Transportation Research Part C 102 (2019) 153–172

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Exploring the impact of foot-by-foot track geometry on the occurrence of rail defects

Reza Mohammadi[b], Qing He[a,b,*], Faeze Ghofrani[c], Abhishek Pathak[c], Amjad Aref[c]

[a] Key Laboratory of High-speed Railway Engineering in Ministry of Education, School of Civil Engineering, Southwest Jiaotong University, Chengdu, China 610031

[b] Department of Industrial and Systems Engineering, University at Buffalo, The State University of New York, 314 Bell Hall, Buffalo, NY 14260, USA

[c] Department of Civil, Structural and Environmental Engineering, University at Buffalo, The State University of New York, 212 Ketter Hall, Buffalo, NY 14260, USA

ARTICLE INFO

Keywords:
Rail defect prediction
Foot-by-foot track geometry
Extreme gradient boosting
Imbalanced dataset
Partial dependence analysis

ABSTRACT

Predicting rail defects is of great importance for safe railway transportation. Using foot-by-foot track geometry and tonnage data, this paper develops a new machine learning based approach to identify the track geometry parameters that contribute most to the prediction of rail defects occurrences. Taking more than 60 types of track geometry measurements into account, this study develops a Recursive Feature Elimination (RFE) algorithm for feature selection and compares its results with Singular Value Decomposition (SVD). In addition, to capture more knowledge from the geometry data, some additional features, including Track Quality Index (TQI), energy spectral density, and time-trend are extracted. This, in turn, facilitates the learning and predicting process. Moreover, since there exists a very limited number of rail defects, the Adaptive Synthetic Sampling Approach (ADASYN) is applied to overcome the issue of imbalance in the dataset. In terms of machine learning algorithms, the proposed approach employs an extreme gradient boosting (XGBoost) algorithm in which the hyper-parameters are optimized using a Bayesian optimization method. Furthermore, the proposed approach investigates the impact of each track geometry parameter as well as a subset of them on rail defects occurrences with Partial Dependence Analysis (PDA). Finally, our approach is implemented on a six-year dataset with over 60 million track geometry records collected from a 100-mile section of a U.S. Class I railroad to demonstrate its applicability and efficiency.

1. Introduction

According to the Association of American Railroads (AAR), railroads account for more than 40% of intercity freight volume in the United States (Andrade and Teixeira, 2015). Safe, reliable, and efficient transportation of freights is the ultimate goal of this mode of transportation. Railroad have implemented numerous advanced technologies and company-wide safety programs to improve the railroad safety. According to the AAR, from 1980 to 2016, American freight railroad invested more than $660 billion in rail infrastructure and equipment to enhance the safety of the freight rail sector. However, rail defects still cause a remarkable number of derailments which are the most common type of train accidents (Liu et al., 2017, Zhao and Khattak, 2017); in 2016, about 450 rail defect-caused derailments were reported by the Federal Railroad Administration (FRA), which accounted for 16–20% of total

* Corresponding author at: Department of Industrial and Systems Engineering, University at Buffalo, The State University of New York, 314 Bell Hall, Buffalo, NY 14260, USA.
E-mail address: qinghe@buffalo.edu (Q. He).

https://doi.org/10.1016/j.trc.2019.03.004
Received 29 September 2018; Received in revised form 28 December 2018; Accepted 4 March 2019
Available online 18 March 2019

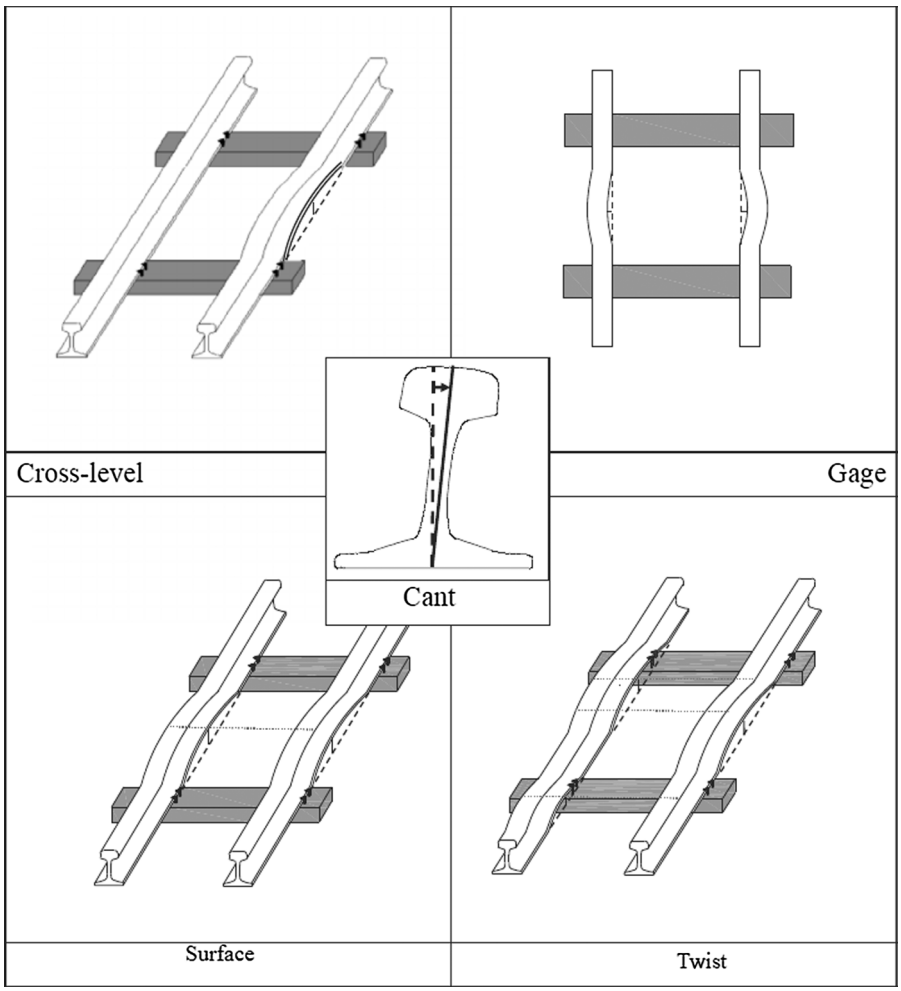

Fig. 1. Track geometry measurements (Sharma et al., 2018).

accidents and resulted in a loss of $103 million.

Rail defects occur as a result of several conditions which originate from the rail manufacturing process, cyclical loading, impact from rolling stock, rail wear, plastic flow, and so on (Cannon et al., 2003). Rail defects are also initiated in rail by fatigue or other failure mechanisms. Fatigue failures include three phases; in the first phase, fatigue crack initiates, then it grows in size, and finally, the rail breaks (Cannon et al., 2003). Although it is very difficult to accurately predict the defect growth, detecting rail defects in the early phases influences the development of them and prolongs the timeframe before the rail breaks. The development of rail defects is identified by three features, including the type of defect, origin, and direction. Based on the identification results, different actions are prescribed to control the defect development, such as rail replacement, applying joint bars, slow orders, and re-inspection.

Past studies have demonstrated that the track geometry irregularities could result in high dynamic wheel/rail loads (Zarembski et al., 2016) and sequentially cause the development of rail fatigue defects and an associated reduction in fatigue rail life. Track geometry describes the geometry of track layouts and track geometry measurements associated with design and maintenance of railroad track (He et al., 2015). Some of these parameters are horizontal and others are vertical. Warp, cross-level, and surface are vertical, whereas alignment and gage are horizontal (Soleimanmeigouni et al., 2018). Fig. 1 describes some of the track geometry measurements. These measurements express the condition of the track; there are some standards that describe the limitation and proper value for them. In other words, these track geometry measurements are regularly inspected and ill-conditioned geometry measurements are considered as track geometry defects. However, literature in rail defects prediction usually neglects the effect of track geometry on the formation process of rail defects. In (Ekberg et al., 2002), an engineering model was proposed to predict the rolling contact fatigue of railway defects. Also, statistical methods have been applied to predict rail defects. A multivariate statistical model was presented to predict the location of rail defects in (Dick et al., 2003). They used rail defects datasets, and not track geometry data. Most of the data-driven approaches use images, not track geometry data, to predict rail defects. A data-driven framework was suggested by (Zhang and Zhou, 2005), to detect the rail surface cracks based on images. Hajizadeh et al., (2016) developed a semi-supervised method to detect the rail defects using image data (Hajizadeh et al., 2016). Based on rail images, a double-layer data-driven approach was proposed to identify the location of the cracks (Zhuang et al., 2018). This approach also used

a feature-based linear iterative crack aggregation method to obtain the boundary of the cracks.

As one can see, very few previous studies explore the inherent relationship between track geometry and rail defects. Recently, investigating the possible relationship between these two types of track defects have attracted a considerable amount of interest, and has been supported by the railroad industry (Lasisi and Attoh-Okine, 2018) (Zarembski et al., 2016). The very basic advantage of analyzing this kind of relation is cost-effective maintenance planning. Track defect behavior was studied by Zarembski et al. (2016) by applying a multivariate regression spline. In fact, track defect data was used, and significant variables were identified to predict the lifespan of rail defects (Zarembski et al., 2016). Building a mathematical function based on those key variables, their model showed a reduction of 30% in rail life (MGT) when track geometry defects are present. However, massive foot-by-foot track geometry data was not taken into consideration in that study.

Besides the aforementioned studies that aimed at improving rail transportation safety, nowadays, an emerging research stream applies machine learning methods to track geometry data to help prevent rail defects and provide a cost-effective maintenance plan. A two-phase method to combine track geometry parameters and decrease the dimension of data was proposed in (Lasisi and Attoh-Okine, 2018). Combining the principal component analysis (PCA) and track quality index (TQI), they studied significant track geometry parameters. Similar research was conducted in (Martey et al., 2017). They also grouped the track geometry data into classes that differ by surface and alignment features. However, both in (Lasisi and Attoh-Okine, 2018) and in (Martey et al., 2017), they only studied 1 mile of rail data and did not consider the impact of MGT.

Motivated by the need to explore the influence of track geometry parameters on the occurrence of rail defects, as well as the fact that this topic is understudied, this paper develops a data-driven approach to investigate the relationship between track geometry parameters and rail defects. More precisely, this approach predicts the rail defects based on the track geometry and tonnage data to identify the significant track geometry parameters that play an undeniable and crucial role in the occurrence of rail defects. This approach is built based on the massive geometry datasets provided by a US Class I railroad. However, without loss of generality, it is applicable to other track geometry and tonnage datasets.

There exist several major challenges when modeling the massive foot-by-foot track geometry data. First, the available track geometry data is massive due to foot-by-foot inspections. Therefore, there is a pressing need to extract valuable knowledge from this raw dataset. The more knowledge is extracted, the easier the prediction task becomes. To address this challenge, feature extraction is performed to obtain significant features from the datasets. Based on both mathematical and technical aspects, some innovative features are extracted, including, but not limited to, energy spectral density, TQI, and time-trend. Second, although there are a large number of features available, they are not of the same importance. Specifically, some of them may not provide valuable information about the defects occurrences. Thus, we use Singular Value Decomposition (SVD) and develop a Recursive Feature Elimination (RFE) to select crucial features, and this also results in dimension reduction. Third, another critical challenge is the issue of imbalance; after merging the track geometry and rail defects data, the dataset become highly imbalanced. That means the track segments with rail defects are extremely outnumbered by the ones without. Although machine learning techniques are well-developed to be applied to the classification problems, classifying imbalanced datasets are still very challenging (Guo et al., 2017). To be more specific, handling an imbalance dataset is challenging for the following reasons:

(1) A minority class, the class with fewer records, may probably be detected as noise, or the opposite is also possible. That is, noises may be treated as a minority class (Beyan and Fisher, 2015).
(2) Standard machine learning classifiers and prediction techniques often focus more on the majority class than they do on the minority class. This results in a suboptimal classification (López et al., 2013).
(3) The most common performance metrics, such as prediction accuracy, are misleading when the dataset is imbalanced. In other words, the majority class is predicted or classified correctly, while the minority class is ignored (Loyola-González et al., 2016).

Moreover, track geometry data includes many features, and issues with imbalanced datasets are doubled when there are more features in the dataset (Branco et al., 2016). To address these issues, Adaptive Synthetic Sampling Approach (ADASYN) is applied to make the training dataset balanced.

As one of the popular machine learning algorithms (Wu et al., 2018) (Ghofrani et al., 2018), gradient boosting algorithms have been applied to railway and transportation applications. In few studies, different versions of gradient boosting are applied to estimate complex functions. (Ahmed and Abdel-Aty, 2013) developed a real-time risk assessment framework for freeways using real-time weather and road geometry. They applied stochastic gradient boosting to estimate a complex nonlinear function in their prediction framework. Gradient boosting is also applied to study non-linear effects of the built environment on driving distance (Ding et al., 2018). In (Ding et al., 2016a,b), a gradient boosting logit is applied to traffic and signal data to handle different types of predictor variables and investigate the behavior. (Zhang and Haghani, 2015) applied a gradient boosting regression tree to model freeway travel time to improve the prediction accuracy and model interpretability. Other studies employed gradient boosting to analyze the influential factors in different transportation-related problems. Using crash data, one study analyzed the relationship between crash severities and a set of heterogeneous risk factors by applying a gradient boosting algorithm (Zheng et al., 2018). In (Ding et al., 2016a,b), the relationship between short-term subway ridership and its influential factors was investigated for short-term subway ridership prediction. Ma et al. (2017) studied the relationship between incident clearance time and some explanatory variables. Using gradient boosting, they found that the greatest contributor is incident response time. While aforementioned work implemented basic gradient boosting, extreme gradient boosting algorithm (XGBoost), as a recent modification of gradient boosting, has demonstrated very successful results than other boosting algorithms (Chen and Guestrin, 2016). Thus, in this paper, we applied XGBoost for both predicting the rail defects using track geometry measurements and identifying the influential factors in rail defects occurrence. Unlike

other gradient boosting methods, it selects trees using a regularized objective function that results in simple but effective models. In addition, feature interaction constraints are defined to incorporate the domain-specific knowledge into the XGBoost and avoid unnecessary interaction of track geometry parameters even if they contribute to the model accuracy.

XGBoost has several hyper-parameters that play a crucial role in the performance of the algorithm and tuning them is very challenging. To improve the accuracy of the algorithm, the hyper-parameters are tuned by a Bayesian optimization method. More details about the discussed components are provided in Section 3. Moreover, we apply Partial Dependence Analysis (PDA) to discover the impact of single or multiple of track geometry parameters on rail defect occurrences based on the XGBoost prediction model, (Friedman and Popescu, 2008). Finally, the proposed approach is applied to the dataset of a 100-mile section to demonstrate the applicability and efficiency.

The key contributions of this paper lie as follows:

(1) The proposed methodology builds a rail defect prediction framework based on foot-by-foot track geometry data. New features such as energy, TQI and time trend are defined to capture more knowledge from the data.
(2) Given massive foot-by-foot track geometry data, an RFE algorithm, which takes feature correlation into account, is developed to identify the most contributing features and reduce the dimension of the dataset.
(3) This study incorporates Bayesian optimization to tune the hyper-parameters of XGBoost. Furthermore, feature interaction constraints are applied to handle the interaction between track geometry parameters in decision trees.

The rest of the paper is organized as follows. Section 2 describes the data. This is followed by the description of the methodology in Section 3. The authors demonstrate, via an extensive analysis in Section 4, the effectiveness of the proposed approach and highlight the benefits of using the proposed approach in railway. Finally, conclusions are drawn in Section 5.

2. Data description

The datasets were provided by a US Class I railroad, which is one of the largest freight railroads in North America. Track geometry measurement dataset includes foot-by-foot inspection records for 100 miles from 2012 to 2017. The inspections were not performed in a constant time interval, and there are a different number of annual inspections for every foot, which varies from 1 to 4. In addition, for every foot inspection, there are 61 measures such as gage, twist, cross-level, surface, and so on. In fact, it is a huge dataset with more than 60 million records. Furthermore, two separate datasets for rail defects and annual tonnage were collected from 2012 to 2017 for the same 100 miles. Annual tonnage is the total weight of goods and cars that passes each segment is reported in Million Gross Ton (MGT).

Our research attempts to perform a segment-based prediction rather than a point-based one. Hence, we divide a track into 0.1-mile segments and provide data analysis and prediction for each segment. Based on the availability of data, different section's length could be applied.

After data cleaning and preprocessing, the datasets are merged based on the unique attributes such as milepost, inspection date, and track number. It turns out that the merged dataset is heavily imbalanced, and the proportion of segments which have rail defects records is about 0.08.

As mentioned before, the dataset includes 61 features. Initial feature selection is performed based on the expert's knowledge and previous studies (Sharma et al., 2018). Table 1 shows the list and description of some of these features.

2.1. Exploratory data analysis

This section analyzes foot-by-foot raw track geometry data to understand the properties of the data both before and after extracting the features and dimension reduction. Fig. 2 depicts some of the boxplots for different track geometry parameters. These boxplots show the variation of each of the parameters. In addition, it is interesting to see that some of them include outliers (i.e.,

Table 1
The description of track geometry parameters.

Geometry measurement	Description
Gage	Gage is the distance between right and left rail measured 5/8″ below the railhead.
Xlevel	Cross-level is the difference in the elevation between the top surfaces of the rails at a single point in a tangent track segment.
CRV	Curvature, the degree to which a curve deviates from a straight line
Warp	Warp is the difference between two cross-level or elevation measurements up to a certain distance apart.
Twist	Twist is the difference between two cross-level measurements a certain distance apart.
Surface	Uniformity of rail surface measured in short distances along the tread of the rails. Rail surface is measured over a 62-foot chord, the same chord length as the FRA specification.
DIP	DIP is the largest change in elevation of the center line of the track within a certain moving window distance. Dip may represent either a depression or a hump in the track and approximates the profile of the center line of the track.
CANT	Rail cant (angle) measures the amount of vertical deviation between two flat rails from their designed value
Align	Align or straightness is the projection of the track geometry of each track centerline onto the horizontal plane.
Unbalance	Unbalance is the difference between actual and equilibrium cant.

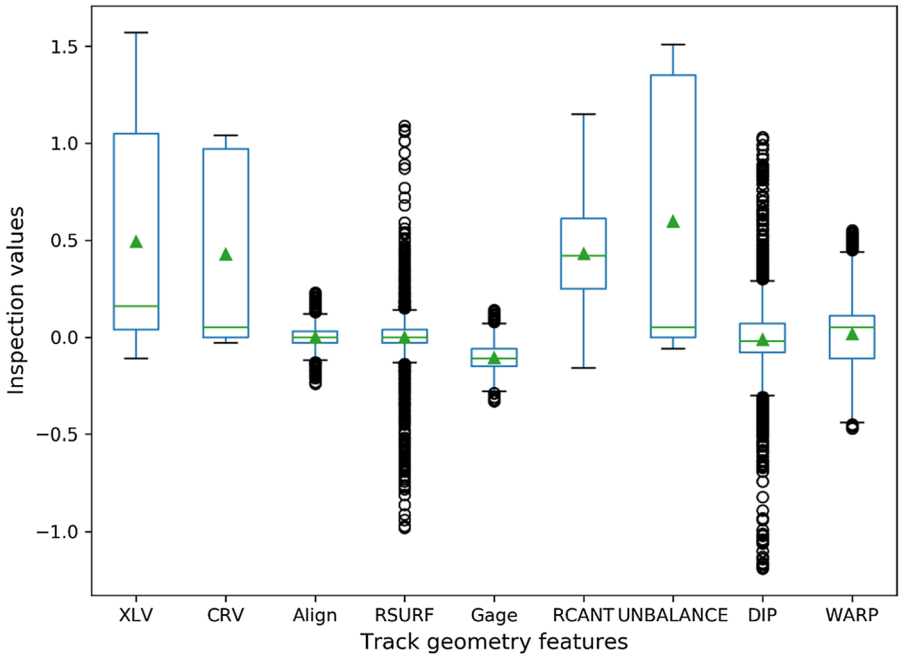

Fig. 2. Box plots for some of the track geometry measurements.

gage) that should be handled carefully. In terms of the range of values, DIP has the widest range; however, the concentration of that is around zero. CANT and CRV seem to have more uniform values than others do. The boxplot provides insight for data preprocessing. It shows the distribution of the variables and if there are outliers or not. Green triangular in the plot also shows mean values for each variable. Initially, these plots can be useful for imputation of missing values or outliers. Based on this plot, for Align, RSURF, CANT, and Gage mean value is a good candidate for replacing the missing values. However, for some others, such as CRV, since the difference between the mean and median is high, the mean is not a proper value to replace missing values.

As mentioned before, some of the parameters are vertical and some are horizontal. Thus, correlation analysis is conducted to study the strength of the correlation between different parameters. This will uncover how different parameters positively or negatively impact each other. Figs. 3 and 4 show the correlation plots for 2016 and 2017 for 100 miles, respectively. These plots indicate some reasonable relationships which can be used to select geometry features to be used in prediction. If two features have a high correlation, one of them should be chosen in the prediction model. Furthermore, comparing the correlation plots for two years demonstrates that the correlated parameters are the same. In both plots, three pairs, including DIP and surface, twist and warp, and Xlevel and CRV, are more correlated than the others. The results of correlation plots are used in both feature selection algorithm and XGBoost. Highly correlated features are prevented from interacting with each other. This is discussed in details in Sections 3.2.2 and 3.4.

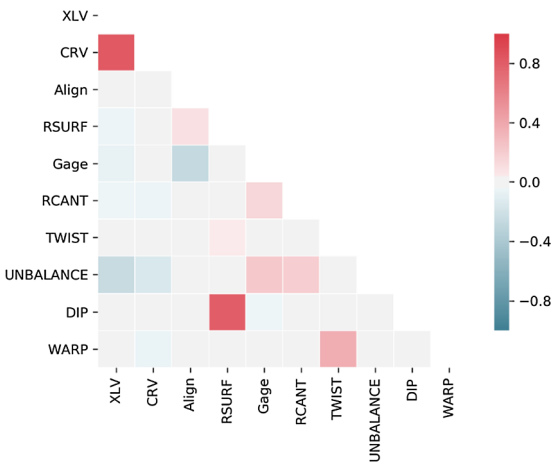

Fig. 3. Correlation plot for track geometry measurements for 100 miles in 2017.

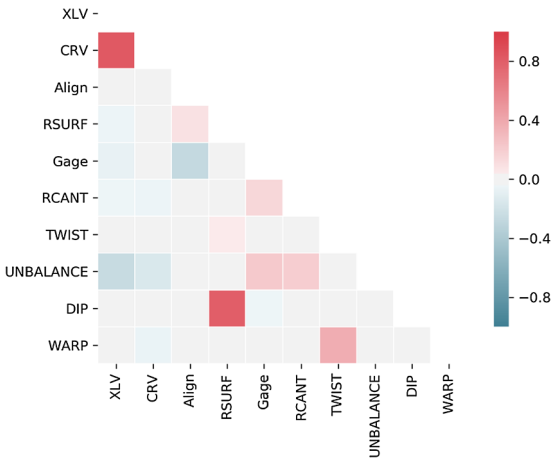

Fig. 4. Correlation plot for track geometry measurements for 100 miles in 2016.

The time interval between rail defects occurrence and the last inspection date also provides significant practical insights about the data. Also, such interval indicates the prediction horizon in this study. Fig. 5 shows the frequency of these time intervals in terms of the number of days. For practical inspection schedules, the time intervals are not equal. Time intervals vary between 7 and 60 days. However, mostly it is distributed between 8 days to 35 days. These time intervals indicate that predicting rail defects even one inspection before can allow railroads to prevent the rail defects by performing remedy or preventive actions.

3. Methodology

This section is devoted to the explanation of the proposed approach. It begins by explaining the components in details, and also, a schematic flowchart that presents the approach is provided in Fig. 6. This figure depicts different components of the approach and shows how they relate to each other. After data preprocessing, this study extracts the features and merges the datasets. However, the final dataset is extremely imbalanced. To resolve the imbalance issue, this paper applies ADASYN which is an adaptive over-sampling method.

Moreover, some of these features may not add much information to the prediction model. Moreover, a large number of features exacerbate the imbalance issue. Thus, dimension reduction is essential to overcome this issue. While dimension reduction resolves some issues, it can also cause some issues if it is not conducted cautiously. For this reason, in the proposed methodology, based on two identical datasets, dimension reduction is conducted in two different ways using SVD and RFE. SVD reduces the dimension based on the variance of the data, whereas RFE tries to reduce the dimension by selecting the features according to their contribution to the accuracy. Therefore, applying SVD and RFE results in two different datasets, and by implementing the same algorithm to these datasets, one can figure out if RFE outperforms SVD.

Once the dataset is prepared, we apply different machine learning algorithms and, finally, propose XGBoost which is a gradient boosting algorithm and recently has shown successful results. Further, we tune hyper-parameters of XGBoost using a Bayesian optimization method. To obtain practical results, PDA investigates the interaction behavior of each individual and a subset of features in rail defect occurrence and identifies the critical range of values for them.

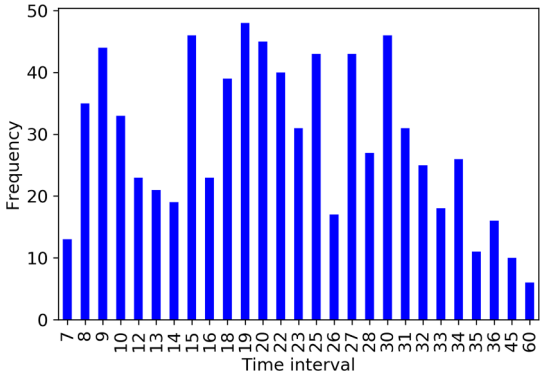

Fig. 5. Time interval between rail defects occurrence time and the last inspection time.

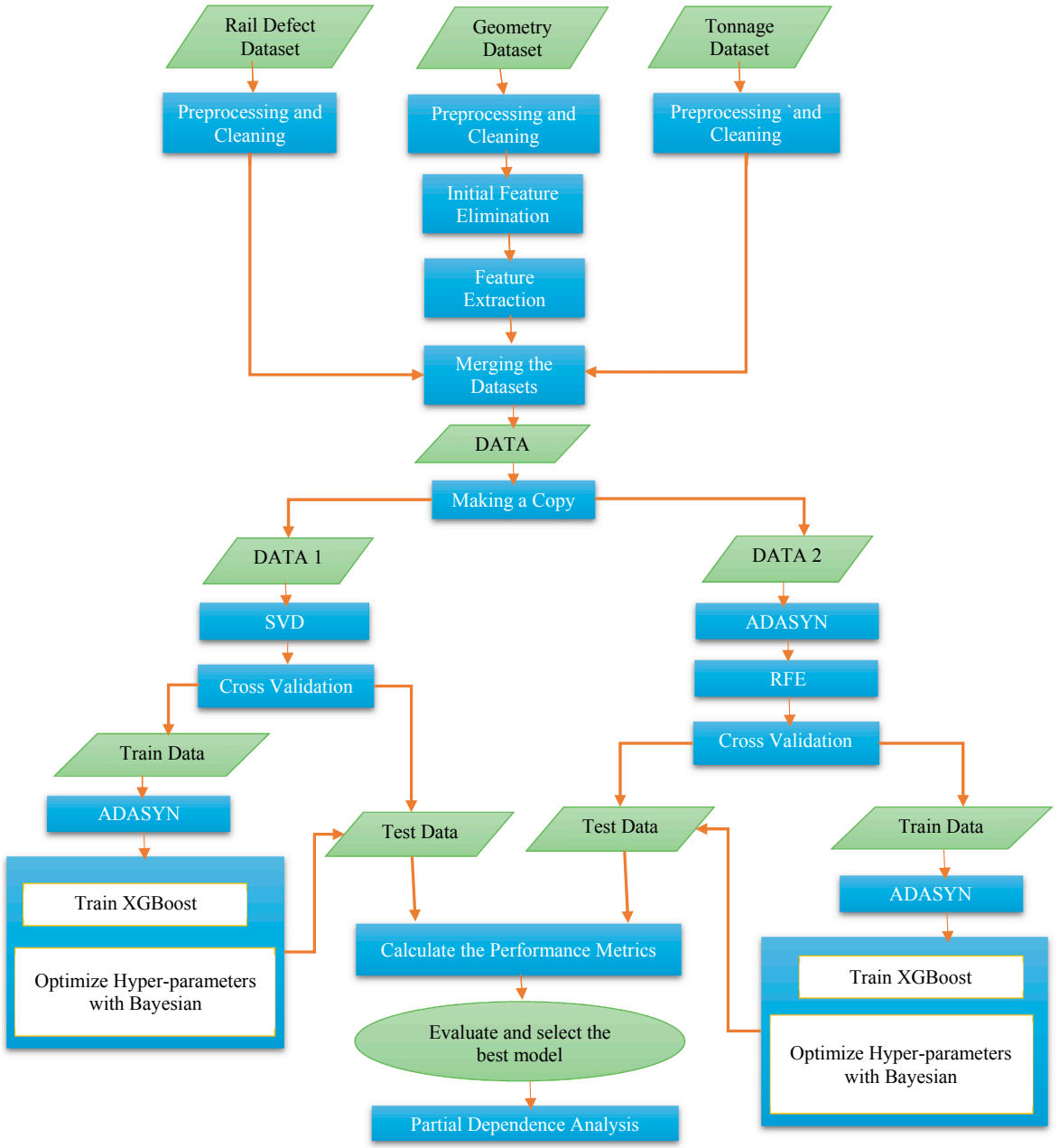

Fig. 6. Flowchart of the proposed approach.

3.1. Feature extraction

Feature extraction is a process of deriving new features such as TQI and the time-trend from the original features (Pereira, 2010). In fact, these features are representative of the original data, which are extracted to increase classifier accuracy or efficiency and reduce dimensions of data (Lee and Landgrebe, 1993). It should be noted that feature extraction and feature selection are different tasks. Feature selection provides a subset of original features, while feature extraction uses different methods to generate new features (Motoda and Liu, 2002). There is no general rule for extracting features, and it depends on the context and the nature of the data.

3.1.1. Energy spectral density

Fourier transform is applied to explain the signal's energy distribution over a range of frequencies by converting the time domain into the frequency domain. The output of this transformation is a complex number which includes the signal's magnitude and phase in each frequency. The magnitude represents the energy (strength) of the signal. We use Fourier transform to extract a feature,

namely energy spectral density. While the time-rend feature, which is introduced in Section 3.1.3, captures the time domain characteristics of the data, the energy spectral density is introduced to investigate the frequency domain. For each segment, the energy spectral density is the sum of the magnitude of the sub-segments.

Given data $x_1, x_2, \cdots, x_n$. Discreet Fourier Transformation (DFT) is defined to be

$$d(w_j) = n^{-\frac{1}{2}} \sum_{t=1}^{n} x_t e_t^{-2\pi i w j t} \tag{1}$$

for $j = 0, 1, \cdots, n-1$,where the frequencies $w_j = j/n$ are called the Fourier or Fundamental frequencies (Hamilton, 1994). If n is a highly composite integer (i.e. it has many factors), the DFT can be computed by the Fast Fourier transform (FFT) introduced in (Cooley and Tukey, 1965). However, different packages and programing languages may use a slightly different formula to calculate the FFT. Considering X_T as a single realization of a stochastic process in the time domain, a Fourier transform of that is as follows:

$$\widehat{X_T}(f) = \int_{-\frac{T}{2}}^{\frac{T}{2}} X(t) e^{-2\pi i f t} dt \tag{2}$$

The amplitude spectrum is the modulus of $\widehat{X_T}$ and the energy spectral density ofX_T is the squared amplitude spectrum which is$|\widehat{X}_T|$. In addition, power spectral density is derived from the following equation:

$$\lim_{T \to \infty} T^{-1} \, |\widehat{X_T}(f)|^2 \tag{3}$$

The energy spectral density feature for each segment is the sum of energy spectral density of sub-segments. For example, in foot-by-foot data for a segment with a length of 0.1 mile, the energy spectral density of each 0.1 mile is the sum of the energy spectral density of 528 feet.

3.1.2. Track quality index

TQI is a numerical measure derived from the track geometry measurements to quantify the quality of the track condition. TQI has been widely used in track deterioration studies (El-Sibaie and Zhang, 2004, Sharma et al., 2018) and has shown successful results.

There are different types of TQIs used both domestically and internationally. In this paper, FRA length-based TQI for each geometry track parameter is calculated and used as a feature. In order to calculate the TQI, each section is divided into smaller subsections. Then, for each section and each track geometry parameter, TQI is calculated using the following equation (Lee, 2005):

$$\mathrm{TQI} = \left(\frac{\mathrm{L_s}}{\mathrm{L_o}} - 1 \right) \times 10^6 \tag{4}$$

where the value of L_o is fixed and equals the length of the track segment. L_s is the traced space curve length, and it is calculated by summing the distances between any two points within the track segment. Also, L_s is calculated using the equation below:

$$L_s = \sum_{i=1}^{n} \sqrt{(\Delta y_i^2 + \Delta x_i^2)} \tag{5}$$

where i is a sequential number, $\Delta y_i$is the difference between two adjacent measurements (ft.), and Δx_i is sampling interval along the track.

In this paper, once the TQI is calculated for each track geometry parameter, it is normalized using the 95th percentiles.

Furthermore, standard deviation and maximum values of raw track geometry parameters are the other features that are extracted from the data. The reason why standard deviation is selected as a feature is that it is included in the TQI formula used in some other countries (e.g. China). Hence, including standard deviation as a feature will help to capture another quantified index of track condition.

3.1.3. Time-trend feature

Another key feature that should be considered is the variations of the extracted features (e.g. energy spectral density, TQI, max, and standard deviation) over time. Our hypothesis is that the way that the feature change over time may impact rail defects growth. For every milepost (0.1 mile) and every track number in six-month intervals, a linear regression model is fitted to capture the variation and time-trend of each feature. In order to be able to use this linear regression model as a feature, we employ the coefficient of the independent variable in each linear regression model as a feature called the time-trend feature. These coefficients incorporate the trending of the extracted features into the model, which results in better learning by the algorithm.

3.2. Dimension reduction

3.2.1. Single value decomposition

Recently, dimension reduction algorithms have gained much attention. Dimension reduction is the act of transforming data into a lower dimensional space and keeping informative variables such that the variability of the dataset is kept within a certain level. Principal component analysis (PCA) (Celik, 2009) and SVD (Cong et al., 2013) are the two most popular dimension reduction methods.

If A is a $M \times N$ matrix of rank r, the SVD of matrix A is the factorization of A into the product of three matrices $A = UDV^T$ where U is a matrix of$M \times M$, D is a matrix of $M \times N$, and V^T(the transposition of V) is a matrix of$N \times N$. The columns of U and V are orthonormal and are called left singular and right singular vectors, respectively. The matrix D is diagonal with positive real entries.

According to the SVD, the closes rank-l matrix to A is

$$A^l = \sum_{k=1}^{l} u_k d_k v_k^T \tag{6}$$

In other words, A^l minimizes the sum of the square of the difference of the elements of A and A^l as follows:

$$\sum_{ij} |x_{ij} - x_{ij}^l|^2 \tag{7}$$

Track geometry data includes lots of parameters, and not all of them provide a significant amount of information to increase the accuracy of the model. Ignoring these parameters without analyzing the importance of them would cause a significant data loss. Thus, we employ SVD to reduce the dimension of the data. That is, SVD attempts to retain maximum variance of the data with the fewest possible number of uncorrelated parameters. Since track geometry parameters are of a different range of values, applying SVD will lead to large biasing for parameters with high variance; to avoid this, the data is scaled and prepared for applying SVD. Specifically, SVD builds a new track geometry data which is approximated using selected uncorrelated features. Finally, the approximated track geometry data is more prepared and tractable for machine learning algorithms to process.

3.2.2. Recursive feature elimination (RFE) algorithm

Feature selection is the process of selecting a subset of k features from the set of original ones to optimize the performance of classification with respect to a certain criterion (Cai et al., 2018). In addition, feature selection is also another way to reduce the dimension of data. One of the most prevalent methods for feature selection is selecting the features based on the importance criteria or feature rank that is given by the machine learning algorithms such as XGBoost. The main idea behind the importance measure is that the more often a feature is selected in the branching point of a tree, the more important it is for accurate prediction. Another criterion that can be considered in feature selection is the correlation between the features. Since highly correlated features do not add much information to the dataset one of them can be removed. Considering correlations in the feature selection algorithms results in robust feature selection (Sageder et al., 2016).

In this paper, RFE is developed based on the feature importance provided by a machine learning algorithm and correlation values. To be more specific, RFE selects the features not only based on the feature importance but also the algorithm's accuracy and the correlation values. It begins with all of the features (S) to train the model. Then, based on the feature importance, the algorithm selects α% of the features (S') that have the lowest feature importance. InS', it drops the features that their correlations with all of the remaining feature in ($S - S'$) are above the correlation lower bound(LT_c),the dropped set is called(S''). This guarantees that features with a certain value of correlation are retained to train the model. Moreover, to avoid the highly correlated features in the remaining set ($S - S''$), features with a correlation value of greater than UT_c, are not allowed to interact with each other. That is, they are not selected in the same decision tree. In fact, instead of removing highly correlated features, the algorithm keeps them but prevents them from interacting with each other. This results in robust feature selection since it does not remove the feature because of a high correlation value.

Finally, the algorithm trains the model with the features in the set of($S - S''$), and avoiding the interaction between highly correlated features. The performance measures of the two successive iterations are compared, and if the difference between the performance measures exceeds a certain boundary, the current set of the features is selected as the best set of the features. In other words, the algorithm stops when there is a significant drop in the performance measure or the maximum iteration is reached. RFE algorithm is summarized in Fig. 7. Calibrations of the parameters are completely problem-dependent. In RFE, α depends on both the number of features and importance rates. If there is a number of features that have significantly lower importance rate, then the value of α could be defined accordingly to remove them. Otherwise, starting from lower values of α ensures no important features will be excluded. β is a type of performance measure; it could be any performance measure such as prediction accuracy, precision, recall, and so on. Also, correlation boundaries UT_c and LT_c are very problem-dependent and a correlation plot that shows the range of correlation will be helpful in determining those values.

As mentioned before, different decision tree based algorithms provide the features' importance as an output. Here, since a decision tree based XGBoost algorithm is applied to classify the binary variable, it is also applied to calculate the features' importance. However, other classification algorithms can be used instead of XGBoost, given the assumption that selected features should not vary considerably. In addition, regarding the algorithm and the context of the problem, different performance measures can be taken into account.

3.3. Imbalanced data handling

The strategies for handling imbalanced data fall into two main categories: preprocessing techniques and cost-sensitive learning (Guo et al., 2017). The preprocessing techniques include resampling techniques and feature selection and extraction. The main idea behind the cost-sensitive approaches is considering higher costs for the misclassifications of minority samples. These approaches are less popular than the preprocessing ones. Preprocessing techniques are briefly explained in the following subsection.

Pseudocode of the RFE for feature selection

Initialization: Set the parameters
α: percentage of weak features
δ: difference parameter as a stopping criteria
I: maximum number of iterations
UT_c: Correlation upper bound
LT_c: Correlation lower bound

Calculate the correlation for each pair of the features
Train the model using all of the features S
Calculate the performance measure (e.g. accuracy) β_0 and the importance of the features
for $i \epsilon I$
  $S_0 = S$
  Select α% of the features (S'_i) that have the lowest importance, (the remaining are S_i)
  for $j \epsilon S'_i$
    if Correlation $\left(f_j \epsilon S'_i, f \epsilon (S_i - S'_i)\right) < LT_c$
      return f_j to S_i
      $D_j = S'_i - f_j$
    End
  $D_j = D_{j-1} - f_j$
  End
 $S_i = S_{i-1} - D_j$
 for $k \epsilon S_i$
    IS ={}
    if Correlation $(f_k \epsilon S_i,\ f'_k \epsilon S_i) > UT_c\ , f_k \neq f'_k$
      $IS = IS + (f_k, f'_k)$
    End
  End
  Train the model avoiding the interaction for each IS set
  Evaluate the accuracy β_{i+1}
 if $\beta_i - \beta_{i+1} > \delta$
   return S_i associated with the best β_i as the selected feature set
 else
  $i = i + 1$
 End
End
return S_i associated with the best β_i as the selected feature set

Fig. 7. Pseudocode of the RFE algorithm.

3.3.1. Resampling approaches

The first category of preprocessing techniques is resampling. They are developed to rebalance the imbalanced datasets and categorized into three groups:

- Under-sampling methods: they remove some of the samples from the majority of classes. The most common method is randomly discarding the samples from the majority of classes, which is called random under-sampling.
- Over-sampling methods: these methods resolve the imbalance issue by over-sampling the minority class. Randomly duplicating the minority class is the easiest available over-sampling method.
- Hybrid methods: they are a combination of under- and over-sampling methods.

In this paper, the ADASYN (He et al., 2008) is applied to overcome the issue of the imbalanced dataset. The ADASYN adaptively generates minority samples according to their distributions. Hence, it falls under the category of over-sampling. It tries to generate more synthetic data for minority classes, which are more difficult to learn in comparison to the samples are easier to learn. Mainly, this approach enhances the learning process regarding the data distribution in two ways. Not only does it reduce the biases caused by the imbalance issue, but also it adaptively moves the classification boundary toward the difficult samples, to focus on learning them.

In addition, one of the key decisions in making the dataset balanced is the number of synthetic samples from the minority class that is needed to be resampled. To make this decision, the ADASYN defines distribution density r_i as a criterion in this equation:

$$r_i = \frac{\Delta_i}{K} i = 1, \cdots..,n \tag{8}$$

where K is the number of nearest neighbors according to the Euclidean distance, n is the number of samples, and Δ_i is the number of samples in the K nearest neighbors of sample i in the minority class.

To be more specific, according to the level of difficulty, the normalized distribution density is used as a measurement of the distribution of weights for samples in the minority class. This leads the algorithm to focus on the samples that are difficult to learn.

3.4. Extreme gradient boosting

Extreme gradient boosting is a recently developed gradient boosting approach which was proposed by (Chen and Guestrin, 2016). It is a modified and advanced implementation of gradient boosted framework (Friedman, 2001) that demonstrated remarkable results in the Kaggle machine learning competition. Boosting algorithms sequentially build models in which the subsequent predictors learn from the mistakes of the previous predictors. These predictors could be decision tree models, regression models, or any other classifiers.

We applied XGBoost to the prepared dataset to predict rail defects occurrence for each segment. Therefore, this is a binary classification problem in which 1 stand for rail defects occurrence and 0 means there is no rail defect, and each segment was described by selected important features. XGBoost, as a tree-based model, partitions the feature space X into a set of T non-overlapping regions $R_1, \cdots, R_T$ and fits a simple model for each region. It improves the accurracy by developing multiple models in sequence and emphasizing on those segments that are difficult to estimate. In these models, segments that are difficult to estimate based on the previous models appears more than other segments, and each additional model is aimed to correct the previous model.

One of the problems with decision trees is the complexity of them that results in over-fitting. Complexity depends on the size of the regions $R_1, \cdots, R_T$, depth of the tree, and relative difference in the leaf nodes $w_1, \cdots, w_T$. Unlike the other boosting algorithms, to resolve this issue, XGBoost incorporates a regularization term into the objective function.

$$L_k(F_k(x_i)) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \varphi(f_k) \tag{9}$$

$$\varphi(f) = \gamma T + \frac{1}{2}\rho \, ||w||^2 \tag{10}$$

The objective function is a loss function that includes two parts. The first part, which is common in boosting algorithms, minimizes the difference between prediction and the target variable. The second term, $\varphi(f)$, is introduced in XGBoost as a regularization term. It penalizes the complexity and helps to avoid over-fitting by smoothing the final learnt weights. That is, the regularized objective will tend to select a simple, yet predictive model. That is, trees are pruned and only important track geometry parameters that play a crucial role in the occurrence of rail defects retain in the tree. In addition, interpreting the relationship between track geometry parameters and rail defects in a simple model is very beneficial for practical purposes.

Prediction in the decision tree is the product of the interaction between track geometry parameters. XGBoost may create deep trees to minimize the loss function. In these trees, track geometry parameters might be added to the decision trees in order to minimize the objective function and capture the knowledge-based relationship between track geometry parameters and rail defects. However, some of the track geometry parameters are highly correlated or technically related to each other and should not interact to predict the defects in the decision trees.

To address this issue, we define feature interaction constraints to avoid the interaction between those features that should not interact. For example, Figs. 3 and 4 show that CRV and Xlevel are highly correlated. By applying feature interaction constraints, they

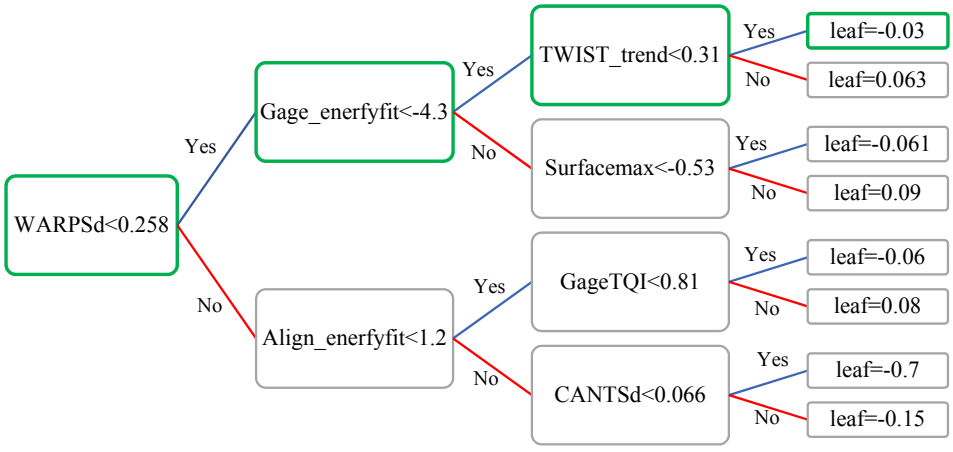

Fig. 8. A sample decision tree.

will not interact in trees to predict the rail defects. This improves the prediction performance through domain-specific knowledge. Fig. 8 shows a sample decision tree. In this decision tree, the highlighted prediction path shows that the highlighted leaf (prediction) is the product of the interaction of WARPSd, Gage_energyfit, and TWIST_trend. Also, feature interaction constraints prevent two or more specific features from interacting with each other in a path.

To the best of our knowledge, very few papers have applied XGBoost to transportation datasets. XGBoost (Martey et al., 2017) is a supervised learning method that includes a model, parameters, and hyper-parameters. The model refers to the mathematical structure of the algorithm, and the parameters are learned from the training dataset. Unlike the parameters that are related to the internal configuration of the algorithm, hyper-parameters are the external configuration of the algorithm (e.g. learning rate, tree depth, number of boosts, etc.) and cannot be estimated from the training data. However, they play a crucial role in the learning process since they help the model to estimate the parameters. The more hyper-parameters an algorithm has, the more challenging the tuning process is. XGBoost has numerous hyper-parameters that make the tuning more challenging. Although simple methods such as random search and grid search have been mostly used to optimize the hyper-parameters, for algorithms such as XGBoost that has numerous hyper-parameters, other advanced methods such as Bayesian hyper-parameter optimization can be more effective.

3.5. Bayesian hyper-parameter optimization

Hyper-parameters are parameters of training algorithms that cannot be learned directly from inputs. They directly control the behavior of the optimization algorithm and influence the performance of it. Thus, they should be determined precisely. The hyper-parameters of XGBoost include learning rate (l), maximum tree depth (m), subsample ratio (s), Gamma (γ), number of boosts (n), maximum delta step (δ), and minimum child weight (c). Also, α and λ are regularization coefficients for the linear model.

Hyper-parameter optimization is the problem of optimizing a loss function over a graph-structured configuration space. The most popular methods to tune the hyper-parameters are grid search, random search, and automatic hyper-parameter tuning. Sequential Model-based Global Optimization (SMBO) algorithm, which uses a surrogate function to estimate the true black box function (Hutter et al., 2011), falls under the automatic tuning category. Fig. 9 depicts the pseudocode of the SMBO algorithm (Hutter et al., 2011). It constructs a model M to transform the hyper-parameter setting x into the loss function L. The loss function and corresponding setting are recorded in H. The inner loop of the algorithm proceeds as follows. Based on the modelM_{t-1}, x^* which is a local optimum hyper-parameter setting is determined, and then, L is calculated. According to the obtained values of L, a new model M_t is built until the algorithm reaches the maximum iteration T.

Different criteria have been used to determine the local optimum. In this paper, Expected Improvement (EI) (Jones, 2001) is selected as a criterion, which is defined as follows:

$$EI(x) = \int_{-\infty}^{\infty} \max(L^* - L, 0) p_M(L|x) dL \tag{11}$$

Two different Bayesian optimization approaches including Gaussian process (GP) (Snoek et al., 2012) and Tree-structure Parzen Estimator (TPE) (Bergstra et al., 2011) have been proposed to approximate L by modeling H. Since the TPE has demonstrated better results, as in (Thornton et al., 2013, Xia et al., 2017), the authors incorporate a TPE-based approach into the SMBO algorithm to optimize the hyper-parameters.

In the TPE approach, the configuration space is restricted to a tree-structured space. In particular, it provides simple, yet efficient, solutions to determine the model M and to find the local optimum hyper-parameter settings. The main difference between the GP and TPE lies in the calculation of$p(x|L)$. While the GP calculates it directly, the TPE indirectly obtains that based on$p(x|L)$ and$p(L)$. Using density estimations, it defines $p(x|L)$ as follows (Bergstra et al., 2011):

$$p(x|L) = \begin{cases} l(x) L < L^* \\ g(x) L \geq L^* \end{cases} \tag{12}$$

where $l(x)$ is the density which is estimated using the observations given that $L < L^*$, and $g(x)$ is the density formed by using the rest of the observations. Also, $L^*$is some quantile (γ) of the observed L values in which $p(L < L^*) = \gamma$, and $\gamma = 0.15$. To facilitate the optimization of EI, (Bergstra et al., 2011) proved that EI is proportional to (11):

Pseudocode of the SMBO

$SMBO(L, M_0, T, S)$
Initialization: H=0
For t=1 to T
  $x^* = argminS(x, M_{t-1})$
  Evaluate $L(x^*)$,
  $H = H \cup (x^*, L(x^*))$
  Fit a new model M_t based on the updated $\boldsymbol{H}$
End for
Return $\boldsymbol{H}$

Fig. 9. Pseudocode of the SMBO algorithm.

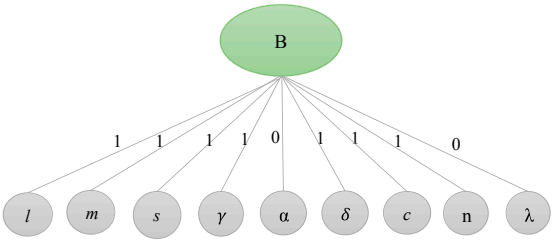

Fig. 10. A sample tree structure in the TPE.

$$EI(x) \propto \left(\gamma + \frac{g(x)}{l(x)}(1-\gamma)\right)^{-1} \tag{13}$$

The $l(x)$ and $g(x)$ are hierarchical processes including discrete and continuous variables. The Parzen estimator is applied to approximate them using a probability density function in the vicinity of N observations. Given that $(x_1, x_2, \cdots, x_N)$ are independent and identically distributed samples, the Parzen estimator approximate the probability density function f using the following equation:

$$f_h(x) = \frac{1}{Nh}\sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) \tag{14}$$

where *K(.)* is the kernel function, and h is the bandwidth. For each valid node in the tree, a Parzen estimator is created to approximate the probability density function. Each node includes either a discrete hyper-parameter or a continuous one. Discrete hyper-parameters are approximated using the probabilities proportional to the occurrence of the corresponding choice in H. On the other hand, for continuous hyper-parameters *1-D* Parzen estimator is calculated by using the Gaussian as a density.

We used the TPE approach to optimize the hyper-parameters of XGBoost algorithm. Fig. 10 represents a sample tree structure in the TPE. There are two variations of XGBoost: tree-based model and linear model. In the sample tree, B = 1 indicates the tree-based model, and B = 0 represents the linear model.

Starting from the root node, the TPE moves down to the valid leaf nodes and estimates the density of the hyper-parameter. Since leaves are mutually independent, the joint density function *f(x)* is calculated by multiplication of the individual density estimations.

3.6. Partial dependence analysis (PDA)

Machine learning algorithms such as XGBoost are very complex and interpreting them is not as easy as other parametric methods such as linear regression. PDA facilitates the interpretation of complex machine learning algorithms and provides a very beneficial and practical analysis that uncovers the effect of each feature as well as subsets of features on the predicted outcome (Friedman, 2001). Unlike the importance factor that is calculated for each feature individually, PDA investigates the interactions of the subsets of variables. In addition, PDA analysis examines the behavior of each feature with respect to the prediction model and target variable.

In terms of mathematical definition, for any subset x_s, $s = \{1, 2, \cdots, N\}$of the features, the partial dependence function is defined as follows:

$$F_s(x_s) = E_{x_c}[F(x_s, x_c)] \tag{15}$$

where x_c is the set of all features excluding x_s. This function is used to visualize the relation between the predictive model and a subset of variables. However, for more than two variables it is not easy to graphically examine the dependence. To facilitate the calculation of the partial dependence function, it is estimated using the Eq. (14):

$$\widehat{F}_s(x_s) = \frac{1}{N}\sum_{i=1}^{N} \widehat{F}_s(x_s, x_{ci}) \tag{16}$$

In this equation, x_{ci} is the data value of the features, and N is the number of records. If two features do not interact, the partial dependence of them is the sum of the corresponding partial dependences of each of them. For more details, interested readers are referred to Friedman (2001) and Friedman and Popescu (2008). Partial dependency plots (PDP) use this estimation to graphically depict the dependencies and interactions.

In our methodology, PDA extracts practical information from XGBoost prediction model about the interaction behavior of the track geometry parameters in rail defects occurrence. This information includes behavior, critical ranges, and the interaction between the track geometry parameters.

4. Results and discussion

4.1. Initial feature analysis

The main goal here is to study the effect of the track geometry parameters and tonnage data on rail defects. Furthermore, the

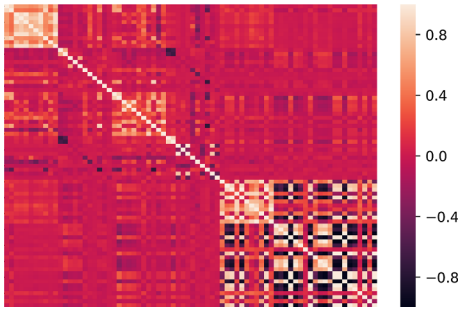

Fig. 11. Correlation analysis before feature selection.

model is aimed to predict the rail defect based on track geometry parameters, extracted features, and total MGT.

As discussed before, the dimension of the dataset is reduced in two different ways, and the proposed model is applied to both of them separately to compare the efficiency of both methods. Therefore, there are two datasets; the first one is created by applying SVD to retain the parameters that correspond to 92% of the variation of the dataset. The second one is obtained by using the RFE algorithm to select the significant features.

Classification algorithms calculate the importance rate based on the contribution of each feature in each decision tree. In each decision tree, for each variable, the importance rate is calculated by the amount that feature's split point improves the performance measure. Then, the importance rate for the corresponding feature is averaged across all of the decision trees. Hence, the higher the importance rate, the more important the feature is. In other words, a feature with higher importance rate is more closely related to defects occurrence. Given the feature importance obtained, RFE algorithm is implemented with the parameters value of $\alpha = 0.035$, $\delta = 0.12$, and $I = 35$.

To analyze the impact of considering correlation in the algorithm, correlation analysis is performed both before and after running the RFE. Fig. 11 shows the correlation between the defined features before running the RFE. There are some positively and negatively correlated features in the dataset. The RFE handled these features using feature interaction constraints and importance rate. However, based on RFE, all of the correlated features are not removed and some of them are kept but prevented from interactin with each other. Fig. 12 shows that the correlation among the selected features is not very high. However, if there are few features that are of high correlation, it indicates that they are of high importance rate. In fact, if a feature is selected by the algorithm, it adds knowledge to the dataset.

Moreover, Fig. 13 shows the number of features and the value of accuracy in each iteration of the RFE algorithm. In most of the iterations, removing a feature improves the accuracy. However, when iteration is at 33, removing a feature reduces the accuracy significantly. Therefore, RFE stops at iteration 33 and 16 features are selected. It is worth noting that the impact of removing features on the accuracy of the algorithm is problem-dependent.

Fig. 14 lists the chosen 16 features from RFE. In this figure, note that trend only refers to the time-trend of the energy spectral density since the time-trend of other extracted features does not show any significance. Total MGT shows the highest importance rate, and 7 out of 16 features are time-trend features, which implies that the historic trend of track geometry parameters plays a crucial role in the rail defect occurrences. The TQI feature for DIP also impacts the occurrence of the rail defect. In general, total MGT, DIP, Gage, and CANT are the geometry parameters which have more significant features in this study. In addition, some of the features such as Xlevel and unbalance are not among the selected features that indicate they contribute less than other features to the rail defects occurrence.

For practical purposes, results of the RFE is of paramount importance for the railroad. Initially, it identifies the features that the

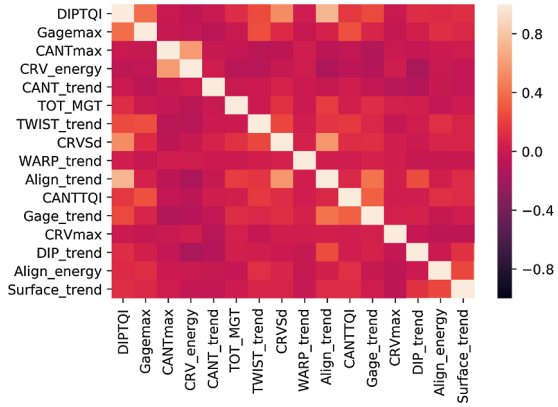

Fig. 12. Correlation analysis after feature selection.

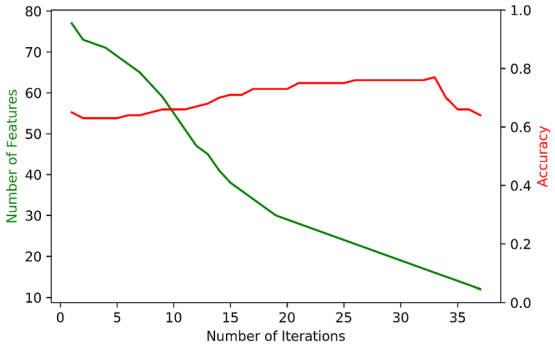

Fig. 13. Number of features and values of accuracy in each iteration of the RFE algorithm.

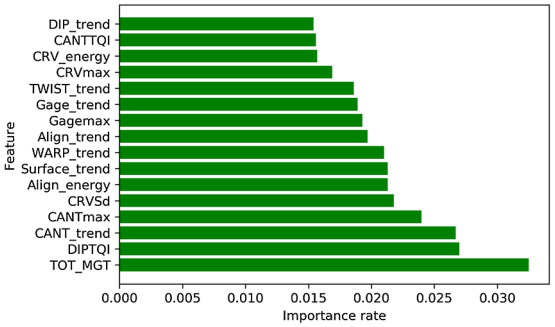

Fig. 14. Importance rate of selected features.

railroad should track and analyze them more than other features. Particularly, MGT and DIPTQI have the highest importance rate. Therefore, based on inspection data TQI for DIP can be calculated and tracked to prevent or at least delay the occurrence of rail defects. Another practical insightful outcome is that almost 44% of the features are time-trend features. This demonstrates that variation of the features such as Gage and Alignment play a crucial role in rail defects occurrence. To complete these results and provide more practical insights, PDA is performed and explained in Section 4.3 to identify the critical ranges.

4.2. Model selection

This subsection compares the performance of the applied machine learning algorithms. To evaluate the accuracy of the model, various performance criteria can be taken into account. Regarding the nature of the problem, the railroad pursues two main objectives: correctly predicting the rail defects when they exist and identifying the track geometry parameters that contribute the most to the occurrence of rail defects. In machine learning terminology, the correct prediction of defects when they exist is measured by True Positive Rate (TPR) and True Negative Rate (TNR). In addition, two other measures including False Positive Rate (FPR) and False Negative Rate (FNR) are taken into account. Also, accuracy is defined as the proportion of the number of truly predicted observations in the total number of observations. For imbalanced datasets, obtaining a high value of TPR is extremely challenging. However, the proposed approach focuses on that and attempts to obtain promising results. To satisfy the second objective, the model provides importance rates for each track geometry parameter, which are the representative of the relative contribution of the corresponding features.

In order to evaluate the model, 5-fold cross-validation is performed. The dataset is randomly divided into five disjoint pieces of equal size, where each part has roughly the same ratio of imbalance. In addition, the model is trained five times, each time with one piece set aside. SVM (Li et al., 2014), Logistic Regression, Random Forest (Li and He, 2015), and the proposed XGBoost model are applied to both datasets separately. To clarify the contribution of each of the proposed components such as the RFE algorithm, SVD, and Bayesian hyper-parameter optimization method to the model accuracy, a comprehensive comparison results are provided. In addition, in order to demonstrate the efficiency of the proposed Bayesian Optimization (BO) method, a random search (RS) hyper-parameter tuning method (Bergstra and Bengio, 2012) is applied. Table 2 shows the search space of hyper-parameters for these methods. Computation time for BO to optimize the hyper-parameters of XGBoost is about 4 s, while it is less than 1 s for RS. Therefore, the computation time is acceptable. In addition, as the results in Fig. 15 show, BO method performs better than RS and increases the accuracy of the XGBoost about 2.5%. Also, for other machine learning algorithms, BO performs better than RS.

Fig. 15 shows the results of applying the algorithms. Generally, as the results represent, all of the algorithms except Logistic Regression perform better on the dataset in which the RFE algorithm is used to select the features. This indicates that, in this study, feature selection with the RFE algorithm prepares a better dataset than SVD and results in higher accuracy. In addition, considering

Table 2
The range of hyper-parameters for two optimization methods.

Hyper-parameter	Random search	Bayesian optimization
Number of boosts (n)	70	70
Subsample ratio (s)	(0.8, 1)	(0.8, 1)
Maximum tree depth (m)	(2, 10)	(2, 10)
Minimum child weight (c)	(0, 5)	(0, 5)
Maximum delta step (δ)	(0, 1)	(0, 1)
Gamma (γ)	(0, 0.02)	(0, 0.02)
learning rate (l)	(0.1, 0.4)	(0.1, 0.4)

Fig. 15. Results of the applied methods.

the other performance measures, XGBoost outperforms the other algorithms with the highest TPR of 82% and the lowest FNR rate of 16%. Considering the high imbalance rate of the dataset, these values are competitive and acceptable. Although the FPR for Random Forest is lower than XGBoost, the TPR for Random Forest is very low. Another interesting criterion that confirms the outperformance of XGBoost is FNR which indicates that the segments without rail defects are wrongly predicted as defects. Thus, XGBoost generally performs better than other algorithms.

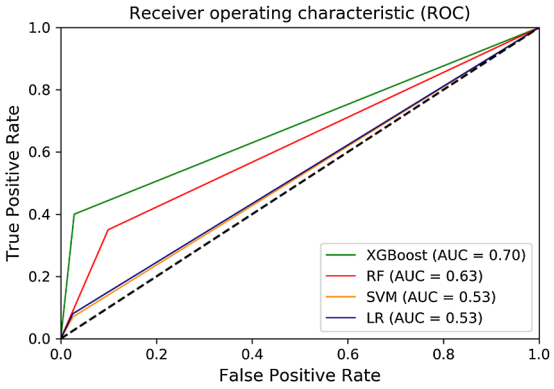

Fig. 16. ROC curve for the applied methods.

Finally, to demonstrate the efficiency of the proposed methodology, machine learning algorithms without any feature selection and hyper-parameter optimization are also applied to the prepared dataset. The results verify that the proposed methodology significantly improves the accuracy by resolving the issues such as a large number of features and imbalance.

Finally, to compare the performance of algorithms, Receiver Operating Characteristic (ROC) curve is also employed. Since the methodology with Bayesian hyperparameter optimization and RFE algorithm outperforms the other methods, the ROC curve is only provided to compare them together and not with other methods. ROC curves compare the algorithms based on TPR and FPR. That is, it shows the capability of the algorithms in discriminating between different classes; segments with rail defects and without rail defects. The farther the curve is from the diagonal line, the better the algorithm is. In addition, the Area Under the Curve (AUC) is another statistic that is beneficial to compare different algorithms. Fig. 16 shows the ROC curve for the algorithms. It confirms the results presented in Fig. 15 that XGBoost outperforms the other models. Since the dataset is highly imbalanced, the highest AUC that was obtained is about 0.7 for XGBoost.

According to the results and discussion, XGBoost outperforms the others. However, it is a complex machine learning algorithm; the prediction model should be interpreted, and the role and interactions of features should be identified for practical purposes.

4.3. Partial dependence analysis (PDA) results

In order to thoroughly analyze the dependencies and interactions among the track geometry parameters and target variable, we first draw the PDPs for those features that have a high importance rate. This provides the railroad with deep knowledge about the impact of each feature on the occurrence of rail defects. Fig. 17 depicts the PDP for MGT and DIPTQI features. In these plots, Y-axis is the Logit of the probability. Generally, the higher the total MGT, the higher the probability of defects occurrence. Specifically, an increase in the amount of MGT until 100 doesn't impact the rail defects occurrence significantly, but once the total MGT exceeds 100, the probability of the rail defects occurrence increases drastically. It is worth noting that a sharp decrease after the MGT of 150 is due to the lack of observations with the total MGT more than 150. In terms of railroad application, it is of paramount importance for the railroad to figure out the critical range of values for MGT. Also, for DIPTQI, an increase from 0.98 causes a sharp increase in the probability of defect occurrence. However, comparing the impact of MGT with DIPTQI on the defect occurrence probability demonstrates that the MGT plays the more important role than DIPTQI in rail defects occurrence. This, in turn, confirms the consistency of the proposed approach.

In addition, Fig. 18 shows the PDPs for two other features that are less significant than others in predicting the rail defect.

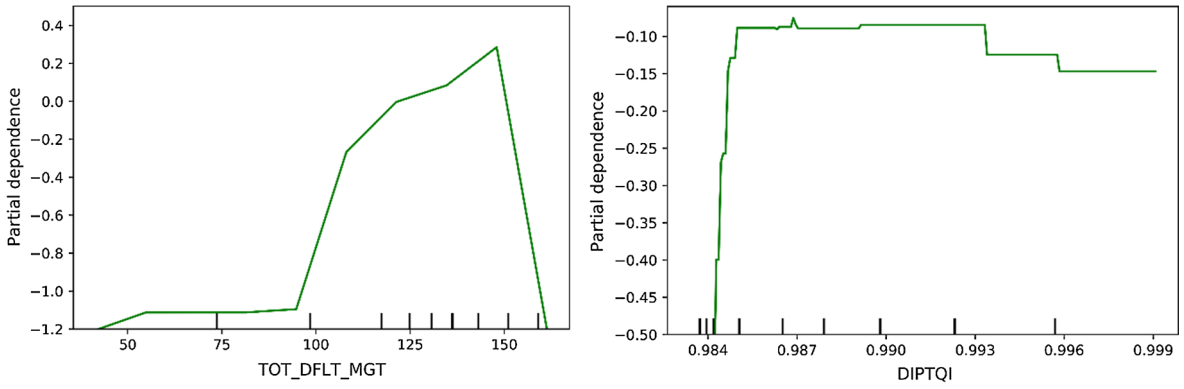

Fig. 17. PDP for MGT and DIPTQI.

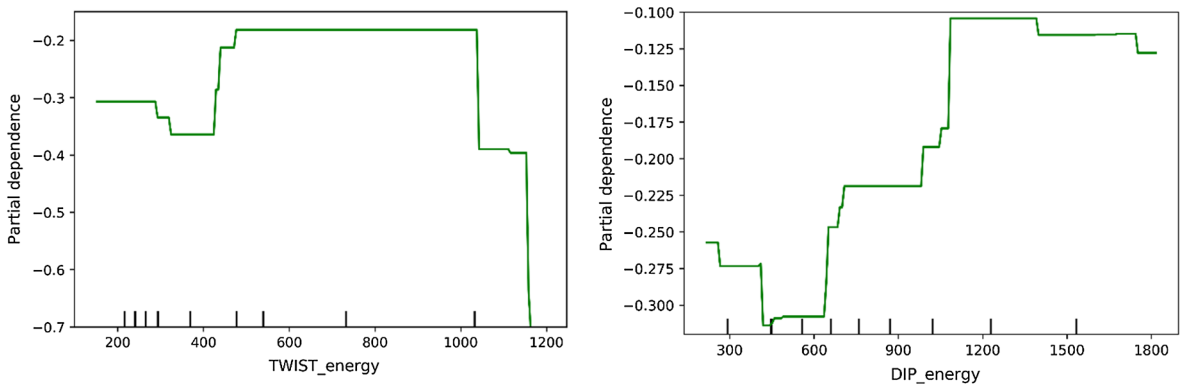

Fig. 18. PDP for Twist Energy spectral density and DIP Energy spectral density.

Comparing the range of the Y-axis of these plots with those of Fig. 11, shows less variation than the Y-axis of Fig. 12. This demonstrates that the variations of these variables do not significantly influence the occurrence of rail defect.

Our methodology identifies the important features that contribute to the rail defects happening. Most of the aggregated features such as TQI, SD, and Max of the track geometry parameters are very easy to calculate and available for railroads. Therefore, these features can be incorporated into track renewal and maintenance planning to prevent rail defects from happening or identify the risky segments. For example, DIPTQI is the second important feature. Based on inspection data, DIPTQI can be calculated and compared to the provided range by PDA method to identify the risky segments.

As mentioned before, the partial dependence function can also be applied to study the interactions between track geometry parameters. For every subset of features, very valuable information with respect to the other features can be extracted. However, for a subset of more than two variables, it is very challenging to visualize. For example, Fig. 19 shows the PDP for Align-Trend and Gage-Trend. According to this figure, the highest interaction between Align-Trend and Gage-Trend happens when Align-Trend is between 3 and 7 and Gage-Trend is between 6 and 9. In other words, the aforementioned values of Align-Trend and Gage-Trend influence the occurrence of rail defects more than other values. Analyzing the interactions between total Align-Trend and Gage-Trend, the railroad can acquire knowledge about the impact of the MGT on Gage-Trend and devise an appropriate maintenance plan based on that.

5. Concluding remarks

This paper has developed a new data-driven approach to analyze the impact of foot-by-foot track geometry on the occurrences of rail defects. This approach involves various components, each following a certain objective. Different features such as time-trend, TQI, and energy spectral density were extracted to ease the learning process. Moreover, a Recursive Feature Elimination (RFE) algorithm is developed to reduce the dimensionof the data. This study also merged the datasets and applied a decision tree based XGBoost algorithm to predict the defect and identify the significant track geometry parameters. To ensure the highest accuracy of the XGBoost, a Bayesian optimization is applied to tune its hyper-parameters.

The results of our case study show that MGT is the most significant parameter that impacts the occurrence of rail defects. Since MGT is known as one of the significant factors in practice, this confirms that the proposed methodology is capable of correctly identifying the relationship and significant track geometry parameters. Also, aggregated measures such as Track Quality Index (TQI) and time-trend were identified as important features that can help to predict rail defects. Without the loss of generality, by applying

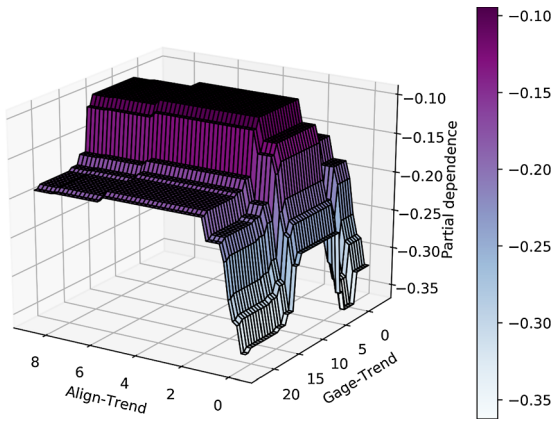

Fig. 19. PDP for the interaction analysis between MGT and Gage Trend.

the approach, important factors can be determined for other similar datasets. In terms of track geometry parameters, DIP, CANT, Gage, and surface played a more important role than the others. we also applied partial dependence plots (PDP) to investigate the influence of multiple track geometry parameters on rail defects occurrence. For example, the segments with the Align-Trend values between 3 and 7 and Gage-Trend between 6 and 9 are more prone to grow rail defects than others. Thus, using PDP these critical ranges can be identified for other features.

Finally, the approach correctly predicts about 83% of rail defects based on tonnage and track geometry data. Considering the issue of imbalance, this amount of accuracy is acceptable, and having a balanced dataset will result in better prediction.

Implementing the proposed approach, the railroad can identify the occurrence of rail defects in their early stage. Moreover, focusing on those key geometry-related factors can prevent the occurrence of rail defects. The significant factors and critical ranges that have been identified by the proposed methodology can be used as a measure to determine the grinding need for each segment. In addition, track geometry data is used for other preventive maintenance activities such as tamping. Therefore, the findings in this study could also possibly direct the railroad toward how to jointly conduct preventive maintenance for both tamping and grinding. Moreover, critical ranges for significant factors are appropriate measures for track renewal decision making. That is, the risk of rail defect occurrences can be estimated using the significant factors and segments with higher risk could be replaced in a timely manner."

Acknowledgment

This study was funded by FRA under contract NO. DTFR5317C00003. The data was provided by BNSF. Authors would like to express their sincere thanks for the support from FRA and BNSF.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.trc.2019.03.004.

References

Ahmed, M., Abdel-Aty, M., 2013. A data fusion framework for real-time risk assessment on freeways. Transport. Res. Part C: Emerg. Technol. 26, 203–213.
Andrade, A.R., Teixeira, P.F., 2015. Statistical modelling of railway track geometry degradation using Hierarchical Bayesian models. Reliab. Eng. Syst. Saf. 142, 169–183.
Bergstra, J., Bengio, Y., 2012. Random search for hyper-parameter optimization. J. Mach. Learn. Res. 13 (Feb), 281–305.
Bergstra, J.S., Bardenet, R., Bengio, Y., Kégl, B., 2011. Algorithms for hyper-parameter optimization. Adv. Neural Inform. Process. Syst.
Beyan, C., Fisher, R., 2015. Classifying imbalanced data sets using similarity based hierarchical decomposition. Pattern Recogn. 48 (5), 1653–1672.
Branco, P., Torgo, L., Ribeiro, R.P., 2016. A survey of predictive modeling on imbalanced domains. ACM Comput. Surv. (CSUR) 49 (2), 31.
Cai, J., Luo, J., Wang, S., Yang, S., 2018. Feature selection in machine learning: a new perspective. Neurocomputing 300, 70–79.
Cannon, D., Edel, K.O., Grassie, S., Sawley, K., 2003. Rail defects: an overview. Fatigue Fract. Eng. Mater. Struct. 26 (10), 865–886.
Celik, T., 2009. Unsupervised change detection in satellite images using principal component analysis and $ k $-means clustering. IEEE Geosci. Rem. Sens. Lett. 6 (4), 772–776.
Chen, T., Guestrin, C., 2016. Xgboost: a scalable tree boosting system. Proceedings of the 22nd acm sigkdd International Conference on Knowledge Discovery and Data Mining. ACM.
Cong, F., Chen, J., Dong, G., Zhao, F., 2013. Short-time matrix series based singular value decomposition for rolling bearing fault diagnosis. Mech. Syst. Sig. Process. 34 (1–2), 218–230.
Cooley, J.W., Tukey, J.W., 1965. An algorithm for the machine calculation of complex Fourier series. Math. Comput. 19 (90), 297–301.
Dick, C.T., Barkan, C., Chapman, E., Stehly, M., 2003. Multivariate statistical model for predicting occurrence and location of broken rails. Transport. Res. Rec.: J. Transport. Res. Board 1825, 48–55.
Ding, C., Cao, X.J., Næss, P., 2018. Applying gradient boosting decision trees to examine non-linear effects of the built environment on driving distance in Oslo. Transport. Res. Part A: Pol. Pract. 110, 107–117.
Ding, C., Wang, D., Ma, X., Li, H., 2016a. Predicting short-term subway ridership and prioritizing its influential factors using gradient boosting decision trees. Sustainability 8 (11), 1100.
Ding, C., Wu, X., Yu, G., Wang, Y., 2016b. A gradient boosting logit model to investigate driver's stop-or-run behavior at signalized intersections using high-resolution traffic data. Transport. Res. Part C: Emerg. Technol. 72, 225–238.
Ekberg, A., Kabo, E., Andersson, H., 2002. An engineering model for prediction of rolling contact fatigue of railway wheels. Fatigue Fract. Eng. Mater. Struct. 25 (10), 899–909.
El-Sibaie, M., Zhang, Y.-J., 2004. Objective track quality indices. Transport. Res. Rec.: J. Transport. Res. Board 1863, 81–87.
Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. Ann. Stat. 1189–1232.
Friedman, J.H., Popescu, B.E., 2008. Predictive learning via rule ensembles. Ann. Appl. Stat. 2 (3), 916–954.
Ghofrani, F., He, Q., Goverde, R.M., Liu, X., 2018. Recent applications of big data analytics in railway transportation systems: a survey. Transport. Res. Part C: Emerg. Technol. 90, 226–246.
Guo, H., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., Bing, G., 2017. Learning from class-imbalanced data: review of methods and applications. Expert Syst. Appl. 73, 220–239.
Hajizadeh, S., Núnez, A., Tax, D.M., 2016. Semi-supervised rail defect detection from imbalanced image data. IFAC-PapersOnLine 49 (3), 78–83.
Hamilton, J.D., 1994. Time Series Analysis. Princeton University Press Princeton, NJ.
He, H., Bai, Y., Garcia, E.A., Li, S., 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. IEEE International Joint Conference on Neural Networks, 2008. IJCNN 2008 (IEEE World Congress on Computational Intelligence). IEEE.
He, Q., Li, H., Bhattacharjya, D., Parikh, D.P., Hampapur, A., 2015. Track geometry defect rectification based on track deterioration modelling and derailment risk assessment. J. Operat. Res. Soc. 66 (3), 392–404.
Hutter, F., Hoos, H.H., Leyton-Brown, K., 2011. Sequential model-based optimization for general algorithm configuration. International Conference on Learning and Intelligent Optimization. Springer.
Jones, D.R., 2001. A taxonomy of global optimization methods based on response surfaces. J. Global Optim. 21 (4), 345–383.
Lasisi, A., Attoh-Okine, N., 2018. Principal components analysis and track quality index: a machine learning approach. Transport. Res. Part C: Emerg. Technol. 91, 230–248.

Lee, C., Landgrebe, D., 1993. Feature extraction and classification algorithms for high dimensional data.
Lee, S., 2005. Development of objective track quality indices. Federal Railroad Administration, Research Results, RR: 05-01.
Li, H., Parikh, D., He, Q., Qian, B., Li, Z., Fang, D., Hampapur, A., 2014. Improving rail network velocity: a machine learning approach to predictive maintenance. Transport. Res. Part C: Emerg. Technol. 45, 17–26.
Li, Z., He, Q., 2015. Prediction of railcar remaining useful life by multiple data source fusion. IEEE Trans. Intell. Transp. Syst. 16 (4), 2226–2235.
Liu, X., Saat, M.R., Barkan, C.P., 2017. Freight-train derailment rates for railroad safety and risk analysis. Accid. Anal. Prev. 98, 1–9.
López, V., Fernández, A., García, S., Palade, V., Herrera, F., 2013. An insight into classification with imbalanced data: empirical results and current trends on using data intrinsic characteristics. Inf. Sci. 250, 113–141.
Loyola-González, O., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., García-Borroto, M., 2016. Study of the impact of resampling methods for contrast pattern based classifiers in imbalanced databases. Neurocomputing 175, 935–947.
Ma, X., Ding, C., Luan, S., Wang, Y., Wang, Y., 2017. Prioritizing influential factors for freeway incident clearance time prediction using the gradient boosting decision trees method. IEEE Trans. Intell. Transp. Syst. 18 (9), 2303–2310.
Martey, E.N., Ahmed, L., Attoh-Okine, N., 2017. Track geometry big data analysis: a machine learning approach. 2017 IEEE International Conference on Big Data (Big Data). IEEE.
Motoda, H., Liu, H., 2002. Feature selection, extraction and construction. Communication of IICM (Institute of Information and Computing Machinery, Taiwan), vol. 5, 67–72.
Pereira, J., 2010. Handbook of Research on Personal Autonomy Technologies and Disability Informatics. IGI Global.
Sageder, G., Zaharieva, M., Breiteneder, C., 2016. Group feature selection for audio-based video genre classification. International Conference on Multimedia Modeling. Springer.
Sharma, S., Cui, Y., He, Q., Mohammadi, R., Li, Z., 2018. Data-driven optimization of Railway maintenance for track geometry. Transport. Res. Part C: Emerg. Technol. 90, 34–58.
Snoek, J., Larochelle, H., Adams, R.P., 2012. Practical bayesian optimization of machine learning algorithms. Adv Neural Inform. Process. Syst.
Soleimanmeigouni, I., Ahmadi, A., Kumar, U., 2018. Track geometry degradation and maintenance modelling: a review. Proc. Inst. Mech. Eng., Part F: J. Rail Rapid Transit 232 (1), 73–102.
Thornton, C., Hutter, F., Hoos, H.H., Leyton-Brown, K., 2013. Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms. Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM.
Wu, X., Guo, J., Xian, K., Zhou, X., 2018. Hierarchical travel demand estimation using multiple data sources: a forward and backward propagation algorithmic framework on a layered computational graph. Transport. Res. Part C: Emerg. Technol. 96, 321–346.
Xia, Y., Liu, C., Li, Y., Liu, N., 2017. A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. Expert Syst. Appl. 78, 225–241.
Zarembski, A.M., Einbinder, D., Attoh-Okine, N., 2016. Using multiple adaptive regression to address the impact of track geometry on development of rail defects. Constr. Build. Mater. 127, 546–555.
Zhang, D., Zhou, Z.-H., 2005. (2D) 2PCA: two-directional two-dimensional PCA for efficient face representation and recognition. Neurocomputing 69 (1–3), 224–231.
Zhang, Y., Haghani, A., 2015. A gradient boosting method to improve travel time prediction. Transport. Res. Part C: Emerg. Technol. 58, 308–324.
Zhao, S., Khattak, A.J., 2017. Factors associated with self-reported inattentive driving at highway-rail grade crossings. Accid. Anal. Prev. 109, 113–122.
Zheng, Z., Lu, P., Lantz, B., 2018. Commercial truck crash injury severity analysis using gradient boosting data mining model. J. Saf. Res. 65, 115–124.
Zhuang, L., Wang, L., Zhang, Z., Tsui, K.L., 2018. Automated vision inspection of rail surface cracks: a double-layer data-driven framework. Transport. Res. Part C: Emerg. Technol. 92, 258–277.