The Impact of Different Kernel Functions on the Performance of Scintillation Detection Based on Support Vector Machines †

Caner Savas *[ID] **and Fabio Dovis**[ID]

Departments of Electronics and Telecommunications, Politecnico di Torino, 10129 Torino, Italy; fabio.dovis@polito.it

* Correspondence: caner.savas@polito.it

† This paper is an extended version of "Comparative Performance Study of Linear and Gaussian Kernel SVM Implementations for Phase Scintillation Detection" published in the Proceedings of the 2019 9th International Conference on Localization and GNSS (ICL-GNSS), Nuremberg, Germany, 4–6 June 2019.

check for updates

Abstract: Scintillation caused by the electron density irregularities in the ionospheric plasma leads to rapid fluctuations in the amplitude and phase of the Global Navigation Satellite Systems (GNSS) signals. Ionospheric scintillation severely degrades the performance of the GNSS receiver in the signal acquisition, tracking, and positioning. By utilizing the GNSS signals, detecting and monitoring the scintillation effects to decrease the effect of the disturbing signals have gained importance, and machine learning-based algorithms have been started to be applied for the detection. In this paper, the performance of Support Vector Machines (SVM) for scintillation detection is discussed. The effect of the different kernel functions, namely, linear, Gaussian, and polynomial, on the performance of the SVM algorithm is analyzed. Performance is statistically assessed in terms of probabilities of detection and false alarm of the scintillation event. Real GNSS signals that are affected by significant phase and amplitude scintillation effect, collected at the South African Antarctic research base SANAE IV and Hanoi, Vietnam have been used in this study. This paper questions how to select a suitable kernel function by analyzing the data preparation, cross-validation, and experimental test stages of the SVM-based process for scintillation detection. It has been observed that the overall accuracy of fine Gaussian SVM outperforms the linear, which has the lowest complexity and running time. Moreover, the third-order polynomial kernel provides improved performance compared to linear, coarse, and medium Gaussian kernel SVMs, but it comes with a cost of increased complexity and running time.

Keywords: GNSS; scintillation; support vector machines; kernel; Gaussian; polynomial

1. Introduction

Trans-ionospheric communication of the radio waves while traveling from transmitter to user is affected by the ionosphere that is highly variable and dynamic in both time and space [1]. The ionosphere, highly varied propagation medium, has an irregular structure due to plasma instabilities and scintillation is basically random fluctuations of the parameters of trans-ionospheric waves. Observation of the scintillation has been used in many research fields such as astronomy, geophysics, atmospheric physics, ocean acoustics, telecommunications, etc. so as to identify the irregular structure of the propagation medium [1]. Moreover, it has been found that the atmospheric structure has the basic characteristics of fluid turbulence in the equilibrium range, which is characterized by the well-known properties of Kolmogorov turbulence [2]. It has led to the development of the theoretical characterization of the scintillation effect [3,4] besides the observational

point of view of the scintillation effect. However, although scintillation phenomena are well understood and widely exploited, it is difficult to find a definitive treatment of the theory of scintillation [2], and it is a fact of life for a number of communication and radar systems that have to operate through the auroral or equatorial ionosphere [5].

Global Navigation Satellite Systems (GNSS) signals also undergo severe propagation effects such as phase shifts and amplitude variations while they propagate through the Earth's upper atmosphere, the ionosphere [6]. Ionospheric irregularities affect the GNSS signals in two ways, and both refractive and diffractive effects are gathered under the name of scintillation since they cause large-scale variations in both signal power and phase [7]. Ionospheric scintillation severely may degrade GNSS receiver performance by causing signal power loss and increasing measurement noise level, causing loss of lock of the tracking of GNSS signals [8,9]. In case of strong amplitude and phase scintillation events, it has been observed that even acquisition of the signals can be prevented [10].

Under a scintillation event, proper countermeasures could be undertaken at signal processing level, enabling either more robust signal acquisition and tracking or alternate resources to decrease the effect of disturbed signal propagations. Therefore, detecting and monitoring the scintillation effects in order to estimate the ionospheric scintillation in its early stages and measure the scintillation parameters gains importance. In this sense, GNSS signals provide an excellent means for measuring scintillation effects due to the fact that they are available all the time and can be acquired through many points of the ionosphere simultaneously [11]. With the coming new GNSS systems, a greater number of signals are available for monitoring signals and the advantages of new modernized signals and constellations for scintillation monitoring can be found in [12]. Moreover, an interested reader can find useful material about the design of monitoring stations in [6,13–16] and alternative receiver architectures for monitoring of ionospheric scintillations by means of GNSS signals in [17,18].

With the evolving of the artificial intelligence world, machine learning (ML) algorithms started to be applied also for scintillation detection. One of the proposed methods is based on a support vector machines (SVM) algorithm [19] that belongs to the class of supervised machine learning algorithms [20]. Supervised algorithms require large data sets to properly train the algorithm so that it is able to recognize the scintillation presence during the analysis of new measurements. In [21], another type of supervised learning algorithm, namely, the decision tree, is applied to enable early scintillation alerts. In [22], the performance of SVM implementations for phase and amplitude scintillation detection have been evaluated. The main weakness of these works is that the impact of the design parameters of the different algorithms on the achieved performance has not been carefully analyzed in detail as witnessed by the limited literature addressing the application of machine learning to GNSS. Since the success of the SVM algorithm can be attributed to the joint use of a robust classification procedure and of a versatile way of pre-processing the data, the parameters of the machine learning phase must be carefully chosen [23].

The SVM algorithm is the most widely used kernel learning algorithm [19]. The kernel method enables the SVM algorithm to find a hyperplane in the kernel space by mapping the data from the feature space into higher dimensional kernel space and leading to achievement of nonlinear separation in the kernel space [24]. Kernel representations offer an alternative solution to increase the computational power of the linear learning machines [25]. In SVM implementations, the kernel functions are linear, Gaussian radial basis function (RBF), and polynomial are widely used. Hence, the problem of choosing an architecture for an ML-based application is equivalent to the problem of choosing a suitable kernel for an SVM implementation [25]. Moreover, when training an SVM algorithm, besides choosing a suitable kernel function, a number of decisions should also be made in the preparation of the data, by labeling them, and setting the parameters of the SVM [26]. Otherwise, uninformed choices might result in degraded performance [27].

In this paper, we extend our previous work [28] which involves the performance comparison of linear and Gaussian kernels for phase scintillation detection versus the analysis of both amplitude and phase scintillation events. Linear and Gaussian kernels, the implementation of different order

polynomial kernels, and the performance comparison on the cross-validation results are the original core of this paper. The impact of the kernel function on the scintillation detection performance by considering the related design parameters (e.g., scale parameter, polynomial order) is discussed. Performance is assessed by exploiting the receiver operating characteristics (ROC) curves, confusion matrix results and the related performance metrics. This study is performed using real GNSS signals that are affected by significant phase and amplitude scintillation effect, collected at the South African Antarctic research base (SANAE IV, 71.67° S, 2.84° W) and Vietnam (Hanoi, 21.00° N, 105.85° E).

The paper is organized as in the following. In Section 2, we introduce the scintillation measurements and the analysis of the real data. In Section 3, we provide an overview of the SVM algorithm. Then, implementation and performance analysis of the SVM algorithm with different kernel functions for scintillation detection and the comparison of the test results are discussed in Section 4. Finally, Section 5 draws conclusions.

2. Ionospheric Scintillation Data and Analysis

The amount of amplitude and phase scintillation that affect the GNSS signal can be monitored and measured by exploiting the signal tracking stage correlator outputs in the GNSS receiver. Specialized Ionospheric Scintillation Monitoring Receivers (ISMRs) or software defined radio (SDR) based receivers can be used for monitoring purposes [6,13–16,29].

Figure 1 shows an example of scintillation monitoring and data collection setup. A part of the data that are used throughout this paper were collected in the Antarctic station and in an Equatorial site by means of data grabbers of this kind. The data-collecting setup is a custom-designed solution based on a multi-constellation and multi-frequency GNSS data grabber and a software-defined radio receiver [15,30]. Data sets acquired in Antarctica grow constantly, and several levels of the data availability such as public access, limited access, and data license are offered [30].
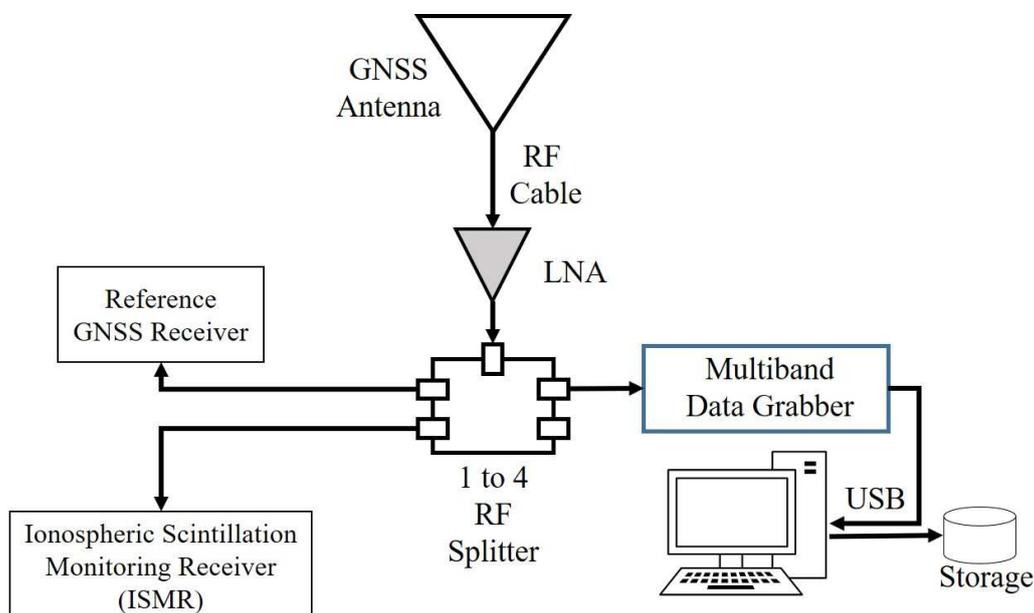

Figure 1. Experimental scintillated GNSS data collection configuration.

Fourtune, which is the type of multiband data grabber shown in Figure 1, is a multi-frequency data collection unit, and it is able to perform medium-complexity signal processing (e.g., decimation, digital filtering, quantization, etc.). It has been developed by the researchers at Joint Research Center (JRC) of the European Commission in Italy and an interested user can find more information about its architecture in [16]. Scintillated data sets that are collected via Fourtune and used in this paper are summarized in Table 1. Pseudo-Random Noise (PRN) codes are ranging code components of

the transmitted satellite signals and are unique for each satellite signal. In Table 1, they refer to the satellites from which the scintillation effect observed in the received signals.

Table 1. Specifications of the scintillated data sets with visible Pseudo-Random Noise (PRN) codes and location of the stations.

	Dates	PRNs	Station	Coordinates
1	21 January 2016 3 February 2016 8 February 2016 17 August 2016	11, 14, 22 3, 6 1, 12, 14 9	South African Antarctic Research Base (SANAE-IV), Antarctic	Lat.: 71.67278° S Long.: 2.840556° W
2	10 April 2013 12 April 2013 16 April 2013 4 October 2013	11 17 1, 20, 28 15, 21, 24	Hanoi, Vietnam	Lat.: 21.004556° N Long.: 105.843917° E

Typically, there are two parameters to indicate the amount of scintillation effect. Amplitude scintillation is monitored by computing the S_4 index which corresponds to the standard deviation of the detrended signal intensity [15]. In an ISMR, S_4 metric is calculated as [16]

$$S_4 = \sqrt{\frac{\langle I^2 \rangle_T - \langle I \rangle_T^2}{\langle I \rangle_T^2}}, \tag{1}$$

where I is the detrended signal intensity, $\langle \cdot \rangle_T$ is the average operation over a fixed period T. Thus, as to compute the scintillation effect on the signal amplitude and to remove the variations due to other effects, detrending operations that commonly correspond to processing the signal intensity through cascaded low-pass filters are employed. On the other hand, phase scintillation monitoring is achieved by computing the σ_ϕ index corresponding to the standard deviation of the detrended phase measurements [16]:

$$\sigma_\phi = \sqrt{\langle \varphi^2 \rangle_T - \langle \varphi \rangle_T^2}, \tag{2}$$

where φ is the detrended phase measurement that can be obtained by processing the carrier phase measurements through three cascaded second-order high pass filters or Butterworth high-pass filter [16]. Moreover, useful material in the design of filters for phase and amplitude detrending algorithms can be found in [11].

Figure 2a,b show an example of the computed amplitude and phase scintillation indices of GPS L1 signal that belongs to the data collected at Antarctic station on 21 January 2016. In Figure 2b, a sharp increase starting around 12:50 a.m. can be noticed, which indicates that the GPS signal broadcast from PRN-14 satellite experiences strong phase scintillation. On the other hand, in Figure 2a, there is no increase observed in the computed S_4 indices, which could be an indicator for amplitude scintillation. Due to polar location of SANAE IV, phase scintillation statistically occurs more often than amplitude scintillation [31]. Owing to the fact that the occurrence of the ionospheric scintillation depends on the several factors such as solar and geomagnetic activity, geographic location, the season of the year, and local time [15], it is not straightforward to model the occurrence of the event, and statistical analyses have been exploited in order to characterize the intensity, duration, and occurrence frequency of the scintillation events observed in different locations [31,32].

Figure 3a,b show an example of the computed amplitude and phase scintillation indices of GPS L1 signal that is broadcast from the PRN-20 satellite. It is seen that both amplitude and phase scintillations occur at the same time in this data that were collected in Hanoi on 16 April 2013. In the scintillation events observed at the equatorial region, typically both phase and amplitude scintillations occur with faster and deeper signal power fadings that take longer durations [32].
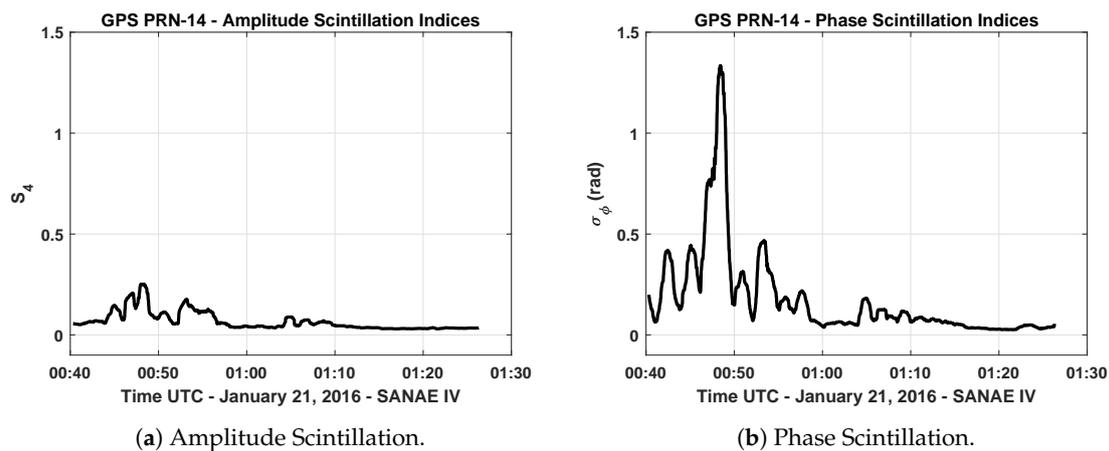
(**a**) Amplitude Scintillation.　　　　　　　　　　　(**b**) Phase Scintillation.

Figure 2. Scintillation index values of GPS L1 C/A PRN-14 signal—21 January 2016 (SANAE IV).

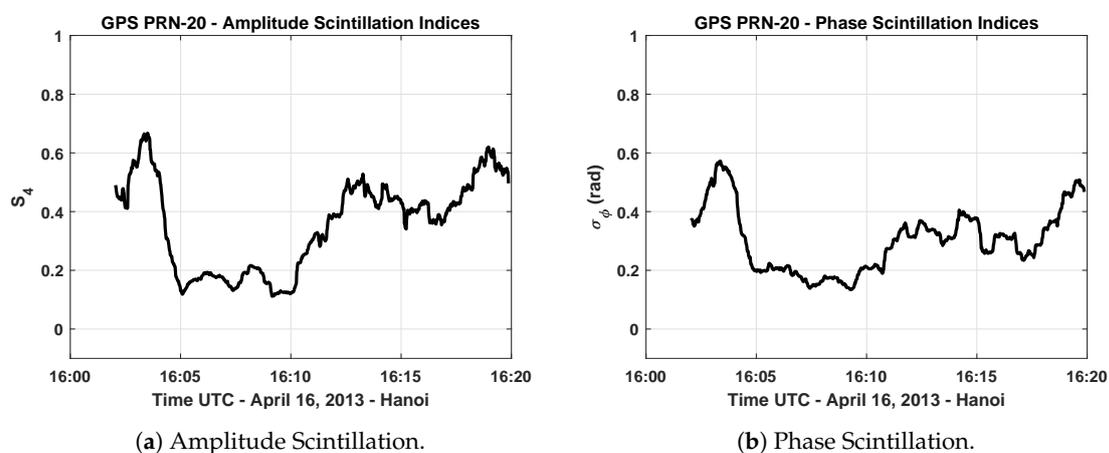(**a**) Amplitude Scintillation.　　　　　　　　　　　(**b**) Phase Scintillation.

Figure 3. Scintillation index values of GPS L1 C/A PRN-20 signal—16 April 2013 (Hanoi).

After having analyzed the scintillated data, the detection of the scintillation and the test results will be explained in the following sections.

3. Overview of Support Vector Machines Algorithm

An SVM algorithm classifies the data by finding the best hyper-plane that separates all the data of one class from those of the other class [33]. Figure 4 shows a pictorial example of two data sets that can be separable into two classes. However, as it can be seen in Figure 4a, there could be an infinite number of separating hyper-planes. The classes can be separated by the linear boundaries as well as nonlinear boundaries. SVM approaches this problem through the concept of the smallest distance between the decision boundary and any of the data samples [34]. In Figure 4b, as an example of linear classifiers, the best hyper-plane that corresponds to the one providing the largest margin between the classes is depicted. The margin is the maximum width of the slice, parallel to the hyper-plane that has no data samples within [33]. The data samples that are closest to the separating hyper-plane are called support vectors as shown in Figure 4b.

In SVM linear classification, the idea is to take the projection of an unknown vector x_i along vector $\bar{\omega}$, which has to be perpendicular to the decision boundary medium (e.g., it has to be in the third dimension if the decision boundary is spanned in two dimensions), and to check whether it crosses the boundary or not in order to decide the classification. The implementation starts with the derivation of the optimal hyper-plane of SVM.
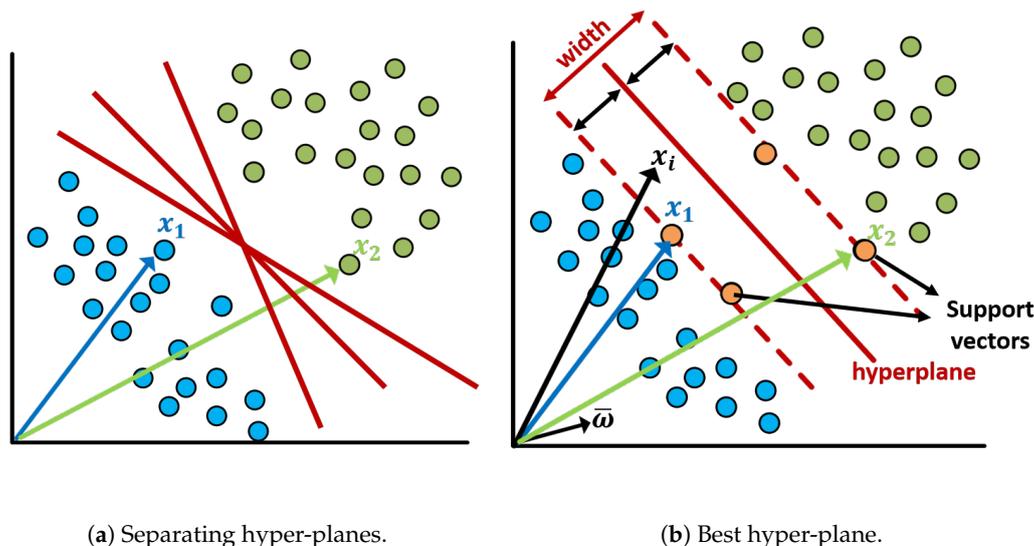
(**a**) Separating hyper-planes.　　　　　　(**b**) Best hyper-plane.

Figure 4. An overview sketch of Support Vector Machines (SVM) algorithm linear classifier.

3.1. Derivation of the Optimum Hyper-Plane

With the given input data-set $\{x_1, x_2, ..., x_N\}$ where $x_i \in R^n$, the two-class data classification problem using linear models are written in the following form [34]:

$$y_i = f(x_i) = \omega_0^T x_i + b_0, \tag{3}$$

where $\omega_0^T x$ corresponds to the projection of x over the direction spanned by ω_0. ω_0 and b_0 are parameters of the optimum hyperplane [20]. Corresponding target values $\{t_1, t_2, ..., t_N\}$ to the input values are decided according to y_i:

$$class(x_i) = \begin{cases} C_1, & \omega_0^T x_i \geq -b_0 & \rightarrow & t_i = +1, \\ C_2, & \omega_0^T x_i < -b_0 & \rightarrow & t_i = -1, \end{cases} \tag{4}$$

where $t_i \in \{-1, +1\}$ defines the class labels. If all the points are classified correctly, it leads to $t_i y_i > 0$ for all i. The distance of x_i from the hyperplane is computed as [35]

$$d(x_i) = \frac{|f(x_i)|}{\|\omega\|} = \frac{t_i \left(\omega_0^T x_i + b_0\right)}{\|\omega\|}. \tag{5}$$

Up to this point, there are not enough constraints to fix specific values for b and ω. Therefore, in order to define a decision rule and have an optimum hyper-plane, the parameters should be optimized by considering the maximum width for the margin that can be obtained by solving [34]

$$\underset{\omega_0, b_0}{\arg\max} = \left\{ \frac{1}{\|\omega\|} \min_i \left[t_i \left(w^T x_i + b \right) \right] \right\}. \tag{6}$$

The direct optimization problem would be very complex and this constrained optimization problem is solved by using Lagrange Multipliers [34]. The cost function is written as

$$L = \frac{1}{2} \|\omega\|^2 - \sum_{i=1}^N a_i \left\{ t_i \left(\omega^T x_i + b \right) - 1 \right\}, \tag{7}$$

where $a = (a_1, a_2, ... a_N)^T$ are Lagrangian multipliers with $a_i \geq 0$. The solutions are saddle points of the cost function L. There is a minus sign before the multipliers because saddle points are obtained by

minimizing with respect to ω and b and maximizing with respect to a [20,34]. The derivatives of the cost function with respect to ω and b are then set to zero:

$$\frac{\partial L}{\partial \omega} = 0 \rightarrow \omega_0 = \sum_{i=1}^{N} \alpha_i t_i x_i, \tag{8}$$

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^{N} \alpha_i t_i = 0. \tag{9}$$

Replacing Equations (8) and (9) in Equation (7) leads to the dual representation of the maximum margin problem [34]

$$\tilde{L}(a) = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} a_i a_j t_i t_j \kappa(x_i, x_j) \quad \text{subject to} \tag{10}$$

$$a_i \geq 0, \quad i = 1, ..., N, \tag{11}$$

where $\kappa(x_i, x_j) = x_i^T x_j$ is kernel function. A necessary and sufficient condition for a function $\kappa(x_i, x_j)$ to be a valid kernel, the function $\kappa(x_i, x_j)$ should be positive semi-definite for all the possible choices, and this also ensures that the Lagrangian function $\tilde{L}(a)$ is bounded [34]. The Lagrange multipliers are obtained by solving

$$\hat{a} = \arg\max_{a} \tilde{L}(a), \tag{12}$$

with respect to a subject to the constraints

$$\hat{a} \geq 0, i = 1, 2, ..., N, \tag{13}$$

$$\sum_{i=1}^{N} \hat{a}_i t_i = 0, \tag{14}$$

where all the points that have $\hat{a} \neq 0$ are on the margin; in other words, they are support vectors [24]. For all the points that do not lie on the margin, the corresponding Lagrange multiplier is $\hat{a} = 0$. Now, y_i, defined by Equation (3), can be evaluated as

$$y_i = f(x_i) = \sum_{j=1}^{N} \hat{a}_j t_j \kappa(x_i, x_j) + b. \tag{15}$$

The optimization of this form satisfies the following Karush–Kuhn–Tucker (KKT) conditions that are both necessary and sufficient for optimum SVM solution [33–35]:

$$\begin{aligned} \hat{\alpha}_i &\geq 0, \ \forall i, \\ t_i y_i - 1 &\geq 0, \ \forall i, \\ \hat{\alpha}_i \{t_i y_i - 1\} &= 0, \ \forall i. \end{aligned} \tag{16}$$

After obtaining the Lagrange multipliers $\hat{a}$ by solving Equation (12), the optimal parameters w_0 and b_0 can be computed by considering the set of support vectors S because any data point for which $\hat{a} = 0$ plays no role in making predictions for the new data points. Having obtained the $\hat{a}$ and support vector x_i values that satisfy KKT conditions and Equation (15), the value of the threshold parameter b can be determined [34]

$$w_0 = \sum_{i \in S} \hat{a}_i t_i x_i, \tag{17}$$

and

$$b_0 = \frac{1}{N_S} \sum_{i \in S} \left(t_i - \sum_{j \in S} \hat{a}_j t_j \kappa(x_i, x_j) \right), \tag{18}$$

where N_S is the total number of support vectors in the set S and averaging over all support vectors provides a numerically more stable solution for b_0 [34]. Eventually, the decision function of SVM is computed as

$$\hat{y}(x_i) = \text{sign}\left(\sum_{j=1}^{N_s} \left[\hat{a}_i t_i \kappa(x_j, x_i)\right] + b_0\right). \tag{19}$$

3.2. Kernel Extension

A kernel is a function κ that for all $x, y \in X$ satisfies [36]

$$\kappa(x, y) = \langle \phi(x), \phi(y) \rangle, \tag{20}$$

where ϕ is a mapping from X to F_κ that is an inner product feature space associated with the kernel κ:

$$\phi : x \in X \rightarrow \phi(x) = \kappa(x, \cdot) \in F_\kappa. \tag{21}$$

Any finite subset of the space X is positive semi-definite, and the kernel function satisfies a positive semi-definite condition as mentioned. Actually, corresponding space F_κ is referred to as Reproducing Kernel Hilbert Space (RKHS), which is a Hilbert space containing the Cauchy sequence limit condition [36]. The theory of reproducing kernels was published by Aronszajn in 1950, and detailed theory can be found in [37]. Moreover, the kernel concept was introduced into the pattern recognition field by Aizerman in 1964 [34].

In most cases, the samples may not be linearly separable. As it is seen in Equation (10), linear kernel can be expressed as

$$\kappa(x_i, x_j) = x_i^T x_j. \tag{22}$$

If the classification problem is not linearly separable, SVM can be powered up by a proper kernel function. The kernel method enables SVM to find a hyperplane in the kernel space and hence nonlinear separation can be achieved in that feature space [24]. An example of nonlinear kernels is Gaussian RBF, which can be written as

$$\kappa(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$
$$= \exp\left(-\gamma \|x_i - x_j\|^2\right), \tag{23}$$

where σ defines the width of the kernel. If the parameter σ is close to zero, SVM tends to over-fitting, which means all the training instances are used as support vectors [24]. Assigning a bigger value to σ may cause under-fitting, leading all the instances to be classified into one class. Therefore, a proper value must be selected for the kernel width. In the same manner, kernel scale parameter corresponds to γ parameter in the RBF definition as being different from the σ representation.

Another most commonly used kernel function is the polynomial that can be represented as

$$\kappa(x_i, x_j) = \left(1 + x_i^T x_j\right)^p, \tag{24}$$

where p is the order of the polynomial kernel. The lowest degree polynomial corresponds to the linear kernel, and it is not preferred in case of having a nonlinear relationship between the features. The degree of the polynomial kernel controls the flexibility of the classifier and higher-degree allows a more flexible decision boundary compared to linear boundaries [38].

The performance of linear, Gaussian, and polynomial kernels are compared in our implementation in the next section.

4. Experimental Tests

In this section of the paper, the implementation and performance analysis of the scintillation detection based on the SVM method that exploits different kernel functions are provided through the collected data.

4.1. Training Data Preparation and Labeling

The preparation of the data is the most important step in the machine-learning implementations. Amplitude and phase scintillation indices have to be put into a format so that the SVM algorithm can detect the scintillation in the correct way. As it is mentioned, it is difficult to model the occurrence of scintillation due to temporal and spatial variabilities of the ionosphere [32]. Statistical analysis has been highly benefited and also chosen fixed period (T) for the computations of S_4 and σ_ϕ indices is quite important, given in Equations (1) and (2). Generally, T is adjusted to 60 s in ISMR receivers. In this case, by considering only one value to feed the algorithm, early detection of the scintillation seems not to be possible. Moreover, according to performed analysis of high-latitude and equatorial ionospheric scintillation events in [32], phase scintillation lasts around 5.6 min at high latitude regions and 10.2 min in the equatorial region. On the other hand, it has been observed that amplitude scintillation events last around 3.1 min at the high-latitude region and 12.4 min in the equatorial region. Therefore, so as to enable early scintillation detection, the training data are put into a format by partitioning the data into three-minutes blocks via a moving time window.

In this work, only two class labels, namely, scintillation and no-scintillation, are assigned as follows:

$$class(x_i) = \begin{cases} C_1 & \{\text{Scintillation}\} & , t_i = 1\,(\sigma_\phi \geq 0.3 \text{ or } S_4 \geq 0.4), \\ C_2 & \{\text{No-Scintillation}\} & , t_i = 0\,(\sigma_\phi < 0.3 \text{ or } S_4 < 0.4). \end{cases} \tag{25}$$

The class definitions in Equation (25) have been decided according to the limit values of σ_ϕ and S_4 indices given in [39,40] that are observed in high-latitude and equatorial scintillation events. The list of the training data segments is reported in Table 1. The data collected on different days at SANAE IV and Hanoi stations are put in the defined structure and labeled according to the class definition above. Figure 5a,b show an example of labeling for both amplitude and phase scintillation data sets.
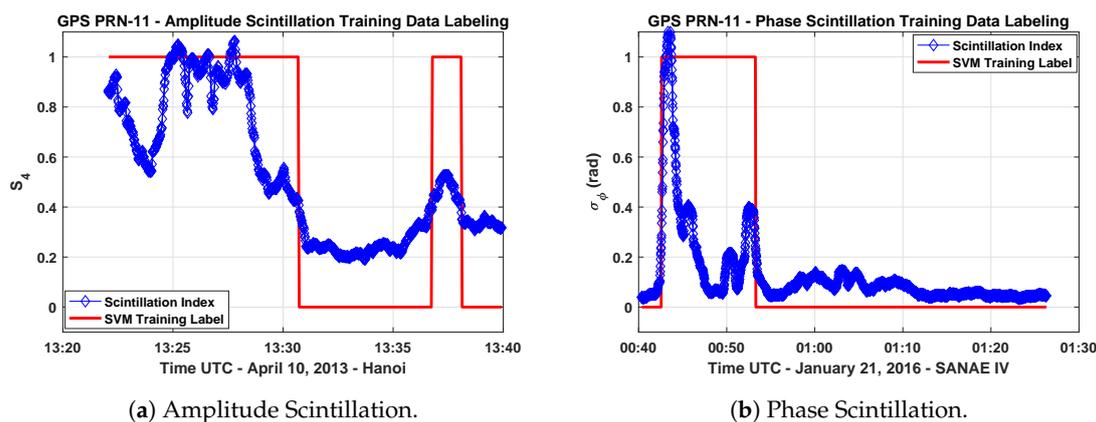
(a) Amplitude Scintillation.   **(b)** Phase Scintillation.

Figure 5. Labeling of the amplitude and phase scintillation index values in the training data sets.

In Figure 5a, S_4 indices computed from the received GPS L1 signal broadcast from the satellite PRN-11 are plotted. It belongs to the data collected on 10 April 2013 in Hanoi. The figure indicates that amplitude scintillation events occurred starting around 1:20 p.m.

Another event that is observed in the data collected on 21 January 2016 at the Antarctic station SANAE IV is analyzed, and the computed σ_ϕ indices are plotted in Figure 5b. The figure shows the phase scintillation event occurred starting from 12:40 a.m. as denoted by the sharp increase in the indices. Although the σ_ϕ values go to values lower than 0.3 around 12:50 a.m., such a time interval

is still considered to be part of the scintillation event and then labeled accordingly. In fact, the data portion between the consecutive scintillation events is affected anyway by a residual scintillation effect that can be observed in the receiver tracking outputs and GNSS measurement observables. Therefore, labeling has been done manually, by inspection, for all the training data sets.

4.2. Cross Validation

After class labels are assigned to the data sets for amplitude and phase scintillation events, SVM methods with different kernel functions are trained. In this section, the performance of validation of the methods is evaluated in terms of ROC curves.

Figure 6 shows an example of the ROC graph. It is a two-dimensional plot of a classifier indexed in one dimension by the false positive rate (FPR) and in the other by the true positive rate (TPR). An ROC graph depicts relative trade-offs that a classifier makes between benefits (true positives) and costs (false positives) [41]. True positive rate (i.e., sensitivity) and the true negative rate (i.e., specificity) are the terms that split the predictive performance of the classifier into the proportion of positives and negatives correctly classified, respectively.
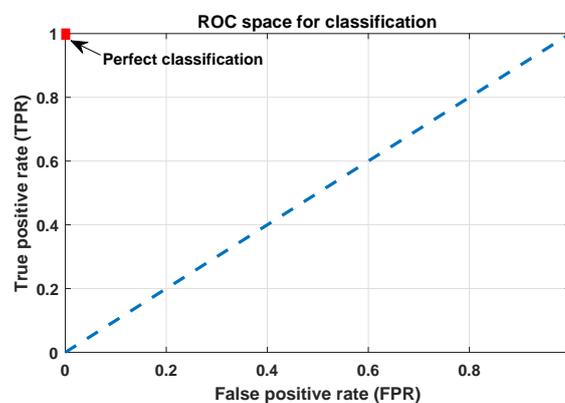

Figure 6. An example of Receiver Operating Characteristics (ROC) space for classification evaluation.

In ROC space, the point $(0, 1)$ represents the perfect classification as shown in Figure 6. The dashed diagonal line represents the case of random assignment of an element to a class. If one point is closer to the upper left corner (i.e., higher TPR, lower FPR), its classification performance is better than another. Classifiers appearing on the left side of the ROC space are called conservative such that they make classification with strong evidence so they have small FPR, but they generally have low TPR [42]. On the other hand, the classifiers on the right side of the ROC space are thought as liberal. They make classifications with low evidence so they classify all the positives correctly with a drawback of high FPR. Each point on the ROC curve represents a trade-off, in other words, a cost ratio [43]. Cost ratio is equal to the slope of the line tangent to the ROC curve at a given point.

SVM implementation for both phase and amplitude scintillation detection has been done by employing MATLAB's statistics and machine learning toolbox (R2016b, The Mathworks, Inc., Natick, MA, USA) [44]. In order to evaluate the performance of the implemented SVM methods, a 10-fold cross-validation technique has been applied. In this technique, the partitions are put into 10 randomly chosen subsets of equal size. Then, each subset is used to validate the model by using the trained remaining nine subsets. This process is repeated 10 times so that each subset is used once for the validation.

Figure 7a,b show the ROC curves of SVM methods with different kernel functions for amplitude and phase scintillation, respectively. In general, it shows that, in an amplitude scintillation case, the performance of the SVM methods is better than the phase scintillation case. However, due to the fact that SVM performance depends on the data sets, we evaluate each scintillation case separately and compare the performance of the kernel functions under the same conditions.
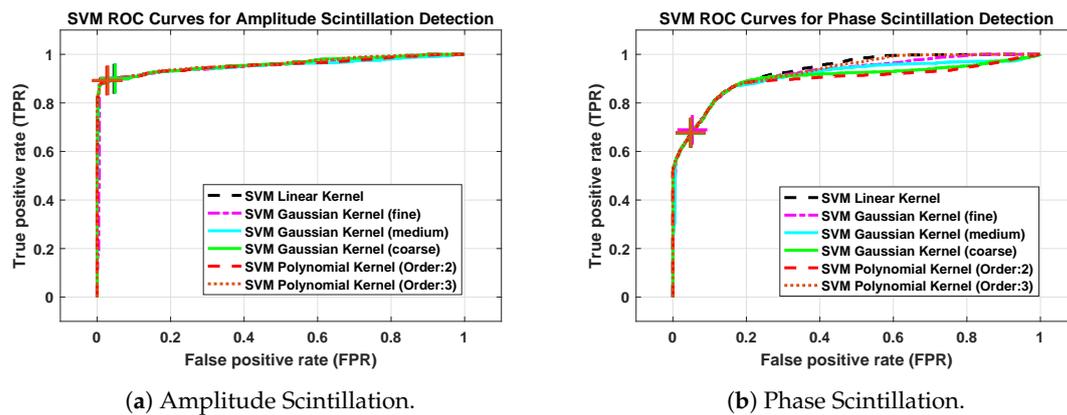
(**a**) Amplitude Scintillation.

(**b**) Phase Scintillation.

Figure 7. Receiver Operating Characteristics (ROC) curves of Support Vector Machines (SVM) methods with different kernel functions.

In the analysis, Gaussian RBF kernel scale parameter γ in Equation (23) is adjusted to different values according to the following assumptions [45]:

$$\begin{aligned}
\gamma_{fG} &= \sqrt{n/4}, \text{ for fine Gaussian,} \\
\gamma_{mG} &= \sqrt{n}, \text{ for medium Gaussian,} \\
\gamma_{cG} &= 4\sqrt{n}, \text{ for coarse Gaussian,}
\end{aligned} \quad (26)$$

where n is the number of features or the dimension size of x_i in Equation (3). Moreover, second and third-order polynomial kernels are included in the analysis. The different colored plus symbols on the figures indicate the operating points of each method in Figure 7. Since the ROC curves are close to each other and it is not easy to differentiate the differences between each method, the results are summarized in Tables 2 and 3.

Table 2. Phase scintillation detection performance comparison in terms of complexity, True Positive Rate (TPR), False Positive Rate (FPR), and Area Under Curve (AUC) under 10-fold cross-validation test.

SVM Method	Kernel Scale	Running Time	Validation Accuracy (%)	Operating Point TPR	Operating Point FPR	AUC (%)
Linear	1	t_p	86.01	0.6772	0.0468	91.98
Coarse Gaussian	6.9	$1.28t_p$	86.29	0.6755	0.0482	90.10
Medium Gaussian	1.7	$1.55t_p$	86.16	0.6751	0.0480	90.85
Fine Gaussian	0.43	$1.70t_p$	85.95	0.6890	0.0530	93.16
Polynomial (Order:2)	1	$1.37t_p$	86.26	0.6768	0.0485	89.38
Polynomial (Order:3)	1	$3.20t_p$	86.04	0.6779	0.0488	92.67

In Tables 2 and 3, we compare the time complexity values that are dedicated time to both training and testing. The complexity of a classifier is divided into two kinds of complexity, namely, time complexity and space complexity [46]. While time complexity deals with the time spent on the execution of the algorithm, space complexity considers the amount of memory used by the algorithm [46]. For example, run-time complexity of linear and RBF kernels differ from each other. While the complexity of the RBF kernel is shown to be $O(n_{SV} \times d)$, which is dependent on the number of support vectors (n_{SV}) and the input dimension (d), linear method has $O(d)$ prediction complexity [47]. Therefore, as it is seen in both Tables 2 and 3, running time increases in the cases of the kernel functions in which the samples are uplifted into higher dimensions, and it also becomes dependent on the number of samples.

The parameter, Area Under Curve (AUC), represents the estimated area under the ROC curve, and it is used as a peformance measure for the machine learning algorithms [48]. AUC is accepted as an indicator for the overall accuracy of the classifier; both Tables 2 and 3 show the importance of kernel

scale parameter. While the overall accuracy of coarse and medium Gaussian kernel SVM methods are less than the linear SVM, fine Gaussian SVM outperforms the linear. Moreover, the third-order polynomial kernel provides improved performance compared to linear, coarse, and medium Gaussian kernel SVMs, but it comes with a cost of increased complexity and time.

Table 3. Amplitude scintillation detection performance comparison in terms of complexity, True Positive Rate (TPR), False Positive Rate (FPR), and Area Under Curve (AUC) under 10-fold cross-validation test.

SVM Method	Kernel Scale	Running Time	Validation Accuracy (%)	Operating Point TPR	Operating Point FPR	AUC (%)
Linear	1	t_a	90.44	0.8990	0.0460	95.37
Coarse Gaussian	6.9	$1.02t_a$	90.72	0.9004	0.0487	95.18
Medium Gaussian	1.7	$1.18t_a$	91.65	0.9004	0.0487	95.19
Fine Gaussian	0.43	$1.33t_a$	91.56	0.9018	0.0378	96.01
Polynomial (Order:2)	1	$1.08t_a$	91.42	0.8920	0.0265	95.61
Polynomial (Order:3)	1	$1.80t_a$	91.56	0.8927	0.0292	95.88

4.3. Tests and Evaluation

In this section of the paper, as a performance cross-check, we also used the collected data which are not included in the training sets to evaluate the performance of scintillation detection methods. Figure 8a,b show the decisions of the different kernel SVM methods for the data sets for both amplitude and phase scintillation.
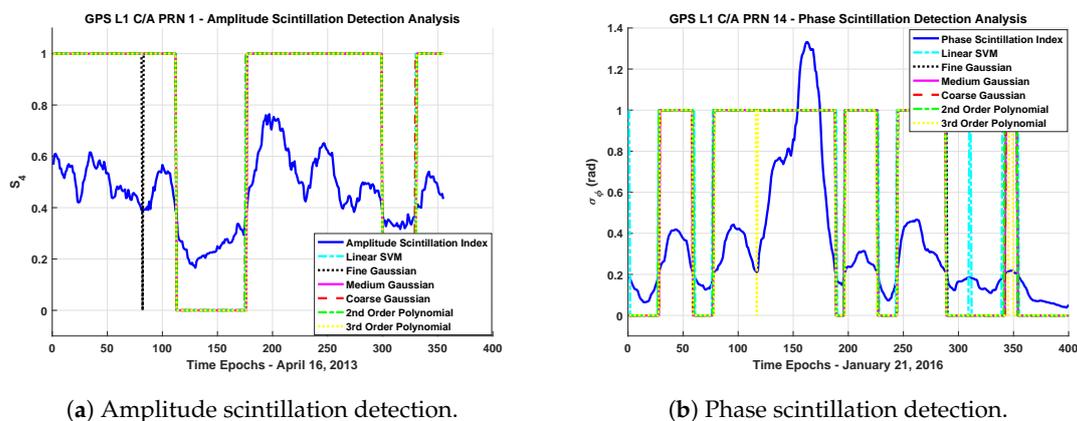

(**a**) Amplitude scintillation detection.　　　　(**b**) Phase scintillation detection.

Figure 8. Scintillation detection results based on Support Vector Machines (SVM) with different kernel functions. "1" corresponds to the points in which the related method points out the scintillation event and "0" means no-scintillation event is detected. Both amplitude and phase scintillation indices synchronized in time can be evaluated as ground truth in the graph to evaluate the performances of different kernels.

In order to evaluate the performance comparison, the confusion matrix technique, which is a two-dimensional matrix indexed in one dimension by the true class of an object and in the other by the class that the classifier assigns [49] as it is seen in Table 4, is applied. A confusion matrix represents the dispositions of a set of instances (i.e., test data) according to a defined classification model that maps the set of instances to predicted classes [42]. Actually, each different point in a ROC curve corresponds to a confusion matrix.

In the two-class data classification problem, the four cells of the confusion matrix correspond to the values of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). By considering the numbers of the measures, this matrix forms the basis for the terms, namely, accuracy, precision, sensitivity, specificity, and error rate. Accuracy is the ratio of the correct predictions

(i.e., the sum of true positives and true negatives) to the total predictions made by the classifier. In the same sense, the error rate is the ratio of the incorrectly classified objects to the total objects. In Table 5, the performance of different kernels for the data sets in terms of accuracy and error rate are summarized.

Table 4. Confusion matrix.

CONFUSION MATRIX		ACTUAL	
		Scintillation	**No-Scintillation**
PREDICTION	Scintillation	True Positive (TP)	False Positive (FP)
	No-Scintillation	False Negative (FN)	True Negative (TN)

As it is expected, the results in Table 5 are consistent with the cross-validation results. With the correct setting of the kernel scale parameter, the fine Gaussian SVM method outperforms in terms of accuracy rate. Furthermore, although the differences in the accuracy of the methods seem not to be at a considerable level, in terms of early detection, the method that provides higher accuracy gains importance. For example, in the cases of the computation rate of the scintillation indices being around one minute, the method having a higher accuracy rate will provide quite advantageous conditions in terms of early detection. However, the third order polynomial kernel provides an improvement in the accuracy compared to the linear kernel, but its performance should be evaluated with increased time and space complexity.

Table 5. Accuracy and error rate performances of different kernel SVMs for scintillation detection

	Phase Scintillation		Amplitude Scintillation	
	Accuracy	**Error Rate**	**Accuracy**	**Error Rate**
Linear	90.98%	9.02%	94.46%	5.54%
Coarse Gaussian	92.61%	7.39%	94.18%	5.82%
Medium Gaussian	92.61%	7.39%	94.18%	5.82%
Fine Gaussian	92.50%	7.50%	95.02%	4.98%
Polynomial (Order:2)	92.28%	7.72%	94.46%	5.54%
Polynomial (Order:3)	93.59%	6.41%	94.74%	5.26%

5. Conclusions

In this paper, we provided performance analysis comparing different kernel methods of SVM and we reviewed and analyzed the linear, Gaussian, and polynomial kernel SVM algorithms for both phase and amplitude scintillation detection. Performance comparison was assessed by exploiting the ROC curves, confusion matrix results, and the performance metrics associated with the confusion matrix. It has been observed that, if the kernel scale parameter of Gaussian RBF kernel SVM algorithm is optimized, the performance of the RBF kernel SVM method outperforms the linear kernel SVM method in terms of overall accuracy. Moreover, although third order polynomial kernel SVM performs better than the linear kernel, it comes with a cost of increased time and space complexity.

For a possible scintillation detection implementation that could be based on unsupervised algorithms in the future, the effect of the multipath has to be considered because the presence of the multipath causes an increase in the amplitude scintillation index values indicating scintillation alarm falsely.

Author Contributions: Conceptualization, F.D.; Formal analysis, C.S.; Funding acquisition, F.D.; Investigation, C.S.; Methodology, C.S. and F.D.; Supervision, F.D.; Writing—original draft, C.S.; Writing—review and editing, F.D.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

GNSS	Global Navigation Satellite Systems
SVM	Support Vector Machines
ROC	Receiver Operating Characteristics
RBF	Radial Basis Function
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative
TPR	True Positive Rate
FPR	False Positive Rate
AUC	Area Under the Curve

References

1. Priyadarshi, S. A review of ionospheric scintillation models. *Surv. Geophys.* **2015**, *36*, 295–324. [CrossRef] [PubMed]
2. Rino, C.L. *The Theory of Scintillation with Applications in Remote Sensing*; IEEE Press, Wiley, Inc. Publication: Hoboken, NJ, USA, 2011.
3. Tatarskii, V.I. *The Effects of the Turbulent Atmosphere on Wave Propagation*; National Technical Information Service: Springfield, VA, USA, 1971.
4. Yeh, K.C.; Liu, C.-H. Radio wave scintillation in the ionosphere. *IEEE Proc.* **1982**, *70*, 324–360.
5. Crane, R.K. Ionospheric scintillation. *IEEE Proc.* **1977**, *65*, 180–199. [CrossRef]
6. Cristodaro, C.; Dovis, F.; Linty, N.; Romero, R. Design of a configurable monitoring station for scintillations by means of a GNSS software radio receiver. *IEEE Geosci. Remote. Sens. Lett.* **2018**, *15*, 325–329. [CrossRef]
7. Kintner, P.M.; Humphreys, T.E.; Hinks, J.C. GNSS and ionospheric scintillation – how to survive the next solar maximum. *Inside GNSS* **2009**, 22–29. Available online: http://www.insidegnss.com (accessed on 15 September 2019).
8. Savas, C.; Dovis, F.; Falco, G. Performance evaluation and comparison of GPS L5 acquisition methods under scintillations. In Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2018), Miami, FL, USA, 24–28 September 2018; pp. 3596–3610.
9. *Fugro Satellite Positioning—The Effect of Space Weather Phenomena on Precise GNSS Applications*; White Paper; Fugro N.V.: Leidschendam, The Netherlands, 2014. Available online: https://fsp.support (accessed on 26 November 2019).
10. Savas, C.; Falco, G.; Dovis, F. A comparative analysis of polar and equatorial scintillation effects on GPS L1 and L5 tracking loops. In Proceedings of the International Technical Meeting of the Institute of Navigation, Reston, VA, USA, 28–31 January 2019; pp. 632–646.
11. Van Dierendonck, A.J.; Klobuchar, J.; Hua, Q. Ionospheric scintillation monitoring using commercial single frequency C/A code receivers. In Proceedings of the 6th International Technical Meeting of Satellite Division of the institute of Navigation (ION GPS 1993), 22–24 September 1993; pp.1333–1342.
12. Romero, R.; Linty, N.; Cristodaro, C.; Dovis, F.; Alfonsi, L. On the use and performance of new Galileo signals for ionospheric scintillation monitoring over Antarctica. In Proceedings of the International Technical Meeting of The Institute of Navigation, Monterey, CA, USA, 25–28 January 2016; pp. 989–997.
13. Curran, J.T.; Bavaro, M.; Morrison, A.; Fortuny, J. Developing a multi-frequency for GNSS-based scintillation monitoring receiver. In Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014), Tampa, FL, USA, 8–12 September 2014; pp. 1142–1152.

14. Peng, S.; Morton, Y. A USRP2-based reconfigurable multi- constellation multi-frequency GNSS software receiver front end. *GPS Solut.* **2013**, *17*, 89–102. [CrossRef]

15. Linty, N.; Dovis, F.; Alfonsi, L. Software-defined radio technology for GNSS scintillation analysis: Bring Antarctica to the lab. *GPS Solut.* **2018**, *22*, 96. [CrossRef]

16. Curran, J.T.; Bavaro, M.; Fortuny, J. Developing an Ionospheric Scintillation Monitoring Receiver. *Inside GNSS* **2014**, 60–72. Available online: http://www.insidegnss.com (accessed on 15 September 2019).

17. Curran, J.T.; Bavaro, M.; Fortuny, G.J.J. An open-loop vector receiver architecture for GNSS-based scintillation monitoring. In Proceedings of the European Navigation Conference (ENC-GNSS), Rotterdam, The Netherlands, 15–17 April 2014; pp. 1–12.

18. Linty, N.; Dovis, F. An open-loop receiver architecture for monitoring of ionospheric scintillations by means of GNSS signals. *Appl. Sci.* **2019**, *9*, 2482. [CrossRef]

19. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]

20. Jiao, Y.; Hall, J.J.; Morton, Y.T. Automatic equatorial GPS amplitude scintillation detection using a machine learning algorithm. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 405–418. [CrossRef]

21. Linty, N.; Farasin, A.; Favenza, A.; Dovis, F. Detection of GNSS ionospheric scintillations based on machine learning decision tree. *IEEE Trans. Aerosp. Electron. Syst.* **2019**, *55*, 303–317. [CrossRef]

22. Jiao, Y.; Hall, J.J.; Morton, Y.T. Performance evaluation of an automatic GPS ionospheric phase scintillation detector using a machine-learning algorithm. *NAVIGATION: J. Inst. Navig.* **2017**, *64*, 391–402. [CrossRef]

23. Bousquet, O.; Herrmann, D.J.L. On the complexity of learning the kernel matrix. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Vancouver, BC, Canada, 9–14 December 2002; pp. 399–406.

24. Liu, Z.; Zuo, M.J.; Zhao, X.; Xu, H. An analytical approach to fast parameter selection of Gaussian RBF kernel for support vector machine. *J. Inf. Sci. Eng.* **2015**, *31*, 691–710.

25. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and Other KERNEL-Based Learning Methods*; Cambridge University Press: Cambridge, UK, 2014.

26. Asa, B.H.; Weston, J. A user's guide to support vector machines. *Methods Mol. Biol.* **2010**, *609*, 223–239.

27. Hsu, C.-W.; Chang, C.-C.; Lin, C.-J. *A Practical Guide to Support Vector Classification*; Technical Report; Department of Computer Science, National Taiwan University: Taipei City, Taiwan, 2003.

28. Savas, C.; Dovis, F. Comparative performance study of linear and gaussian kernel SVM implementations for phase scintillation detection. In Proceedings of the IEEE International Conference on Localization and GNSS (ICL-GNSS), Nuremberg, Germany, 4–6 June 2019; pp. 1–6.

29. Linty, N.; Romero, R.; Dovis, F.; Alfonsi, L. Benefits of GNSS software receivers for ionospheric monitoring at high latitudes. In Proceedings of the 1st URSI Atlantic Radio Science Conference (URSI AT-RASC 2015), Las Palmas, Spain, 16–24 May 2015; pp. 1–6.

30. Alfonsi, L.; Cilliers, P.J.; Romano, V.; Hunstad, I.; Correia, E.; Linty, N.; Dovis, F.; Terzo, O.; Ruiu, P.; Ward, J.; et al. First, observations of GNSS ionospheric scintillations from DemoGRAPE project. *Space Weather* **2016**, *14*, 704–709. [CrossRef]

31. Jacobsen, K.S.; Dahnn, M. Statistics of ionospheric disturbances and their correlation with GNSS positioning errors at high latitudes. *J. Space Weather Space Clim.* **2014**, *4*, 1–10. [CrossRef]

32. Jiao, Y.; Morton, Y.; Taylor, S. Comparative studies of high-latitude and equatorial ionospheric scintillation characteristics of GPS signals. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium—PLANS, Monterey, CA, USA, 5–8 May 2014; pp. 37–42.

33. Gutierrez, D.D. *Machine Learning and Data Science: An Introduction to Statistical Learning Methods with R*; Technics Publications: Basking Ridge, NJ, USA, 2015.

34. Bishop, C.M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*; Springer: Berlin/Heidelberg, Germany, 2006.

35. Laface, P.; Cumani, S. *Support Vector Machines*; Class Lecture; Machine Learning for Pattern Recognition (01SCSIU), Politecnico di Torino: Turin, Italy, 19 March 2018.

36. Shawe-Taylor, J.; Cristianini, N. *Kernel Methods for Pattern Analysis*; Cambridge University Press: Cambridge, UK, 2004.

37. Aronszajn, N. Theory of reproducing kernels. *Trans. Am. Math. Soc.* **1950**, *68*, 337–404. [CrossRef]

38. Asa, B.-H.; Ong, C.S.; Sonnenburg, S.; Scholkopf, B.; Ratsch, G. Support vector machines and kernels for computational biology. *PLoS Comput. Biol.* **2008**, *4*, e1000173.

39. Kinrade, J.; Mitchell, C.N.; Yin, P.; Smith, N.; Jarvis, M.J.; Maxfield, D.J.; Rose, M.C.; Bust, G.S.; Weatherwax, A.T. Ionospheric scintillation over Antarctica during the storm of 5–6 April 2010. *J. Geophys. Res.* **2012**, *117*, A05304. [CrossRef]

40. Moraes, A.O.; Costa, E.; De Paula, E.R.; Perrella, W.J.; Monico, J.F.G. Extended ionospheric amplitude scintillation model for GPS receivers. *Radio Sci.* **2014**, *49*, 315–329. [CrossRef]

41. Provost, F.; Fawcett, T. *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2013.

42. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [CrossRef]

43. Japkowicz, N.; Shah, M. *Evaluating Learning Algorithms: A Classification Perspective*; Cambridge University Press: New York, NY, USA, 2011.

44. The Mathworks, Inc. *MATLAB and Statistics and Machine Learning Toolbox R2016b*; The Mathworks, Inc.: Natick, MA, USA, 2016.

45. Wang, T. Development of the Machine Learning-Based Approach to Access Occupancy Information through Indoor CO_2 Data. Master's Thesis, Delft University of Technology, Delft, The Netherlands, 2017. Available online: http://repository.tudelft.nl/ (accessed on 15 September 2019).

46. Abdiansah, A.; Wardoyo, R. Time complexity analysis of support vector machines (SVM) in LibSVM. *Int. J. Comput. Appl.* **2015**, *128*, 28–34. [CrossRef]

47. Claesen, M.; De Smet, F.; Suykens, J.A.; De Moor, B. Fast prediction with SVM models containing RBF kernels. *arXiv* **2014**, arXiv:1403.0736v3.

48. Bradley, A.P. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit.* **1997**, *30*, 1145–1159. [CrossRef]

49. Ting, K.M. Confusion Matrix. In *Encyclopedia of Machine Learning*; Sammut, C., Webb, G.I., Eds.; Springer: Boston, MA, USA, 2011.