Universiteit Leiden

Master Computer Science

Evaluating and Optimizing Electric Bicycle Designs by Coupling Machine Learning and Evolutionary Strategy

Name:	Mingkang Wang
Student ID:	s1941941
Date:	19/11/2019
Specialisation:	Computer Science and Advanced Data Analytics
1st supervisor:	Dr. Hao Wang
2nd supervisor:	Prof. Dr. Thomas Bäck

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

In the business field, companies are always looking to discover the relationship between consumer preferences and product features to promote the design of their own products. However, due to rapid changes in market trends, it has become very difficult to discover patterns of consumer preferences in real-time. Companies want a system to dynamically learn consumer preferences to evaluate and optimize product designs.

Machine learning methods and evolutionary strategies (ESs) provide the possibility to evaluate and optimize product designs based on learning consumer preferences. Machine learning models can be used to predict the consumer preferences of new product designs by learning the relationship between quantified consumer preferences and product features. Machine learning models require feature vectors and labels for training, so we based on some online traffic data to quantify consumer preference into numerical scores to be the labels, and we selected the relevant product features and encoded them into the numerical feature vectors. In order to reduce the error caused by the product feature encoding method, we compared the effect of two different encoding methods, then chose the one-hot encoding method to encode product features. For ensuring the machine learning process can be dynamic, we analyzed the trend of a machine learning model predicting performance in different time periods to find out the suitable time span for dynamic machine learning. For selecting the most suitable machine learning method to design our machine learning model, we built 3 different supervised learning models (artificial neural networks, random forest, and k nearest neighbor) and optimized their hyperparameters with covariance matrix adaptation evolution strategy (CMA-ES), then we compared their predicting performance to select the best-performed model to be our product evaluation model. Finally, for achieving the target of product optimization, we regarded the product evaluation model as the fitness function and also based on CMA-ES to establish the product optimization model.

Acknowledgements

This master thesis was done at LIACS and International Bike Group B.V., under the supervision of Prof. Dr. Thomas Bäck and Dr. H. Wang and Olav Jorgensen and Fubin Lu and Pieter Booij and Taeke Bijlsma and Lucas Kuin and Jeroen Wilmink. I would like to thank them for their generous help, advice and supervision during this project.

Contents

1 Introduction **5**
- 1.1 Background . . . 5
- 1.2 Interactions Between IBG and Consumers . . . 5
- 1.3 Available Data Sources . . . 6
  - 1.3.1 Web Traffic Data . . . 7
  - 1.3.2 Online Sales Data . . . 8
  - 1.3.3 Consumer Reviews Data . . . 8
  - 1.3.4 E-bike Component Information Data . . . 8
- 1.4 Project Overview . . . 9
- 1.5 Thesis Structure . . . 10

2 Data Source Selection **11**
- 2.1 Criteria for Data Source Selection . . . 11
- 2.2 Data Sources Comparison . . . 12
- 2.3 Consumer Preference Scores Calculation Based on Exhibition Platform Traffic Data 13

3 Methodology **14**
- 3.1 Machine Learning Techniques . . . 14
- 3.2 Covariance Matrix Adaptation Evolution Strategy . . . 15
  - 3.2.1 Hyperparameter Optimization . . . 16
  - 3.2.2 E-bikes Optimization . . . 16
- 3.3 Time Series Analysis . . . 17
- 3.4 E-bike Component Information Encoding . . . 19
  - 3.4.1 Feature Selection Techniques . . . 20
- 3.5 Implementation . . . 23

4 Experiments **24**
- 4.1 Time Series Analysis . . . 24
- 4.2 Encoding Methods Comparison . . . 26
- 4.3 Features Selection Methods Comparison . . . 26
- 4.4 Comparison of the 3 Machine Learning Methods . . . 30
- 4.5 Optimization Function Performance . . . 34

5 Conclusion **35**
- 5.1 Summary . . . 35
- 5.2 Limitations and Future Works . . . 36

Bibliography **39**

List of Figures

1.1 The insight of e-bike design . . . 6
1.2 Schematic diagram of the generation and storage of each data source . . . 7
1.3 Google Analytics web traffic data snippet (e-bike names have been replaced by code) 8
1.4 An Trustpilot review example . . . 8
1.5 E-bike component information data snippet (e-bike names have been replaced by code) . . . 9

3.1 The example of e-bike optimization . . . 17
3.2 The example of evaluating RF model's predicting performance on a certain time periods . . . 18
3.3 Example of dummy encoding . . . 19
3.4 Example of one-hot coding . . . 20
3.5 The example construction of the feature vectors . . . 21
3.6 The feature selection process with wrapper method . . . 22
3.7 Implementation overview Y . . . 23

4.1 The relationship between the number of decision trees(n) and the average $mse_{n,s_t,D}$ 25
4.2 The relationship between time span(s_t) and the average $mse_{n,s_t,D}(n = 40)$ . . . 25
4.3 The relationship between the duration settings and the $mse_{n,s_t,d_t}(n = 40)$ . . . 26
4.4 The mse_n $(n = \{1, ..., 100\})$ trend of one-hot and dummy encoding . . . 27
4.5 The mse trend when iteratively add the filter method sorted features . . . 28
4.6 The mse trend when iteratively add the filter method sorted features . . . 28
4.7 The mse changing curve along the feature selection process with the CMA-ES and wrapper method . . . 29
4.8 The feature importance ranking of RF . . . 29
4.9 The mse trend when iteratively add the embedded method sorted features . . . 30
4.10 The mse trend during the ANNs hyperparameters optimization with CMA-ES . . . 31
4.11 The relationship between the decision trees number and the mse_n of RF . . . 32
4.12 The average mse trend of the KNN models during the growth of k . . . 32
4.13 The mse of the 3 machine learning models with the best settings on the same test set 33
4.14 The prediction performance of the 3 machine learning models on the same test set 33
4.15 An e-bike design optimization sample . . . 34

List of Tables

2.1 Characteristics of the 4 data sources 12

4.1 Encoding methods overall comparison 26
4.2 Feature Selection Results 30

Chapter 1

Introduction

1.1 Background

International Bike Group B.V. (IBG) has businesses in many European countries and cooperates with many external brands, new bike models are released frequently. According to the company's statistics, in every month there are on average 31 new bike models are released online, on the contrary, an average of 27 bicycle models are offline monthly. Thus, IBG is always considering the new strategy for bike designs, especially the electric bikes (e-bikes) contribute a huge part of IBG's profit, so IBG starts a project to analyze consumer preferences to support e-bike designing.

Understanding the consumer's preference in depth can significantly help for making reasonable procurement plans and designing new bicycles, which not only can better serve the needs of consumers but also can reduce the storage burden and management costs of the warehouse and increase the sales and turnover of bicycles. In this thesis, we will discuss the feasibility of many different data sources and select the most reliable data to quantify consumer preferences in numerical form for analysis.

The e-bike designing is a complex job, as Figure 1.1 shows the insight of e-bike design, we can see that there are too many e-bike features should be considered during the designing process, so IBG wants to build up a system to dynamically learn the newest patterns of how the consumers like the e-bike features combination. And they also hope the system can according to the learned patterns to evaluate and optimize the e-bike designs automatically. Supervised machine learning and optimization of evolutionary strategy are the main methods we used to build the system.

There are 3 main problems can be solved by the consumer-driven product evaluation and optimization system:

- What kind of features in the e-bike designs have strong relationships with the preferences of consumers?
- How to estimate an e-bike's consumer preference score while the e-bike does not exist in the previews market?
- How to optimize the design of an e-bike?

Since our analysis results involved confidential information, the product names and component information in this thesis will be replaced by code.

1.2 Interactions Between IBG and Consumers

The initial preferences of a consumer do not always result in purchasing the most preferred bike, in order to minimize the impact of the company's operating strategy on consumer preferences, we

Figure 1.1: The insight of e-bike design

first need to understand the interaction between the consumers and IBG, which can help us to choose the reliable data source during analysis and find out strong connections between consumer preferences and product designs.

IBG is the fastest growing omnichannel specialty platform for e-bikes in Benelux and Scandinavia, IBG has a strong online and offline sales platform and serves consumers in the Benelux region, more than 85% of Dutch consumers have visited the websites of IBG when orientating to purchase new bicycles. IBG's consumer base is comprehensive, the consumer base includes different nationalities, gender and age segments. IBG serves consumers with a seamless omnichannel journey through the online platform assisted by 42 physical stores and more than 250 after-sales service points. IBG's business scope includes centralized bicycle online sales platform, transportation, assembly, after-sales, promotion, retail, experience store, bicycle insurance. IBG currently offers 22 external brands and 4 privately owned brands, namely Brinckers, Victesse, Vantuyl, Laventino.

The following are the service forms of IBG:

- **Offline Interactions** Factors such as stock availability can be of great influence, as well as incentives to nudge consumers towards a certain choice that reflects a more commercially attractive outcome for the company. Hence, the actual offline purchase is biased and we are not able to research the initial wills and preferences of consumers from offline interactions.
- **Online Interactions** We found the online interaction consists out of a combination of three different stages. First, many orientating consumers visit an IBG-owned exhibition platform, that offers product information on every well-known bike brand in the Dutch market, bicycles can't be purchased at the platform, and there is no online consultation service, and no discount information is included. Afterward, the potential consumers visit the IBG's online sales platform to seek for a commercially interesting proposition. The online sales platform contains discount information, bike-expert online consultation. After defining a number of interesting bikes, most consumers also visit IBG-owned physical stores, to take a look at the bikes in real life and test them out, before finally proceeding to actually purchase a bike.

1.3 Available Data Sources

In Section 1.2 we introduced the ways IBG interacts with consumers, but the offline sales are not suitable to be our analysis basis. Since limited e-bikes are available in each retail store, which causes

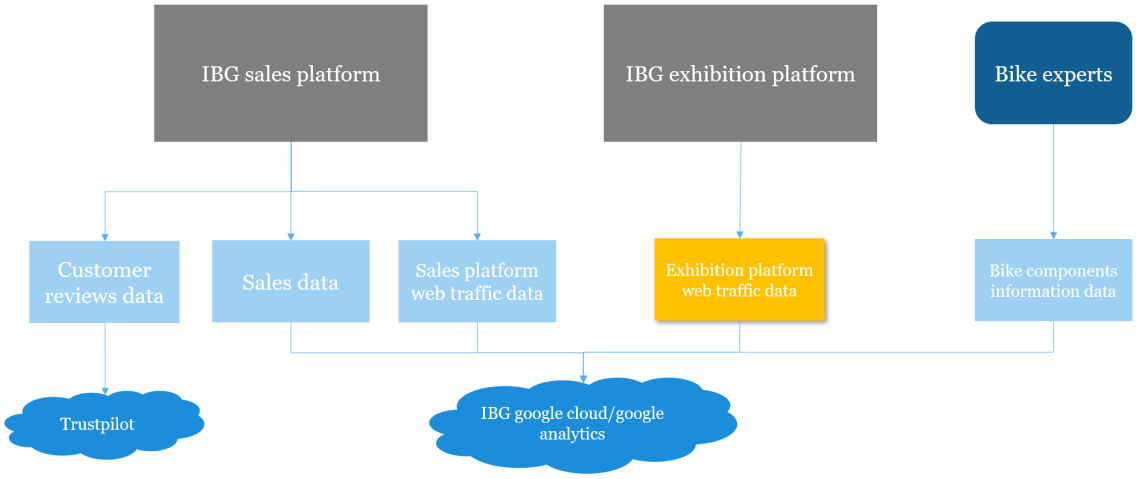

Figure 1.2: Schematic diagram of the generation and storage of each data source

limitations on consumer options, and a large number of consumers do not have enough knowledge of e-bikes and e-bike components, so the guidance from the clerks also could cause some impact on their choices, so the offline sales data can not objectively represent the consumers' preferences.

In comparison, online interaction is more suitable to be our analysis basis. As Figure 1.2 shows, during online interaction, there are many kinds of data that have been recorded, while most kinds of these recorded data are potential to represent consumer preferences, but finally, we chose to use the online exhibition platform web traffic data to quantify consumer preferences. Here we will introduce the basic information about these different kinds of data and we will explain the reason for our choice in Chapter 2.

1.3.1 Web Traffic Data

The web traffic data of the online sales platform and the online exhibition platform are recorded with Google Analytics separately. "Provided by Google Inc. , Google Analytics is used for a *R&D* resource devoted to cultural tourism, about the number of visits on a website and the traffic source, which includes organic results in search engines, links from referral web pages or direct access" [1], we can use Google Analytics recorded online traffic data to analyze the consumer's online behavior. Google Analytics records real-time data, which can be used to statistically analyze web traffic in different time periods, in different time period the web traffic data of the same web page will be different, and it is also possible that some new web pages are uploaded or some old web pages are removed in different time period.

Because IBG's web developers and employees visit the websites very often, and this amount of traffic data contribution is meaningless for the consumers'online behavior analysis, IBG's filters web traffic data, eliminating web data generated from the company's IP address.
Figure 1.3 shows a snippet of the Google Analytics web traffic data, as we can see Google Analytics includes these kinds of attributes to measure online visitors' behavior:

- **Page Views** The total number of times the page has been loaded, note that the same user may load one page multiple times in the same session, which will cause an increase in the number of page views of this page.
- **Unique Page Views** The total number of times the page has been loaded, note that no matter how many times one page has been loaded by the same user in the same session, the number of unique page views of this page just will increase 1.
- **Average Time On Page** The average amount of time a user spends on the single-page.

Page path level 2	Page Views	Unique Page Views	Avg. Time on Page	Bounce Rate	% Exit
	18,309 % of Total: 28.67% (63,871)	16,196 % of Total: 30.96% (52,310)	00:00:59 Avg for View: 00:01:14 (-19.83%)	64.30% Avg for View: 51.00% (26.08%)	27.37% Avg for View: 33.20% (-17.57%)
1. E-bike 1	**3,635** (19.85%)	3,465 (21.39%)	00:00:31	24.93%	12.02%
2. E-bike 2	**826** (4.51%)	595 (3.67%)	00:00:46	43.68%	17.80%
3. E-bike 3	**105** (0.57%)	87 (0.54%)	00:01:24	83.33%	36.19%
4. E-bike 4	**93** (0.51%)	80 (0.49%)	00:01:18	75.00%	41.94%
5. E-bike 5	**88** (0.48%)	74 (0.46%)	00:01:21	77.78%	29.55%
6. E-bike 6	**86** (0.47%)	81 (0.50%)	00:01:32	80.00%	30.23%

Figure 1.3: Google Analytics web traffic data snippet (e-bike names have been replaced by code)

- **Bounce Rate** Bounce means that when a visitor accesses the single-page without any interaction with this page or the session duration on this page is 0 second. The bounce rate is the percentage of the bounce sessions of this page.
- **Exit Rate** The percentage of site exits that occurred from a specified page or set of pages.

1.3.2 Online Sales Data

From the online sales platform, IBG records both online sales data in Google Cloud, and uses Google Big Query for querying jobs. "Google Cloud is a Google Cloud Storage component, the Big Query is powerful with storing structured data and querying in an efficient way" [2][3]. The sales data includes the sales time stamp, sales location, sales channel, bike names, and final sales price of each bicycle.

1.3.3 Consumer Reviews Data

From the online sales platform, IBG's consumer reviews data is recorded by Trustpilot, The Trustpilot corpus will collect consumers reviews of the products, as Figure 1.4 shows an example of the consumer review, which includes the bike name, the text form review, and four one to five stars rating of each product, which are quality rating, design rating, performance rating, and overall rating [4].

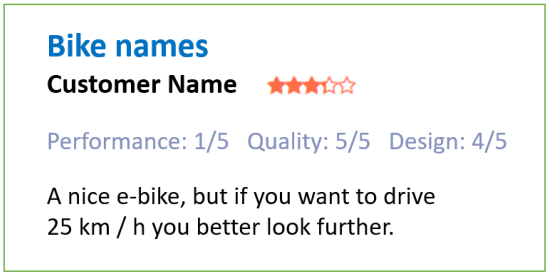

Figure 1.4: An Trustpilot review example

1.3.4 E-bike Component Information Data

The component information data contains important characteristics of each e-bike design. The component information data of IBG e-bikes is generated by the bicycle experts in accordance with

IBG's Component Information Recording Standards during the procurement and design process. IBG records each e-bike's component information with a total of 37 attributes, Figure 1.5 shows a snippet of the component information data, with the multi-attribute component information we are able to identify and differentiate among very similar e-bikes. Conversely, we also can base on component information to find out common points among very different e-bikes. But currently, the IBG bicycle experts just have 270 days of e-bike component information records, so the number of e-bikes we can analyze is very limited.

Bike name	Bike brand	Frame color	Saddle color	With front carrier?	Wheels color	.....
e-bike 1	victesse	black	black	no	black	.....
e-bike 2	brinckers	black	orange	no	black	.....
e-bike 3	brinckers	black	brown	no	black	.....
e-bike 4	brinckers	black	brown	no	black	.....
e-bike 5	cortina	green	black	yes	black and orange	.....
e-bike 6	cortina	green	black	yes	black and orange	.....
e-bike 7	cortina	black	brown	yes	gray	.....
e-bike 8	cube	gray	black	no	orange	.....
e-bike 9	sparta	silver gray	black	no	black	.....
e-bike 10	sparta	silver gray	black	no	black	.....
e-bike 11	stromer	black	black	no	black	.....
e-bike 12	victesse	black	brown	no	black	.....
e-bike 13	cube	black	black	no	black	.....
e-bike 14	gazelle	orange	black	no	black	.....
e-bike 15	cortina	brown	black	yes	black	.....
e-bike 16	cortina	black	black	yes	black	.....
e-bike 17	cortina	sky blue	black	yes	black	.....
.....	.....	.....	.....	.....	.....	.....

Figure 1.5: E-bike component information data snippet (e-bike names have been replaced by code)

1.4 Project Overview

In this thesis, we firstly will discuss the feasibility and potential bias of each available data source then select the most relevant data source to quantify consumer preferences to numerical scores. And we will also discuss how to encode [5] the text form e-bike component information data into the numerical features, the numerical features together with the consumer preference scores will be the feature vectors (X) and labels (Y) for supervised machine learning [6].

Based on machine learning models' predicting performance, we will analyze how to select the proper time period for learning useful patterns to predict consumer future preference on certain e-bikes. Also for improving machine learning models' predicting performance, we will conduct feature selection [7][8] to analyze how to select proper features to generate feature vectors which can make the machine learning models get the best capability for predicting.

In order to solve the first problem "What kind of factors in the e-bike designs have strong relationships with the preferences of consumers?" we used the Random Forest (RF) [9] approach to build the machine learning model, and by analyzing how each feature affect the RF model' prediction to figure out the feature importance.

For the problem of "How to estimate an e-bike's consumer preference score while the e-bike does not exist in the previews market?" we based on the knowledge we got from the previous steps to build 3 different machine learning models, which are an artificial neural networks (ANNs) model, an RF model, and a k nearest neighbor (KNN) model. And we select the best-performed model to estimate e-bikes' future popularity. Before the machine learning models' comparison, we based on covariance matrix adaptation evolution strategy (CMA-ES) [10][11] to do hyperparameter op-

timization [12] for 3 models to get them the best performance for comparison. We finally chose the ANNs model to be the e-bikes' future consumer preference score evaluator, which can estimate an e-bike's consumer preference score while the e-bike does not exist in the previews market.

In order to solve the problem of "How to optimize the design of an e-bike?" we used the well-trained ANNs model as the fitness function, and then also used the CMA-ES to optimize the e-bike design. For ensuring the optimized e-bike design is logical, we modified the optimization process of the evolutionary algorithms so that the evolutionary strategy can maintain some parameter variables during the optimization process. For improving both models' practical usage, at last, we designed a friendly interface for the e-bike evaluation model and the e-bike optimization model, which made the users can easily evaluate and optimize e-bike based on consumer preferences.

1.5 Thesis Structure

In Chapter 2 we will explain our criteria for data source selection and compare the advantage and feasibility of each data source. After we decide the data source to use, we will introduce how we process those raw data to become the consumer preference numerical scores. In Chapter 3 we will introduce the methods we used and the implementation details for building the consumer preferences driven product evaluation and optimization system. In Chapter 4 we will discuss the experimental results, besides, the settings and the conditions of each practical step are also included. In Chapter 5 we will summarize the project, the interesting patterns we found will be introduced, and we will compare our project with the other business intelligence projects, the future advice on this project will be given.

Chapter 2

Data Source Selection

In Section 1.2 we mentioned the three key problems we want to study. These three problems are based on the understanding of consumer preferences in the market, thus we need to select reliable data sources to be used to quantify consumer preferences to numerical scores, more precisely, a numerical score can represent the extent of how all consumers in the market like an e-bike. The consumer preference scores will be used as labels for machine learning models. In this chapter, we will introduce the criteria for our data source selection, and we will discuss the characteristics of four different data sources. Finally, we chose to use the online exhibition platform traffic data to calculate the consumer preference scores. At the end of this chapter, we will also discuss the effects of potential biases that are hidden in the online exhibition platform traffic data.

2.1 Criteria for Data Source Selection

Consumer preferences are very subjective, in our case, we need to use other actual values to represent consumer preferences. As we introduced in Section 1.3, we have several different kinds of data sources that have the potential to be used to quantify consumer preferences, and we still need to consider the origin of these data sources and their characteristics to determine if they are appropriate enough to represent consumer preferences. Inspired by the traditional questionnaire design [13][14][15][16], we have summarized the following criteria for evaluating data sources.

- **Can consumers get enough product information before the data is generated?** While in most cases, consumers don't need to know all the features of the product at the time of purchase, they only need to select products based on their unique concerns. But for designers, they want to know as much as possible about the consumer's attitude towards the details of the product. So we have a criterion that the consumer should be able to access almost all the bicycle information before the data is generated.
- **Number of issued e-bikes** Due to the functional differences between platforms, the number of issued e-bikes in each platform is different, so the number of issued e-bikes that appear in the different data sources will also be different. In order to analyze more e-bikes features, we hope that more e-bikes are issued in the selected data source.
- **Human influence** There is always competition and cooperation in all business fields, and these market behaviors will affect the initial will of consumers. In order to design e-bikes that consumers really like, we need to select data sources that are less interfered by human factors for analyzing.
- **Amounts of data** A larger data amount can reduce the bias caused by outliers and small probability random events, so that we can learn more general patterns, which is beneficial to machine learning models' predicting performance.
- **Display order influence** Since our available data sources are generated by users of IBG's web platforms, and a large number of products are displayed on platforms, and these products are

assigned to different pages to be displayed, inevitably, the order of display will cause errors in our analysis.

2.2 Data Sources Comparison

As we introduced in Section 1.3, we have online exhibition platform web traffic data, online sales platform web traffic data, sales data, consumer reviews data and bike component information data available. Wherein bike component information data will be encoded as e-bikes' feature vectors, so we will discuss which of the other 4 data sources is more suitable to be used to calculate consumer preference scores.

The Table 2.1 shows the 4 different data sources characteristics in the available 270 days, the issued e-bikes are the number of different e-bikes exist in each data source. We can find out that the exhibition platform traffic data contains more issued e-bikes data, and richer in data amount, which is the optimal data source to be analyzed.

Data Source Characteristics (in 270 days)		
data source names	issued e-bikes	data amount
online sales data	376	13698 e-bikes were sold
consumer reviews data	376	336 reviews received
sales platform traffic data	376	31134 unique page views
exhibition platform traffic data	**1562**	**491956 unique page views**

Table 2.1: Characteristics of the 4 data sources

But we should not only consider the data diversity, but the unexpected influence also should be considered. Now we will discuss the unexpected influence of each data source to verify if the exhibition platform traffic data is really suited to be used for quantifying consumer preferences.

The advantage of sales data is directly related to the products' market performance. But the downside is that the e-bikes being sold by IBG are just a part of the e-bikes in the Benelux market, which will limit the analysis sample amount. Secondly, sometimes the shortage of bike storage and other human factors will also affect selling. Thirdly, price fluctuations can also affect consumers' choices. So we will not use sales data as a variable for evaluation.

Although the consumer reviews data is the most direct feedback of e-bikes, only a few consumers will give comments on the platform. According to the statistics, only 31% of e-bikes are reviewed, since the consumer reviews number is not sufficient, it is dangerous to evaluate e-bike designs based on a small sample.

The web traffic data of the e-bike exhibition platform is suitable as an e-bikes design evaluation data source, the reason why we should base on the exhibition platform traffic data instead of the sales platform traffic data is that the sales platform includes interference factors such as discount information and online consultation. And the e-bikes number on the sales platform is much lower than the exhibition platform, the exhibition platform contains almost all the e-bike brands in the Benelux market and the average number of on showing e-bikes is around 1562. The exhibition platform web traffic data is very objective, because on the exhibition platform, consumers only can check the e-bike components information and the suggested retail price, and they can not buy e-bikes from the platform. Besides, the exhibition platform has a huge number of page views. In the 270 days, the number of page views is around 500000. we also considered the constraints as bounce rate, average time on page, unique page views to more accurately mining the consumer preference patterns.

Comparing with the other data sources, the exhibition platform web traffic data is the optimal choice to support the consumer preferences analysis, For using the exhibition platform traffic data to quantify consumer preferences, we should understand the consumers' behaviors on the exhibition platform. The exhibition platform offers more than one thousand e-bike models for consumers'

visiting, since the visitors will find their favorite e-bikes through different ways, such as keywords querying, survey-driving redirection, home page's filters, direct landing from the external network or page by page scanning, no matter they access their ideal e-bikes pages in what way, one thing could be sure, before they access the product pages they already take in account some important factors that they care, the factors could be the outlook, recommend sales price, brand, engine and more, so with the huge amount of platform visiting, the e-bike pages' unique page views can show the e-bikes' consumer preference among visitors. And we should also consider the e-bikes' exhibited days. Since the e-bikes are start exhibited on different dates, the old e-bike models generally have more unique page views than the new coming e-bike models. Another factor is the bounce rate, if a visitor access a product page but quit immediately or didn't start any sessions on the product page, maybe the e-bike is not that attractive to the visitor.

There is one factor can affect consumer online behavior and bring bias during quantifying consumer preferences, which is the e-bike's display order on the exhibition platform web pages. The exhibition platform displays an average of 1559 e-bikes in each month. However, each web page of the exhibition platform can only display 17 e-bikes, so the display platform has an average of 92 pages per month for displaying e-bikes. Due to IBG did not record the e-bikes' display order, the influence from the display order is unremovable. Fortunately, the display order of e-bikes on the exhibition platform is frequently changed, and the consumers have several different products sorting options, and the consumers can query the relevant e-bike with multiple filters or directly land on the product pages from the external networks, in this case, the display order influence has been reduced a lot. So in this thesis, we will not consider the display order influence, and the display order influence removing topic will be discussed in the future works.

2.3 Consumer Preference Scores Calculation Based on Exhibition Platform Traffic Data

We chose exhibition platform web traffic data as the data source for calculating consumer preference scores, and consumer preference scores will be used for machine learning labels (Y). We regard the e-bike instances set as I, for each e-bike model i we have $i \in I$, since the web traffic data is data flow, before we start to calculate an i's score we should define the observation time period $Q \in \mathbb{N}$. Note that in the time period we want to analyze, some new e-bikes may be published in the middle of this time period, and some old version e-bikes may be removed, IBG has recorded the exhibited days of each e-bike i which is $q_i \subseteq Q$. We also consider the bounce rate $b_{q_i} \in [0, 1]$ of an e-bike i in Q, so an e-bike i's consumer preference score score y_i is calculated as the equation 2.1 and normalized as the equation 2.2, we need the consumer preference scores set Y to support further analysis.

$$y_i = \frac{p_{q_i} \times (1 - b_{q_i})}{q_i} \tag{2.1}$$

where p_{q_i} represents the unique page views of an e-bike i in Q.

$$y_i = \frac{y_i - \min Y}{\max Y - \min Y}, y_i \in [0, 1] \tag{2.2}$$

Chapter 3

Methodology

In Chapter 2 we introduced the data sources selection criteria and how to calculate the e-bikes' consumer preference scores Y from evaluating the consumer preferences of e-bike designs among the exhibition website visitors, now we can take the Y to some advanced data science models to solve the problems we mentioned in 1.1. In this chapter, we will discuss how to find the relevant and valuable features of e-bikes, and relate those features and the consumer preference scores Y to machine learning models and mining out the hidden patterns of consumer preferences.

3.1 Machine Learning Techniques

Consumers' preferences analysis is not a linear problem, which means we can not simply use a statistical way to calculate out each e-bike feature' consumer preference score and select the top-ranked features to combine together to be the optimal design, for example, If we detect out black is the most popular frame color and orange is the most popular wheel color, but maybe the bike with these two colors together is not well accepted by consumers. So it is valuable to analyze the combination of features rather than every single feature. However, there are too many features that should be considered at the same time, on this occasion, machine learning techniques are necessary to be used, the machine learning techniques are able to handle high-dimensional, multi-variate data and extract implicit relationships within large data-sets in a complex and dynamic environment [17].

Since this project is dealing with a regression problem and the data set is not big, so we decide to base on supervised learning strategy to design our machine learning models, the strategy of supervised learning regression is input a feature vector which can represent the characteristics of a sample into the supervised learning model, and the model will predict a numerical value of the sample, while we know the correct value (label) of the sample, we can use the mean squared error (mse) between the predicted value and correct value to represent the machine learning model's predicting performance, and adjust the model with the mse to improve the predicting model's predicting performance until the model is accurate enough for our requirement.

Indranil and Radha proposed business data mining in a machine learning perspective [18] in 2001. They introduced five kinds of machine learning techniques to achieve business data mining. We tested and compared 3 techniques which are Rule induction (RI), Neural networks (NN) and Case-based reasoning (CBR).

- **RI technique** The decision tree (DT) method is a commonly used RI strategy. "The nodes of the DT are a set of rules, the root node of a DT represents all examples in the training set" [19][18]. The advantages of DT technology are obvious, which is highly interpretable, the DT is very intuitive for decision-makers in the company. But the disadvantage of the DT technology is also significant. DT is easy to overfit to a training set, meaning that the decision tree may have high accuracy on the training set, but the accuracy could be very low on the test set. Fortunately, Random Forest (RF) can avoid this problem, "RF is the bagging [20][21] based machine learning method, unlike most machine learning algorithms, train an

RF just need 2 parameters, one is the number of trees, another is the number of features" [9], which makes RF's sensitivity to parameters is low, RF has a robust performance and is often less prone to overfitting than decision trees, but also because of the small dependence on parameters, RF performance is difficult to improve. For the problem "What kind of factors in the bicycle design have strong relationships with the preferences of the consumers?" We also chose the RF method to deal with. RF can estimate the feature importance by checking how the prediction error change when just edit one feature while keeping the other features unchange [9].

•**NN technique** NN are the most commonly used machine learning models, is also known as Artificial Neural Networks (ANNs), ANNs was originally a machine learning method proposed by simulating the human neural system. ANNs are models that can be used for regression or classification problems [22]. The advantage of ANNs is that even if the patterns in the training set are not obvious, the ANNs still can learn from the training set and have good predicting performance, basically, the ANNs can learn any classification and regression functions. While the drawback of ANNs is the learned patterns are hidden in the network structure and node weights, which is difficult for humans to understand. Secondly, because ANNs are very sensitive to network parameter settings, although this gives ANNs the potential to achieve better performance, it may also result in poor learning performance due to errors in hyperparameter settings. Therefore, the training of ANNs requires the use of hyperparameter optimization strategy [12]. For the problem "How to estimate an e-bike's consumer preferences while the e-bike does not exist in the previews market?" Although we can use RF to estimate the e-bike consumer preference score, since RF is not sensitive to hyperparameters, so we are wondering that if the ANNs model with good hyperparameter settings can be better than the RF model, so we also designed and optimized an ANNs model for comparison.

•**CBR technique** The CBR technique is to store the historical cases in case-base. When predicting a new case, CBR will calculate the similarity between the new case and the historical cases, and select the most similar historical case's label as the output of the new case. The k nearest neighbor (KNN) algorithm is often used in CBR [23]. Although RF and ANNs can learn the hiding patterns from the datasets and make forecasting based on the patterns they learned, and theoretically they are indeed effective predictive models, but from the business perspective, predicting the product performance by the difficult models is less persuasive than predicting from the similar existed cases, so we decided also use CBR strategy to estimate e-bikes' consumer preference scores, k nearest neighbors algorithm (KNN) is most commonly used CBR algorithm, which is effective way to find similar cases to solve the classification or regression problems [24]. We based on Euclidean distance calculation [25] designed a KNN predictive model and compared the performance of the KNN model with the RF and ANNs models. Even though the performance of the KNN model's performance is not perfect, it can provide companies with similar cases to help the company's business decisions. The setting of the k value has a huge impact on the performance of the KNN. A k value that is too small can cause overfitting, while the k value that is too large can cause underfitting of the KNN model. So we should progressively increase the k value while checking the KNN's performance to find the optimal setting of the k.

There are more than a lot of other machine learning algorithms applied in business data mining, but the purpose of this thesis is not to introduce all existing machine learning algorithms, so we mainly discuss how to select appropriate machine learning algorithms to solve the three questions proposed in Section 1.1.

3.2 Covariance Matrix Adaptation Evolution Strategy

The Covariance Matrix Adaptation Evolution Strategy CMA-ES is known for its state-of-the-art performance in continuous functions [26][10], CMA-ES enumerates amongst the most efficient evolutionary methods for continuous parameters optimization [27]. For this project, we will use the CMA-ES to optimize the hyperparameters of predicting models, and after we get the qualified predicting models we will regard them as the fitness function and base on CMA-ES again to search

for the optimal e-bike designs. The CMA-ES is a standard method which is already evaluated in a lot of researches, and the CMA-ES does not require a tedious parameter tuning for its application, so in this project, we regard the CMA-ES as a black-box optimization function, and we only should care about the input and fitness value.

3.2.1 Hyperparameter Optimization

The selection of the hyperparameters settings is the common problem of all the methods we used, especially for ANNs the setting of the number of neural nodes will significantly affect the ANNs model's predicting performance. We should understand that the ANNs models' performance is not only determined by the instances sets but also the models' hyperparameters and structures, it is possible that there are very obvious patterns are hidden in the instances sets, but because of the model is bad initialized, the model can not learn the useful patterns to make good predictions. So it is necessary to try different hyperparameter settings of the ANNs models, the problem is the hyperparameters are continuous and infinite variables, thus we can not test the different settings one by one to find the optimal setting, so we should find an efficient way to search for good hyperparameter combinations.

For avoiding the huge number of hidden layers to make the training process of ANNs models become too expensive, we keep the number of hidden layers of ANNs models as 2, and the hyperparameters we will optimize are the nodes number of each hidden layer. For each step of CMA-ES optimization, we based on the new number of nodes settings to train 5 ANNs models, and we regard the 5 models' average mse on the test set to be the fitness value of CMA-ES optimization process, the reason why for the same setting we trained 5 independent ANNs models is because we want to reduce the randomness of the training and testing process. After 1500 rounds of CMA-ES hyperparameter optimization, we took the number of nodes setting which lead the minimum average mse of the ANNs models on test set to be the best setting of ANNs models.

For RF and KNN models we didn't use CMA-ES to optimize the hyperparameter settings. For RF models the number of decision trees is the hyperparameter that should be optimized, we tested the 100 settings of the number of decision trees, the number of decision trees was chosen from 1 to 100 and for each setting we also trained 5 independent RF models and took the 5 models' average mse on test set to be the fitness value, and we selected the number of decision trees setting which can lead the minimum average mse for the RF models' setting. The searching procedure of KNN's k setting is similar, we tested all the possible k settings to search for the optimal one.

3.2.2 E-bikes Optimization

The CMA-ES is also an efficient way to optimize the e-bike designs, after comparing the three hyperparameter optimized machine learning methods, we found that the ANNs models have the best predicting performance. So we can regard the ANNs model as the fitness function, and the output is the e-bike's predicted consumer preference score, we used the CMA-ES method again to search for feature vectors with a higher consumer preference score to get a better e-bike design [28].

Note that the ANNs model we finally selected was trained by the feature vectors encoded by one-hot encoding method, we will introduce the reason for using one-hot encoding method in the later of this chapter. As Figure 3.5 shows the feature vectors are sparse boolean vectors, a boolean vector contain many features of different attributes, and for each attribute, there is only one feature's value could be 1. Since the feature values of the feature vector should not be simply changed, we can not directly use the feature vector as the input for the CMA-ES optimization model.

As Figure 3.1 shows an example of e-bike optimization process, the CMA-ES input vector has the same dimension as e-bike attributes, since we know the feature dimension of each e-bike attribute, in the optimization steps, the input vector is compiled according to the feature dimensions. The rule of compiling is that if the input vector value smaller than 1, then the compiled value equal to 1, if the input vector value is bigger than the feature dimension of this attribute, then the compiled value equal to feature dimension of this attribute, and if the input vector value is between 1

and the number feature dimension of this attribute, then the compiled value equal to the closest integer. For instance, the feature dimension of the frame color attribute in this figure is 4, and the frame color value of the input vector is 0.5 which is smaller than 1, then the compiled value is 1. After we get the compiled vector, for each e-bike attribute, we generate a sub-feature vector with the attribute's feature dimension and set 1 at the corresponding position, the other value of this sub-feature vector will be 0, then we combine all the sub-feature vectors to become the feature vector for feeding into the fitness function and evaluate the fitness value for the next optimization step.

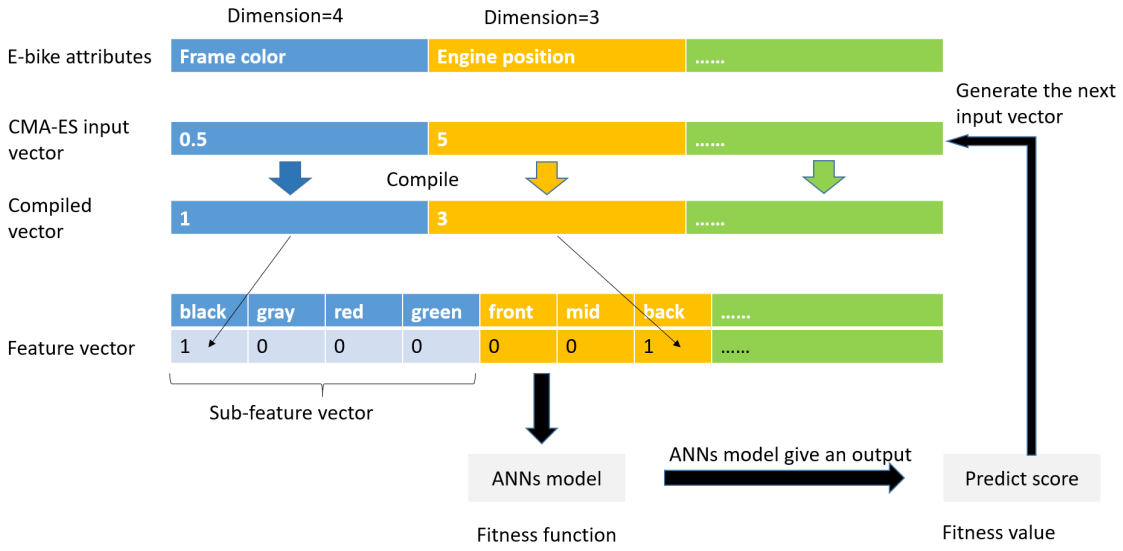

Figure 3.1: The example of e-bike optimization

For avoiding the optimized bike design violate the common sense of e-bike designing, such as after optimized the result suggests that design a transport bike without a front carrier which is not correct. So we also created a function that can keep some attributes' values to be constant in the CMA-ES input vector during the optimization process.

3.3 Time Series Analysis

Analyzing the e-bikes' web traffic data is a time series problem, the different time periods could contain different patterns [29]. Think about that, if we just based on one month's web traffic data to train our machine learning models, then the models just can learn from very few e-bike instances, then the patterns the models learned are neither sufficient or qualified enough to provide precise predictions on the e-bikes which machine learning models had never met. On the contrary, if we feed the machine learning models with twenty years' web traffic data records, then the machine learning models may over-learn the hidden patterns, which also could affect the machine learning models' performances. In summary, in the total time period ($T = 270$) we have, there are two factors should be considered, which are the time span (s_t, $t \in T$) and the start date (d_t, $t \in T$). We want to figure out 3 problems with the analysis of the time series, which are:

- We want to find the proper s_t to enable machine learning models to learn useful patterns and make accurate predictions of e-bike designs' future consumer preference scores.

- We want to prove that d_t will not cause a significant impact on the performance of machine learning models, so we can dynamically analyze consumer preferences.

- We want to prove that with the increase of s_t there are more different e-bike features that can be analyzed.

We did a lot of tests and the results of the tests will be shown in Section 4.1. The problem we faced during the test was that we only had 270 days of e-bike component information data, so the time period we can analyze is very limited. We analyze the first two questions based on the random forest method, because random forest method is not sensitive to the hyperparameter settings, and not easy to be affected by the noise data, and is also an efficient way to reduce the affection of overtraining [30].

As the Example 3.2 shows, we generated the training set and test set from two adjacent equal length time segments, then we used training set to train a random forest model, note that as Figure 3.4 shows that the feature vectors of training set and test set are e-bike component information which encoded with one-hot encoding method, after the random forest model is well trained, we calculated the *mse* of the random forest model on test set. As we know, the *mse* can represent the random forest model's predicting performance, but due to random forest model training is an uncertain process, to reduce the uncertainty, for each pair of the training set and test set, we independently processed the training and test 5 times, and we got 5 independent *mse*s, then we used the average of the 5 *mse*s to represent the random forest models' general predicting performance. With this method, we can test different settings of s_t and d_t to figure out how s_t and d_t will affect machine learning models' predicting performance. And we found that with the same s_t the different d_ts did not cause huge impacts on the predicting performances of machine learning models, which proves that we can dynamically learn consumer preferences through machine learning models. But when choosing the different s_ts with the same d_t the machine learning models' average *mse*s will decrease as s_t increases.

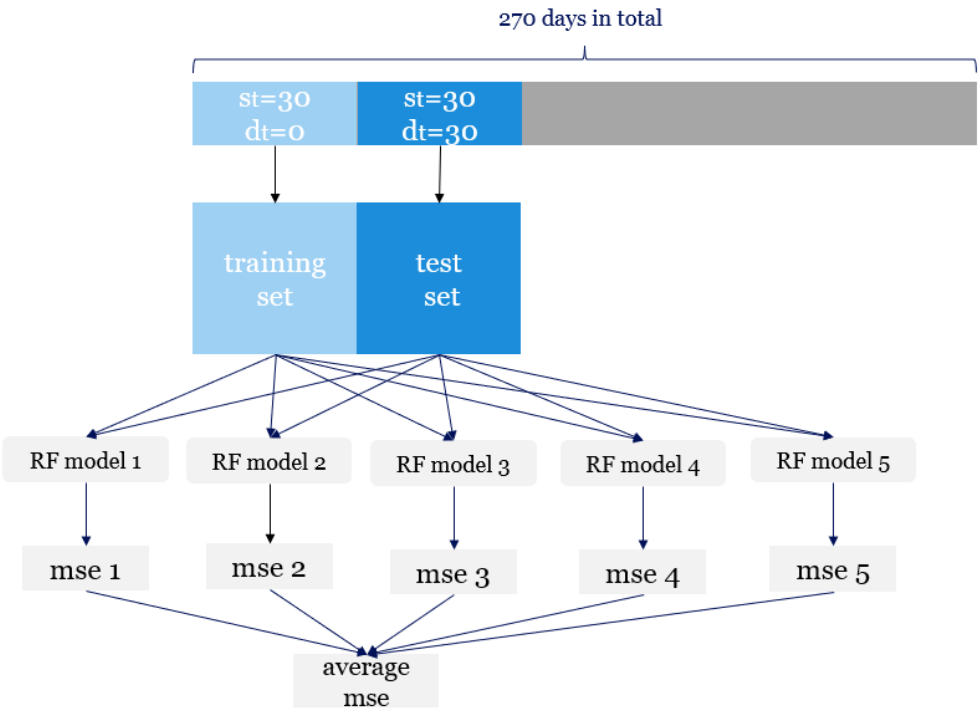

Figure 3.2: The example of evaluating RF model's predicting performance on a certain time periods

RF models' predicting performance is not only affected by the training set and test set, but the

hyperparameters of RF also have affections. RF has two parameters to be set, one is the max features number of each decision tree, and the other is the number of decision trees. So we analyzed RF average mses in different time periods with the max features number of each decision tree is automatically defined and tested 100 different settings of the number of trees which is from 0 to 100. Finally, we found that when the number of trees is greater than 25, all average RF mse will converge. So, in further analysis, we will define the number of trees of all the RF will to be 40.

For having more options for e-bike designs, e-bike designers want to analyze more features' affection on consumer preferences, so when we are choosing s_t and d_t to generate feature vectors and labels for machine learning models' training, we hope more different e-bike features are included in the selected time period. For example, in the period of $d_t = 0$, $s_t = 30$, there is no e-bike in which frame color is red, thus, machine learning models can't learn how red frame color affects consumer preferences. But when $d_t = 0$, $s_t = 60$, there appears some e-bikes with the red frame color, and then the machine learning models can learn the red frame color's influence on consumer preferences. For the same s_t, we chose all the different start dates d_T and counted the appeared features in each time period, and we calculated the average appeared features' number of different s_t, d_T. We counted the average number of features in different s_t, d_T and found that the average number of features was positively correlated with s_t.

Because the longer time span can effectively increase the number of appeared features and reduce the machine learning models' average mses, we will choose the larger s_t within the 270 days. In the subsequent study, since we only had 270 days of e-bike component information data, we selected the time period of $s_t = 120$ with $d_t = 0$ to generate the training set and use the time period of $s_t = 120$ with $d_t = 120$ generates the test set for machine learning models' evaluation.

3.4 E-bike Component Information Encoding

Because IBG's bike experts record the bike component information data in text forms, and the data is discontinuous categorical data, which is difficult for machine learning models to learn from. In order to convert them into numerical form feature vectors X, we need to encode them in a proper way. There are two commonly used methods to convert the text features to numerical, one is dummy encoding, and the other is one-hot encoding [5]. We tried both methods, and finally we chose to use the one-hot encoding method to encode the bike component information data.

Bike name	Bike brand	Frame color	.....
e-bike 1	victesse	black	.....
e-bike 2	brinckers	black	.....
e-bike 3	brinckers	black	.....
e-bike 4	brinckers	black	.....
e-bike 5	cortina	green	.....
e-bike 6	cortina	green	.....
e-bike 7	cortina	black	.....
.....	.....	.....	.....

dummy coding →

Bike name	Bike brand	Frame color	.....
e-bike 1	1	black	.....
e-bike 2	2	black	.....
e-bike 3	2	black	.....
e-bike 4	2	black	.....
e-bike 5	3	green	.....
e-bike 6	3	green	.....
e-bike 7	3	black	.....
.....	.....	.....	.....

Figure 3.3: Example of dummy encoding

For example, as Figure 3.3 shows the process of encoding bike brands with dummy encoding method. In the first 7 rows, there are 3 unique options for bike brand, which are "victesse", "brinckers" and "cortina", then we base on the order of the three brands appear in the "bike brand" column to refer them as 1, 2, 3, and store the text-number pairs in the encoding dictionary, note that there are more unique bike brands in the column, and the other unique bike brands will also be encoded and save in the encoding dictionary. Then we can base on the dictionary to encode the whole "bike brand" column. But the features which original in text form are discontinues fea-

tures, while after dummy encoding those discontinues text features become the continues numerical features, which will affect ML models' performance, for machine learning prediction the predicted value $y_{pred_i} = f(x_i)$ where f is the predicting function, the bike brand feature is included in the x_i, because we don't know if a bike brand is better than the others, all the unique bike brands should be treat equally before training, but with dummy encoding method, the bike brands have the differences in number and that could have impact on model training.

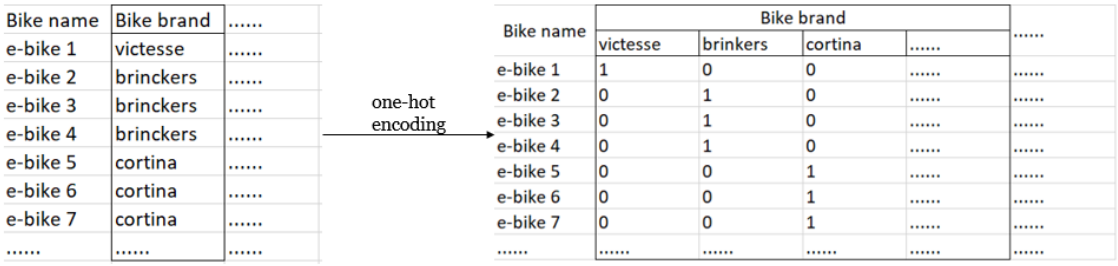

Figure 3.4: Example of one-hot coding

Another way is one-hot encoding, as Figure 3.4 shows the "bike brand" column is converted into a boolean sparse matrix, in the bike brand boolean matrix, each row can represent an e-bike's brand, and the sum of each row is always equal to 1, which won't cause the difference in numerical value while still can represent different brand features.

As we introduced in Section 3.3, we generated the training set in the time period of $s_t = 120$ with $d_t = 0$, and the test set is generated from the time period of $s_t = 120$ with $d_t = 120$, then we based on RF's average mses to evaluate the effects of one-hot encoding method and dummy encoding method on the machine learning process. We increased the RF's number of trees settings from 0 to 100 and compared how the feature vectors generated by the two encoding methods affect RF models' predicting performances. After testing, we found that when the number of trees is small (number of trees < 25), the one-hot encoding generated feature vectors is more conducive to RF to obtain a smaller average mse. While the RF's number of trees becomes larger, there is no obvious dominance relationship between the two encoding methods. However, in order to avoid unnecessary errors caused by encoding methods, in the subsequent research we will train all the machine learning models based on the feature vectors generated by the one-hot encoding method.

3.4.1 Feature Selection Techniques

In IBG's bike components information set there are 37 different attributes of each bike, but there are 7 attributes that are e-bikes' unique information such as e-bikes' names, e-bike's webpage URLs, etc. So we will not choose that unique information to generate feature vectors. After removing these 7 attributes, there are 30 e-bike attributes can be encoded into feature vectors X.

In the previous Section 3.3 and Section 3.4 we took all the 30 e-bike attributes for analysis without feature selection, but it is important to select relevant features to train machine learning models. Since Feature vectors are the inputs of machine learning models, for machine learning models' training, the good features may contain useful patterns which can significantly improve the predicting performance, while the bad features may include the irrelevant or wrong patterns which are meaningless for machine learning and increase the computational complexity of the training process, and even will have some negative affections to make the machine learning models overfitted. For an e-bike design, we can extract thousands of features to represent, although the IBG's components information data set contains the most common attributes of e-bikes, the irrelevant or redundant attribute still exist. For reducing the computational requirement and improving the machine learning models' predicting performance we based on the 3 methods are mentioned in

Section 3.4.1 to conduct the feature selection.

As Figure 3.5 shows the example construction of the feature vectors before feature selection, The feature vectors (X) were generated from time period of $s_t = 120$ with $d_t = 0$ and encoded with the one-hot encoding method, and we also chose the web traffic data of the same time period to produce the labels (Y). We regard the 30 attributes as the attributes set (A) each attribute ($a_i,\ i \in \{1, ..., 30\}$) as one member of A, and each a_i after encoded by one-hot encoding method will become a vector with z_i dimension (z_i is the number of features of a_i), there are just 0 or 1 in the vector of a_i, so each dimension of the vector can represent a feature status, such like the first column can represent each e-bike's frame color is black or not (when the feature value is 1, the e-bike's frame color is black). If we do feature selection, the new generated feature vector can contain several or all features of different attributes.

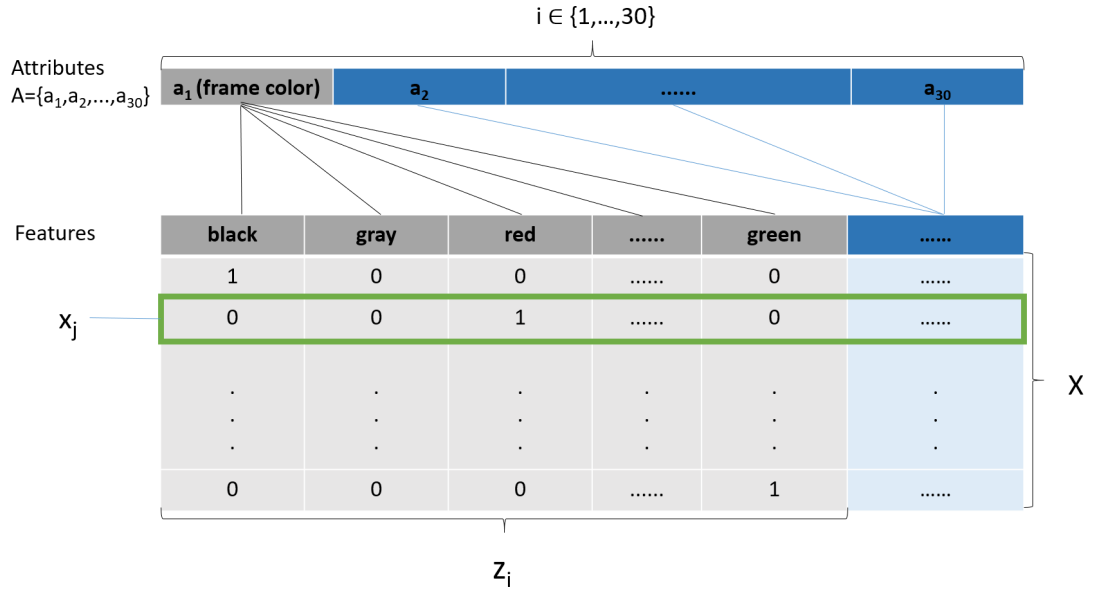

Figure 3.5: The example construction of the feature vectors

In order to meet the designer's needs, during the machine learning models' training in our subsequent study, we actually used the features selected by the designers to form the feature vectors, they selected all the features of 11 attributes. But we still want to know if these features selected by the designer will make the machine learning models perform better or worse than the features selected by standard methods. Thus we tried 3 additional standard feature selection methods (Girish Chandrashekar, 2013), and we based on a random forest machine learning model's predicting performance to evaluate the features selected by 3 standard methods and the features selected by IBG's designers.

- **Integrated Selection** The e-bike designers chose the e-bike attributes that they want to learn, and the feature vectors are generated by all the features of each attribute they selected. And the selected features were finally used to generate feature vectors for machine learning in our later research.

- **Filter Method** The most basic feature selection method, the filters methods just operate on the data set, the aim is to find out the features which are most relevant to the labels, the most common way is to calculate the absolute value of correlation coefficient between each attribute and the labels Y, and select the most relevant attributes to create the feature vectors [7]. The filter method is fast to select features, but easy to bring the inferior result,

because some features are not showing patterns along, but showing patterns by interacting with some other features together.

•**Embedded Method** The embedded method usually uses the internal information of regression or classification models like the feature importance provided by the random forest model, the random forest can provide the features' importance with lower computational complexity. In random forest, for regression problems the importance of features is measured by the predicting error while randomly permuting the features, more important the feature is, the smaller error the prediction will finally provide [31][8].

•**Wrapper Method** The wrapper method is guided by the outcome of the models, they regard the feature set as a search space to search the set of features that shows the best performance of a test model. the wrapper method usually provides a good result, but the computing complexity is really high [32].

In our experiments, we composed different feature vectors by changing the feature combination. Together with the labels Y, each new feature vectors X will be evaluated by RF models' average mse on the test. However, due to the number of possible feature combinations is too big, it is impossible for us to evaluate all the feature combinations, so we used CMA-ES [27] to gradually search for the feature combination which can lead a lower RF models' average mse.

As shown in Figure 3.6, we used a boolean feature selection vector to represent the selection status of features, and we use the RF models' average mse as the fitness value. Based on CMA-ES, during gradually find the better feature combination, the values of the boolean feature selection vector will be changed, and the values of the feature selection vector will no longer be an integer after each change, so before the next evaluation, the feature selection vector will be compiled again as a boolean vector, the rule of compiling is that if the selection value is closer to 0 than 1 then the selection value will be 0, on the contrary, if the value is closer to 1 than 0, then the selection value will be 1.

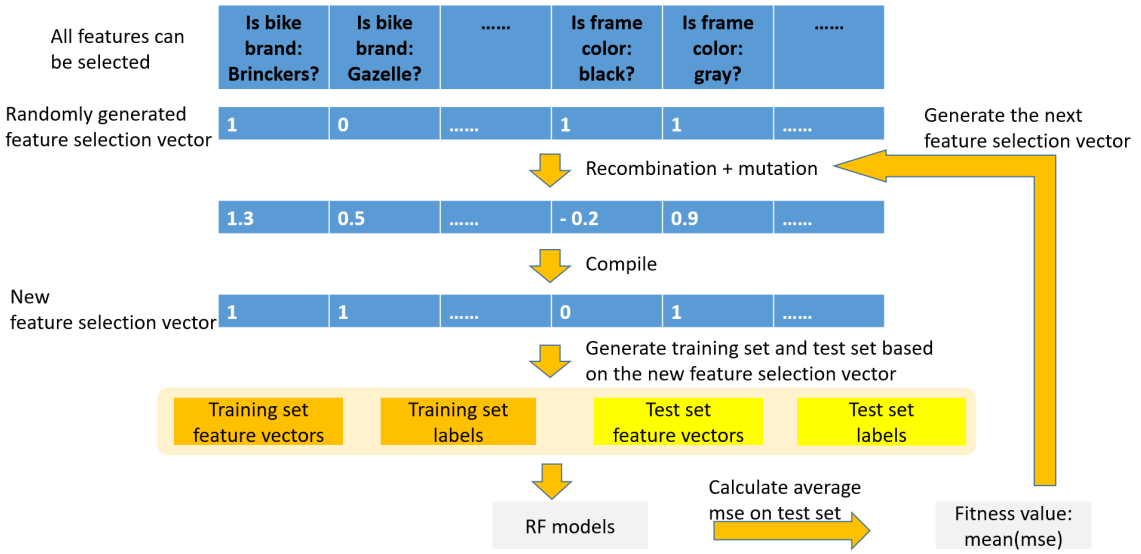

Figure 3.6: The feature selection process with wrapper method

After we tested those feature selection methods and we realized the features selected by bike designers performed not much worse than the features were selected by the other 3 standard feature selection methods, so we finally used the features selected by bike designers to build the feature vectors for machine learning.

3.5 Implementation

As Figure 3.7 shows, we first selected the time duration for analysis, then we queried out the e-bikes are displayed on the exhibition website and got those e-bikes' component information data as the feature sources, next we encoded the feature vectors for machine learning models, and we conducted the time series analysis to search for suitable time period for analysis, then we did the feature selection to select the relevant features. After these steps, we trained 3 different machine learning models and optimized their hyperparameters to get them their best predicting performance and selected the best-performed model to be the fitness function of the optimization model.

During feature selection, we got the feature importance, which can represent the component importance to answer the first question of 1.1. The best-performed machine learning model could be the e-bike evaluation model to predict the new e-bike's consumer preferences among consumers. And the optimization model can give inspirations on the e-bike designs and solved the third question.

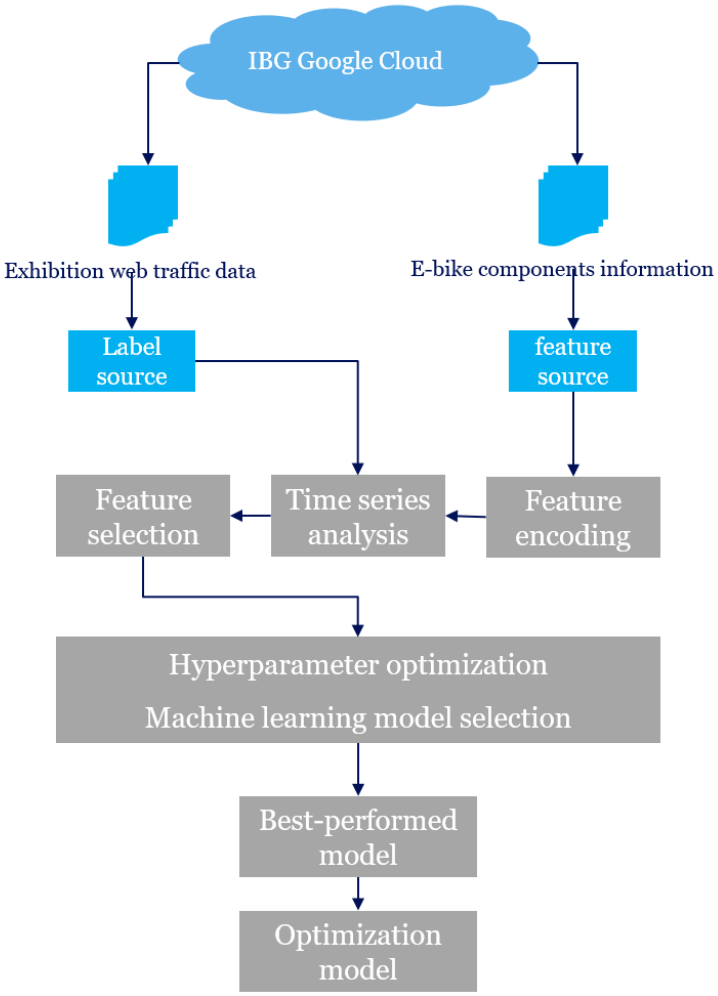

Figure 3.7: Implementation overview Y

Chapter 4

Experiments

4.1 Time Series Analysis

Since the machine learning methods we used are supervised-learning based and analyzing the e-bikes' web traffic data is a time series problem, the different time periods could contain different patterns [29]. So we should consider how the time span s_t and start date d_t affect machine learning models' predicting performance.

As Section 3.3 introduced, we used RF models' average mse on test set to evaluate the predicting performance. The entire time period T that IBG has e-bike component information data records is 270 days, for having the equal time period to generate the training set and the test set to evaluate RF models' predicting performance, so the maximum time span s_t that we can choose is 120 days, for example we choose the 0 to 120 days to generate the training set, and we choose the following 120 to 240 days to generate test set. We will analyze the affections of different time span s_t, $(s_t < T, t \in \{1, ..., 4\}, s_1 = 30, s_{t+1} - s_t = 30)$, and all the possible start date d_t, $(d_t \in T, d_1 = 0, d_{t+1} - d_t = 30)$.

Although the hyperparameter settings of RF models are very simple, there is a factor still should be considered which is the number of decision trees (n), while the n increasing, the mse of a RF model will decrease and converge to a certain level. So firstly we analyzed out how many decision trees n are sufficient for the RF model. We use the $mse_{n,s_t,D}$ to represent that with the certain n the average mse of the RF model which is trained and tested with the instances generated from the s_t with all the possible dates D. As Figure 4.1 shows, the different $mse_{n,s_t,D}$ $(n = \{1, ..., 100\})$s start converge when n reach 25. To ensure the RF models have enough decision trees, we will create RF models with different time series data sources with the $n = 40$.

From Figure 4.1 we also can be told that the time span s_t will lead the $mse_{n,s_t,D}$ to converge in different levels, for more clearly understand the relationship between the average $mse_{n,s_t,D}$ and the $S\,(S = \{s_1, ..., s_4\})$, we set the $n = 40$ to get the $mse_{n,s_t,D}$, as the Figure 4.2 shows, during the changing of the s_t the $mse_{n,s_t,D}$ will linearly reduce, so we can conclude that in this 270 days the training set is generated from longer time span s_t can train the machine learning models to get the better predicting performance. So in the subsequent machine learning training we chose to use the $s_t = 120$ to generate training set and test set.

From Figure 4.3 we can see how the start date d_t affect the RF model's performance. The $mse_{n,s_t,d_{t'}}$ varies with different s_t and $d_{t'}$, we took the average values of $mse_{n,S,d_{t'}}$ which have the different s_t but the same $d_{t'}$ to represent the start dates' influence on machine learning models' predicting performance. As we can see, the mse_{n,S,d_t} didn't get affected by d_t a lot, the variance of the set of mse_{n,S,d_t} is close to zero. So, we conclude that the time series data source can be used for dynamic data mining, the start date d_t is not the main factor to be considered, while the time span s should be carefully selected.

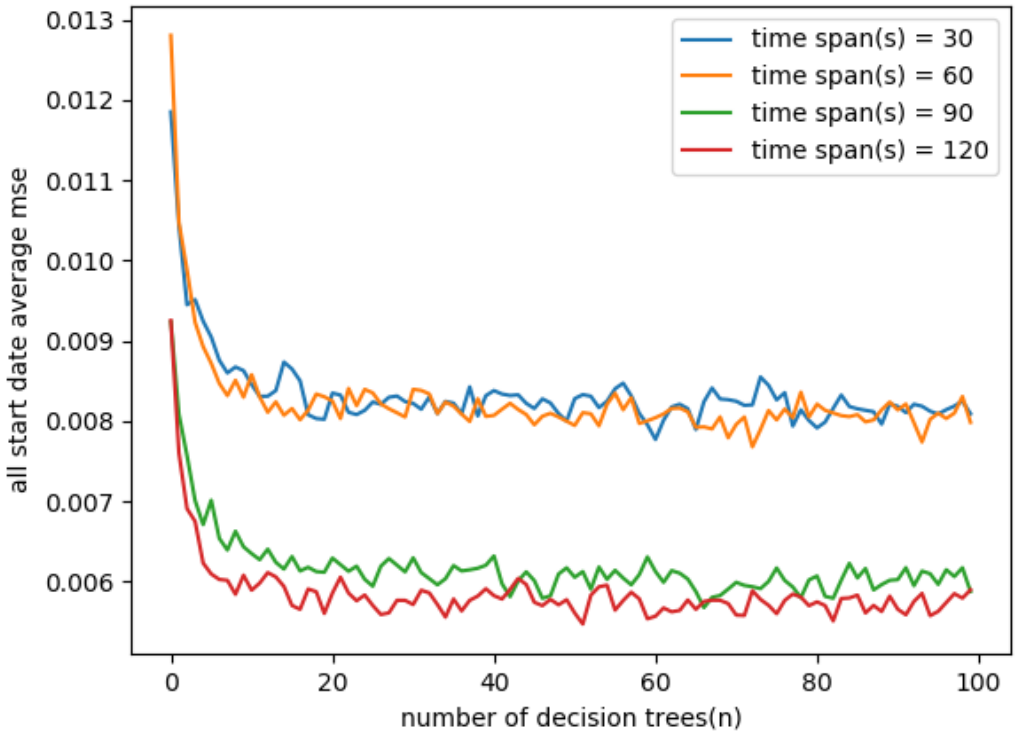

Figure 4.1: The relationship between the number of decision trees(n) and the average $mse_{n,s_t,D}$

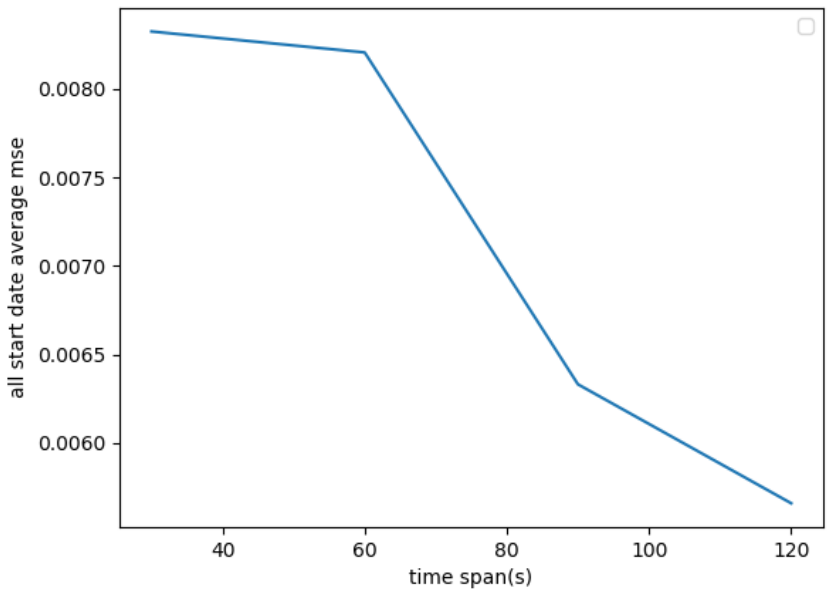

Figure 4.2: The relationship between time span(s_t) and the average $mse_{n,s_t,D}(n = 40)$

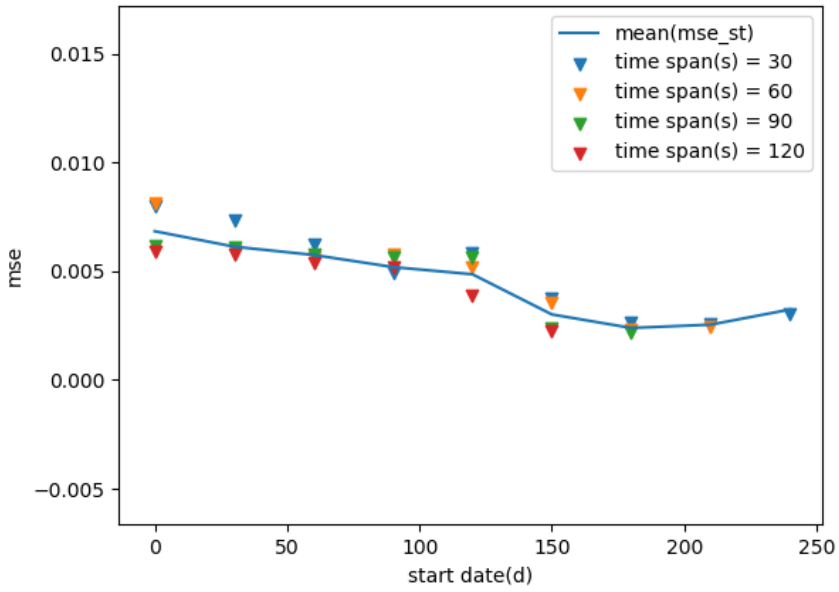

Figure 4.3: The relationship between the duration settings and the $mse_{n,s_t,d_t}(n = 40)$

4.2 Encoding Methods Comparison

In Section 4.1 we showed the affections caused by different time series data based on the features which are encoded by one-hot encoding method. In this section, we will discuss why did we use the one-hot encoding method rather than dummy encoding method to do the features encoding.

As we discussed in Section 4.1, we chose $s_t = 120$ and $d_t = 0$ time period to generate training set and we took $s_t = 120$ with $d_t = 120$ as the time period for generating test set. We still used RF models' average $mse_n(n = \{1, ..., 100\})$ to evaluate the encoding methods' affection on RF model's predicting performance.

As the result is shown in Figure 4.4, while the n increases from 1 to 100, we can see that when the n <25 the one-hot encoding dominated the dummy encoding, but when the n becomes big enough, the mse_n of both methods converged to the almost same level.

And as the Table 4.1 shows, for the average of the mse_n $(n = \{1, ..., 100\})$, the one-hot encoding method still performed slightly better than the dummy encoding method, which means encoding the discontinuous features with the continuous numbers will more or less increase the predicting errors, so this is the reason we chose the one-hot encoding method to encode the e-bike features to sparse boolean vectors.

encoding methods	number of decision trees	time span(s_t)	start date(d_t)	mean mse_N
one-hot encoding	1 to 100	120	0 and 120	0.005959
dummy encoding	1 to 100	120	0 and 120	0.006045

Table 4.1: Encoding methods overall comparison

4.3 Features Selection Methods Comparison

The purpose of feature selection is to select relevant features to reduce the machine learning process complexity and get the machine learning models better predicting performance. As we discussed

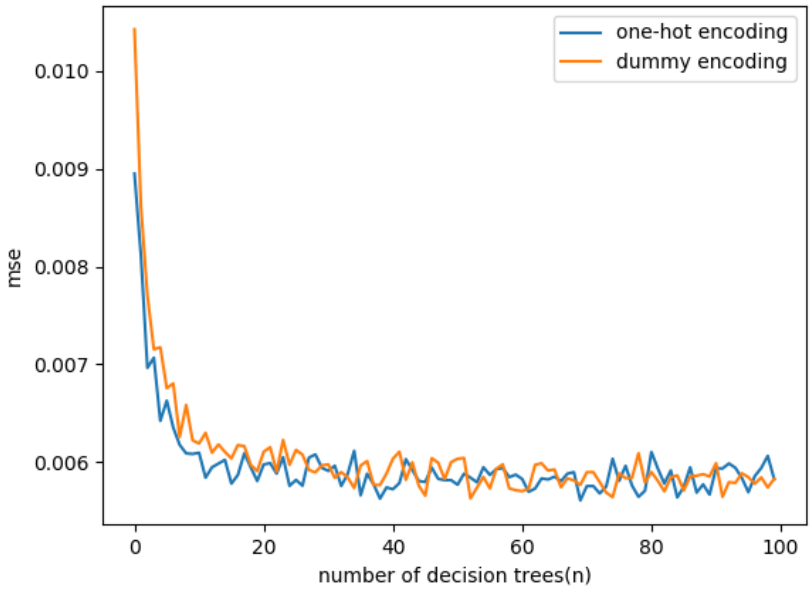

Figure 4.4: The mse_n ($n = \{1, ..., 100\}$) trend of one-hot and dummy encoding

in the last two Sections, we select the time period of $s_t = 120$ and $d_t = 0$ to generate the training set, and use $s_t = 120$ and $d_t = 120$ time period to generate a test set, not that in both training set and test set the feature vectors were encoded by one-hot encoding method. We still use RF models' average mse on the test set to evaluate RF models' general performance with a certain feature selection strategy, the number of trees n setting of the RF models is 40.

The filter method in our experiment is calculating the correlation coefficient η_{f_i} between each feature column a_i and the labels column Y. After we knew all the features' correlation coefficient, we sorted the features with the correlation coefficient ranking which is shown in Figure 4.5, along the features' ranking we iteratively add one more feature to construct the new feature vector set $X^{'}$, and evaluate the RF performance with the new $X^{'}$ in each round, finally we got the $mse_{X'}$ trend as shown in Figure 4.6, from the result we can see while along the features adding process, the RF models' average mse reached minimum when the top 188 features which sorted by the correlation coefficient are selected, but the $mse_{X'}$ does not always decrease, when more features are selected the $mse_{X'}$ even became bigger, which proved that some of the attributes have negative affections for the predicting performance.

The concept of the wrapper method is changing the feature combinations and evaluating RF models' average mse to test out the best-performed features set. Different from the filter method, we don't need to calculate the correlation coefficient between each feature column and the labels column, but it is not possible for us to try all the possible combinations of all the different features, which are too expensive to be tested one by one. But the CMA-ES can help in this occasion, we input a boolean vector with the dimensions of all possible features to the CMA-ES optimization model, the boolean vector represents the features selection status, in the boolean vector "1" means the corresponding feature has been selected, and "0" means the corresponding feature has not been selected. The fitness value of the CMA-ES model is the average $mse_{X'}$ of the RF models which were trained with the feature vectors set $X^{'}$ that selected by a boolean vector. From Figure 4.7 we can see that the average mse of RF models is gradually getting smaller during the CMA-ES optimization, and the minimal average mse reached 0.00504.

The concept of the embedded method is to incorporate the feature selection as part of the machine

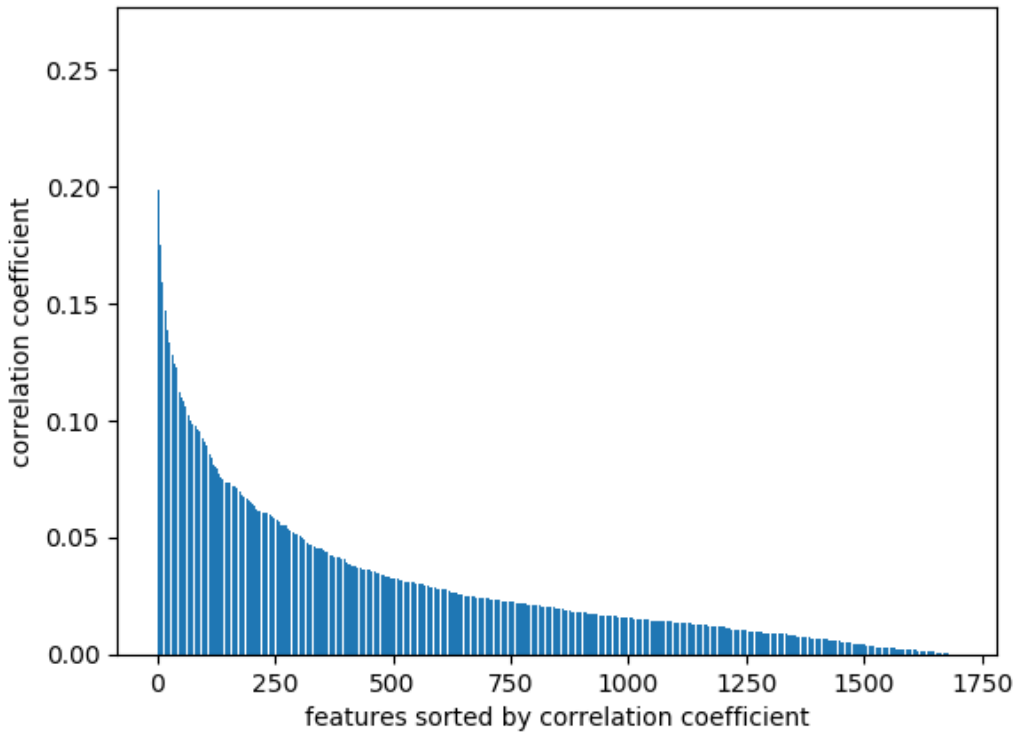

Figure 4.5: The *mse* trend when iteratively add the filter method sorted features

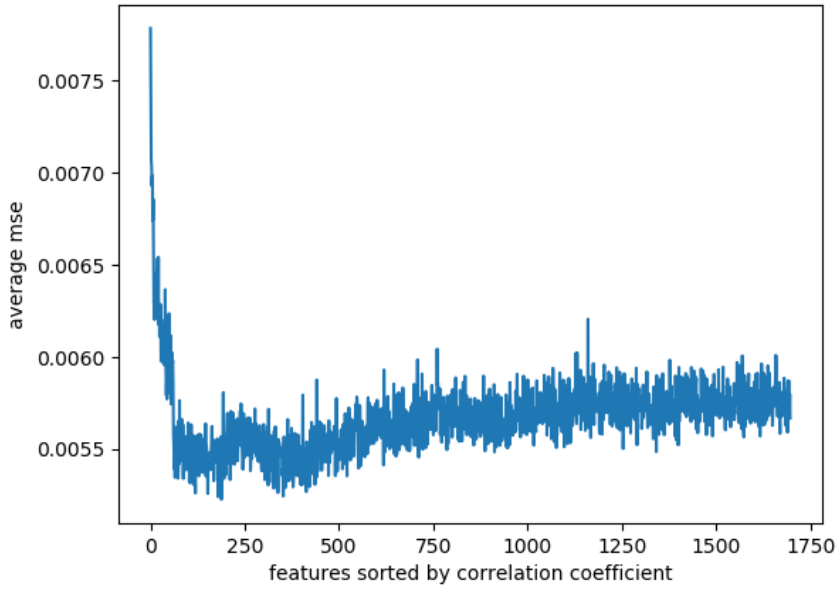

Figure 4.6: The *mse* trend when iteratively add the filter method sorted features

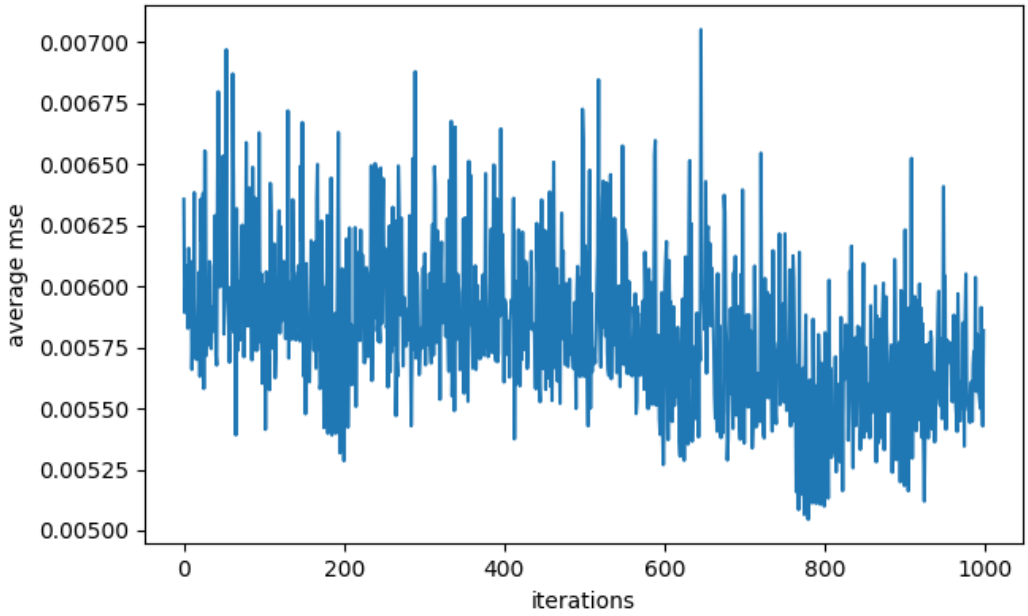

Figure 4.7: The *mse* changing curve along the feature selection process with the CMA-ES and wrapper method

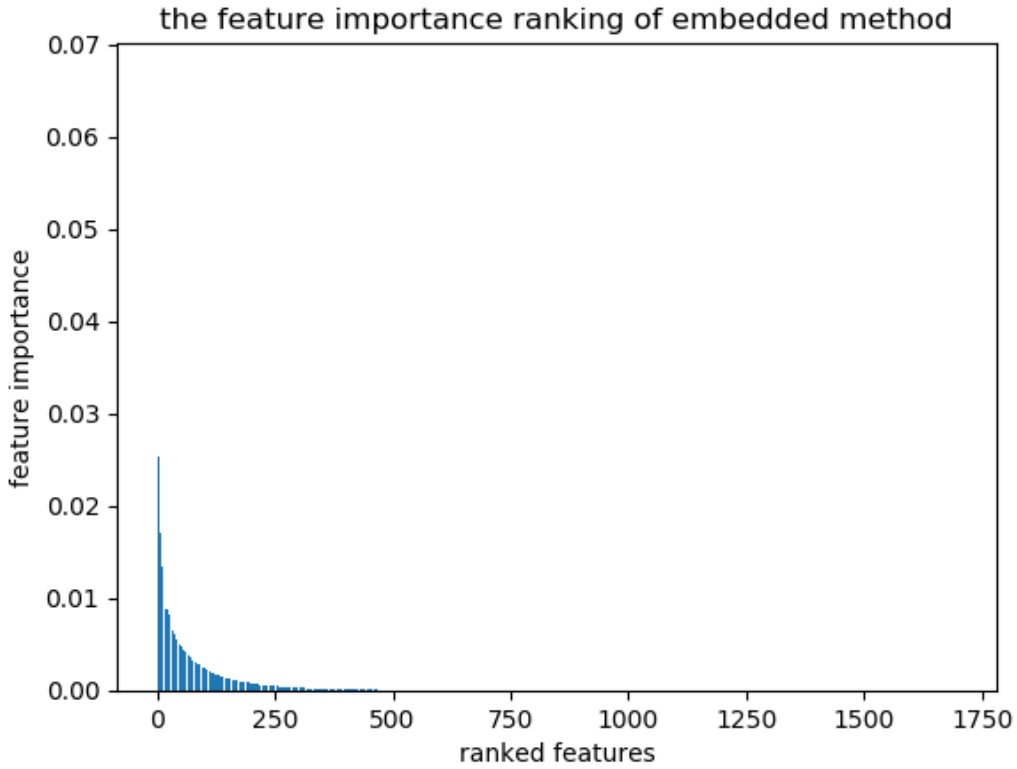

Figure 4.8: The feature importance ranking of RF

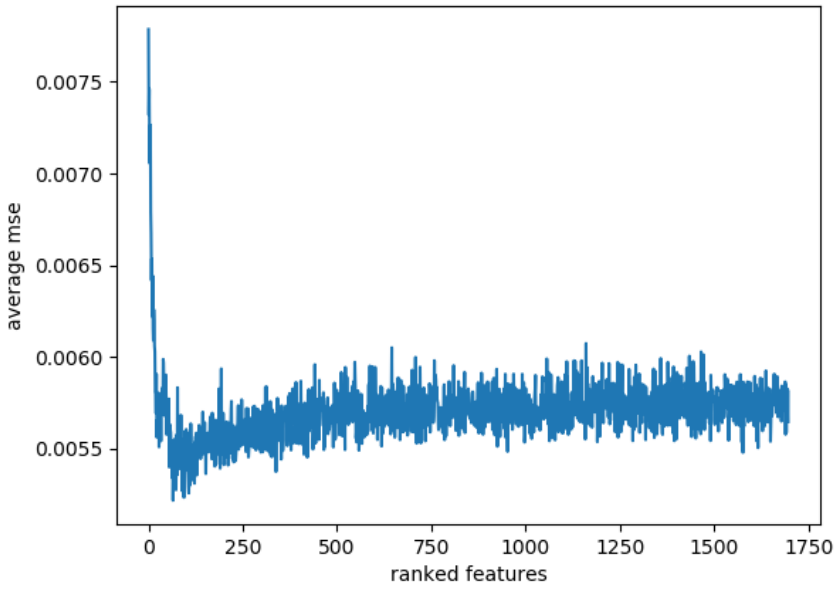

Figure 4.9: The mse trend when iteratively add the embedded method sorted features

learning model training process. In our experiments we used RF model to detect out the features' importance during training, we firstly input the feature vectors which contains every possible features, and trained the RF model, after training we can get the features' importance, as it is shown in Figure 4.8, we ranked the features by the feature importance from high to low, then we based on the feature importance ranking to iteratively add features to build new feature vectors $X^{'}$, and we evaluate the RF performance with the new $X^{'}$ in each round, finally we got the average $mse_{X'}$ trend as shown in Figure 4.9. The minimum average $mse_{X'}$ the embedded method reached is 0.00521.

From Table 4.2 we can know the best features combination for the machine learning model is provided by the wrapper method, but the aim of feature selection is not only about the machine learning models' performance. But since the goal of this project is to support the decision making during e-bike designs, the features which are valuable for the e-bike designers and decision-makers should always be included. Thus, finally from the e-bike designers' integrated selection we selected all the features of 11 attributes for analysis, the selected features can lead the RF models' average mse to 0.00562, although it is slightly worse than the feature combination selected by the other three methods, but it is the most meaningful feature combination to support the e-bike designers decisions, so we chose to use the features which selected by the e-bike designers.

feature selection methods	wrapper	filter	embedded	integrated
minimal mse	0.00504	0.00522	0.00521	0.00562

Table 4.2: Feature Selection Results

4.4 Comparison of the 3 Machine Learning Methods

In the preview sections we compared the feasibility of different time series data sources, feature encoding method, and feature selections, so now we have the proper way to generate the data set to train the e-bike design evaluation models and the e-bike designs optimization model. As we mentioned in Section 3.1, we tried 3 different ways to generate the e-bike designs evaluation model, which are the RF, ANNs, and KNN.

Before comparing the 3 methods on our data set, we should ensure the 3 different models get their best status, which means we should optimize their hyperparameter settings.

The RF method just has one hyperparameter that needs to be optimized which is the number of decision trees n, As Figure 4.11 shows, the RF's mse will converge along the growing of the n, so we will still set the $n = 40$ for the RF e-bike designs evaluation model.

But for the ANNs method, the hyperparameters are more changeable, the number of hidden layers and the number of nodes in each layer should be considered. In theoretical, if there are sufficient hidden nodes in one single hidden layer, the single hidden layer is able to approximate any none linear function, but the ANNs with a huge amount of hidden nodes are very expensive for training. However, the two hidden layers ANNs generally can have more efficient than the single hidden layer ANNs [22]. So we set 2 hidden layers during the initialization of our ANNs, and the number of nodes in each layer is another variable that we should consider. We also based on the CMA-ES to find the optimal layer nodes setting, and the fitness value of the CMA-ES is the average $mse = 0.0041$ on the test sets of the trained ANNs. For the active function of each node, we chose the $relu$ function. As we can see from Figure 4.10, after 3000 times of hyperparameter optimization, the ANNs found the local optimal hyperparameter setting which is 195 nodes in the first hidden layer and 111 nodes in the second hidden layer.

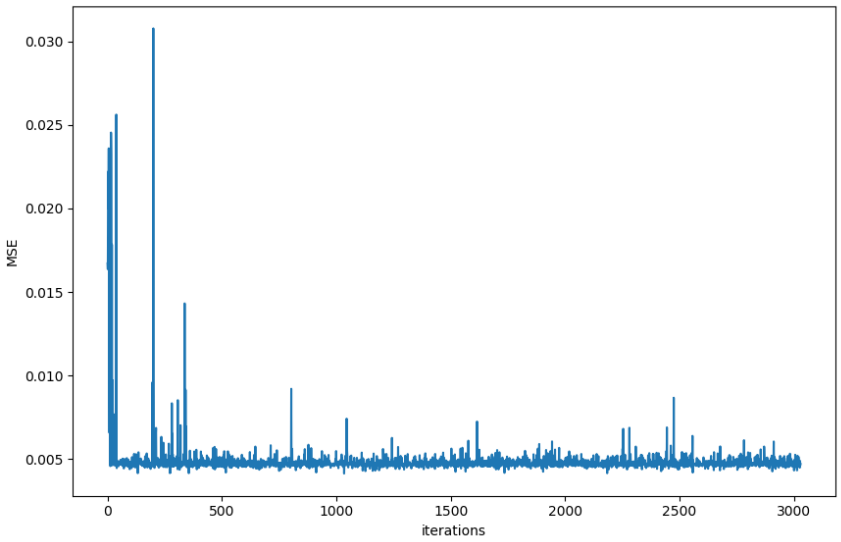

Figure 4.10: The mse trend during the ANNs hyperparameters optimization with CMA-ES

The KNN method has no requirement on training, but we need to find the optimal setting of the number of neighbors k to get the best performance on the KNN predictor. We start at $k = 1$ to the number of all the e-bike instances that exist in the sample set and for each k setting we still use the mse to evaluate the KNN model's performance. As Figure 4.12 shows the mse is increasing during the growing of k, when $k = 1$ the mse got the minimal.

After we get the 3 models' optimal hyperparameters settings, we can train the 3 models with the same training set, and compare their performance on the same test set, as Figure 4.13 shows, the ANNs model got the lowest mse among the 3 models.

And as Figure 4.14 shows the distribution of the predicting values of the 3 machine learning models, in most of the case the ANNs model's predicting results are closer to the real label values. So we decided to use the ANNs model to be the e-bike evaluation model.

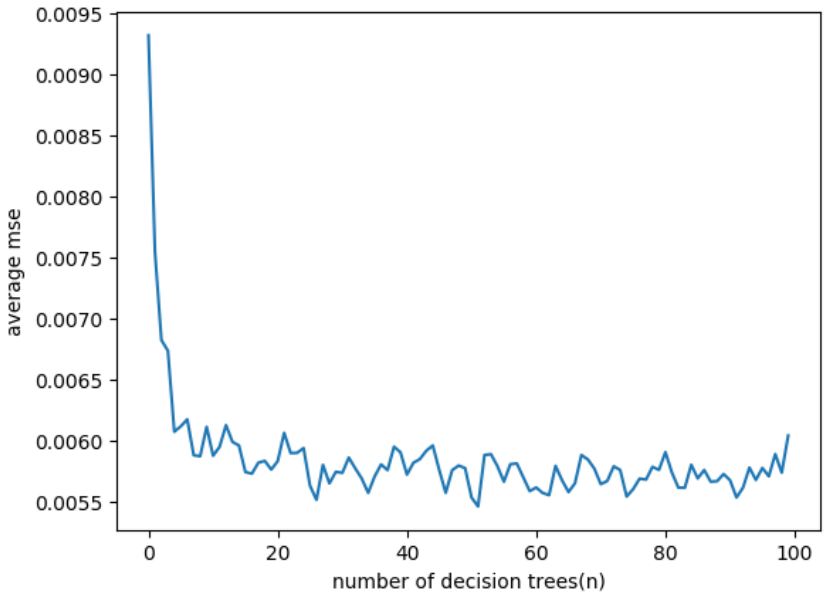

Figure 4.11: The relationship between the decision trees number and the mse_n of RF

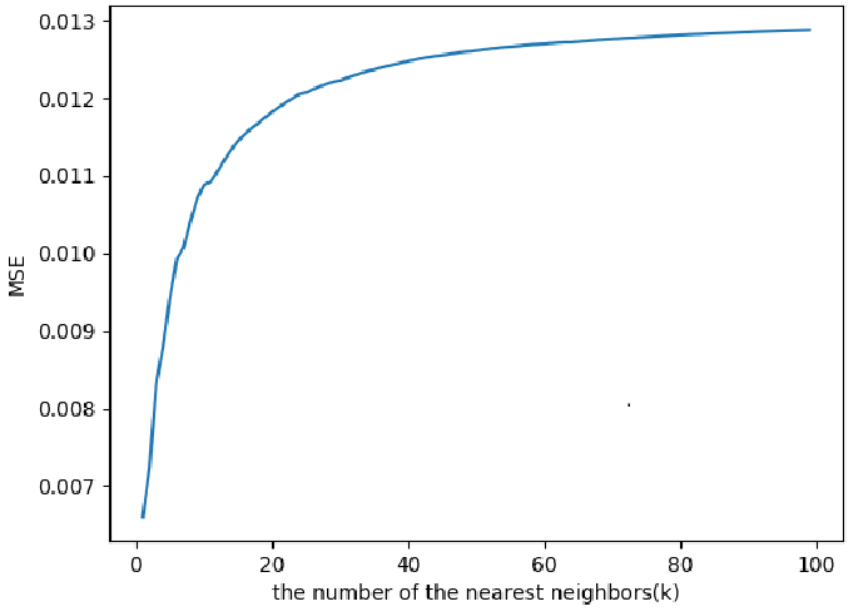

Figure 4.12: The average mse trend of the KNN models during the growth of k

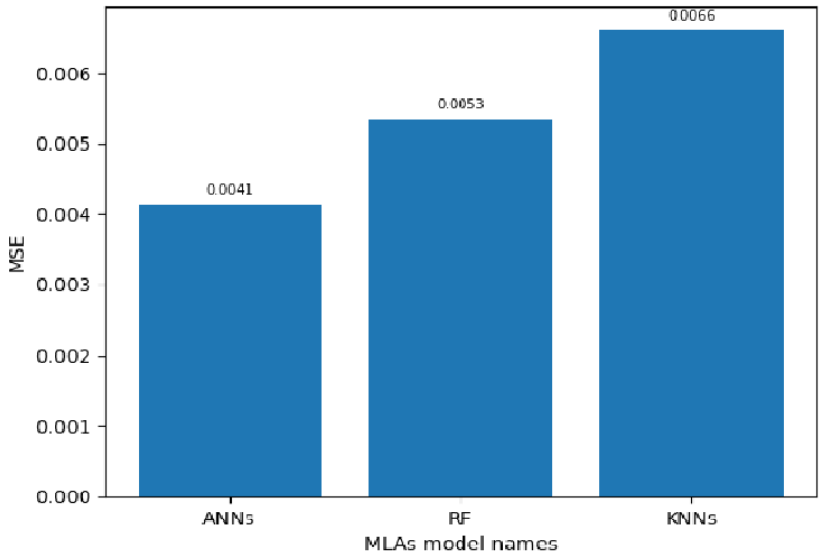

Figure 4.13: The *mse* of the 3 machine learning models with the best settings on the same test set

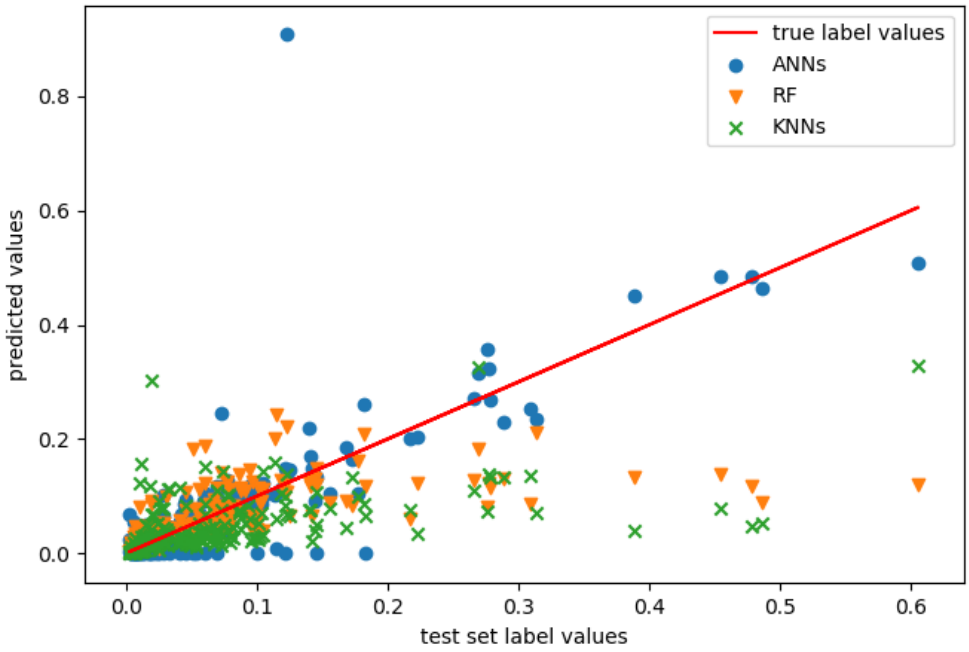

Figure 4.14: The prediction performance of the 3 machine learning models on the same test set

4.5 Optimization Function Performance

As we compared the 3 different MLA bike design evaluating models, the ANNs model has the best performance than the other two, which can provide the most accurate predictions on e-bikes. So we regard the ANNs model as the fitness function for e-bike designs, when we change the feature values of an e-bike's feature vector, we can get different popularity scores of the new designs. We still used the CMA-ES to optimize the e-bike designs.

During the optimization, we should not only focus on the popularity score prediction, but we should also consider if the component combinations are logical, for instance, if the e-bike type is "transport e-bike" then this e-bike should always have a front carrier. So we also created a function that can keep the e-bike designs logical during the CMA-ES optimization process. Figure 4.15 shows an example of the CMA-ES optimization process, the e-bike design is optimized in 50 generations with the recombination and mutation on the e-bike features, and the score has been improved from originally 0.17 to 0.28.

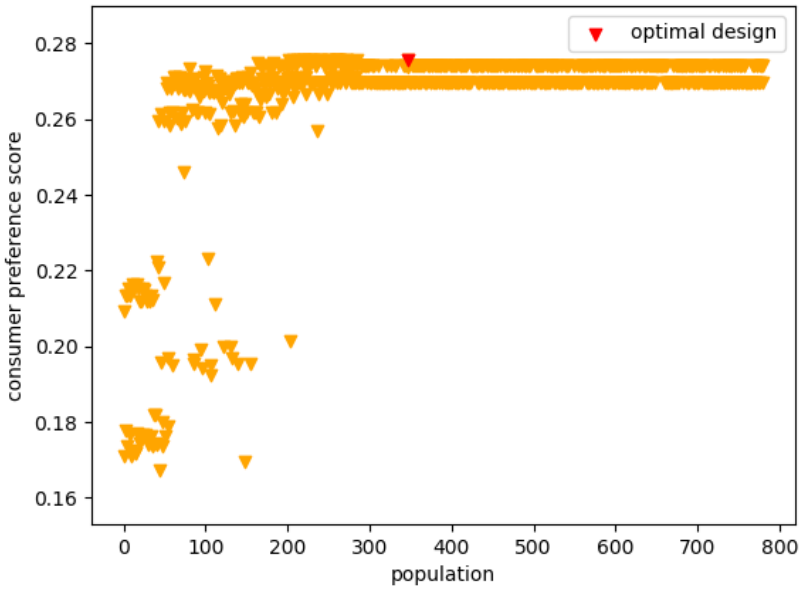

Figure 4.15: An e-bike design optimization sample

Chapter 5

Conclusion

5.1 Summary

In this thesis, we successfully applied the supervised machine learning algorithms, optimization algorithms and data mining methods in the business filed. For choosing the suitable data source, we found out that the exhibition online platform traffic data has good capabilities on dynamic analysis, because the web traffic data analysis is very time-saving and cost-saving, and can show the consumers' initial willingness when choosing products. But the web traffic data still contains some inevitable bias from the product display order, fortunately, the IBG exhibition website has various product display orders, so the bias from display order shouldn't cause a huge effect on product page visits.

For dynamically analyze the e-bike design features' importance, we should figure out how the time duration settings affect our data mining process, from Figure 4.1 and Figure 4.3 we find out the time span s has bigger affections than the start date d, and when the $s = 120$ the RF models get the best predicting performance on the test set, which also proved that the customers' preference is very changeable in a short time duration, when the time duration becomes longer the more stable patterns can be learned. But from Figure 4.3 we can see the start date d does not have huge affections on predicting models performance, thus, as long as we find out the suitable time span s, we can base on the s and any start date d to do patterns learning and machine learning models training.

Since most of the e-bike design features are categorical features, but the machine learning models we used are regression models, so we doubted that the features encoding method will cause huge affections on the machine learning models training, But from Figure 4.4 we can see that when the number of estimators (for RF the estimators are decision trees) is small (n<25), the curve of the one-hot encoding dominated the curve of the dummy encoding, which means the features are encoded by one-hot encoding method can lead better performance in the simple structured predicting models. But when the number of estimators becomes larger, the two encoding methods lost the relationship of dominance, which tells us both of the encoding methods can be used in the well structured predicting models.

The selection of features just has slight affections on the patterns learning. We have tried the 3 most common used feature selection methods to evaluate the RF models‘ performance, as the Table 4.2 shows, the wrapper method found a better features combination than the other two methods, but the wrapper method is really time-consuming, and it is impossible for us to test all the possible feature combinations. The filter method and the embedded method are showing that during adding new features based on the ranking of correlation coefficients or feature importance the performance of RF models will generally become better, but some features even can have negative influence to the predicting models performance, and selecting all the available features is not the optimal choice for machine learning, this phenomenon shows the necessity of feature selection. After all, although the 3 standard feature selection methods provide us the better features combinations for machine learning, but not all the features in these combinations are interesting for e-bike designers, comparing the features selected by designers with the features are selected by the

3 standard feature selection methods, the RF models' predicting performance with the features selected by designers is still acceptable for IBG's commercial needs.

After the feature selection, time series analysis, and encoding method selection we can understand the characteristics of the data that we want to analyze, then we compared and selected the most suitable machine learning method to build up the consumer preference scores predicting model. Although nowadays, there are huge amounts of machine learning methods, there is no method suit for every problem, so it is necessary to try different machine learning strategies to find out the most suitable method. After we did the hyperparameters optimizations on the 3 different machine learning models to get their best predicting status, the ANNs model performed best, although the RF method has the most stable performance and famous with avoiding overfitting, the ANNs method is more potential because of the more hyperparameter choices.

The CMA-ES is the state-of-the-art optimization method, it not only can be used for searching better e-bike designs but also can be used for hyperparameters optimization. With the CMA-ES method, we improved the wrapper feature selection strategy, and we optimized the ANNs models' construction, and we found the conditional best e-bike designs.

In conclusion, this project is a completed machine learning, data mining, and optimization project, with this structure we can achieve automatic feature selection, feature importance extraction, model selection, hyperparameters optimization, and optimal solutions finding. And this structure is easy to extend to the other commercial or academic fields, as long as we determine the label source and feature source, the most suitable models will be trained, and the optimal solutions will be provided.

5.2 Limitations and Future Works

In this thesis, although we tried our best to find out the optimal settings for labels, features, and machine learning methods to simulate the realistic commercial environment to analyze customers' preference, but the hidden bias is always exist, the bias could come from the customers, the web traffic data or the methods we used, so we will discuss the limitations of this project and the future works to reduce bias.

- **The sample set is too small** Currently, in total we just have 1559 e-bikes in our sample set, and the sample set is not balanced, for instance, the number of the e-bikes which have the frame color as black is 700, while the number of e-bikes with red frame color just reaches 39, which cause the black-frame feature has more samples to learn, so the trained machine learning predicting model can learn the black-frame feature's influence on more component combinations, but for the red-frame feature because of the lack of samples, the influence of the red-frame feature is not clear enough. In the future, when more new e-bikes are displayed on the IBG's exhibition website, we can have a bigger sample set, then we can extract out the subset which can keep the balance of each feature, and we can base on the subset to analyze the customers' preference.

- **The rule of components information labeling should be improved** The IBG's bike experts recorded the bike components information in text form, few bikes have the same components but the components are recorded with different descriptions, and there are also some missing data in the component information recording, which will also affect the patterns learning. So in future work, we will improve the components labeling method to avoid the over-classification and the insufficient classification and the missing data in the components recording set.

- **The simulation of the real shopping environment should be improved** The reason why we chose the exhibition website traffic data as the source of the fitness value to evaluate customers' preference for e-bike designs is that the exhibition website can simulate the non-commercial-interference environment for customers to compare e-bikes, but the simulation way can be better. Currently, we just can have the product pages clicks and the bounce rate

to understand the customers' preference, but before they access the product pages they can not know all the features of the e-bikes, such like the "display screen brand" feature they can not figure out before they access the product pages, so some of the features have been ignored in the customers' clicking behavior, but when they are making decision for buying e-bikes those ignored features may be the important factors to affect their decisions. So the design of the exhibition website also needs to be improved, which should enable the customers to consider all the important factors that we interested in before they access the product pages.

- **The e-bikes' pictures should be displayed in a uniform format** Some of the e-bike pictures are displayed in different formats on IBG's exhibition website, which are shown as different display size, this problem will cause an unexpected bias on customers' preference. If the e-bike pictures are displayed in the same format, then we can use the pictures to replace all the e-bikes' appearance features, which can avoid the components information recording mistakes caused by manual classification, and can more accurately understand the customers' preference on the e-bike appearance design.

- **The traditional statistical survey can be used to validate the patterns we learned** In this project we developed the dynamic data mining, predicting models building, optimization structure, which can learn the customers' preference patterns and update the e-bike evaluation and optimization models automatically, but for validating if the patterns we learned really exist in customers' preference we also can conduct a survey to do the cross-validation.

Bibliography

[1] Beatriz Plaza. Google analytics for measuring website performance. *Tourism Management*, 32(3):477–481, 2011.

[2] N Saravanan, A Mahendiran, N Venkata Subramanian, and N Sairam. An implementation of rsa algorithm in google cloud using cloud sql. *Research Journal of Applied Sciences, Engineering and Technology*, 4(19):3574–3579, 2012.

[3] Jordan Tigani and Siddartha Naidu. *Google BigQuery Analytics*. John Wiley & Sons, 2014.

[4] Anders Johannsen, Dirk Hovy, and Anders Søgaard. Cross-lingual syntactic variation over age and gender. In *Proceedings of the nineteenth conference on computational natural language learning*, pages 103–112, 2015.

[5] V. Andersson. Machine learning in logistics: Machine learning algorithms: Data preprocessing and machine learning algorithms. 2017.

[6] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160:3–24, 2007.

[7] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[8] Yvan Saeys, Thomas Abeel, and Yves Van de Peer. Robust feature selection using ensemble feature selection techniques. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 313–325. Springer, 2008.

[9] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[10] Ilya Loshchilov and Frank Hutter. Cma-es for hyperparameter optimization of deep neural networks. *arXiv preprint arXiv:1604.07269*, 2016.

[11] Thomas Back. *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford university press, 1996.

[12] James Bergstra, Dan Yamins, and David D Cox. Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. In *Proceedings of the 12th Python in science conference*, pages 13–20. Citeseer, 2013.

[13] Seymour Sudman and Norman M Bradburn. Asking questions: a practical guide to questionnaire design. 1983.

[14] Ian Brace. *Questionnaire design: How to plan, structure and write survey material for effective market research*. Kogan Page Publishers, 2018.

[15] Liqiong Deng and Marshall Scott Poole. Affect in web interfaces: a study of the impacts of web page visual complexity and order. *Mis Quarterly*, pages 711–730, 2010.

[16] Ann Bowling. Mode of questionnaire administration can have serious effects on data quality. *Journal of public health*, 27(3):281–291, 2005.

[17] Gülser Köksal, İnci Batmaz, and Murat Caner Testik. A review of data mining applications for quality improvement in manufacturing industry. *Expert systems with Applications*, 38(10):13448–13467, 2011.

[18] Indranil Bose and Radha K Mahapatra. Business data mining—a machine learning perspective. *Information & management*, 39(3):211–225, 2001.

[19] Chidanand Apté and Sholom Weiss. Data mining with decision trees and decision rules. *Future generation computer systems*, 13(2-3):197–210, 1997.

[20] Eric Bauer and Ron Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, 36(1-2):105–139, 1999.

[21] David Haussler. *Probably approximately correct learning*. University of California, Santa Cruz, Computer Research Laboratory, 1990.

[22] Guoqiang Zhang, B Eddy Patuwo, and Michael Y Hu. Forecasting with artificial neural networks:: The state of the art. *International journal of forecasting*, 14(1):35–62, 1998.

[23] Janet Kolodner. *Case-based reasoning*. Morgan Kaufmann, 2014.

[24] Sahibsingh A Dudani. The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, (4):325–327, 1976.

[25] Li-Yu Hu, Min-Wei Huang, Shih-Wen Ke, and Chih-Fong Tsai. The distance function effect on k-nearest neighbor classification for medical datasets. *SpringerPlus*, 5(1):1304, 2016.

[26] Nikolaus Hansen. The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016.

[27] S Amir Ghoreishi and Hamid Khaloozadeh. Application of covariance matrix adaptation-evolution strategy to optimal portfolio. *International Journal of Industrial Electronics, Control and Optimization*, 2(2):81–90, 2019.

[28] Micah D Gregory, Zikri Bayraktar, and Douglas H Werner. Fast optimization of electromagnetic design problems using the covariance matrix adaptation evolutionary strategy. *IEEE Transactions on Antennas and Propagation*, 59(4):1275–1285, 2011.

[29] William WS Wei. Time series analysis. In *The Oxford Handbook of Quantitative Methods in Psychology: Vol. 2*. 2006.

[30] V Rodriguez-Galiano, M Sanchez-Castillo, M Chica-Olmo, and M Chica-Rivas. Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. *Ore Geology Reviews*, 71:804–818, 2015.

[31] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[32] Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2):273–324, 1997.