). These trees can be very deep due to over fitting. In classification problem, the leaf node is assigned the label of the majority class. Impurity measures such as gini index or entropy are used for splitting the feature space. The advantage of decision trees over linear algorithms such as logistic regression are it can classify non-linear spaces, and works on numeric and categorical variables.

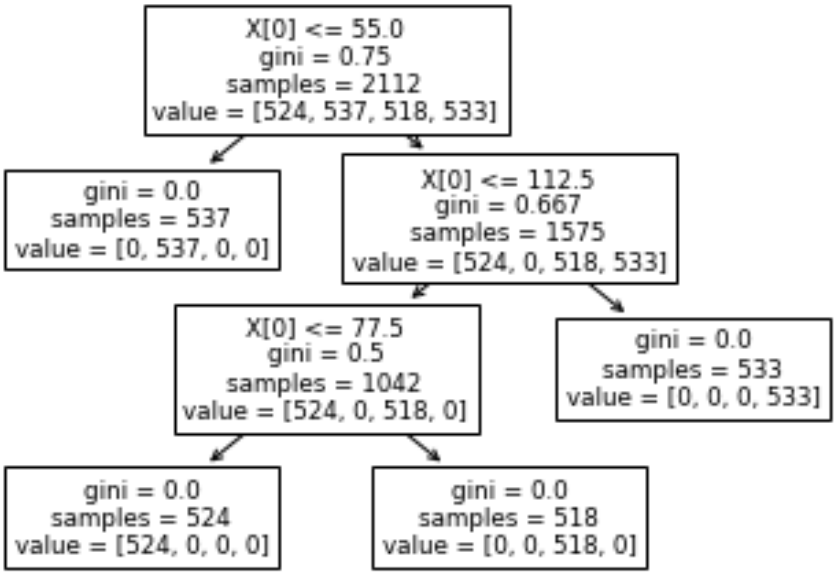

Figure 5.1: Decision tree taken from the model built on sequence 3 data

Random Forest algorithm:

Over fitting in Decision trees can be avoided in Random Forests.

Random forest algorithm is an ensemble algorithm. Ensembles are to build multiple base models and vote the best. In Random Forest we use an ensemble technique called bagging. Bagging is to change the training data for each classifier. Bagging is good for unstable learners because it reduces variance and over fitting. The disadvantage of Bagging is it is hard to interpret.

Random forests build a lot of decision trees. Random forests sample training data with replacement and build each model using a random subset of features. Finally, the strong predictors are voted.

Random forests improve accuracy and incorporate more diversity in the feature selection.

Gradient boosting:

Boosting is a technique which tries to build a large set of classifiers each focusing on observations hard to classify by previous classifiers. Its a sequential procedure. New models are influenced by previously built ones.

In Gradient boosting, the subsequent classifier is trained directly on the error of the preceding classifiers. i.e., only samples having error rate are fed to the subsequent classifier.

Boosted Trees

- Tree-1 is trained with a random subset of the available training data.
- Tree-2 is trained on the residuals(instead of output) of one of the leaf nodes of the Tree-1
- Tree-3 is trained on the residuals(instead of output) of one of the leaf nodes of the Tree-2
- The three classifiers are combined through a three way majority vote.

The advantages of Gradient boosting is it has low bias and if avoided overfitting, Gradient boosted trees prediction is better than single Decision tree or Random forest algorithm.Feature importance's can be generated.

Ada boost Algorithm:

Ada boosting is a type of boosting algorithm. In Gradient boosting, we train the classifiers using the residuals of the previous classifier, whereas in Ada boosting

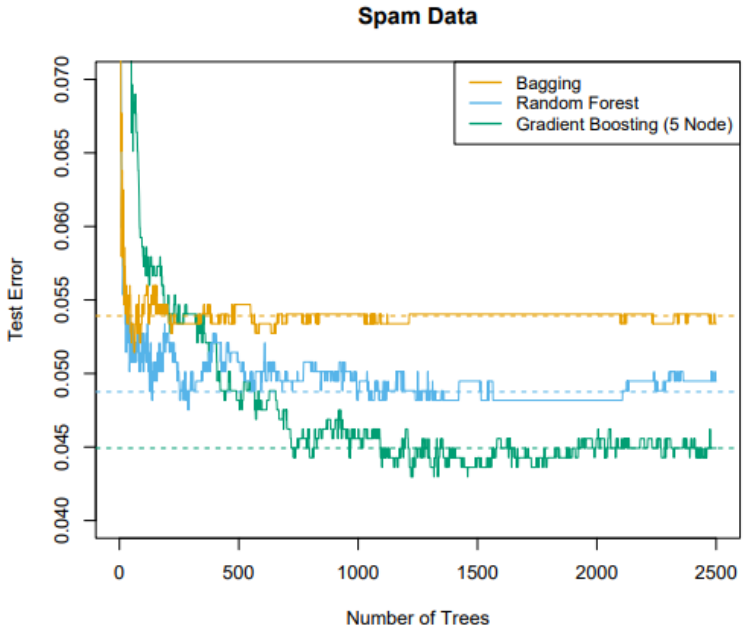

Figure 5.2: comparison of different algorithms on spam data classification:[9]

we train a series of classifiers each focuses on(gives higher weightage) instances that the previous ones got wrong. Then, Ada boosting uses a weighted average of predictions.

Ada boosting has no parameters to tune and it is simple, fast, and easy to program. Ada boost can be susceptible to noise and in case of large number of outliers the weight placed on outliers can hurt the performance.

Support vector Machines with linear and Radial basis function Kernels:

Support vector machines(SVM) construct a hyper plane or a set of hyper planes in a high dimensional space which can be used for classification, regression, or other tasks like outliers detection.

Support vector machines are very effective in high-dimensional spaces. They work well even when the number of features exceeds number of samples. Support vector machines shifts over-fitting problem to the model selection using kernels.

linear kernel: Linear kernel refers to the decision boundary(hyper plane) is linear. Radial basis function kernel: Radial basis function is commonly used in SVM as it supports non linear decision boundaries.

The figure below shows the decision function for a linearly separable problem,

with three samples on the margin boundaries, called “support vectors”.

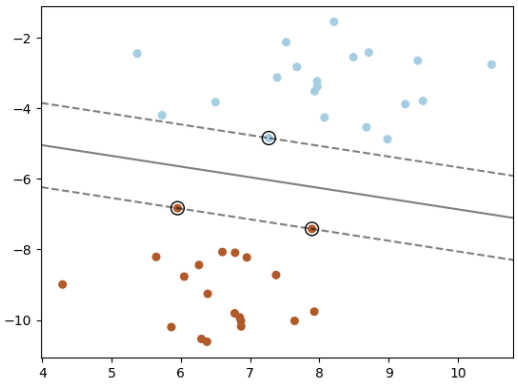

Figure 5.3: SVM classifier with linear kernel credits:[10]

Multi layer perceptron algorithm:

Multi layer perceptron(MLP) is a supervised learning algorithm that learns a function training on the data set.

Multi layer perceptron can learn non-linear functions. It can learn in real time(online-learning).

MLP is sensitive to hyper parameter scaling. It also requires tuning a number of hyper parameters. Optimization is very hard due to several local-minimas in the hidden layers. MLP with large number of hidden layers can lead to over fitting.

K-Nearest Neighbors algorithm:

The principle behind nearest neighbor methods is to find a predefined number of training samples closest in distance to the new point, and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning), or vary based on the local density of points (radius-based neighbor learning). The distance can, in general, be any metric measure: standard Euclidean distance is the most common choice. Neighbors-based methods are known as non-generalizing machine learning methods, since they simply “remember” all of its training data.

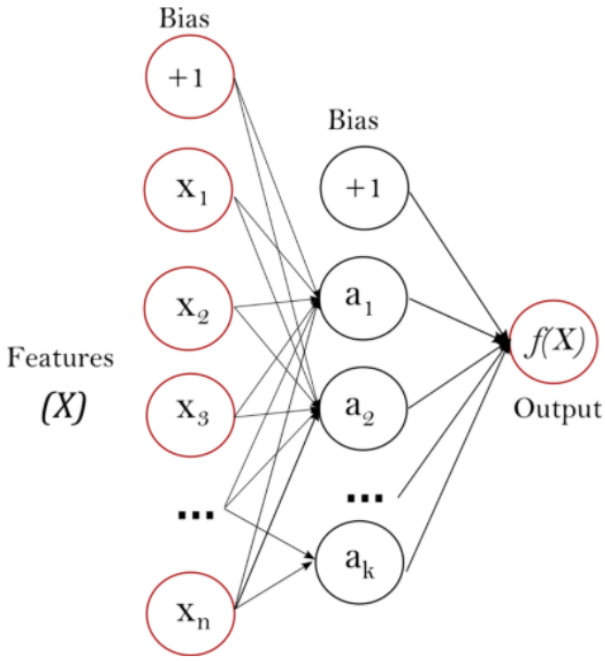

Figure 5.4: 1 hidden layer MLP credits:[10]

5.2 Deep Learning

Deep learning is a class of machine learning algorithms which uses multiple layers of neural networks to progressively extract higher level of features from the raw input. Deep learning is used in most complex problems and with non-linear data such as text, images, audio, video etc.,

Most of the deep learning algorithms are supervised.

Supervised Deep learning algorithms: Convolutional Neural Networks(CNN's), Recurrent Neural Networks(RNN's), Long Short-Term Memory Networks (LSTMs),Stacked Auto-Encoders,Deep Boltzmann Machine (DBM),Deep Belief Networks (DBN),etc.

Unsupervised Deep learning algorithms: Deep Embedded Clustering (DEC), etc.,

Deep learning algorithms used in this internship are Convolutional Neural Networks (CNN's).

5.2.1 Convolutional Neural Networks(CNN)

Convolutional Neural Nets differ from perceptron models in the way they use convolutions instead of Matrix multiplication in one of their inner layers.

A Convolutional Neural Network consists of input layers, output layers, and several hidden layers. The input layers or hidden layers typically consists of convolutional layers that convolve with a matrix of weights. The activation function is commonly a RELU layer, and is subsequently followed by additional convolutions such as pooling layers, fully connected layers and normalization layers, referred to as hidden layers because their inputs and outputs are masked by the activation function and final convolution.

Convolutional Neural Networks are often used in Image recognition systems due to their ease of use without any processing necessary compared to other image classification algorithms. CNN's are used in anomaly detection, Electromyography detection, etc.

In 5.5, a 1 dimensional CNN has been used to detect emotion based on single pulse PPG signals. We have used a similar architecture to build CNN models.

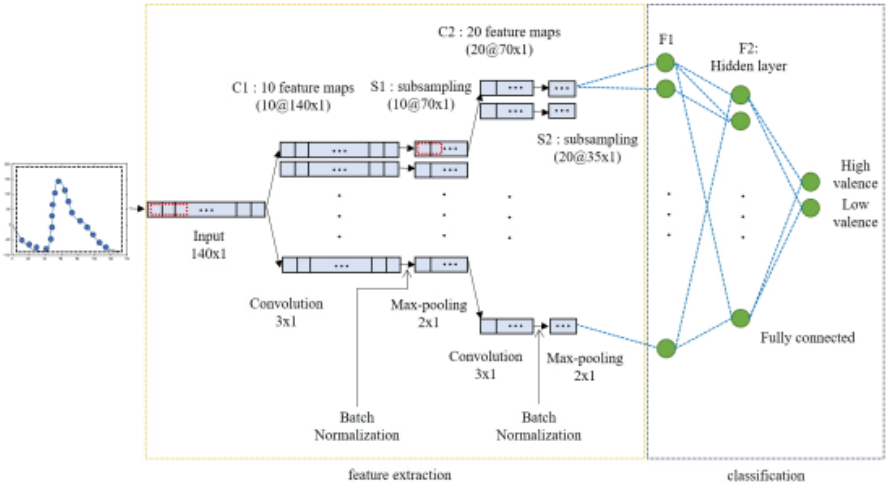

Figure 5.5: 1D CNN used for emotion detection using a PPG signal credits:[11]

5.2.2 An Example CNN architecture from the built models

In 5.6, a three layer CNN model flowchart from one of the CNN models built has been presented.

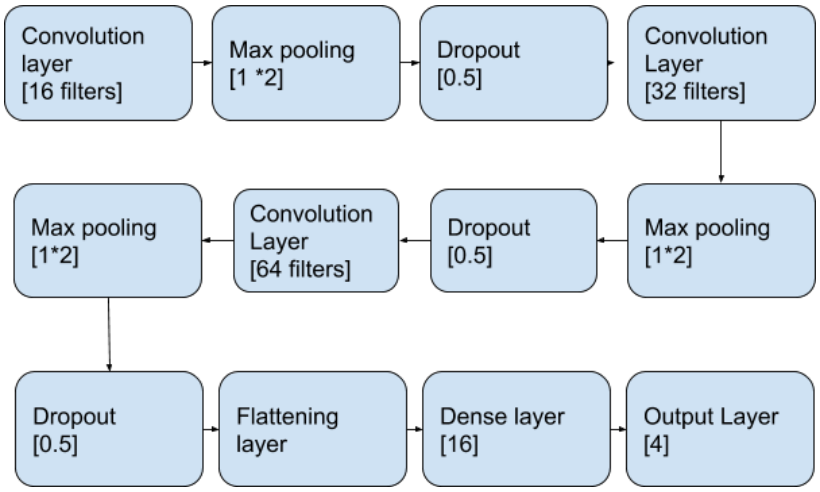

Figure 5.6: 1D CNN with 3 convolutional layers, maxpooling, dropout and dense layers

Different layers of the CNN model

1st Convolutional Layer:

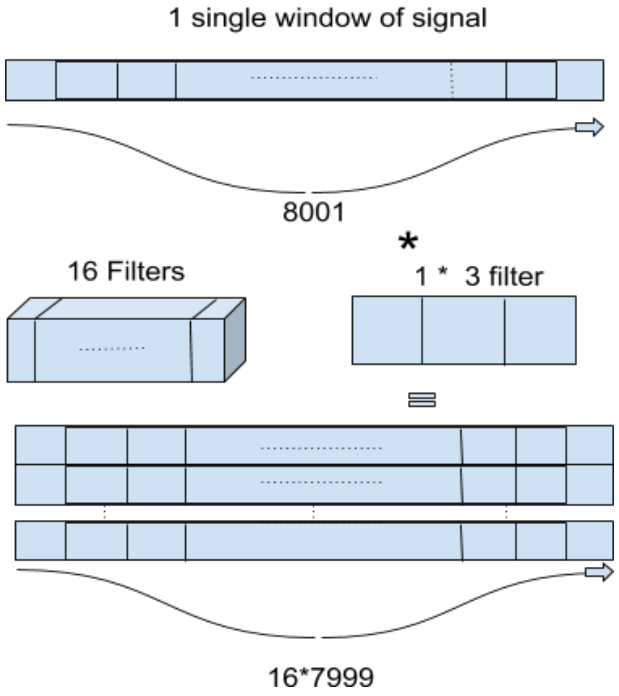

Figure 5.7: 1st layer convolutional neural net with 16 filters

Convolutions of 16 1*3 filters with the 1D signal of length 8001 produce 16*7999 matrix.

Filter weights are chosen by CNN. In practice, when training CNNs we won't be setting the filter weights manually. These values are learnt by the network automatically via back propagation.

Instead of large weight matrices per layer, CNN's learn filter weights. In other words, the network when adjusting its weights (from random values) to decrease the classification errors, comes up with the right filters that are suitable for characterising the output classification we're interested in.

Max pooling:

Pooling layers are another way to reduce the size of the signal interpretation in order to speed up computation, and it makes the detected features more robust.

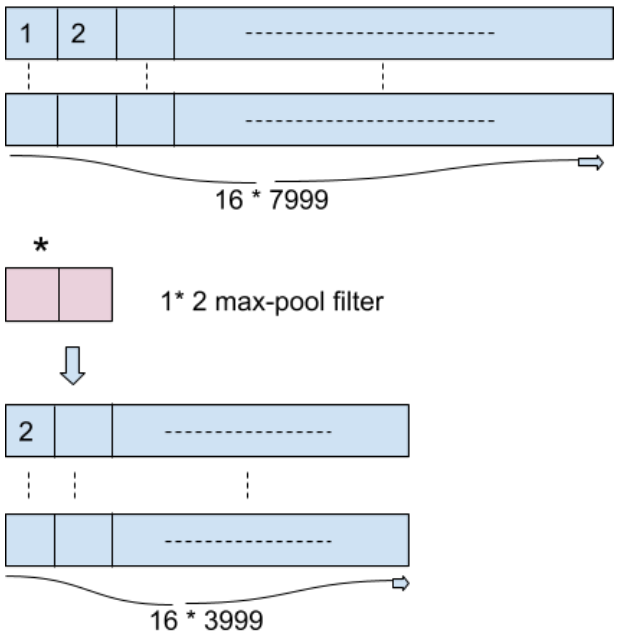

Figure 5.8: 1st layer Max pooling layer

After the First pooling, data was reduced in half.

Dropout:

Drop out is a mechanism to avoid over fitting which is the intrinsic nature of the neural networks. The network iteratively masks some neurons as in 5.9 to avoid over fitting. There is no change in output dimensions.

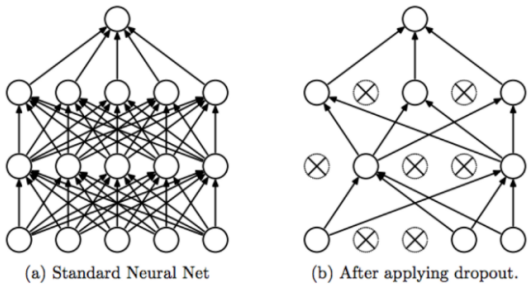

Figure 5.9: Dropout layer illustration [12]

The architecture of the model implemented has 3 layers of convolution neural networks, 3 max pooling layers and 3 dropouts.

Flattening layer:

Flattening layer converts 2D outputs to a single dimension to feed to the next dense layers.

Dense and Output layer:

We choose 4 neurons as output to match our 4 wheel configurations.

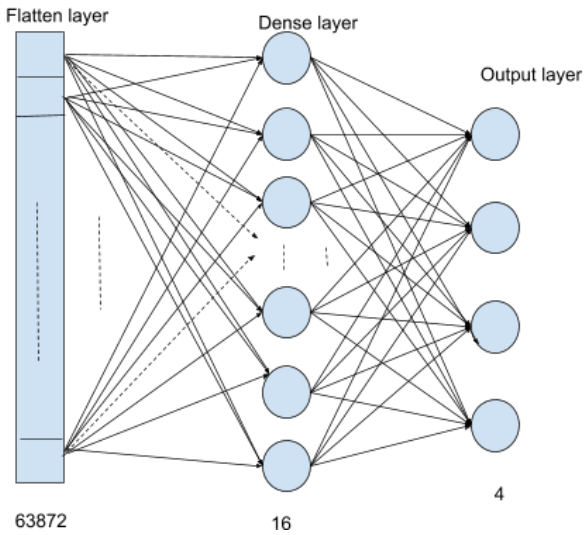

Figure 5.10: Flattening layer, Dense layer, output layer

Hyper parameters and loss metrics:

Hyper parameters are the parameters that are externally chosen to control the learning.

The Hyper parameters of our CNN model are

- Number of filters and filter shape: the number filters of a CNN is to extract as many features as possible.Filter shapes have to carefully choose to capture a certain feature.
- Activation Functions: After every layer there will be some kind of activation function defining the output of a given node in a layer. There are several types of activation functions for example: ReLu, sigmoid, softmax, etc.,
- Max pooling shape: By increasing the size of pooling layer, we might be missing out important learned feature.And, by decreasing the size of pooling layer, we are increasing the computational.

Loss metrics:

Loss metrics are used to judge a model performance. In multi-class classification, there are several loss metrics to choose from, for example: categorical accuracy, categorical cross entropy, etc.

Optimization:

Optimization algorithms are used to improve neural network accuracy. The optimization algorithm's performance is evaluated on its fastness to achieve accurate results. In this internship, we tested multiple optimization algorithms, finally Adam optimizer proved best because of its bias-correction. In the figure 5.11, Adam is compared with other optimization algorithms on MNIST Images.

algorithms: gradient descent, back propagation, adam optimizer, RMS prop, etc.,

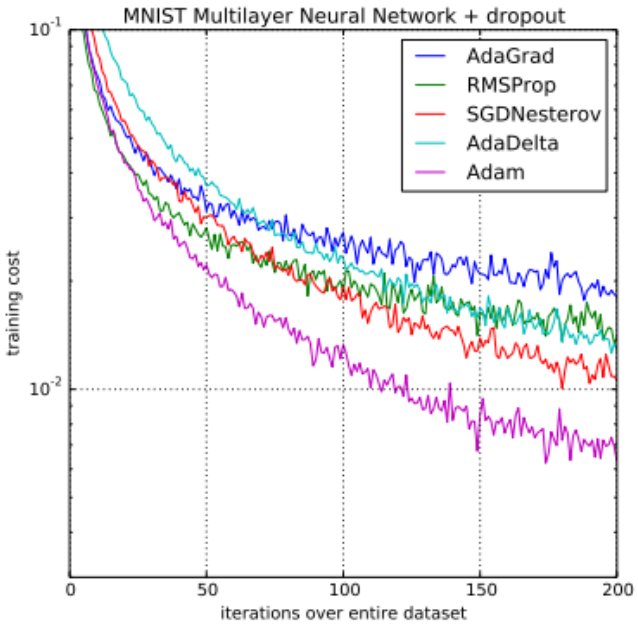

Figure 5.11: Training of Multi layer neural networks using MNIST images credits:[13]

Chapter 6

Results

The data has been grouped based on the geophone locations.

- Geophones at a depth of 40cm and 20cm are in Group 1
- Geophones at a depths of 6cm are in Group 2
- Geophones on the surface are grouped into Group 3
- Geophones which all are placed in Zone 1 i.e., close to the Geophones 1 and 2 (cyan in plot 1.4) are all in Group 4
- Geophones which all are in Zone 2 are placed in Group5

Group-wise models have been built using both Convolutional neural networks on raw data and Machine learning models on Features extracted data.

6.1 Convolutional neural network model results

Several configurations have been trained using Sequence 3 data.

Each combinations of the below selection have been trained, and the best combinations have been presented.

- 1,2,3 convolutional neural nets
- 16,32,64 filters
- 0,1,2 dense layers
- 3,5,7,9 kernel sizes for each filter

For example: 1 convolutional neural net, 16 filters, 0 dense layers, 3 kernel size are one complete neural net architecture.

At first, 10 epochs have been chosen to get the top 3 best models. In the second run, the number epochs have been increased to 20.

6.1.1 Validation curves

To validate a model we need a scoring function, for example accuracy for classifiers. The proper way of choosing multiple hyper parameters of an estimator are of course grid search or similar methods that select the hyper parameter with the maximum score on a validation set or multiple validation sets. Note that if we optimized the hyper parameters based on a validation score the validation score is biased and not a good estimate of the generalization any longer. To get a proper estimate of the generalization we have to compute the score on another test set. plotting these metrics is the best practise to understand and evaluate the model. These plots are called as Validation curves.

Epoch vs Accuracy and Epoch vs Loss have been plotted to understand whether the model is over fitting, under fitting, or training accuracy is followed by validation accuracy. By looking at these validation curves, we validate the model.

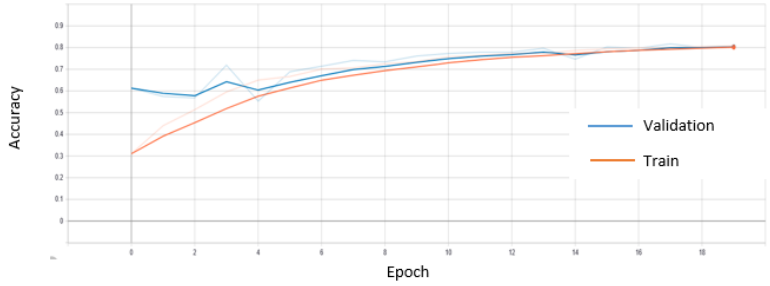

Figure 6.1: Number of Epochs vs Accuracy for Group 3

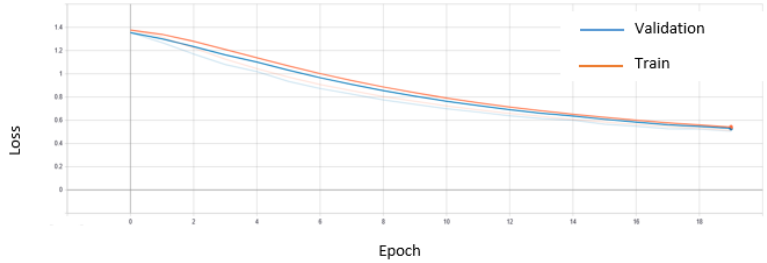

Figure 6.2: Number of Epochs vs Loss for Group 3

The results of best models and their configurations from all the groups have been presented in 6.1.

Confusion matrices have been plotted to understand exactly which wheel configurations are hard to detect using Geophone signals. The observations from confusion matrices are as follows

Group	Architecture	test set accuracy
Group 1	1 conv net, 16 filters, 1 dense layer, 3 kernel size	24.441%
Group 2	1 conv net, 16 filters, 1 dense layer, 3 kernel size	59.198%
Group 3	1 conv net, 32 filters, 0 dense layer, 3 kernel size	79.713%
Group 4	1 conv net, 16 filters, 1 dense layer, 7 kernel size	48.666%
Group 5	1 conv net, 64 filters, 1 dense layer, 3 kernel size	57.960%

Table 6.1: Group wise Convolutional neural nets accuracy

- Group 1: The model misclassifies all the wheel configurations as tridem
- Group 2: The model misclassifies some tandem samples as tridem and all the single wheel and dual wheel samples as tridem.
- Group 3: The model accurately classifies tandem and tridem, but misclassifies some single wheel samples as dual wheel and vice versa.
- Group 4: The model classifies tandem and dual wheel with high accuracy, but misclassifies all single wheel samples as dual wheel, most of the tridem samples as dual wheel, and some tridem samples as tandem.
- Group 5: The model accurately classifies tandem, but misclassifies single wheel samples as dual wheel and tandem samples, some dual wheel samples as tandem samples, and some tridem samples as tandem.

6.2 Feature based model results

6.2.1 With all variables:

At first, Decision tree and Random forest models have been built using all the extracted features from sequence 3 and tested on sequence 7 data. Feature importance plots have been plotted. By looking at these feature importance's 6.3 6.4, we understood that the model is trained using only one variable which is load, and which will not be available in a real scenario. So, we removed load variables from the feature data.

6.2.2 After removing load variables:

Plot 6.5 shows feature importance's after removing load variables.

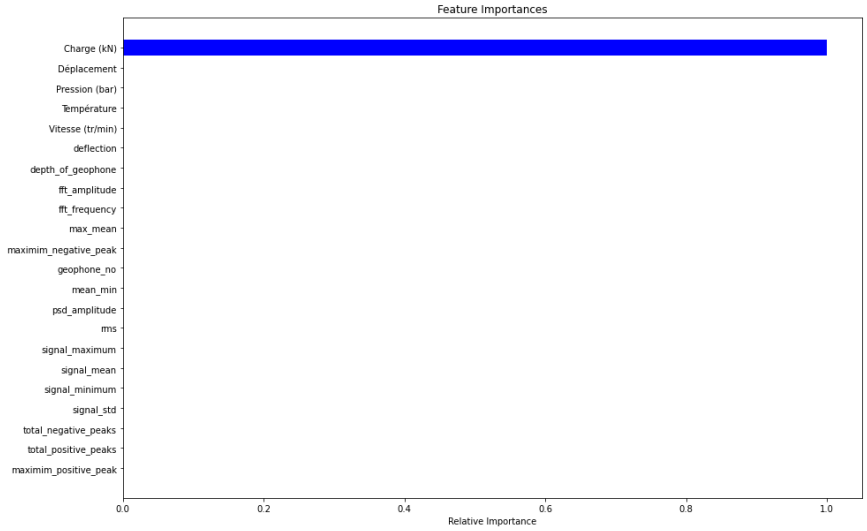

Figure 6.3: Variable importance's for whole data in sequence 3 using Decision Trees

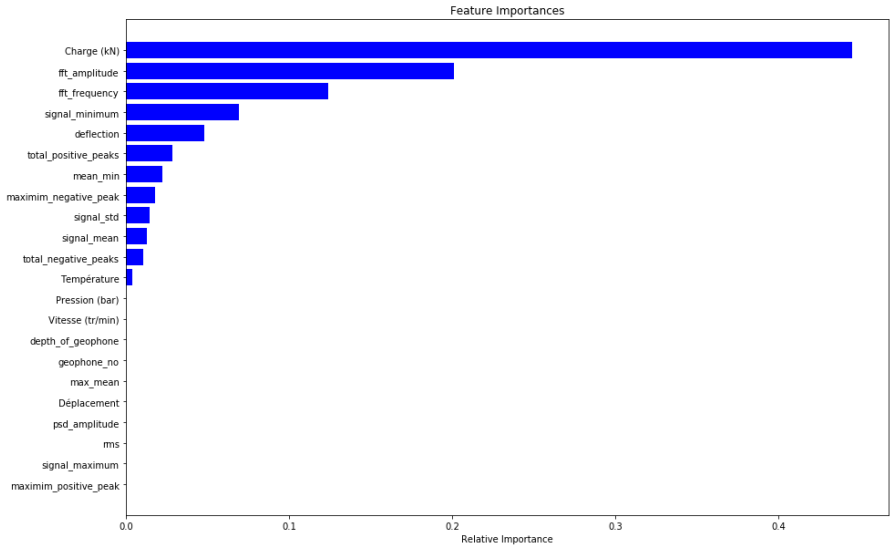

Figure 6.4: Variable importance's for whole data in sequence 3 using Random Forests

Finally, Sequence 7 data has been used to test these model's accuracy.

Looking at the accuracies in the table 6.1, we conclude that with load variables the models are over-fitting.

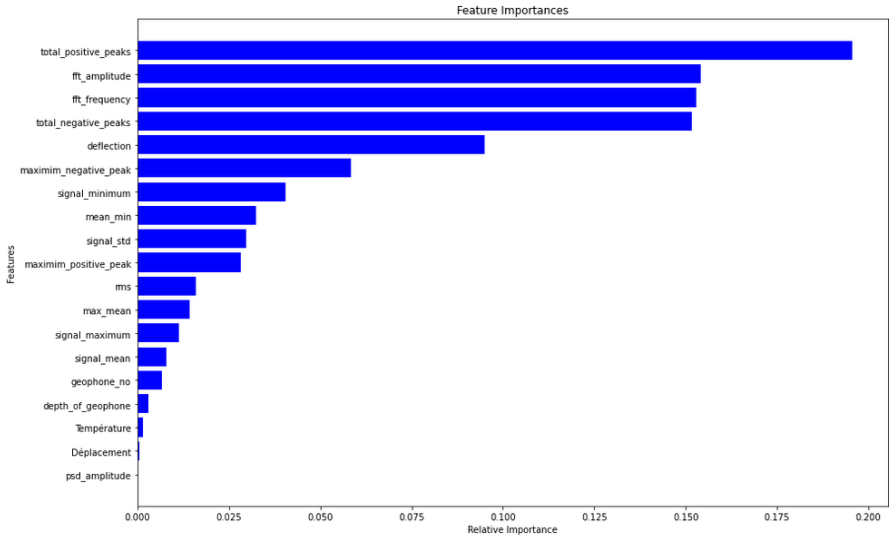

Figure 6.5: Variable importance's after removing load variables for whole data in sequence 3 using Random Forests

Model	Accuracy on test set	Accuracy on sequence 7 data
Decision Tree	100%	75%
Random Forest	84%	50%
Random Forest without load variables	44%	42%

Table 6.2: Accuracies with load and without load parameters

6.2.3 Group wise models

Group wise models have been built in two phases,

- with out one hot encoding the categorical variables and standardizing the numerical variables.
- with one hot encoding the categorical variables and standardizing numerical variables.

Group 1

- Input features: ['Température', 'deflection', 'depth of geophone', 'fft amplitude', 'fft frequency', 'max-mean', 'maximim negative peak', 'maximim positive peak', 'mean-min', 'psd amplitude', 'rms', 'signal maximum', 'signal mean', 'signal std', 'total negative peaks', 'total positive peaks','Déplacement', 'geophone no', 'Time 1(s)','Time 2(s)', 'K1','K2','K3','peaks 10 mean','peaks 3 mean','peaks 2 mean']
- Output Features: ["Type roue"]
- Test set: Sequence 8
- Training data: Sequence 3 and sequence 7

Model	Accuracy on test set	Accuracy on sequence 8 data
Decision Tree	51.89%	38.63%
Random Forest	52.83%	38.63%
Ada Boost	51.89%	45.45%
Gradient Boosting	74.53%	53.97%

Table 6.3: Group 1 model Accuracies with out standardization

Group 2

- Input features: ['Température', 'deflection', 'depth of geophone', 'fft amplitude', 'fft frequency', 'max-mean', 'maximim negative peak', 'maximim positive peak', 'mean-min', 'psd amplitude', 'rms', 'signal maximum', 'signal mean', 'signal std', 'total negative peaks', 'total positive peaks','Déplacement', 'geophone no', 'Time 10(s)','Time 3(s)', 'Time 4(s)', 'Time 5(s)', 'Time 6(s)', 'Time 7(s)', 'Time 8(s)' 'K1','K2','K3','peaks 10 mean','peaks 3 mean','peaks 2 mean']

Model	Accuracy on test set	Accuracy on sequence 8 data
K-Nearest Neighbors	64.15%	44.32%
Support Vector Machines with linear kernel	65.09%	42.61%
Support Vector Machines with RBF Kernel	74.53%	46.02%
Decision Tree	63.20%	43.18%
Random Forest	53.77%	37.5%
Multi-layer perceptron	76.41%	49.43%
Ada Boost	43.40%	40.91%
Gradient Boosting	84.90%	52.84%

Table 6.4: Group 1 model Accuracies with standardization

- Output Features: ["Type roue"]
- Test set: Sequence 8
- Training data: Sequence 3 and sequence 7

Model	Accuracy on test set	Accuracy on sequence 8 data
Decision Tree	58.38%	56.49%
Random Forest	50.27%	49.68%
Ada Boost	51.08%	45.45%
Gradient Boosting	75.40%	75.81%

Table 6.5: Group 2 model Accuracies with out standardization

Model	Accuracy on test set	Accuracy on sequence 8 data
K-Nearest Neighbors	59.20%	67.86%
Support Vector Machines with linear kernel	64.86%	55.84%
Support Vector Machines with RBF Kernel	67.84%	63.47%
Decision Tree	61.35%	53.90%
Random Forest	57.30%	49.51%
Multi-layer perceptron	78.18%	68.51%
Ada Boost	53.24%	47.24%
Gradient Boosting	78.92%	73.05%

Table 6.6: Group 2 model Accuracies with standardization

Group 3

- Input features: ['Température', 'deflection', 'depth of geophone', 'fft amplitude', 'fft frequency', 'max-mean', 'maximim negative peak', 'maximim positive peak', 'mean-min', 'psd amplitude', 'rms', 'signal maximum', 'signal mean', 'signal std', 'total negative peaks', 'total positive peaks','Déplacement', 'geophone no', 'Time 11(s)', 'Time 12(s)', 'Time 13(s)', 'Time 14(s)', 'Time 15(s)' 'K1','K2','K3','peaks 10 mean','peaks 3 mean','peaks 2 mean']
- Output Features: ["Type roue"]
- Test set: Sequence 8
- Training data: Sequence 3 and sequence 7

Model	Accuracy on test set	Accuracy on sequence 8 data
Decision Tree	56.15%	39.58%
Random Forest	59.62%	44.89%
Ada Boost	56.47%	46.97%
Gradient Boosting	77.29%	65.53%

Table 6.7: Group 3 model Accuracies with out standardization

Model	Accuracy on test set	Accuracy on sequence 8 data
K-Nearest Neighbors	55.83%	58.90%
Support Vector Machines with linear kernel	61.83%	46.60%
Support Vector Machines with RBF Kernel	64.98%	52.65%
Decision Tree	53.63%	38.45%
Random Forest	53.94%	40.53%
Multi-layer perceptron	79.18%	59.47%
Ada Boost	53.31%	51.51%
Gradient Boosting	76.34%	63.63%

Table 6.8: Group 3 model Accuracies with standardization

Group 4

- Input features: ['Température', 'deflection', 'depth of geophone', 'fft amplitude', 'fft frequency', 'max-mean', 'maximim negative peak', 'maximim positive peak', 'mean-min', 'psd amplitude', 'rms', 'signal maximum', 'signal mean', 'signal std', 'total negative peaks', 'total positive peaks','Déplacement', 'geophone no', 'Time 9(s)','Time 10(s)', 'Time 1(s)', 'Time 2(s)' 'K1','K2','K3','peaks 10 mean','peaks 3 mean','peaks 2 mean']
- Output Features: ["Type roue"]
- Test set: Sequence 8
- Training data: Sequence 3 and sequence 7

Model	Accuracy on test set	Accuracy on sequence 8 data
Decision Tree	41.98%	37.5%
Random Forest	57.58%	40.34%
Ada Boost	50.94%	38.92%
Gradient Boosting	79.24%	48.86%

Table 6.9: Group 4 model Accuracies with out standardization

Model	Accuracy on test set	Accuracy on sequence 8 data
K-Nearest Neighbors	63.68%	38.92%
Support Vector Machines with linear kernel	65.09%	37.78%
Support Vector Machines with RBF Kernel	64.62%	37.21%
Decision Tree	59.43%	41.48%
Random Forest	60.38%	41.76%
Multi-layer perceptron	76.41%	37.78%
Ada Boost	59.90%	36.36%
Gradient Boosting	84.43%	48.29%

Table 6.10: Group 4 model Accuracies with standardization

Group 5

- Input features: ['Température', 'deflection', 'depth of geophone', 'fft amplitude', 'fft frequency', 'max-mean', 'maximim negative peak', 'maximim positive peak', 'mean-min', 'psd amplitude', 'rms', 'signal maximum', 'signal mean', 'signal std', 'total negative peaks', 'total positive peaks','Déplacement', 'geophone no', 'Time 3(s)','Time 4(s)', 'Time 5(s)', 'Time 11(s)' 'K1','K2','K3','peaks 10 mean','peaks 3 mean','peaks 2 mean']
- Output Features: ["Type roue"]
- Test set: Sequence 8
- Training data: Sequence 3 and sequence 7

Model	Accuracy on test set	Accuracy on sequence 8 data
Decision Tree	74.06%	77.27%
Random Forest	71.69%	71.30%
Ada Boost	52.83%	54.82%
Gradient Boosting	91.03%	84.94%

Table 6.11: Group 5 model Accuracies with out standardization

Model	Accuracy on test set	Accuracy on sequence 8 data
K-Nearest Neighbors	68.39%	75.85%
Support Vector Machines with linear kernel	64.62%	60.51%
Support Vector Machines with RBF Kernel	73.11%	74.43%
Decision Tree	75%	72.44%
Random Forest	66.98%	61.64%
Multi-layer perceptron	79.72%	73.58%
Ada Boost	65.09%	57.38%
Gradient Boosting	87.26%	83.81%

Table 6.12: Group 5 model Accuracies with standardization

6.3 Cross-validation and Optimization of the best model

Cross validation is to test how well model generalizes on out-of-sample estimate.

The best model with lesser number of geophones is Gradient Boosting classifier on Group 5 with accuracy of 91.0% on test set and 84.94% on Sequence 8 data.

After Cross Validation using 3 Stratified folds, we achieved accuracies of 68%, 74%, 79% on the 3 folds.

To optimize, we used three optimization techniques

- Gridsearch CV
- Scikit-optimize
- Optuna

6.3.1 Gridsearch CV

3 Stratified folds exhaustive search on an specified parameter values to find the best estimator.

The parameter space for Gridsearch CV

- "n_estimators": 50,100,250,500
- "max_depth": 1,3,5,7,9
- "learning_rate":0.01,0.1,1,10,100

Best accuracy score achieved on test data is 93%. Confusion matrix 6.7 and the % success plot hass been plotted 6.6

And, the parameters to achieve the best accuracy are

- "n_estimators": 250
- "max_depth": 3
- "learning_rate": 1

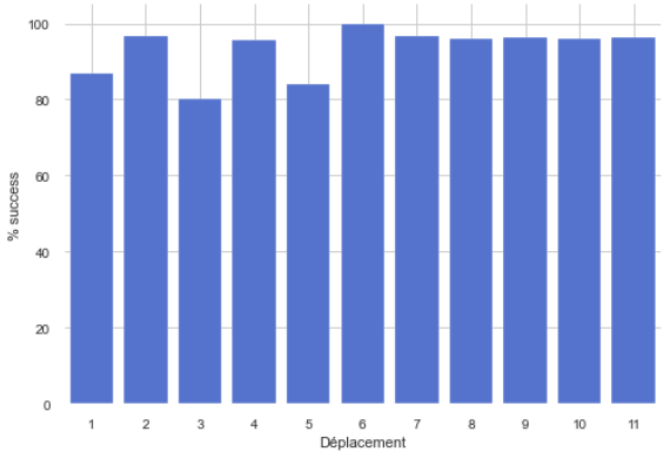

Figure 6.6: % success of prediction of wheel configuration at a particular position

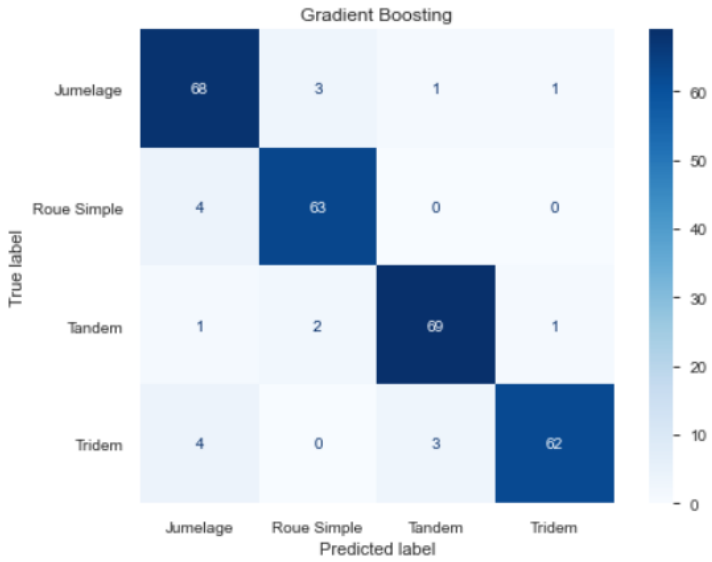

Figure 6.7: Confusion matrix of the test predictions for best model

6.3.2 Scikit-Optimize

Scikit-Optimize is a model based sequential optimization package. In Scikit optimize, we used gp_minimize function which is a Bayesian optimization function using Gaussian Process. The objective of the Scikit optimize is to take black box models and globally optimize (minimize the objective function).

In this optimization, we provide a parameter search space to the algorithm. The parameter space is between a minimum and a maximum for each parameter.

- "max_depth" = [3,15]
- "learning_rate" = [0.01, 2]
- "n_estimators" =[100,600]
- " max_features" [’sqrt’,’log2’,None],
- "subsample" =[0.6, 1.0]

Best accuracy score achieved is 76.22%, and the parameters to achieve best accuracy score are

- "max_depth" = 3
- "learning_rate" = 0.45025995935586915
- "n_estimators" = 600
- " max_features" None
- "subsample" = 0.6

Convergence plot visualizes the number of calls to the optimization function

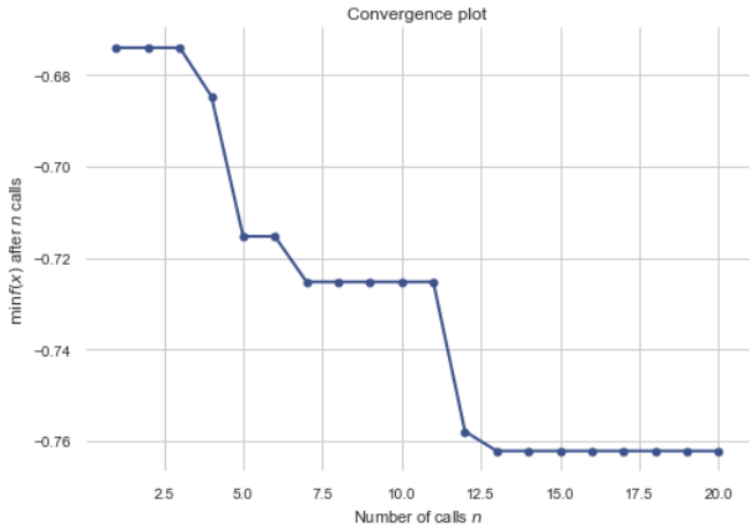

Figure 6.8: Convergence plot of Bayesian Optimization on Gradient descent Algorithm

6.3.3 Optuna

Optuna is an automatic hyper parameter optimization software framework, particularly designed for machine learning

The parameter space for optimization is

- "max_depth" = [1,20]
- "learning_rate" = [0.01, 1]
- "n_estimators" =[50,400]
- " max_features" = ['sqrt','log2'],
- "subsample" =[0.6, 1.0]

Best accuracy acheived is 75%, and the parameters for best accuracy are

- "max_depth" = 11
- "learning_rate" = 0.2843404874991921
- "n_estimators" = 150
- " max_features" = 'sqrt'
- "subsample" = 0.6021849304936285

Chapter 7

Conclusion

We started this internship with bibliographic study to understand AI in pavement research. We understood from the bibliography that the authors have concentrated more on model building using different machine learning techniques to predict pavement roughness index, pavement response e.t.c., than the business outcome of applying particular technique. We formulated a problem of utilizing very few geophones than the costly WIM sensors to detect wheel configurations with high accuracy. We tried both Deep learning and Machine Learning techniques on multiple groups of geophones signals. We cross validated and optimized the best model which could detect wheel configuration with 93% accuracy using 4 geophones(3 at the interface and one at the surface).

Chapter 8

Limitations and Future Work

- The data consists of only 4 wheel configurations[single wheel, dual wheel, tandem, and tridem] geophone signals. Testing external data with other wheel configuration will lead to inconsistent results
- We manually extracted the data for some signals with different speeds and automated the extraction for others, which lead to slightly different signal lengths for different speeds.
- We have not removed any outliers in the data assuming that they all are the best signals.
- Re-training the models will result in slightly different accuracies.

In order to extend this problem to a real pavement, data must be gathered from different types of wheels plying on a real pavement. And, with the data and labels of wheel configurations, this model can be extended to different wheels using that pavement.

Bibliography

[1] P. Marcelino, M. de Lurdes Antunes, E. Fortunato, and M. C. Gomes, "Machine learning approach for pavement performance prediction," *International Journal of Pavement Engineering*, pp. 1–14, 2019. [Online]. Available: https://doi.org/10.1080/10298436.2019.1609673

[2] T. SS, L. B. C, I. A, B. V, D. X, and F. C, "Detection of debondings with ground penetrating radar using a machine learning method," in *2017 9th International Workshop on Advanced Ground Penetrating Radar (IWAGPR),Edinburgh*, Jun. 2017, pp. 1–6, doi: 10.1109/IWAGPR.2017.7996056.

[3] G. OE and A.-Q. IL, "Developing machine-learning models to predict airfield pavement responses," in *Transportation Research Record: Journal of the Transportation Research Board*, Jun. 2018, pp. 23–34, doi: 10.1177/0361198118780681.

[4] R. S. G, K. Mohamed, and B. Behnam, "Development of an intelligent system for automated pavement evaluation," in *Pavement management: data collection, analysis, and storage, 1991*, 1991, pp. 112–119.

[5] H. Ceylan, M. B. Bayrak, and K. Gopalakrishnan, "Neural networks applications in pavement engineering: A recent survey," *International journal of pavement research and technology*, vol. 7, pp. 434–444, 2014.

[6] N. Kargah-Ostadi and S. M. Stoffels, "Framework for development and comprehensive comparison of empirical pavement performance models," *Journal of Transportation Engineering-asce*, vol. 141, p. 04015012, 2015, doi: 10.1061/(ASCE)TE.1943-5436.0000779.

[7] M. Eisenbach, R. Stricker, D. Seichter, K. Amende, K. Debes, M. Sesselmann, D. Ebersbach, U. Stoeckert, and H. Gross, "How to get pavement distress detection ready for deep learning? a systematic approach," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 2039–2047, doi: 10.1109/IJCNN.2017.7966101.

[8] R. Abduljabbar, H. Dia, S. Liyanage, and S. A. Bagloee, "Applications of Artificial Intelligence in Transport: An Overview," *Sustainability*, vol. 11, no. 1, pp. 1–24, January 2019, doi: 10.3390/su11010189.

[9] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[10] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[11] M. Lee, Y. Lee, D. Pae, M. T. Lim, D. W. Kim, and T.-K. Kang, "Fast emotion recognition based on single pulse ppg signal with convolutional neural network," *Applied Sciences*, vol. 9, p. 3355, 08 2019.

[12] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

List of Figures

1.1 Fatigue Machine at IFSTTAR, Nantes 2
1.2 Aerial View of the carrousel 2
1.3 Positioning of the load cells on the CSA overload ring 5
1.4 Positioning of the sensors on the track 5
1.5 Different wheel configurations used in the experiment 7

3.1 plots of Geophones signals from file 306 13
3.2 plots of all the wheel configurations from file 295 and Geophone 4 14
3.3 plots of all the wheel configurations from file 295 and Geophone 9 14
3.4 plots of all the wheel configurations from file 295 and Geophone 2 15
3.5 plots of tandem wheel configuration from file:295 and Geophones 3, 4, 5. 15
3.6 plots of tandem wheel configuration from file:295 and Geophones 1, 2, 9, 10 16
3.7 Overlapping Geophone 2 plots of the three files(file:290,295,306) with tandem wheel configuration 16
3.8 Overlapping Geophone 5 plots of the three files(file 290, 295, 306) with tandem wheel configuration 17
3.9 Overlapping Geophone 10 plots of the three files(file 290, 295, 306) with tandem wheel configuration 17
3.10 Time difference box plot for the file:435 in sequence 7 21

4.1 Three phase approach to develop advanced predictive model-adapted from [8] 22
4.2 Frequency features of Group1 25
4.3 Time differences boxplot of Group1 times 26
4.4 Group1 Boxplot of position vs maximum positive peak 26
4.5 Group1 Boxplot of position vs maximum negative peak 27
4.6 Group1 boxplots of position vs maximum positive peaks and maximum negative peaks 27

4.7 Frequency features of Group5 29
4.8 Time differences boxplot of Group5 times 29
4.9 Group5 Boxplot of position vs maximum positive peak 30
4.10 Group5 Boxplot of position vs maximum negative peak 30
4.11 Group5 boxplots of position vs maximum positive peaks and maximum negative peaks 31
4.12 Group5 files 293, 315 geophone 14 signals [outlier in green] . . . 31
4.13 Group5 files 294, 316 dual wheel geophone 14 signals [outlier in green] 32
4.14 Group5 files 294, 316 tridem geophone 14 signals [outlier in green] 32
4.15 Group5 files 294, 316 single wheel geophone 14 signals [outlier in green] 33
4.16 Group5 files 295, 317 tridem geophone 14 signals [outlier in green] 33
4.17 Correlation between all the variables in sequence 3 34
4.18 example plot of positive peaks extraction using prominence of 0.00005 for single wheel configuration 35
4.19 example plot of positive peaks extraction using prominence of 0.00005 for single wheel configuration 36
4.20 file 306 plots of all the geophone signals 36

5.1 Decision tree taken from the model built on sequence 3 data . . . 39
5.2 comparison of different algorithms on spam data classification:[9] 41
5.3 SVM classifier with linear kernel credits:[10] 42
5.4 1 hidden layer MLP credits:[10] 43
5.5 1D CNN used for emotion detection using a PPG signal credits:[11] 44
5.6 1D CNN with 3 convolutional layers, maxpooling, dropout and dense layers 45
5.7 1st layer convolutional neural net with 16 filters 45
5.8 1st layer Max pooling layer 46
5.9 Dropout layer illustration [12] 47
5.10 Flattening layer, Dense layer, output layer 47
5.11 Training of Multi layer neural networks using MNIST images credits:[13] 49

6.1 Number of Epochs vs Accuracy for Group 3 51

6.2 Number of Epochs vs Loss for Group 3 51
6.3 Variable importance's for whole data in sequence 3 using Decision Trees 53
6.4 Variable importance's for whole data in sequence 3 using Random Forests 53
6.5 Variable importance's after removing load variables for whole data in sequence 3 using Random Forests 54
6.6 % success of prediction of wheel configuration at a particular position 62
6.7 Confusion matrix of the test predictions for best model 62
6.8 Convergence plot of Bayesian Optimization on Gradient descent Algorithm 63

List of Tables

1.1 Characteristics of the load cells installed on the Pavement test bench 4

3.1 Signals in order of acquisition 12
3.2 the number peaks with the change in the threshold value for single wheel configuration 19
3.3 the number peaks with the change in the threshold value for dual wheel configuration 20

4.1 sequence 3 Group1 summary statistics of statistical features . . . 24
4.2 sequence 3 Group1 summary statistics of obtained features 25
4.3 sequence 3 Group5 summary statistics of statistical features . . . 28
4.4 sequence 3 Group5 summary statistics of obtained features 28

6.1 Group wise Convolutional neural nets accuracy 52
6.2 Accuracies with load and without load parameters 54
6.3 Group 1 model Accuracies with out standardization 55
6.4 Group 1 model Accuracies with standardization 56
6.5 Group 2 model Accuracies with out standardization 56
6.6 Group 2 model Accuracies with standardization 57
6.7 Group 3 model Accuracies with out standardization 57
6.8 Group 3 model Accuracies with standardization 58
6.9 Group 4 model Accuracies with out standardization 58
6.10 Group 4 model Accuracies with standardization 59
6.11 Group 5 model Accuracies with out standardization 59
6.12 Group 5 model Accuracies with standardization 60