Article

A Cloud-Based Intelligence System for Asian Rust Risk Analysis in Soybean Crops

Ricardo Alexandre Neves [1,2,*,‡] and Paulo Estevão Cruvinel [1,3,*,‡]

1   Post-Graduation Program in Computer Science, Federal University of São Carlos, UFSCar, São Carlos 13565-905, SP, Brazil
2   Federal Institute of São Paulo, IFSP, São João da Boa Vista 13871-298, SP, Brazil
3   Brazilian Agricultural Research Corporation, Embrapa, São Carlos 13561-206, SP, Brazil
*   Correspondence: ricardo.alexandre@ifsp.edu.br (R.A.N.); paulo.cruvinel@embrapa.br (P.E.C.)
‡   These authors contributed equally to this work.

Abstract

This study presents an intelligent method for evaluating the risk of Asian rust (Phakopsora pachyrhizi) based on its development stage in soybean crops *(Glycine max (L.) Merrill)*. It has been designed using smart computer systems supported by image processing, environmental sensor data, and an embedded model for evaluating favorable conditions for disease progression within crop areas. The approach also includes the use of machine learning techniques and a Markov chain algorithm for data fusion, aimed at supporting decision-making in agricultural management. Rules derived from time-series data are employed to enable scenario prediction for risk evaluation related to disease development. Measured data are stored in a customized system designed to support virtual monitoring, facilitating the evaluation of disease severity stages by farmers and enabling timely management actions.

Keywords: Asian soybean rust; machine learning; digital images; pattern recognition; cloud; big data

1. Introduction

Advances in digital agriculture fostered the development of risk management tools [1–4]. Therefore, considering the different applications in the academic field and marketplace, a significant amount of data is required. Prior to its application to different models, this data needs to be stored and handled.

Various models and algorithms are used to collect, process, and transform this data, and the results obtained aid in decision-making. Different decision support systems are available for various agricultural crops [5]. Among these, significant attention has been paid to the soybean crop *(Glycine max (L.) Merrill)*. Soy is considered one of the most important legumes in the world and generally has high protein content and numerous nutrients and bioactive factors beneficial to human life.

In 2024, the global soybean production was equal to 420.762 million metric tons, with Brazil and the United States leading the way. These two countries are consistently the top producers (Figure 1), followed by Argentina, China, and India [6].

However, in all the productive soybean countries, yield losses due to soybean diseases vary between harvests and have been considerable over time. Moreover, in such a context, Asian soybean rust (ASR) is one of the most serious diseases affecting soybean crops

worldwide. It is known for causing premature defoliation, early maturation, and significant yield losses, potentially reaching up to 90% or total damage in crop areas [7].
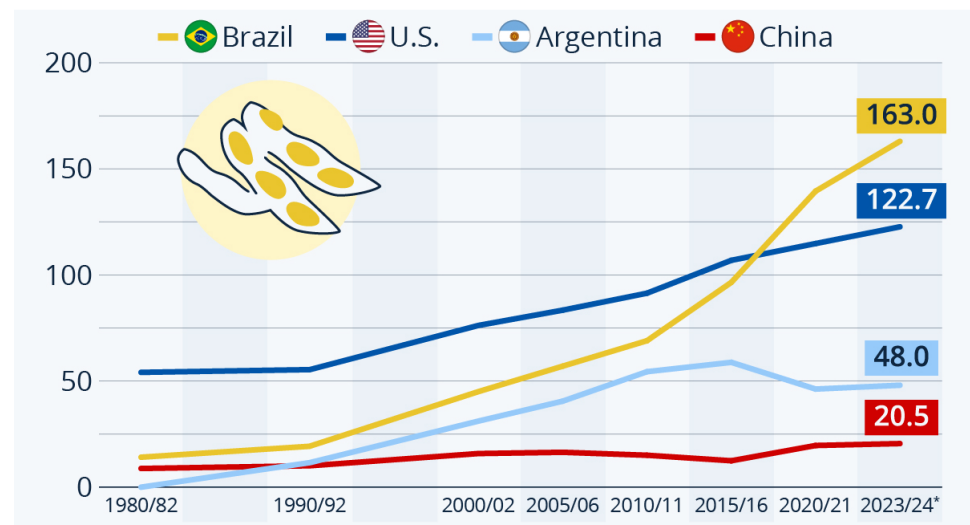

Figure 1. A balance of the harvests related to the world's main soybean-producing countries by year (in millions of tons); forecasts as of June 2023 [8].

In Brazil, according to Godoy and collaborators, in the 1970s, soybeans became an economically important product, and their significance in the global agricultural market has increased ever since. Despite such an opportunity, data from the 2022–2023 harvest periods indicate that ASR occurred in all the soybean-producing regions of the country. In fact, ASR has been reported at different phenological stages of soybean plants, with a predominance of favorable occurrences between reproductive (R) stages, i.e., R4, R5, and R6, denoting the phenological stages [9]. This phenological window typically falls between the 85th and 95th day of the crop's cycle.

Various factors, such as disease spread, climatic conditions, and the influence of other environmental variables, are important for understanding regional severity indices and their direct or indirect impact on losses.

The fungus *Phakopsora pachyrhizi* is the pathogen responsible for ASR [10]. In its initial stage, the disease appears as yellowish or orange spots; in the intermediate stage, these spots expand into larger reddish areas. In the advanced stage, the affected areas become tan, covering large portions of the leaf.

Owing to different climatic conditions, Brazil has a diversity of soybean-growing regions, and 44,062.6 million hectares are currently being used. Thus, making generalized recommendations to control a factor that directly influences the severity of ASR and covers all regions is not possible and solutions must be adaptive and customized. The variables that directly contribute to ASR infection are related to the duration of leaf wetness (6–12 h) at 15–28 °C. Rainfall near the dew point contributes directly to the infection and sporulation of the fungus that causes ASR, accelerating epidemics with regional spread. Ref. [11] reaffirmed that the duration of leaf wetness and night air temperature directly affect the spread of ASR and encouraged the use of methods that can measure or estimate the period of leaf wetness using relative humidity (RH). Similarly, some researchers highlighted rainfall as the leading cause of variation in the severity of ASR epidemics given the correlation between rainfall and disease severity. Lelis et al. used two models to assess conditions favorable to ASR development. One model indicated the number of hours with $RH \leq 90\%$, and the other indicated a dew point depression of <2 °C. In both models, the working temperature range was 18–25 °C, which is considered ideal for the development of the fungus causing Asian rust. Consequently, in Brazil, July and August were identified

as having the least favorable conditions for fungus development, whereas October–April was identified as the period with the most favorable conditions.

According to Bedin [12], plants with nutritional deficiencies are more susceptible to pathogen attacks than adequately nourished ones. Ref. [13] emphasized that models should integrate meteorological data, crop and disease information, and other inoculum sources (e.g., contagion or diffusion), as well as wind direction and speed, temperature, RH, leaf wetness, solar radiation intensity, and crop development stage. Also, mathematical models have been used for predictions of soybean diseases. Researchers used varying parameters, such as epidemiological knowledge and statistical methods [14,15]. In addition, Zagui and co-authors [16] developed a spatio-temporal model based on a fuzzy system to simulate ASR. Their approach integrated input variables into the decision model, including pathogen presence, susceptible plants, and favorable environmental conditions, thereby providing information on the region's vulnerability to the disease. Yu and collaborators [17] introduced a method for recognizing soybean leaf diseases using traditional deep learning models (AlexNet, ResNet18, ResNet50, and TRNet50). They proposed a model based on an enhanced deep learning algorithm, which enabled effective recognition of soybean leaf diseases.

Recent studies have reinforced the role of mathematical models, hyperspectral sensors, and machine learning algorithms in advancing ASR monitoring and control strategies [18–20]. Some authors proposed a mechanistic model based on differential equations to simulate the initial phases of the disease epidemic, incorporating climatic variables and plant characteristics [21–23]. Other authors proposed the DC²Net model, which integrates advanced neural network techniques with hyperspectral imaging, achieving high accuracy in early ASR detection, including asymptomatic stages [24]. In contrast, other studies employed algorithms such as Random Forest (RF) and Support Vector Machine (SVM) to classify disease severity based on spectral data, demonstrating both precision and large-scale applicability [25,26]. Climatic risk assessments were also investigated, as in the study that mapped the Brazilian regions most susceptible to ASR based on historical meteorological data [27]. Complementarily, another applied machine learning techniques to multispectral images obtained via drones to estimate soybean defoliation levels, highlighting the potential of precision agriculture in monitoring symptoms associated with ASR [28]. Likewise, some authors mentioned that digital images, acquired using drones or satellites, can be used to assess severity states [29].

However, in such contexts, relying solely on images—especially those based on partial climatic information—is not sufficient to achieve a complete and precise diagnosis, i.e., in order to reduce and minimize false-positive information.

In fact, based on the literature and state-of-the-art data fusion techniques, it has become possible to observe opportunities to structure a complete rule base that systematically accounts for different situations in which ASR is likely to occur. Then, an intelligent decision support system can also be defined to assist producers in controlling such an important disease problem, including the rational and localized use of fungicide applications. This study aimed to present such a new method for evaluating the stage of favorability of ASR occurrence in a real crop area.

2. Materials and Methods

Accurately diagnosing the potential occurrence and severity of ASR in the field requires the integration of heterogeneous data. In this study, we combined specific climatic data, with patterns observed on digital soybean leaf images and key agronomic parameters (cultivar, plant spacing, and planting period). In such a context, a set of techniques—based on the literature and including advanced computational intelligence and vision

algorithms for data fusion—were developed to support decision-making and operate in a cloud environment.

Thus, as illustrated in Figure 2, in addition to the materials cited below, the techniques employed are as follows. For data storage and analysis, data lake (DL), data warehouse (DW), data mart (DM), relational database (RD), object storage (OS), autonomous database (AD), and extract, transform, load (ETL) were employed. For the computational instances, data science environment (DSE) and analytics cloud service (ACS) were used. For climatic data series interpolator, the cubic spline was used. For image processing, median filtering, segmentation based on histogram equalization and automatic thresholding, clustering based on K-means, feature extraction based on HU moments, Scale-Invariant Feature Transform (SIFT), and Histogram of Oriented Gradients (HOG) were employed. For image pattern classification, principal component analysis (PCA) and SVM were employed. In addition, for data fusion, two different models were evaluated: one of them regarding the state of the art presented in the literature, meaning spatio-temporal modeling and simulation based on fuzzy systems, and one based on hidden Markov chains.
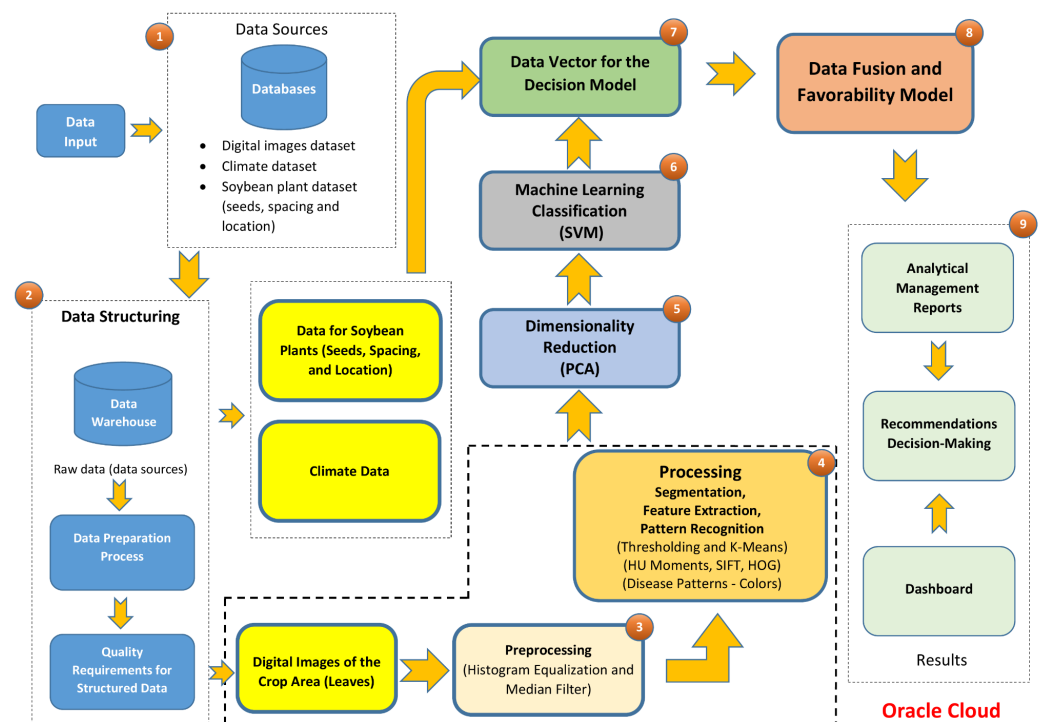
Figure 2. Conceptual diagram.

2.1. Materials

The materials used included a dataset of soybean leaf images collected in a real field crop during cultivation [30], a dataset of climatic data [31], and a dataset containing information on the soybean plant cultivated and used in the experimental pilot [32]. These datasets have the following characteristics:

1. Image dataset: organized according to the protocol established by [33], where soybean leaves were collected from georeferenced plots and imaged under controlled laboratory lighting using a 24-megapixel digital camera. The images were acquired at a 90-degree angle with a 19-centimeter camera-to-leaf distance. The resulting dataset consists of sRGB images showing soybean leaves with various ASR symptoms against a complex background, with dimensions of 4128 × 3096 pixels, a resolution of 12.78 megapixels, and three color channels;

2. Climate data: station name and location; station code; municipality; latitude, longitude; start date; end date; measurement periodicity: daily;
3. Plant data: the crop variety (used the BRS-536), distance between plants and rows, plant height, and number of plants per linear meter.

The primary computing infrastructure, contracted from Oracle Cloud, was configured as follows: a Virtual Cloud Network (VCN) established within a private subnet, contained in a compartment that manages security through policies and security lists. The architecture employs object storage for public data and images in various processing stages. Data processing is conducted by a data science service and a Linux compute instance, with analysis and monitoring provided by the cloud service, which users can access via a Python-based web API and an adequate interface. In addition, it also used a computing infrastructure involving a workstation with the following configurations: x64-based PC architecture; Advanced Micro Devices, Inc. (AMD), Santa Clara, CA, USA, 64-bit processor, 3893 megahertz (MHz); 64 gigabytes (GB) of physical memory; and operating system: Microsoft Windows 10.

2.2. Methods

In relation to the methods, the data source is characterized by input data from public or private sources used in the decision model. These databases may originate from agencies under federal government control, third-sector entities including non-governmental organizations (NGOs), or directly from agricultural producers, provided the relevant variables are measured using sensors.

The daily historical data included the following climatic variables available for public access: total precipitation (mm); maximum temperature ($^\circ$C); minimum temperature ($^\circ$C); compensated average temperature ($^\circ$C); RH (%); and dew point ($^\circ$C).

Data structuring determines the organization of the data from the data source stage. The following components were used for the structuring (Figure 3): (1) different data sources; (2) data lake; (3) data marts; (4) data warehouse; (5) relational database; (6) data preparation; (7) quality requirements; and (8) data vector.
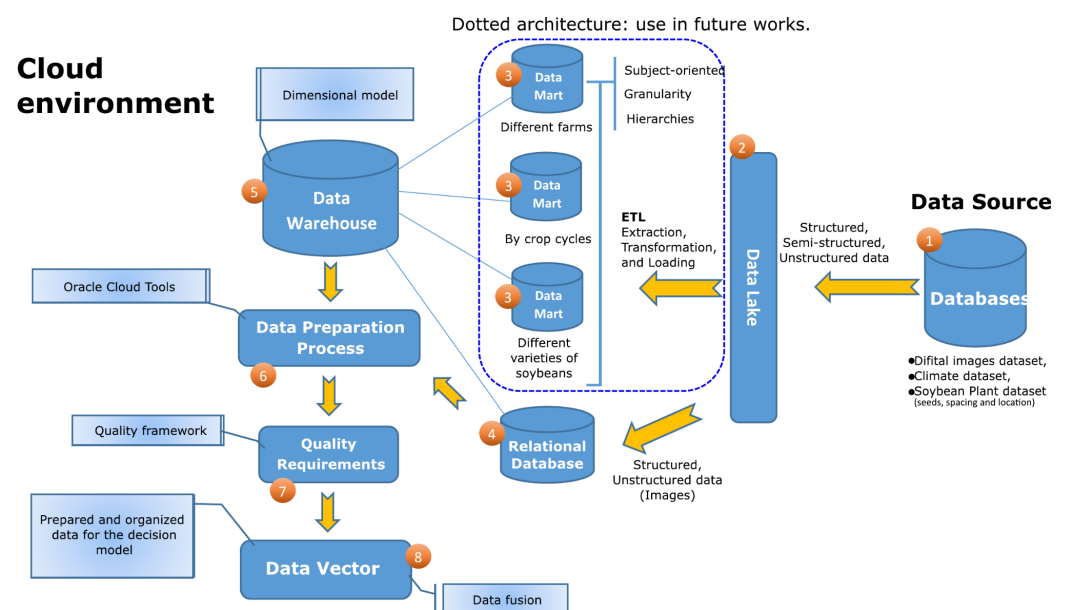

Figure 3. Database structuring diagram.

The infrastructure for organizing the data was planned to meet four possible scenarios: (1) the input of data exported via data marts from legacy systems; (2) the input of semi-structured and unstructured data via data lake; (3) the input of only structured data using data lake and storage in the relational database; and (4) a combination of the three previous scenarios, i.e., the use of input data via data marts and semi-structured, unstructured, and structured data. Algorithm 1 illustrates the steps for structuring the databases in pseudocode.

Algorithm 1: Data Structuring

input : *d*1— climatic data; *d*2—leaf images; *d*3—plant data (seeds, spacing, and location)

output: Data vector

1: $d1 \leftarrow$ climatic data

2: $d2 \leftarrow$ leaf images

3: $d3 \leftarrow$ plant data

4: dimensions $\leftarrow$ integrity, consistency, completeness

5: **procedure** BEGIN

6:    $s1 \leftarrow$ FUNC_RECEIVE_DATA($d1, d2, d3$)

7:    $s2 \leftarrow$ FUNC_DIRECT_DATA($s1$)

8:    $s3 \leftarrow$ FUNC_PREPARE_DATA($s2$)

9:    $s4 \leftarrow$ FUNC_VALIDATE_QUALITY($s3$, dimensions)

10:    $s5 \leftarrow$ FUNC_GENERATE_DATA_VECTOR($s4$)

11: **end procedure**

The complex background of the images constituting the dataset was removed, and the image segmentation technique was automatically used to investigate the color characterization of the disease. In this context, the band-pass thresholding technique (Equation (1)) was used, which consists of selecting a range of threshold values applied uniformly to all the pixels in the image [34,35]. Pixel values within the specified range are assigned to one category, while values outside this range are assigned to another.

$$f(cx, cy) = \begin{cases} 1, & \text{if } LM_{min} \leq I(cx, cy) \leq LM_{max} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $I(cx, cy)$ is the pixel value at position (cx, cy) of the image, LM_{min} is the lower threshold, and LM_{max} is the upper threshold.

For this stage of processing, the following quality indicators of the processed data were considered: image histogram, mean squared error (MSE) metrics (Equation (2)), peak signal-to-noise ratio (PSNR) (Equation (3)), structure similarity index method (SSIM) (Equation (4)), and outliers.

$$\text{MSE} = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} (IA[i,j] - IB[i,j])^2 \tag{2}$$

where *m* and *n* are the width and height of the images, respectively; $IA[i,j]$ and $IB[i,j]$ are the values of the pixels at positions i, j in images IA and IB, respectively.

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{MVP^2}{\text{MSE}}\right) \tag{3}$$

where *MVP* represents the maximum value of a pixel in an image. In images with 8 bits per channel, as is the case with RGB color images, the maximum pixel value is 255; *MSE* is the mean squared error found between the reference and processed images.

$$\text{SSIM}(IA, IB) = \frac{(2 \cdot \mu_{IA} \cdot \mu_{IB} + \alpha_1) \cdot (2 \cdot \sigma_{IAIB} + \alpha_2)}{(\mu_{IA}^2 + \mu_{IB}^2 + \alpha_1) \cdot (\sigma_{IA}^2 + \sigma_{IB}^2 + \alpha_2)} \tag{4}$$

where IA and IB are the two reference and processed images, respectively; μ_{IA} and μ_{IB} are the averages of the pixel values in IA and IB, respectively; σ_{IA} and σ_{IB} are the standard deviations of the pixel values in IA and IB, respectively; σ_{IAIB} is the covariance between the pixel values in IA and IB; α_1 and α_2 are small constants that are added to avoid division by zero and stabilize the calculation, such that $\alpha_1 = (k_1 \cdot L)^2$ and $\alpha_2 = (k_2 \cdot L)^2$, where L is the dynamic range of the pixel values (e.g., 255 for 8-bit images per channel); k_1 and k_2 are predefined constants.

The MSE, PSNR, and SSIM are frequently applied to evaluate image quality in various image-processing tasks. These matrices are used to compare images to meet different sensitivities and degradation contexts irrespective of the consideration of human perception [36–38].

In this study, a pattern recognition technique was used to extract ASR characteristics from soybean leaf images. For this, a set of color descriptors defining the patterns were obtained using the Scale-Invariant Feature Transform (SIFT) technique [39] (Equations (5)–(8)) and HU invariant moment technique [40] (Equations (9)–(20)), and texture descriptors were obtained using the Histogram of Oriented Gradients (HOG) technique [41] (Equations (21)–(25)).

$$G(cx, cy, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(cx^2 + cy^2/2\sigma^2)} \tag{5}$$

$$L(cx, cy, \sigma) = G(cx, cy, \sigma) \star I(cx, cy) \tag{6}$$

where $G(cx, cy, \sigma)$ is the key location function at a point (cx, cy) in the image and at a specific scale (σ). This function represents the response of the Gaussian filter at that position and scale; cx and cy are the horizontal and vertical coordinates, respectively, for calculating the response of the Gaussian filter; σ is the Gaussian scaling parameter that controls the size of the Gaussian filter. The higher the value of σ, the larger the Gaussian filter and the smoother the response. The smaller the value of σ, the sharper the response, but the greater the sensitivity of the details.

$$M_{ij} = \sqrt{(A_{ij} - A_{i+1,j})^2 + (A_{ij} - A_{i,j+1})^2} \tag{7}$$

where M_{ij} is the magnitude of the gradient at position (i, j) of the image, which represents the change in pixel intensities around that position. Next, the magnitude of these differences was calculated using the Pythagorean theorem. A_{ij} is the value of the pixel in position (i, j) of the original image, representing the intensity or color value of the pixel in that position; $A_{i+1,j}$ and $A_{i,j+1}$ are the values of the pixels in the adjacent positions to the right of $(i+1, j)$ and below $(i, j+1)$ the pixel at (i, j), respectively.

$$R_{ij} = ATAN2(A_{ij} - A_{i+1,j}, A_{i,j+1} - A_{ij}) \tag{8}$$

where R_{ij} is the orientation of the gradient at position (i, j) of the image, representing the direction in which the greatest change in intensity occurs in the vicinity of the pixel (i, j); A_{ij} is the value of the pixel at (i, j) of the original image, which represents the intensity or

color value of the pixel in that position; $A_{i+1,j}$ and $A_{i,j+1}$ are the values of the pixels in the adjacent positions, to the right of $(i+1,j)$ and below $(i,j+1)$ the pixel at (i,j), respectively.

As for the geometric descriptors, the two-dimensional, central, and normalized central moments need to be calculated to calculate the seven HU invariant moments [42].

$$mb_{pq} = \sum_{cx=0}^{M-1} \sum_{cy=0}^{N-1} cx^p cy^q f(cx, cy) \tag{9}$$

where $p = 0, 1, 2, \ldots$ e $q = 0, 1, 2, \ldots$ are integers.

$$\mu_{pq} = \sum_{cx=0}^{M-1} \sum_{cy=0}^{N-1} (cx - \overline{cx})^p (cy - \overline{cy})^q f(cx, cy) \tag{10}$$

where $p = 0, 1, 2, \ldots$ e $q = 0, 1, 2, \ldots$ are integers.

$$\overline{cx} = \frac{m_{10}}{m_{00}} \text{ and } \overline{cy} = \frac{m_{01}}{m_{00}} \tag{11}$$

where $\overline{cx}$ and $\overline{cy}$ are the coordinates of the center of mass of the image, $f(cx, cy)$. Along with these central and two-dimensional moments, the other moments that constitute the set of HU invariant moments, given by Equations (12) and (13), were also considered.

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\varsigma}} \tag{12}$$

$$\varsigma = \frac{p+q}{2} + 1 \tag{13}$$

where $p + q = 2, 3, \ldots$

$$\varnothing_1 = \eta_{20} + \eta_{02} \tag{14}$$

where $\varnothing_1$ is the orthogonal invariant that refers to the first invariant moment; η_{20} is the second-order central moment, which is calculated from the image or region of interest (ROI) and represents the dispersion of the distribution of pixels or voxels along the X axis; η_{02} is the second-order central moment, calculated from the image or ROI, and represents the dispersion of the distribution of pixels or voxels along the Y axis.

$$\varnothing_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{15}$$

where $\varnothing_2$ is the second invariant orthogonal to the rotation that refers to a measure of the geometric characteristics of the image or ROI; η_{11} is the second-order central moment between the X and Y axes and represents the covariance between the axes of the distribution of pixels or voxels.

$$\varnothing_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{16}$$

where $\varnothing_3$ is the third orthogonal rotation invariant that refers to the measure of the geometric characteristics of the image or ROI; η_{30} is the third-order central moment along the principal X axis and represents the dispersion of the pixel or voxel distribution along that axis; η_{12} and η_{21} are third-order central moments involving displacement mixtures along the principal X and Y axes and represent the dispersion of the pixel or voxel distribution due to interactions between the axes; η_{03} is the third-order central moment along the Y axis and represents the dispersion of the pixel or voxel distribution along that axis.

$$\varnothing_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \tag{17}$$

where $\varnothing_4$ is the fourth invariant orthogonal to the rotation that refers to the measure of the geometric characteristics of the image or ROI.

$$\begin{aligned}\varnothing_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{18}$$

where $\varnothing_5$ is the fifth invariant orthogonal to the rotation that refers to the measure of the geometric characteristics of the image or ROI.

$$\begin{aligned}\varnothing_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})\end{aligned} \tag{19}$$

where $\varnothing_6$ is the sixth invariant orthogonal to the rotation that describes the geometric characteristics of an image or ROI.

$$\begin{aligned}\varnothing_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]\end{aligned} \tag{20}$$

where $\varnothing_7$ is the seventh invariant orthogonal to the rotation used to describe the geometric characteristics of an image or ROI.

$$\begin{cases} Gr_{cx} = \frac{\partial I}{\partial cx} \\ \\ Gr_{cy} = \frac{\partial I}{\partial cy} \end{cases} \tag{21}$$

where Gr_{cx} and Gr_{cy} represent the derivatives of the image I with respect to the coordinates cx and cy, respectively; the gradients are calculated using Sobel differentiation operators.

$$Mag = \sqrt{Gr_{cx}^2 + Gr_{cy}^2} \tag{22}$$

where Mag is the gradient magnitude calculated by the gradients Gr_{cx} and Gr_{cy}.

$$\Theta = \arctan\left(\frac{Gr_{cy}}{Gr_{cx}}\right) \tag{23}$$

where Θ is the orientation of the gradient, calculated using the arc tangent function (arctan).

$$Hist(\theta) = \sum_{pixels\ per\ cell} w(\theta - \Theta) \tag{24}$$

where $Hist(\theta)$ represents the orientation histogram for a given cell. It is a distribution that shows the number of gradients with orientations in different angular ranges within the cell; θ represents the orientation of the gradient at a given pixel within the cell; Θ represents the predominant orientation of the gradients in the cell. This is often calculated from θ and can be used to weigh the contribution of each pixel to the histogram; $w(\theta - \Theta)$ represents a weighting function that determines the weighing of the contribution of a specific gradient to the histogram based on the angular difference between θ and Θ; $\sum_{pixels\ in\ the\ selected\ cell}$ indicates the sum of all the pixels in the cell.

$$v = \frac{v}{\sqrt{\|v\|_2^2 + \epsilon^2}} \tag{25}$$

where ϑ represents the concatenated vector of orientation histograms in a block; $||\vartheta||_2$ indicates the Euclidean norm (or length) of the vector ϑ, calculated as the square root of the sum of the squares of the vector elements; ϵ is a small constant added inside the square root to avoid possible divisions by zero.

Thus, these descriptors were used to recognize patterns that constitute the image variable for the fusion model, and the recognition of pixels or even clusters of pixels in green, yellow, and brown was considered.

As part of the deliverables of this stage, the process considers feature vectors organized into green, yellow, and brown.

For this stage of processing, the following quality indicators of the processed data were considered: missing values and dimensionality reduction. Missing values were assessed at the point in the process where the feature data vectors were joined.

For the high dimensionality indicator, the dimensionality of the feature vector was reduced to 130 columns.

A machine learning technique was employed to classify the patterns identified in the images, corresponding to each crop leaf. The SVM classifier was applied to process the feature vectors extracted from these patterns. The SVM classifier utilizes functions known as kernels, as shown in Equation (26). A kernel represents abstract spaces and receives two objects, xo_i and xo_j, in the input space to compute their scalar product in the feature space $\Im$, which may reach very high dimensions, where the computational cost factor Φ can be substantial [43–46].

$$K(xo_i, xo_j) = \Phi(xo_i).\Phi(xo_j) \tag{26}$$

For the kernel to represent mappings that facilitate the calculation of scalar products, according to the function defined in Equation (26), the conditions provided by Mercer's theorem were considered, which is characterized by giving rise to semidefinite matrices k, where each element K_{ij} is defined by $K_{ij} = K(xo_i, xo_j)$ for all $i, j = 1, ..., n$, where $\Phi(xo_i)$ and $\Phi(xo_j)$, respectively, represent xo_i and x_j after applying the feature mapping function $\Phi(x)$.

In this study, the SVM technique was applied using grid search to obtain the best hyperparameter configurations using the dataset of characteristics originating from soybean leaves. From the processing being performed in machine learning, metrics are generated for model evaluation. Various statistical indicators and classification metrics were used, which are fundamental for understanding the quality of predictions and the robustness of the model. These indicators allow an analysis ranging from data dispersion to the effectiveness of classifications, providing a comprehensive view of the model's performance. Each metric used is presented individually below.

The machine learning model was evaluated based on the following metrics [47]: variance, standard deviation, precision, accuracy, support and revocation, F1-score, and area under the ROC curve (involving the measures true positive rate (TVP), true negative rate (TFP), and confusion matrix).

The confusion matrix represents the distribution of classifications made by the model, comparing predicted values with actual values, and involves the measures true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Although not a metric itself, it provides the necessary data for calculating key performance metrics such as precision, recall, and F1-score, which, in turn, compose the classification report.

After using the classifier, the dimensionality of the feature vector was reduced owing to the use of the principal component analysis (PCA) technique. PCA is an unsupervised technique for dealing with high-dimensional data and is also known as the Karhunen–Loève transformation [48], Hotelling transformation [49], or singular value decomposition [50].

For this stage of the processing, the following quality indicators of the processed data were considered: accuracy, precision, F1-score, recall (the classifier report), the TP, TN, FP, and FN of the confusion matrix, and the area under the ROC curve.

Algorithm 2 illustrates the steps involved in processing, such as segmentation, pattern recognition and feature extraction, dimensionality reduction, and machine learning in pseudocode.

Algorithm 2: Image Processing

    input : *images*—soybean leaf images; *vetor_dados*—data vector
    output: Data vector with classified images

1 *images* ← soybean leaf images
2 *qi* ← number of images
3 *b1* ← bucket for images without cloud background
4 *b2* ← bucket for segmented images with cloud background
5 *b3* ← bucket for cloud pattern data
6 *c* ← colors: green, yellow, brown
7 *qual_1* ← metrics: PSNR, MSE, SSIM
8 *qual_2* ← metrics: missing values
9 *qual_3* ← metrics: dimensional reduction
10 *qual_4* ← metrics: SVM classifier report
11 **begin**
12   **for** *i in* 1 *to qi* **do**
13   *s1* ← *Func_background_segmentation(imagens)*
14   *b1* ← *s1*
15   **end**
16   **for** *j in* 1 *to* 3 **do**
17   *s2* ← *Func_background_segmentation(b1)*
18   *s3* ← *Func_process_quality(s2, qual_1)*
19   *b2* ← *s2*
20   **end**
21   **for** *k in* 1 *to qi* **do**
22   **for** *l in* 1 *to* 3 *in c* **do**
23   *s4* ← *Func_extract_patterns(b2)*
24   *s5* ← *Func_process_quality(s4, qual_2)*
25   *b3* ← *s4*
26   **end**
27   **end**
28   *s6* ← *Func_dimensionality(b3)*
29   *s7* ← *Func_process_quality(s6, qual_3)*
30   *s8* ← *Func_pattern_classification(b3)*
31   *s9* ← *Func_process_quality(s8, qual_4)*
32   *s10* ← *Func_insert_classification_into_data_vector(s8)*
33 **end**

A structured data vector was adopted for data fusion, combining variables from the climatic time series with those derived from the processing of digital images of soybean crop leaves. When structuring this variable vector (Figure 4), all time-series data are checked for gaps within the ten-day time windows considered for analysis. If any gap is found, data interpolation with a cubic spline is used (Equations (27)–(29)).
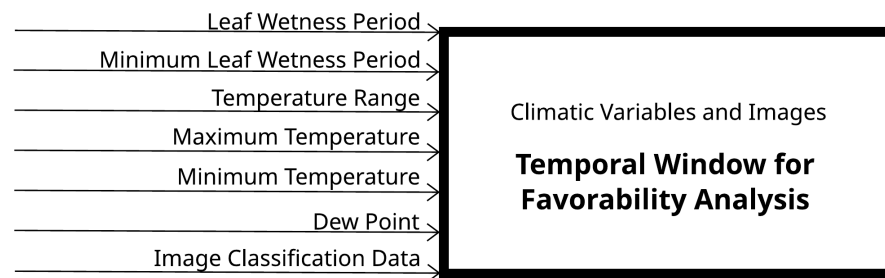
Leaf Wetness Period

Minimum Leaf Wetness Period

Temperature Range

Maximum Temperature

Minimum Temperature

Dew Point

Image Classification Data

Climatic Variables and Images

Temporal Window for Favorability Analysis

Figure 4. Set of variables considered for analysis in a temporal window.

The cubic B-spline is a polynomial function consisting of continuous parts. Therefore, each part is composed of a 3rd degree polynomial in the interval $[xi_{k-1}, xi_k], k = 1, 2, 3, .n$. In addition, it obtains an interpolation formula that is smooth and continuous in the first and second derivatives, respectively, both within an interval and on its boundaries [51].

$$\Gamma(xi) = \sum_{i=0}^{n-1} c_i B_{i,g;t}(\iota) \tag{27}$$

where c_i are the coefficients, g represents the order of the B-spline, t represents the nodes, and $B_{i,g}(\iota)$ is defined by Equations (28) and (29).

$$B_{i,0}(xi) = \begin{cases} 1, & se \quad t_i \leq \iota < t_{i+1} \\ 0, & se \quad \text{otherwise} \end{cases} \tag{28}$$

$$B_{i,k}(xi) = \frac{x - t_i}{t_{i+k} - t_i} B_{i,k-1}(\iota) + \frac{t_{i+k+1} - xi}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(xi) \tag{29}$$

Furthermore, after using the Equations (27)–(29) to complete all the climatic series of data, the rules for decision-making could be established. Such rules describe the set of conditions associated with the favorability definition for ASR occurrences.

2.3. Description of Data Fusion Process

This study compares two data fusion methods for addressing ASR: the first is based on the hidden Markov technique, while the second is a fuzzy logic approach, considered state-of-the-art in the literature [16].

The data fusion process using the hidden Markov chain technique [50,52,53] is based on the integration of variables from different sources and normalized physical quantities, as can be observed from data listed in Table 1.

Table 1. Variables and physical quantities: data fusion.

ID	Description of Variables	Physical Quantity
$V1$	Leaf Wetness Period	Percentage (%)
$V2$	Minimum Leaf Wetness Period	Millimeters (mm)
$V3$	Temperature Range	Degrees Celsius (°C)
$V4$	Maximum Temperature	Degrees Celsius (°C)
$V5$	Minimum Temperature	Degrees Celsius (°C)
$V6$	Dew Point	Degrees Celsius (°C)
$V7$	Image Classification Data	Classification Unit (0 or 1)

In addition, this study considers a general rule base that integrates the main climate data and image patterns recognized from soybean leaves since they can be correlated, enabling risk assessment of disease severity and favorability diagnoses. Table 2 presents the general rule base for the decision-making process.

Table 2. Integral rule base for ASR favorability [54].

Climatic Conditions for Asian Soybean Rust Favorability		
Description	**Variable**	**Estimated Value**
Known Climatological Data		
Leaf Wetness Period	Hours Quantity	Relative humidity greater than or equal to 90%
Dew Point	Temperature	Difference less than 2 °C
Temperature Range Favorable for Fungus Development	Temperature	Range between 18 °C and 25 °C
Minimum and Maximum Temperature during Leaf Wetness Period	Temperature Range	Range between 18 °C and 26.5 °C
Minimum Leaf Wetness Period	Time	6 h
New Presented Data		
Soybean Leaf Cultivar Data	Classification	Pixel analysis
Phenomenology of Asian Soybean Rust Problem	Discovery of Color Classes	Analysis of green, yellow, and brown pixels
Disease Stage Identification	Percentage occurrence of classes	Quantity of pixels for each class
Favorability Probability	Set of variables from indicators	Low, Median, and High

The data fusion based on fuzzy logic, as presented in the literature, is defined by [55]. In such an arrangement, four main types of decision functions are considered as follows [56,57]: (1) Gaussian, (2) trapezoidal, (3) triangular, and (4) singleton. For this development, the triangular function was used as described by Equation (30) given that $L(\chi)$ is a continuous strictly increasing function with $L(a) = 0$ and $L(b) = 1$ and $R(\chi)$ is a continuous strictly decreasing function with $R(b) = 1$ and $R(c) = 0$.

$$\mu_\alpha(\chi) = \begin{cases} 0, & \text{if } \chi < a \\ L(\chi), & \text{if } a \leq \chi \leq b \\ R(\chi), & \text{if } b \leq \chi \leq c \\ 0, & \text{if } \chi > c \end{cases} \tag{30}$$

Additionally, like a discrete universe $\mathbb{X}$, it was defined according to Equation (31), i.e., following Prokopowicz and collaborators [58].

$$\alpha = \sum_{\chi \in \mathbb{X}} \mu_\alpha(\chi)/\chi \tag{31}$$

where $\mu_\alpha(\chi)$ and χ represent the membership degree of the pair object χ, with the "/" symbol denoting the pair separator and $\sum$ representing idempotent summation.

In fact, the concepts of fuzzy logic are organized into a fuzzy model given by the configuration of the antecedent and consequent variables. The formation (if <antecedent> then <consequent>) is used, adhering to conditions that can be fully or partially satisfied, according to the fuzzy inference mechanism, which defines the rule firing. It should also be noted that the rules were constructed according to the Mamdani inference model [59], as

presented in Table 3. These descriptions contain descriptions of the constructed inferences, considering low, medium, and high favorabilities. Additionally, it represents the number of rule combinations generated for each inference.

The combinations arise from the variations, translated by the phenomenological knowledge of the ASR problem, expressed by the seven variables, V1 to V7 (antecedents), that feed the fuzzy model. These are necessarily composed of "OR" and "AND" conjunctions, forming unique rules. Therefore, by summing all combinations of the three favorability possibilities, one may find 120 constructed rules, which comprise the rule base to be submitted to the fuzzy inference engine for data fusion and the support decision method.

Table 3. Fuzzy inferences.

If	Favorability	Combinations
If Favorability is TRUE for up to two variables THEN 1 option: V1 or V2 or V3 or V4 or V5 or V6 or V7	Low	1
If Favorability is TRUE for up to two variables THEN 2 options: V1 or group (V2 or V3 or V4 or V5 or V6 or V7)	Low	8
If Favorability is TRUE for up to four variables THEN 3 options: V1 AND V2 AND group (V3 or V4 or V5 or V6 or V7)	Medium	21
If Favorability is TRUE for up to four variables THEN 4 options: V1 AND V2 AND V3 AND group (V4 or V5 or V6 or V7)	Medium	35
If Favorability is TRUE for more than four variables THEN 5 options: V1 AND V2 AND V3 AND V4 AND group (V5 or V6 or V7)	High	35
If Favorability is TRUE for more than four variables THEN 6 options: V1 AND V2 AND V3 AND V4 AND V5 AND group (V6 or V7)	High	20

Moreover, to have the conditional fuzzy rules defined, in order to reach both the minimum *t-norm* function ($\wedge$) and the maximum *t-norm* function ($\wedge$), as presented by Equations (32) and (33):

$$\alpha \star_T \beta = \min(\alpha, \beta) = \alpha \wedge \beta \tag{32}$$

where α and β are fuzzy variables or sets being combined using the minimum *t-norm* function; $\star_T$ is the *t-norm** operator representing the minimum operation (or minimum *AND*) used to combine the fuzzy sets α and β; $\alpha \wedge \beta$ means the function's output is the minimum membership value between the two sets for a given element of the universe of discourse.

$$\alpha \star_S \beta = \max(\alpha, \beta) = \alpha \vee \beta \tag{33}$$

where α and β are fuzzy variables or sets being combined using the maximum *s-norm* function and can be either a scalar value or a fuzzy set; $\star_S$ is the *s-norm* operator representing the maximum operation (or maximum *OR*) used to combine the fuzzy sets α and β; $\alpha \vee \beta$ means the function's output is the maximum between the membership values of α and β for a given element of the universe of discourse.

Conversely, the defuzzification process consists of calculating a representative numerical output, where $\beta_0 \in \mathbb{Y}$, from the resulting fuzzy set $B'(\beta)$ in $\mathbb{Y}$. Therefore, it involves mapping fuzzy sets from the space $\mathbb{Y}$ to a single numerical value in $\mathbb{Y}$, where $\mathfrak{F}(\mathbb{Y}) \to \mathbb{Y}$. Thus, the numerical result is calculated using the Center of Gravity (COG) method, utilizing Equations (34) and (35) [58].

$$\beta_0 = \frac{\int_{\mathbb{Y}} \beta \mu_{B'}(\beta) d\beta}{\int_{\mathbb{Y}} \mu_{B'}(\beta) d\beta} \tag{34}$$

$$\mu_{B'}(\beta) = \bigvee_{i=1}^{m} \left[F^{(i)}(\chi_0) \wedge \mu_{B^{(i)}}(\beta) \right] \tag{35}$$

where $\mu_{B'}(\beta)$ represents the membership of β to a fuzzy set B'; β is the output variable for which the membership in the fuzzy set B' is calculated; $\bigvee_{i=1}^{m}$ represents the "supremum" or maximum operation to calculate the supreme membership among the m fuzzy sets resulting from the fuzzy inference; i is the index used to iterate from 1 to m through the fuzzy sets participating in the inference; $F^{(i)}(x_0)$ is the membership function of the fuzzy set $\alpha^{(i)}$ with respect to the input variable χ_0, representing the membership of χ_0 to the fuzzy set $\alpha^{(i)}$; $\mu_{B^{(i)}}(\beta)$ is the membership function of the fuzzy set $B^{(i)}$ with respect to the output variable β, representing the membership of β to the fuzzy set $B^{(i)}$.

After the calculation is performed by the defuzzification process, a 5% error rate is computed on the resulting numerical value so that the favorability can be known. The value of the "favorability" consequence ranges from 0 to 100%, which maintains the standard used in the figure of merit approach. Then, the favorability result, given the numerical defuzzification value, is low favorability from 0 to 33.3%, medium favorability from 33.4 to 66.6%, and high favorability from 66.7 to 100%. Algorithm 3, presented below, uses methods from the *Scikit-Fuzzy* library [60].

Algorithm 3: Data Fusion with Fuzzy Logic Approach

Input : Crisp values for leaf wetness, precipitation, temperature, etc.
Output: Favorability: {low, medium, high}

```
/* 1.  Fuzzification Stage                                      */
```
1 Fuzzify all crisp inputs to determine their membership degrees in the relevant fuzzy sets;
```
/* 2.  Inference Stage                                          */
```
2 Let C be the crisp output of the fuzzy inference system;
3 **if** *(leaf_wetness $\geq$ 90%)* $\vee$ `/* ...OR if the next condition is true         */`
4   *(min_wetness $\geq$ 6h $\wedge$ precipitation $\leq$ 25%)* $\vee$ `/* ...OR if the next condition is true  */`
5   *(temp_range $\in [18, 25]$)* $\vee$ `/* ...OR if the next condition is true          */`
6   *(avg_temp $-$ dew_point < 2)* $\vee$ `/* ...OR if the next condition is true         */`
7   *(image_class $= 1$)* **then**
8   Apply inference engine (e.g., Min–Max aggregation) and defuzzify the result to calculate C;
9 **endif**
10 **else**
11     $C \leftarrow 0$ `/* Default to a non-favorable value                 */`
12 **endif**
```
/* 3.  Final Classification                                     */
```
13 **if** $C \leq 33.3\%$ **then**
14   **return** *low*;
15 **else if** $C > 33.3\% \wedge C \leq 66.6\%$ **then**
16   **return** *medium*;
17 **else**
18   **return** *high*;

Finally, system validation was conducted with input from five specialists in agronomy and phytopathology, focusing on soybean diseases and particularly Asian rust. A questionnaire was designed for expert evaluation, including various occurrence scenarios observed in a soybean cultivation area, along with corresponding tables of climatic data and digital

images of the crop leaves. This setup allowed the consulted experts to assess the presence or absence of Asian rust, as well as its severity stage when applicable. Regarding the hidden Markov chain [61], it is a dual stochastic process that has both observable and unobservable components. Hidden Markov chains are an extension of the Markov chains [62,63] defined as a stochastic model $\{X_n, n \in \mathbb{N}\}$ that describes a sequence of events where the probability of a future event depends only on the current state and not on previous states. This Markovian property is expressed as

$$Pr(\zeta_n = \xi_n | \zeta_{n-1} = \xi_{n-1}, \ldots, \zeta_0 = i_0) = Pr(\zeta_n = \xi_n | \zeta_{n-1} = \xi_{n-1}) \tag{36}$$

where $P = p_{(ij)}$ is the transition matrix governing the Markov chain; if ζ_n denotes the state of the Markov chain at time n, then $p_{i,j} = Pr(\zeta_n = j | \zeta_{n-1} = \xi)$; that is, every entry of P satisfies $p_{ij} \geq 0$ and every line of P satisfies $\sum_j p_{ij} = 1$.

For the model developed, Markov chains have discrete states representing the possible conditions or configurations of the combination of variables constituting the input data vector. Each state in the discrete-time Markov chain corresponds to a discrete representation of the system's situation at a given time. According to the probability model, changes in state are referred to as transitions. Transition probabilities describe the likelihood of these transitions between stages of favorability within a given period (time window).

In addition, considering the hidden Markov chains to be characterized by elements N [64] is important. The number of hidden states in the model, denoted by the individual states, is

$$SO = \{so_1, so_2, \ldots, so_N\} \tag{37}$$

where SO represents the set of individual possible states that a Markov chain can assume; $s_1, s_2, \ldots, s_N$ are the individual state variables that constitute the set S, where each s_i represents a specific state in the Markov chain. The subscription i ranges from 1 to N, where N is the total number of states in the Markov chain.

For the result delivery stage, a set of reports was considered. This set was used to construct the decision-making recommendations based on information from management reports and by visualizing the method's processing content available on the user interface via dashboard.

For this processing stage, the quality indicators (accuracy and precision) of the processed data were used to evaluate the outcomes obtained from the application of Markov chains. These quality indicators were based on autocorrelation theory, where expected values were estimated from the observations [65]. Thus, from a time series of N measurements for the Markov process (Equation (38)), e_i represents the configurations generated for the time series, and i is the temporal order measured between observations. The estimator of the expected value for $\hat{\jmath}$ is shown in Equation (39), where the symbol (dash) represents the sample mean.

$$\jmath_i = \jmath_i(e_i), \; i = 1, \ldots, N \tag{38}$$

$$\bar{\jmath} = \frac{1}{N} \sum \jmath_i \tag{39}$$

The autocorrelation function for an observable $\hat{\jmath}$ was defined (Equation (40)) considering the translation invariance in time for the equilibrium of the dataset established in the process. According to Equation (41), the variance of $\jmath$ is a special case of autocorrelation.

$$\hat{C}(t) = \hat{C}_{ij} = \langle (\jmath_i - \langle \jmath_i \rangle)(\jmath_j - \langle \jmath_j \rangle) \rangle = \langle \jmath_i \jmath_j \rangle - \langle \jmath_i \rangle \langle \jmath_j \rangle = \langle \jmath_0 \jmath_t \rangle - \hat{\jmath}^2 \tag{40}$$

where $\hat{C}(t)$ is the autocorrelation value for an observable at a given time t; $\hat{C}_{ij}$ is the autocorrelation value between two variables J_i and J_j; J_i and J_j are the observations of the variables. Each of them can be viewed as a time series of data; $\langle J_i \rangle$ and $\langle J_j \rangle$ are the averages of the time series J_i and J_j, respectively; $\langle (J_i - \langle J_i \rangle)(J_j - \langle J_j \rangle) \rangle$ is a measure of the covariance between the time series; $\langle J_i J_j \rangle$ is the mean of the product of the time series J_i and J_j, representing the raw covariance between the two variables; $\hat{J}^2$ is the mean square of the variable J, i.e., $\langle J \rangle^2$, which represents the variance of J; $\langle J_0 J_t \rangle$ is the average of the product of the variables J_0 e J_t, where J_0 is an observation at a given time and J_t is an observation at a later time t in the time series. It represents the covariance between J_0 and J_t.

$$\hat{C}(0) = \sigma^2(J) \tag{41}$$

where $\hat{C}(0)$ is the value of autocorrelation at $t = 0$ for an observable, i.e., the covariance of the variable J with itself at the same instant of time; $\sigma^2(J)$ represents the variance of the variable J. The variance measures the dispersion of the values of the variable J in relation to its mean, calculated as the average of the squares of the differences between each value and the mean of J.

Another point to consider in the theory [65] is the analysis of self-consistency versus reasonable error. This involves examining the system's equilibrium aspects by evaluating the time series within the context of the Markov chain and monitoring the integrated autocorrelation times obtained from different measurements of J. Equations (42)–(44) define the calculation of the error $\Delta \bar{J}$, the variance of the estimator J, and the integrated correlation time τ_{int}, respectively.

$$\Delta \bar{J} = \sqrt{\sigma^2(\bar{J})} \; with \; \sigma^2(\bar{J}) = \tau_{int} \frac{\sigma^2(\bar{J})}{N} \tag{42}$$

where $\Delta \bar{J}$ is the standard error that measures the uncertainty associated with estimating the sample mean $\bar{J}$; $\sigma^2(\bar{J})$ is the variance that indicates the spread of the values of the sample means relative to the true population mean; τ_{int} represents the integration time or integrated correlation time. This parameter describes the autocorrelation of the data; $\frac{\sigma^2(\bar{J})}{N}$ is the estimate of the variance of the mean $\bar{J}$ based on the sample size N.

$$\sigma^2(\bar{J}) = \sigma^2 \frac{(J)}{N} \left[1 + 2 \sum_{t=1}^{N-1} \left(1 - \frac{t}{N} \right) \hat{\epsilon}(t) \right] com \; \hat{\epsilon}(t) = \frac{\hat{C}(t)}{\hat{C}(0)} \tag{43}$$

where $\hat{\epsilon}(t)$ is the autocorrelation function normalized at $t = 0$, i.e., $\hat{C}(0) = 1$, and measures the autocorrelation of the variable J at different points in time t, normalized with respect to autocorrelation.

$$\tau_{int} = \left[1 + 2 \sum_{t=1}^{N-1} \left(1 - \frac{t}{N} \right) \hat{\epsilon}(t) \right] \tag{44}$$

The use of the variable rule base in Algorithm 4 is innovative. In the current agronomic literature, as previously observed, such variables are generally considered individually. This approach thus enables the simultaneous consideration of all conditions that may lead to the occurrence of Asian rust in soybeans.

After the execution of Algorithm 4, the flow of methodological steps concludes by integrating data structuring, image processing, and the fusion of the involved variables. Each algorithm fulfills a specific function: Algorithm 1 organizes the data; Algorithm 2 performs image processing and pattern extraction; and Algorithm 4 integrates this information with climatic data to identify the risk of Asian rust occurrence. The combination of

these methods enables the analysis of disease favorability within time windows, providing support for decision-making in soybean crop management.

Algorithm 4: Data Fusion with Hidden Markov Chain Approach

input : *v*—data vector; *window*—temporal data window; *rules*—rule base;
            chain—hidden Markov chain

output: Result of occurrence and favorability

  1: *v* ← data vector

  2: *rules* ← rule base

  3: *chain* ← hidden Markov chain

  4: *quality* ← Precision $\sigma(\bar{f})$, Accuracy $\sigma^2(\bar{f})\hat{C}(t)$

  5: **procedure** BEGIN

  6:      *s1* ← FUNC_PROCESS_RULE_BASE(*rules*)

  7:      *s2* ← FUNC_PROCESS_FUSION(*s1*, *chain*)

  8:      *s3* ← FUNC_PROCESS_MARKOV_QUALITY(*s2*, *quality*)

  9:      *s4* ← FUNC_GENERATE_RESULT(*s2*, *s3*)

10: **end procedure**

For data fusion, the evaluation of the two selected models, as described below, was considered by the following metrics: accuracy, precision, and performance.

3. Results and Discussion

Results related to the performance of the computational architecture and the effectiveness of the image processing and classification techniques were obtained. Also, results of the outcomes related to probabilistic modeling using both the fuzzy and hidden Markov models were evaluated for ASR's risk analysis and the system validation.

3.1. Implementation of the Cloud Architecture and Interfaces of the Intelligent System

For the Oracle Cloud platform, a study was conducted considering three possible architecture scenarios. Among them, the most suitable option identified (Figure 5) featured access to both private and public networks, interconnection of object storage components, infrastructure for compute instances, a data science environment, services for analytical data processing, and support for both transactional and multidimensional databases. Additionally, the computational infrastructure for hosting WEB services aimed at user monitoring was also highlighted.

Figure 6 illustrates the resources used for system development, without detailing network configurations, users, and access permissions, which were nonetheless implemented. In this context, a compute instance was also utilized, sized to host the code developed in Python technology, and configured to provide external access via public IP for the data fusion stage through a WEB framework. The compute instance was set up with the Oracle Linux 8.0 operating system, a configuration of 1 OCPU on an AMD architecture, and 16 GB of RAM. Access was established via the SSH protocol using the PuTTY application and 2048-bit public and private key encryption. Additionally, the object storage menu featured buckets, organized according to the processing structure to provide storage for both data source input and processing output. Similarly, appropriate configurations were applied to the Oracle database (Figure 7) for the transactional (relational) and multidimensional (DW) databases, respectively.

The instance configured to process the Python code in the data science environment was prepared using AMD architecture, with four OCPUs, 64 GB of RAM, and a 250 GB disk for storing processing results in the form of a VM.Standard.E3.Flex compute shape.

The technologies provided by Oracle Cloud enabled seamless integration of the architecture modules, which included the data science environments and the Linux Computing Instance. Consequently, the implementation of Python algorithms and their deployment on a web platform were facilitated.

The cloud-based intelligent system for Asian soybean rust risk analysis in soybean crops was designed to present results in a dashboard format. The system's main interface (Figure 8) supported both the fusion stage processing and the visualization of results through a clean, intuitive navigation layout. Accordingly, tabs were positioned at the top of the interface, ensuring clear and organized information display.

The results of the processing performed on the cloud infrastructure were stored in databases when structured or in buckets when unstructured or semi-structured. These were analyzed using the Analytics Cloud Service, which also supplied the decision support system. The analyses were made available for user monitoring via a web interface through the Linux compute instance.
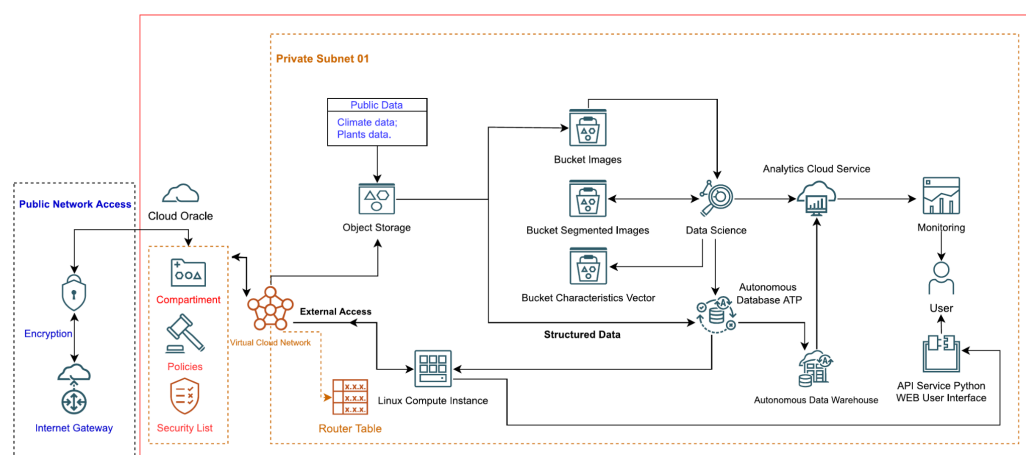
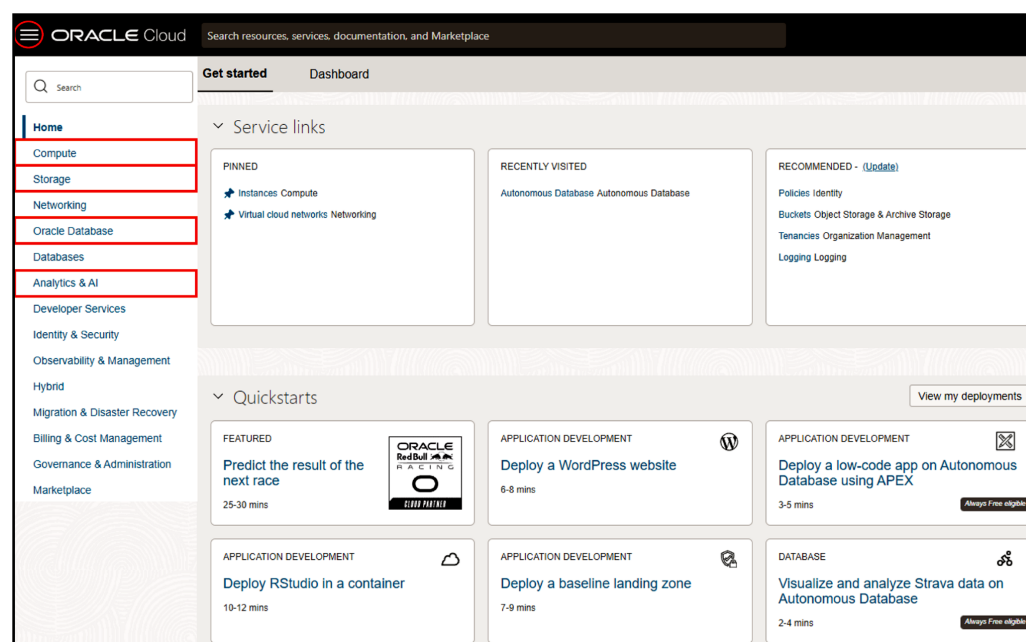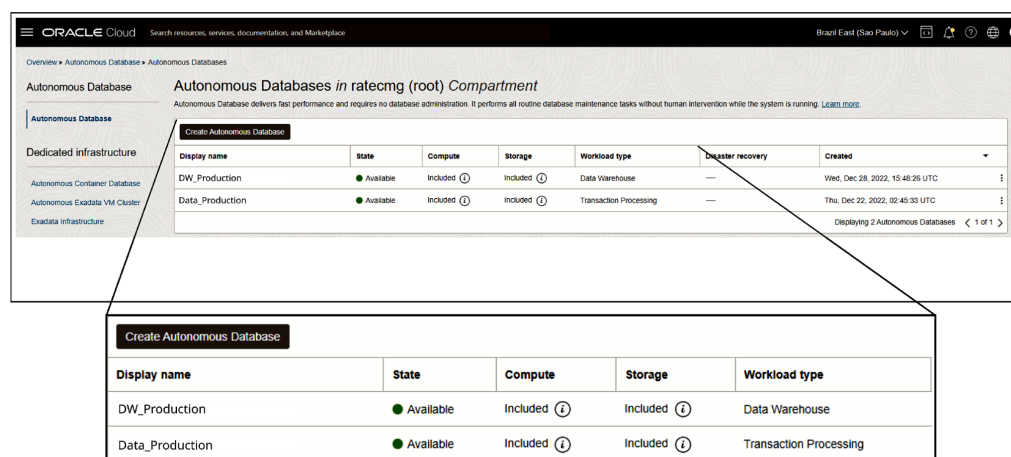**Figure 5.** Oracle Cloud architecture for the intelligent system.

Figure 6. Oracle Cloud architecture for the intelligent system: initial screen.

Figure 7. Oracle Cloud architecture for the intelligent system: autonomous database.

Figure 8. Main interface (input).

Additionally, recommendations for the soybean producer were relationally included based on the favorability results obtained from the processing. Thus, when the result indicates low favorability, a corresponding set of considerations is presented, as also occurs for medium- and high-favorability scenarios (Figure 9).

Further aspects were also taken into account in the recommendation reports, such as the inclusion of a link to the Phytosanitary Pesticide System (Agrofit) for consulting registered fungicide options for disease control, in accordance with the technical recommendations issued by the Brazilian Ministry of Agriculture, Livestock, and Supply (MAPA/Brazil).
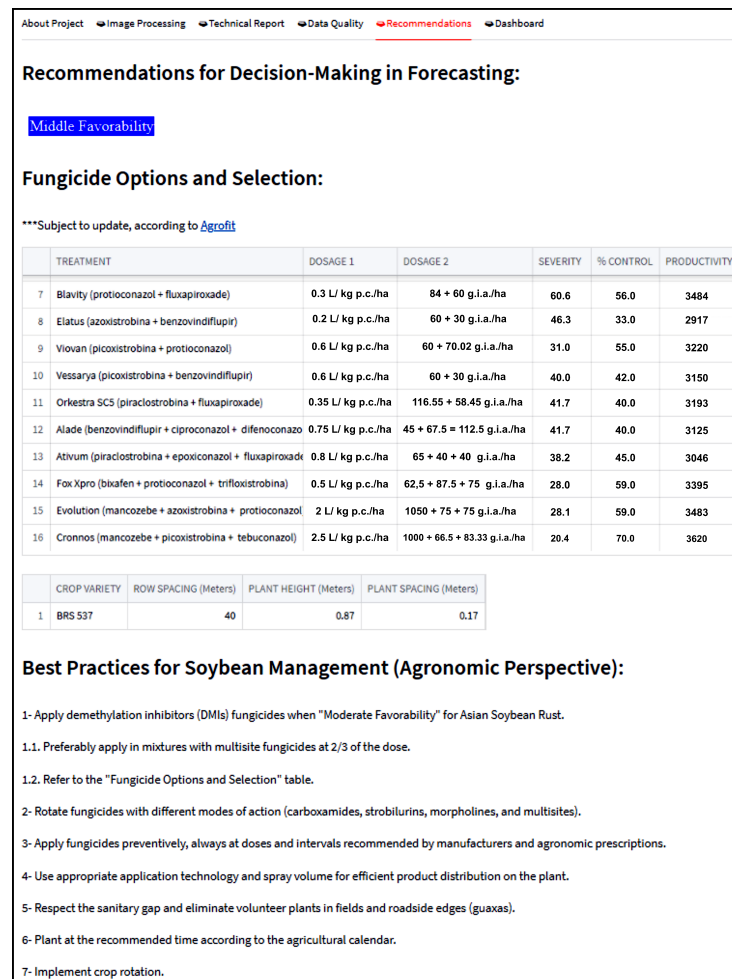
Figure 9. Recommendation interface (output).

3.2. Image Processing and Classification Performance

The results of processing the developed model, based on the established cloud infrastructure, include the organization of climatic data, the processing of soybean leaf images, and the fusion of variables through hidden Markov chains. The processing sequence involves interpolation of the time series, image processing and classification results, dimensionality reduction, and ultimately the assessment of disease favorability based on the generated analytical reports.

During the data reading stage, within the established windows (Table 4), interpolation was required in some cases to fill in missing records. Thus, the records were completed using cubic B-spline interpolation, as shown in one of the analyzed cases (Figure 10), which illustrates the arrangement adopted for organizing and using the time-series data of the variables considered for decision support.

Regarding the used interpolation, it was also observed that the correlation coefficients, obtained with the application of the B-spline function, were of the order of 0.66 for the precipitation data series, 0.78 for the maximum temperature data series, 0.82 for the minimum temperature data series, 0.63 for the relative humidity data series, 0.82 for the dew point data series, and 0.72 for the compensated average temperature data series.

Regarding the processing of leaf images collected in a real field via imaging, a dataset of sRGB images of soybean leaves exhibiting various ASR symptoms and containing complex backgrounds was used for method validation; dimensions: 4128×3096 pixels; resolution: 12,780,288 pixels. Thus, after splitting the RGB channels, the green channel was selected for processing as it exhibited a wavelength closest to the effects expected due to

the potential presence of the rust pathogen. Image histogram techniques were applied to this channel, resulting in the minimization of background effects.

Table 4. Data series temporal window.

N.	Precip.	Max. Temp.	Min. Temp.	Relative Humidity	Dew Point	Comp. Average Temperature	Status
1	4.20	35.50	24.00	72.75	23.08	28.44	Original
2	0.00	32.50	24.40	88.75	23.80	25.80	Original
3	18.00	33.30	22.50	79.00	22.85	26.80	Original
4	0.00	33.00	23.20	84.00	22.62	25.52	Original
5	0.00	33.60	23.80	88.25	24.02	26.12	Original
6	3.00	34.50	23.40	83.00	23.06	26.18	Original
7	0.00	33.50	24.00	84.25	23.47	26.34	Original
8	4.20	35.50	24.00	72.80	23.10	28.40	Interpolated
9	6.10	32.80	24.90	88.70	23.80	25.90	Interpolated
10	5.40	32.60	23.90	86.80	23.70	26.00	Interpolated

Next, a median filter with a 3×3 window was applied to smooth the image for better feature extraction. After this step, a highlight, as an automation point of the process, was the identification of the seed pixel, according to the disease reference colors and the threshold definition process (Figure 11), using statistical techniques such as median calculation, standard deviation, and outlier removal, considering a maximum associated error $\leq 5\%$.
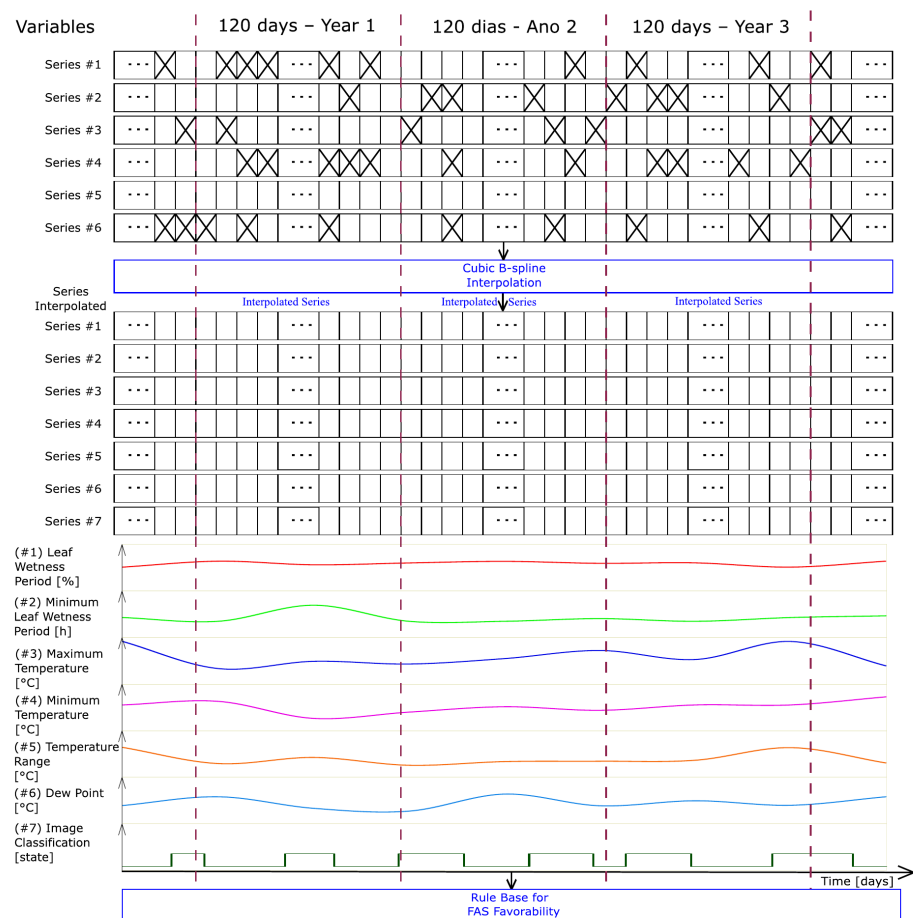

Figure 10. Arrangement of the time series of data from the set of variables for the decision support system.

The choice in thresholds involved analyzing image histograms and evaluating regions to segment the object of interest. The background exhibited a significant number of colors similar to those of the object of interest, i.e., the leaf. The adopted procedure for histogram evaluation was supervised, aiming to identify two thresholds capable of segmenting the largest possible background area without compromising the leaf region, which, due to ASR, displayed a variety of color tones. The histogram analysis focused on six different ranges: (a) 0 to 85, (b) 31 to 165, (c) 70 to 159, (d) 83 to 159, (e) 100 to 130, and (f) 18 to 200. Tests conducted with ranges (b) and (f), like the others, resulted in substantial pixel loss in the object of interest. The threshold range (b), from 31 to 165, yielded favorable results and was adopted as the standard for processing the image dataset.
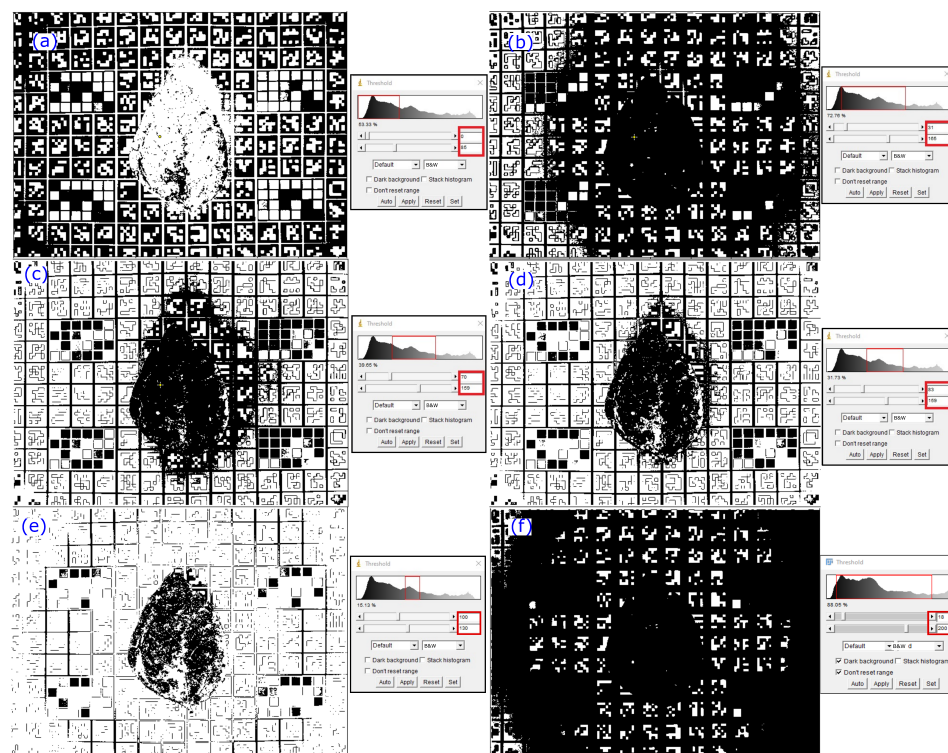
Figure 11. Examples of results obtained based on differents threshold selection, where (**a**) $0 \leq$ *threshold values* ≤ 85, (**b**) $31 \leq$ *threshold values* ≤ 165 (**c**) $70 \leq$ *threshold values* ≤ 159, (**d**) $83 \leq$ *threshold values* ≤ 159, (**e**) $100 \leq$ *threshold values* ≤ 130, (**f**) $18 \leq$ *threshold values* ≤ 200.

The result of applying segmentation techniques (Figure 12) was organized in stages, i.e., first removing the background and then associating the result based on the reality of the absence, appearance, and presence of the disease in a soybean cultivation area, in other words, considering segmentation in green, yellow, and brown colors, respectively.

Additionally, after thresholding, the k-means technique (Figure 12d) was applied to cluster the image pixels according to the established color class definitions. These results indicated the need to consider up to six different labeled clusters, with label number four being used as it was indeed associated with identifying the occurrence of ASR, both in its intermediate and advanced stages.

Regarding the quality metrics of all the images analyzed: the MSE values (Equation (2)) ranged from 0.01 to 0.06, with a median of 0.03; the SSIM values (Equation (4)) ranged from 0.87 to 0.97, with a median of 0.94; the PSNR values (Equation (3)) ranged from 18.98 to 20.04, with a median of 14.29.

For the ranges of pixel values of an ROI related exclusively to green, yellow, and brown colors, the values summarized in Table 5 were observed for these metrics.

Figure 12. An example of results obtained with the application of segmentation technique, where (**a**) is an RGB original image, (**b**) is the green band from the original one, (**c**) is the histogram processing equalization result, (**d**) are the processed labels from 0 to 5, related to the output results obtened after equalization, also showing into a red retangle the selected one, (**e**) is the selected label, (**f**) is the segmented values related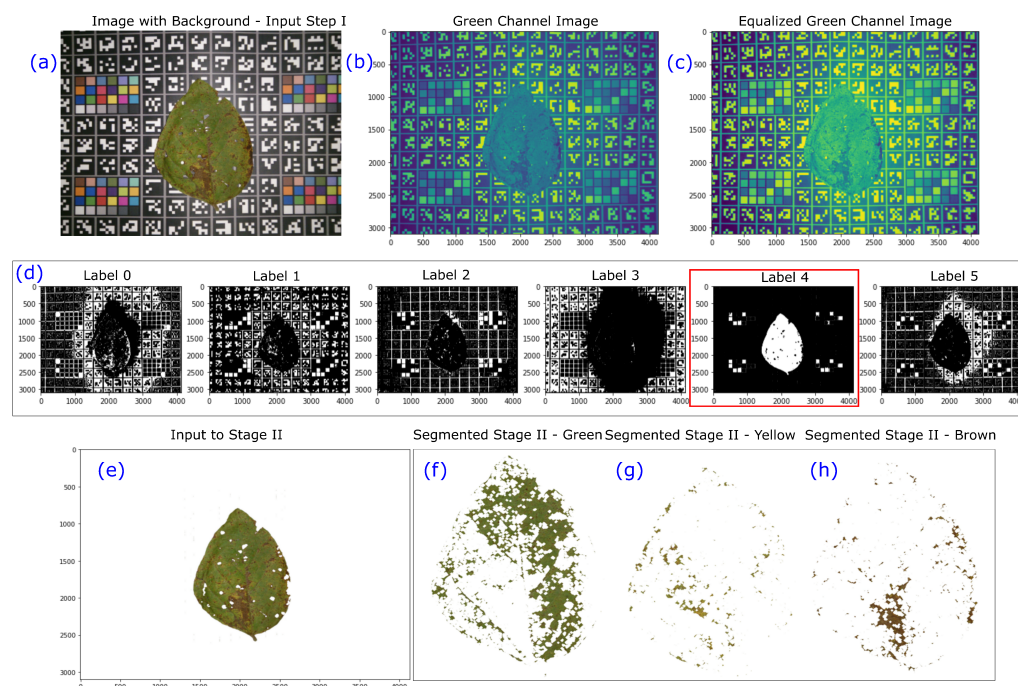 to green pixels, (**g**) is the segmented values related to yellow pixels, (**h**) is the segmented values related to brown pixels.

Table 5. Segmentation quality analysis: metrics and outliers.

Segmented Images	Metrics			Outliers	
	MSE	PSNR (dB)	SSIM	Seeds	Calculation
Green	0.05	13.35	0.91	0	3
Yellow	0.06	12.59	0.91	14	14
Brown	0.05	12.94	0.91	1	1

The OpenCV and Skimage libraries were used to extract the features and recognize the patterns. This was achieved by applying SIFT (Equations (5)–(8)), HOG, and HU moments (Equations (9)–(20)), with algorithms written in Python 3.6.8., based on the default parameters of these libraries. Each process generated a file with the characteristics of each color, and its storage was considered in the Oracle Cloud bucket.

For example, part of the processing can be observed in Figure 13, which illustrates the results for texture, color, and geometric shape.

The processing of these features using the HOG, SIFT, and HU invariant moment algorithms resulted in vectors with 130 features. The PCA technique was then used to reduce this vector to that with five features, as shown in Figure 14.

The choice in the ideal number of principal components was based on the total variance (Table 6), adopting a minimum threshold of 70% explained variance as the criterion. To ensure an efficient representation of the information in the feature vectors, nineteen principal components were sufficient to explain 70.79% of the total variance. In contrast, reducing to eighteen components explained 69.56%, which is slightly below the established threshold.

Based on PCA dimensionality reduction, different classifiers were evaluated (Decision Tree, K-Nearest Neighbor, Naïve Bayes, and Support Vector Machine (SVM)), with the latter selected for yielding the best results. For SVM, three kernels were tested, chosen

according to the behavior of the data to be classified: linear, polynomial, and RBF. These configurations are shown in Table 7.

	Hog	SIFT_0	SIFT_1	SIFT_2	⋯	SIFT_126	SIFT_127	HU
Yellow Color	0.264443670	0.0	9.0	32.0	⋯	5.0	18.0	3.16163556
	0.124659943	0.0	0.0	1.0	⋯	7.0	20.0	7.42741412
	0.670443637	0.0	0.0	1.0	⋯	2.0	4.0	15.2891546
	0.670443637	135.0	44.0	21.0	⋯	0.0	0.0	14.82113771
	0.124659943	11.0	19.0	116.0	⋯	0.0	23.0	−29.88527428
	0.260114127	0.0	0.0	0.0	⋯	6.0	0.0	−18.54458951
	0.122618977	0.0	0.0	0.0	⋯	0.0	6.0	30.57225104
Brown Color	0.223057618	38.0	80.0	21.0	⋯	1.0	4.0	3.15794589
	0.505066400	0.0	0.0	0.0	⋯	2.0	2.0	7.42023486
	0.803366993	1.0	0.0	1.0	⋯	0.0	0.0	15.12093478
	0.223057618	0.0	1.0	5.0	⋯	1.0	9.0	14.57657125
	0.116746823	0.0	0.0	1.0	⋯	4.0	10.0	−29.57097774
	0.264348281	0.0	0.0	1.0	⋯	0.0	7.0	−18.41325536
	0.543600703	0.0	0.0	0.0	⋯	1.0	3.0	29.58081213
Green Color	0.99999999875	0.0	0.0	0.0	⋯	0.0	2.0	3.14125232
	0.99999999875	0.0	0.0	0.0	⋯	3.0	19.0	7.37795624
	0.99999999875	0.0	0.0	0.0	⋯	0.0	1.0	13.67007204
	0.99999999875	0.0	0.0	0.0	⋯	1.0	4.0	13.21509213
	0.99999999875	0.0	0.0	0.0	⋯	0.0	1.0	−26.6611286
	0.577350268	1.0	0.0	5.0	⋯	0.0	6.0	−16.90808909
	0.577350268	0.0	0.0	0.0	⋯	3.0	9.0	−27.55859066

Figure 13. Non-normalized Hu, Hog, and SIFT descriptors.

A0	A1	A2	A3	A4	C
−0.0702396	−0.766969	0.0866682	−0.201177	−0.509347	0
−0.2053394	0.6626499	−0.0720878	−0.2039496	−0.6448856	0
0.3305015	−0.1212144	−0.043059	−0.2817163	0.9337489	0
0.3328465	0.100635	−0.0407547	−0.3681225	0.7903825	0
−0.4678113	0.3018566	0.3787054	0.0617785	−0.6318964	0
−0.4937295	0.2704155	−0.3002524	0.3750716	0.0643537	0
−0.1510889	−0.0943005	0.1885314	−0.6940842	−0.1595316	0
−0.3268209	0.0193354	0.1061983	−0.8234989	−0.0287219	0
⋮	⋮	⋮	⋮	⋮	⋮
−0.3818587	0.6444977	0.106373	−0.0248139	−0.1106327	1
−0.2292507	−0.6489568	0.1398214	0.0468033	0.113593	1
0.2152551	0.0143186	0.5020942	0.0104236	−0.1853552	1
−0.6380817	−0.1859785	−0.3054753	0.0423496	−0.0392558	1
−0.6848561	0.2017993	−0.3023047	−0.068399	0.0068871	1
−0.4408378	−0.1016725	−0.2260899	−0.2273353	−0.152463	1
−0.4693166	−0.001434	−0.2640595	−0.2928893	0.1600137	1
−0.3119296	−0.4323564	0.4848472	0.0542812	0.0589734	1
⋮	⋮	⋮	⋮	⋮	⋮

Figure 14. Feature data after PCA processing.

The data for training and testing, intended for selecting the SVM classifier, was organized considering three configuration aspects, namely percentages of 80–20%, 50–50%, and 70–30%, respectively, for the training and testing stages.

From the analyses performed, the third-order polynomial kernel presented the best result (Figure 15), where the best combination evaluated for the training and testing data, according to the classification report metrics (Table 8), was 80–20% (Table 9). That is, it presented the best metrics regarding accuracy, precision, recall, F1-score, area under the curve, and lower mean squared error (Equation (2)). In this context, the final configuration for the polynomial kernel is presented in Table 10.

Table 6. Explained variance per principal component–19 components.

PC	Eigenvalue	% of Variance	Cumulative Variance (%)
1	0.64	12.30	12.30
2	0.57	11.02	23.31
3	0.33	6.33	29.65
4	0.28	6.28	35.93
5	0.28	5.31	41.24
6	0.18	3.56	44.79
7	0.18	3.59	48.18
8	0.13	2.79	50.97
9	0.14	2.56	53.53
10	0.12	2.47	56.00
11	0.13	2.57	58.52
12	0.11	1.87	60.39
13	0.09	1.63	62.02
14	0.09	1.65	63.66
15	0.08	1.57	65.23
16	0.08	1.54	66.76
17	0.07	1.41	68.17
18	0.06	1.31	69.48
19	0.06	1.23	70.79

Table 7. Hyperparameter settings–grid search

Polynomial Kernel Settings
kernel: polynomial, Degree: 3, 5, 7, Parameters C: 1, 10, 100, 1000, Gamma: 0.001; 0.01; 0.1; 1, Class_Weight: (balanced, 0: 0.1 | 1: 0.9)
RBF Kernel Settings
kernel: RBF, Degree: 3, 5, 7, Parameters C: 1, 10, 100, Gamma: 0.001; 0.01; 0.1; 1, Weight: (0: 0.3 | 1: 0.7) (0: 0.1 | 1: 0.9)
Linear Kernel Settings
kernel: linear, Parameters C: 1, 10, 100, Gamma: 0.01; 0.1; 1, Class_Weight: (0: 0.1 | 1: 0.9)
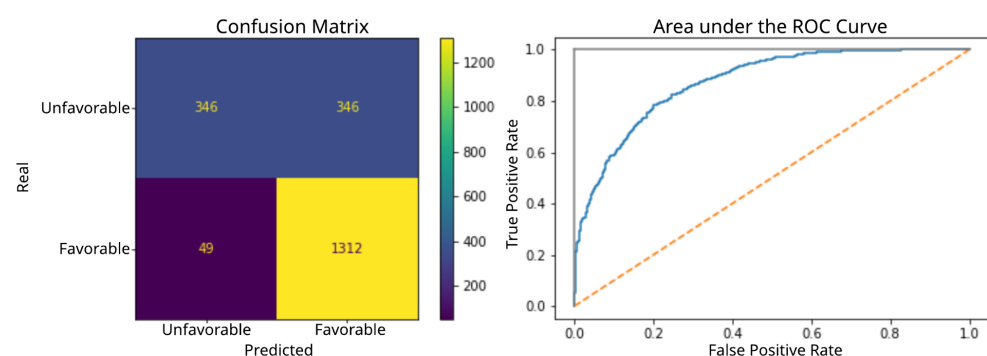
Figure 15. Result obtained with an SVM classifier based on a polynomial kernel.

Table 8. Classifier report data—polynomial kernel.

	Precision	Recall	F1-Score	Support
0	0.88	0.50	0.64	692
1	0.79	0.96	0.87	1361
Accuracy			0.81	2053
Macro Average	0.83	0.73	0.75	2053
Weighted Average	0.82	0.81	0.79	2053

Table 9. Comparative data—SVM classifier.

Descriptive Statistics	Configuration 80-20			Configuration 70-30			Configuration 50-50		
	Acc.	MSE	AUC	Acc.	MSE	AUC	Acc.	MSE	AUC
SVM Classifier—Linear Kernel									
Minimum	0.692	0.000	0.440	0.684	0.000	0.480	0.687	0.000	0.490
Maximum	1.000	0.308	0.690	1.000	0.316	0.690	1.000	0.313	0.640
Mean	0.787	0.213	0.587	0.792	0.208	0.588	0.777	0.223	0.583
Standard Error	0.011	0.011	0.008	0.011	0.011	0.006	0.011	0.011	0.003
Variance	0.008	0.008	0.004	0.007	0.007	0.002	0.007	0.007	0.001
Standard Dev.	0.089	0.089	0.062	0.085	0.085	0.050	0.085	0.085	0.025
Median	0.750	0.250	0.590	0.761	0.239	0.590	0.741	0.259	0.590
25th Percentile	0.731	0.167	0.540	0.729	0.179	0.550	0.724	0.212	0.570
75th Percentile	0.833	0.269	0.640	0.821	0.271	0.620	0.788	0.276	0.600
SVM Classifier—Polynomial Kernel									
Minimum	0.692	0.000	0.820	0.795	0.034	0.800	0.769	0.041	0.800
Maximum	1.000	0.308	1.000	0.966	0.205	1.000	0.959	0.231	0.990
Mean	0.790	0.210	0.917	0.860	0.140	0.916	0.844	0.156	0.900
Standard Error	0.011	0.011	0.006	0.005	0.005	0.005	0.005	0.005	0.005
Variance	0.008	0.008	0.002	0.002	0.002	0.001	0.001	0.001	0.001
Standard Dev.	0.088	0.088	0.043	0.042	0.042	0.039	0.036	0.036	0.039
Median	0.756	0.244	0.915	0.850	0.150	0.910	0.841	0.159	0.900
25th Percentile	0.731	0.167	0.900	0.829	0.128	0.890	0.815	0.133	0.870
75th Percentile	0.833	0.269	0.948	0.872	0.171	0.940	0.867	0.185	0.928
SVM Classifier—RBF Kernel									
Minimum	0.709	0.000	0.570	0.709	0.000	0.570	0.687	0.000	0.460
Maximum	1.000	0.291	1.000	1.000	0.291	1.000	1.000	0.313	1.000
Mean	0.794	0.206	0.820	0.794	0.206	0.820	0.779	0.221	0.769
Standard Error	0.011	0.011	0.015	0.011	0.011	0.015	0.011	0.011	0.018
Variance	0.007	0.007	0.014	0.007	0.007	0.014	0.007	0.007	0.020
Standard Dev.	0.084	0.084	0.119	0.084	0.084	0.119	0.084	0.084	0.143
Median	0.765	0.235	0.830	0.765	0.235	0.830	0.744	0.256	0.755
25th Percentile	0.729	0.173	0.713	0.729	0.173	0.713	0.728	0.210	0.653
75th Percentile	0.827	0.271	0.930	0.827	0.271	0.930	0.790	0.272	0.878

Table 10. Hyperparameters—polynomial kernel.

Hyperparameters	Values
C	100
Weight (Class 0)	0.3
Weight (Class 1)	0.7
Degree	3
Gamma	0.1

When the polynomial kernel was used and the confusion matrix was analyzed, the main diagonal of the matrix correctly indicated 346 cases belonging to class "0", i.e., absence of favorability to ASR. These records were indeed class "0". However, 346 false positives were observed in the upper right quadrant, corresponding to cases that actually belonged to class "1", i.e., favorable to the occurrence of ASR.

In the second quadrant along the main diagonal, 1312 cases were recorded. The model correctly classified these cases as belonging to class "1", which they indeed did.

However, in the lower left quadrant, 49 false negatives were identified. These cases actually belonged to class "0".

3.3. Results of Variable Fusion and Fuzzy Modeling for Favorability Prediction

Table 11 shows the fuzzy variable settings and the corresponding membership functions for the seven variables considered for risk analysis. In addition, Figure 16 shows the obtained results, as indicated for each interval of the membership functions, i.e., with the associated error of approximately $\pm5\%$, corresponding to each transition zone between the low, medium, and high favorability levels.

Table 11. Configuration of the membership functions.

| Description	Configuration
Antecedent: Leaf Wetness Period	
Humidity below threshold	0, 43, 89
Humidity at threshold	88, 90, 94
Humidity above threshold	93, 96, 100
Antecedent: Minimum Leaf Wetness Period	
Time below threshold	0, 14, 24
Time at threshold	22, 46, 70
Time above threshold	66, 83, 100
Antecedent: Soybean Leaf Image Classification Data	
Unfavorable	0, 0, 1
Favorable	1, 1, 1
Antecedent: Dew Point	
Temperature below threshold	$-2, -1, 0$
Temperature at threshold	0, 1, 2
Temperature above threshold	2, 3, 4
Antecedent: Temperature Range	
Initial: range below threshold	0, 7, 15
Initial: range at threshold	14.4, 18, 21.4
Initial: range above threshold	21, 24, 27
Final: range below threshold	14, 19, 24
Final: range at threshold	23.4, 26, 28.4
Final: range above threshold	28, 36, 44
Antecedent: Minimum Temperature	
Minimum temperature below threshold	0, 7, 15
Minimum temperature at threshold	14, 18, 22
Minimum temperature above threshold	21, 24, 27
Antecedent: Maximum Temperature	
Maximum temperature below threshold	14, 19, 24
Maximum temperature at threshold	23, 26, 28
Maximum temperature above threshold	27, 35, 43
Consequent: Favorability	
Low	0, 17.15, 33.3
Medium	32.3, 50, 67.6
High	66.6, 84, 100
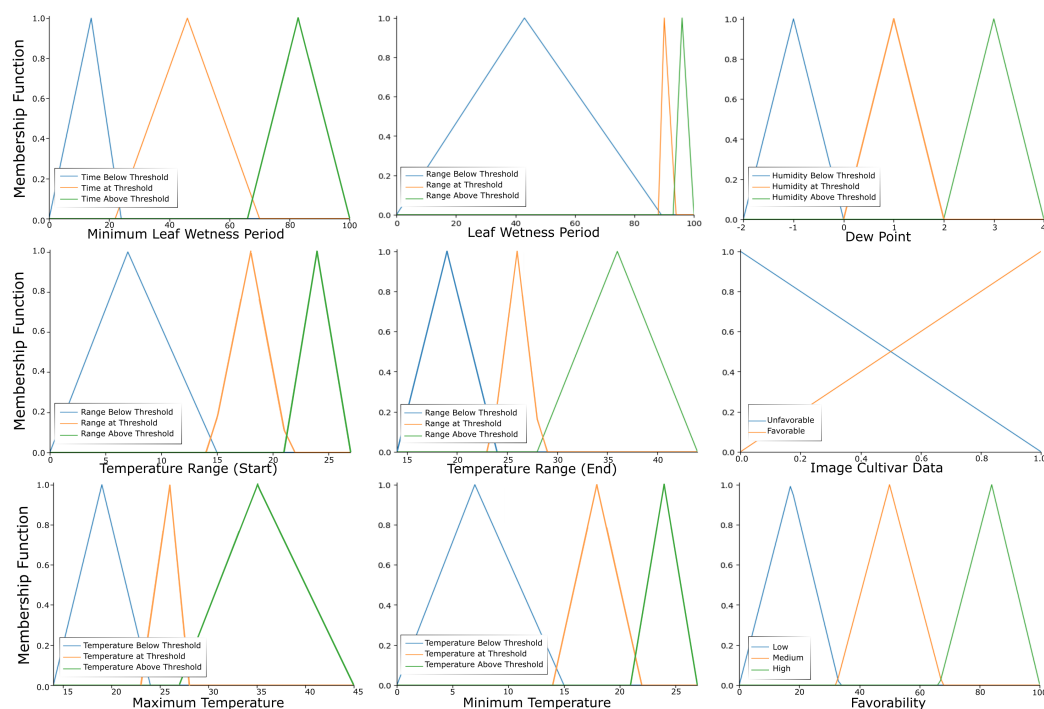
Figure 16. Results based on the membership functions.

3.4. Results of Variable Fusion and Markovian Modeling for Favorability Prediction

Based on the structuring of the cloud architecture, the data for the variable fusion stage were selected. This dataset encompassed the considered time series period, enabling validation of the method using predefined ten-day temporal windows shifted along the series. Classification information was derived from the analysis of image processing using the Embrapa dataset.

Once the climatic time-series data were structured, and considering the set of images with their classified patterns, the variable fusion algorithm based on the Markovian model was applied, as shown in Figure 17.
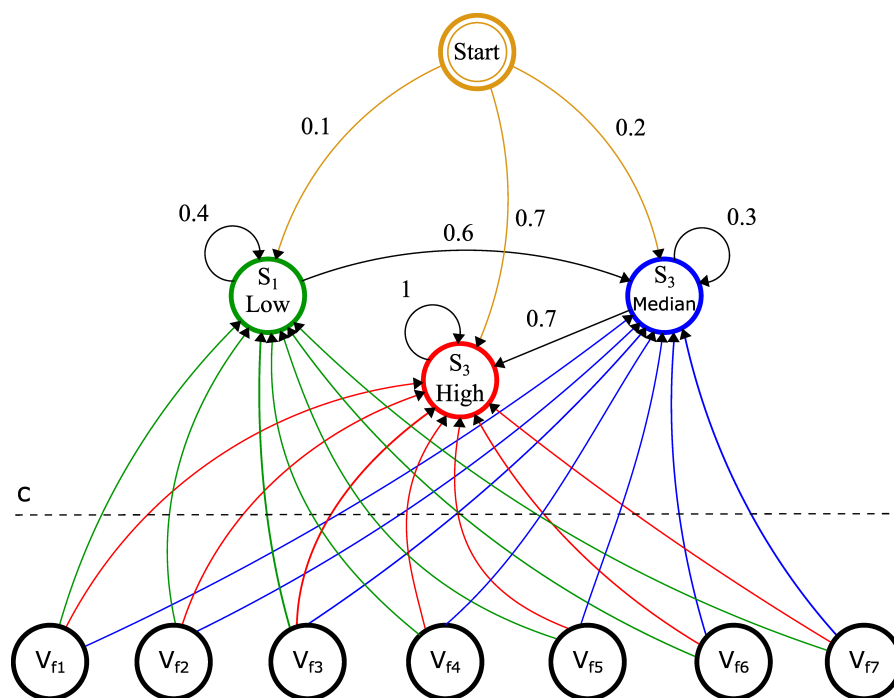

Figure 17. The Markov hidden chain's model for Asian rust risk analysis in soybean crops.

For the favorability of ASR occurrence, the probability values for low, median, and high occurrences were considered to be 0.1, 0.3, and 0.7, respectively. The combinations denoted by "C" represented 27 possibilities generated from the variables V_{f1} (leaf wetting period), V_{f2} (minimum leaf wetting period), V_{f3} (temperature range), V_{f4} (maximum temperature), V_{f5} (minimum temperature), V_{f6} (dew point), and V_{f7} (results of the image classification based on the soybean leaf color related to field truth), totaling 128 combinations. To evaluate disease occurrence, the hidden Markov chain observations were defined by the combinations of these seven variables (V_{f1}–V_{f7}) and their associated probabilities within the windowing period, corresponding to the time-series data and this classification variable.

The transition probabilities, representing changes in disease favorability states, associated with each variable also comprised the hidden Markov chain and were identified by the percentages indicated in each observation.

The emission probabilities were derived from the state transitions of the observations within the hidden Markov chain. The combinations were selected through a data collection process using a time window, guided by the ASR favorability rule for different stages: (1) transition to the "Low" favorability state, when the set of variables corresponded to the 0–33% range according to the observations; (2) transition to the "Median" favorability state, when the identified variables were within the 34–66% range; and (3) transition to the "High" favorability state, when the variables exceeded the 66% range. The hidden Markov chain is summarized schematically in Table 12.

Table 12. The hidden Markov chain data.

C	V_{f1}	V_{f2}	V_{f3}	V_{f4}	V_{f5}	V_{f6}	V_{f7}	S	P_V_{f1}	P_V_{f2}	P_V_{f3}	P_V_{f4}	P_V_{f5}	P_V_{f6}	P_V_{f7}
1	0	0	0	0	0	0	0	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0	0	0	0	0	0	1	1	0.00	0.00	0.00	0.00	0.00	0.00	1.00
3	0	0	0	0	0	1	0	1	0.00	0.00	0.00	0.00	0.00	1.00	0.00
4	0	0	0	0	0	1	1	1	0.00	0.00	0.00	0.00	0.00	0.50	0.50
5	0	0	0	0	1	0	0	1	0.00	0.00	0.00	0.00	1.00	0.00	0.00
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9	0	0	0	1	0	0	0	1	0.00	0.00	0.00	1.00	0.00	0.00	0.00
10	0	0	0	1	0	0	1	1	0.00	0.00	0.00	0.50	0.00	0.00	0.50
11	0	0	0	1	0	1	0	1	0.00	0.00	0.00	0.50	0.00	0.50	0.00
12	0	0	0	1	0	1	1	2	0.00	0.00	0.00	0.33	0.00	0.33	0.33
13	0	0	0	1	1	0	0	1	0.00	0.00	0.00	0.50	0.50	0.00	0.00
14	0	0	0	1	1	0	1	2	0.00	0.00	0.00	0.33	0.33	0.00	0.33
15	0	0	0	1	1	1	0	2	0.00	0.00	0.00	0.33	0.33	0.33	0.00
16	0	0	0	1	1	1	1	2	0.00	0.00	0.00	0.25	0.25	0.25	0.25
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
128	1	1	1	1	1	1	1	3	0.14	0.14	0.14	0.14	0.14	0.14	0.14

The probability for the "Start" state in the Markovian model application was randomly assigned. Additionally, at the model's onset, low favorability (S_1) was set to 10%, median favorability (S_2) to 20%, and high favorability (S_3) to 70%. For state S_1, the probability of remaining in S_1 was set at 40%, while the probability of transitioning to state S_2 was 60%. For state S_2, the probability of remaining in the same state was 30%, whereas the probability of evolving to state S_3 was 70%. Once state S_3 was reached, the probability of remaining in this state was set to 100%, meaning a return to states S_1 or S_2 was not possible.

The hidden Markov chain customized for the process was obtained by using Equation (45) to calculate the probabilities (Table 12) of each combination in the hidden Markov chain.

$$PRc = \frac{1}{\sum_{var=1}^{n}(R_{var} + \beta)} \tag{45}$$

where PRc is the total probability of the hidden Markov chain combination; n is the quantity; V_f represents the variables involved; and β is an adimensional constant used to avoid division by zero.

After defining the vector of occurrences corresponding to one of the windowings applied to the time series of climate and classification data, it was used as input for the Markovian algorithm.

As part of the method, the number of values within the same time window that satisfied the rule was counted for each variable. This involved transforming the number of occurrences relative to the values of the "V_f" variables. The transformation function was defined as follows: when the number of occurrences of $V_f \geq 1$, then $V_f = 1$; and, when the number of occurrences of $V_f \leq 1$, then $V_f = 0$.

After this, the occurrences were transformed to form the input vector for the Markovian algorithm. Table 4 presents an example of the data mapped over a ten-day window, in which the leaf wetness period variable registered seven occurrences of favorability for ASR. In this example, the values for minimum leaf wetting period, maximum temperature, minimum temperature, dew point, and image data were "1", while no occurrences were recorded for the temperature range variable.

Based on the graph of considered rules (Figure 18) and data input vector (Table 13), the transformed occurrences indicated high favorability, considering values of six variables as "1".
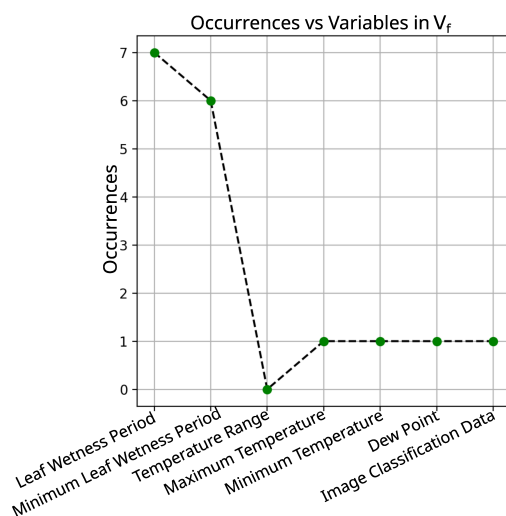

Figure 18. Chart of accounted rules.

Table 13. Data input vector.

Input (Algorithm)	V_{f1}	V_{f2}	V_{f3}	V_{f4}	V_{f5}	V_{f6}	V_{f7}
Occurrences:	7	6	0	1	1	1	1
Transformed Occurrences:	1	1	0	1	1	1	1

Thus, based on this input vector, the Markovian model generated an output vector that translated the information on the stage of favorability of ASR, as summarized in Table 14.

Table 14. Markov chain result.

Selected Hidden Chain:	1	1	0	1	1	1	1
Selected Probability:	0.17	0.17	0.0	0.17	0.17	0.17	0.17
State (S):				3			
Favorability:				High			

It is noteworthy that, depending on the input variables, the state of favorability was defined as low, medium, or high.

Examples of data processed in different time windows and originating from each process cycle are listed in Table 15. To assess the quality of processing, the time windows were selected via the web interface using the hidden Markov chain technique.

Table 15. Markovian quality data.

Favorability	V_{f1}	V_{f2}	V_{f3}	V_{f4}	V_{f5}	V_{f6}	V_{f7}	$\hat{c}(t)$	$\sigma^2(\overline{f})$	σ	Accuracy	Precision
	0	0	0	0	0	0	0	1.00	0	0	1	1
	0	0	0	0	0	0	1	1.00	0.12	0.35	0.88	0.65
Low	0	0	0	0	0	1	0	1.00	0.12	0.35	0.88	0.65
	0	0	0	0	0	1	1	1.00	0.20	0.45	0.80	0.55
	0	0	0	0	1	0	0	1.00	0.12	0.35	0.88	0.65
	0	0	0	0	1	1	1	1.00	0.24	0.49	0.76	0.51
	0	0	0	1	0	1	1	1.00	0.24	0.49	0.76	0.51
Median	0	0	0	1	1	0	1	1.00	0.24	0.49	0.76	0.51
	0	0	0	1	1	1	0	1.00	0.24	0.49	0.76	0.51
	0	0	0	1	1	1	1	1.00	0.24	0.49	0.76	0.51
	0	1	1	1	1	0	1	1.00	0.20	0.45	0.80	0.55
	0	1	1	1	1	1	0	1.00	0.20	0.45	0.80	0.55
High	0	1	1	1	1	1	1	1.00	0.12	0.35	0.88	0.65
	1	0	0	1	1	1	1	1.00	0.20	0.45	0.80	0.55
	1	1	0	1	1	1	1	1.00	0.12	0.35	0.88	0.65

The result indicated an error of <1%, demonstrating a high-quality index for the data fusion process. Next, for low favorability, the first point was notable, presenting an error of "0" and accuracy and precision values of "1".

Regarding the standard deviation (Table 15), the observed error differences were minor, making the obtained autocorrelation values reasonable. The calculated autocorrelations depended on two main factors: the input variables and the value of ($\hat{c}(t)$), which represents the size of the processing time window. The variation in ($\hat{c}(t)$) was minimal, affecting only the third or fourth decimal place as the processing was executed under the same infrastructure configuration. Consequently, under these conditions, this variable contributed minimally to the differences observed in the standard deviation of the reasonable errors for the calculated autocorrelations.

However, the variation in combinations of input variables from V_{f1} to V_{f7} more significantly influenced the differences in the standard deviation values of the reasonable errors for the calculated autocorrelations. The combinations of variables from V_{f1} to V_{f7}, as shown in Figure 19, indicate that the increase in standard deviation values was due to the presence of variables with a value equal to "1". A noteworthy observed behavior was that the standard deviation value reached its peak with up to three variables equal to "1" and

stabilized at the fourth. From the fifth variable equal to "1" onward, the standard deviation of the reasonable errors for the calculated autocorrelations began to decrease, indicating greater consistency in information processing.
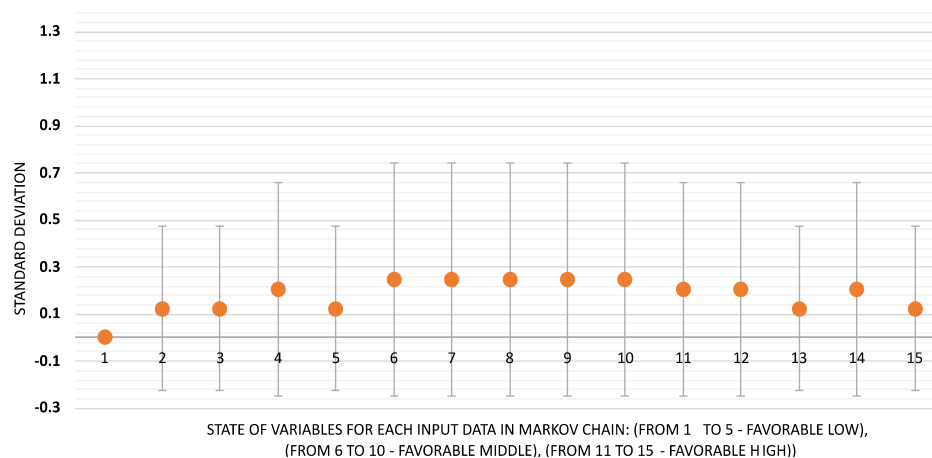

Figure 19. Standard deviation versus status of the variables related to each input in the hidden Markov chain applied to ASR risk evaluation.

3.5. Comparative Evaluation of the Results Between Modeling Based on the Fuzzy System and the Hidden Markov Chain

To compare the data fusion models, an evaluation framework based on two distinct scenarios was established, i.e., using for the first one 29 combinations for low-, 29 combinations for medium-, and 29 combinations for high-favorability occurrences. On the other hand, for the second one, there were only 41 combinations for medium favorability, and zero combinations for the low and high occurrences. Table 16 presents the final comparative results, where it is possible to observe, for both scenarios, the best behavior of the model based on the hidden Markov chain, which presented accuracy equal to 100% matching.

Table 16. Performance comparison of the data fusion models.

Favorability Analysis—Evaluation						
Fuzzy Logic (Category)	**Scenario 1**			**Scenario 2**		
	Samples	**Correct (Count)**	**Accuracy (%)**	**Samples**	**Correct (Count)**	**Accuracy (%)**
Low Favorability	29	8	27.59	0	N/A	N/A
Medium Favorability	29	12	41.38	41	25	60.98
High Favorability	29	18	62.07	0	N/A	N/A
Hidden Markov Model (Category)	Samples	Correct (Count)	Accuracy (%)	Samples	Correct (Count)	Accuracy (%)
Low Favorability	29	29	100.00	0	N/A	N/A
Medium Favorability	29	29	100.00	41	41	100.00
High Favorability	29	29	100.00	0	N/A	N/A

In such a context, the processing output was displayed on a dashboard panel (Figure 20), which, in summary, included the main information from the executed procedures, such as segmented images, visualization of climatic variables, data fusion and

favorability results, as well as access to a container with decision-support reports, prepared based on the historical data cube from the data warehouse. Accordingly, reports (Subject 1), (Subject 2), and (Subject 3) could be displayed in separate containers at the top of the dashboard panel.
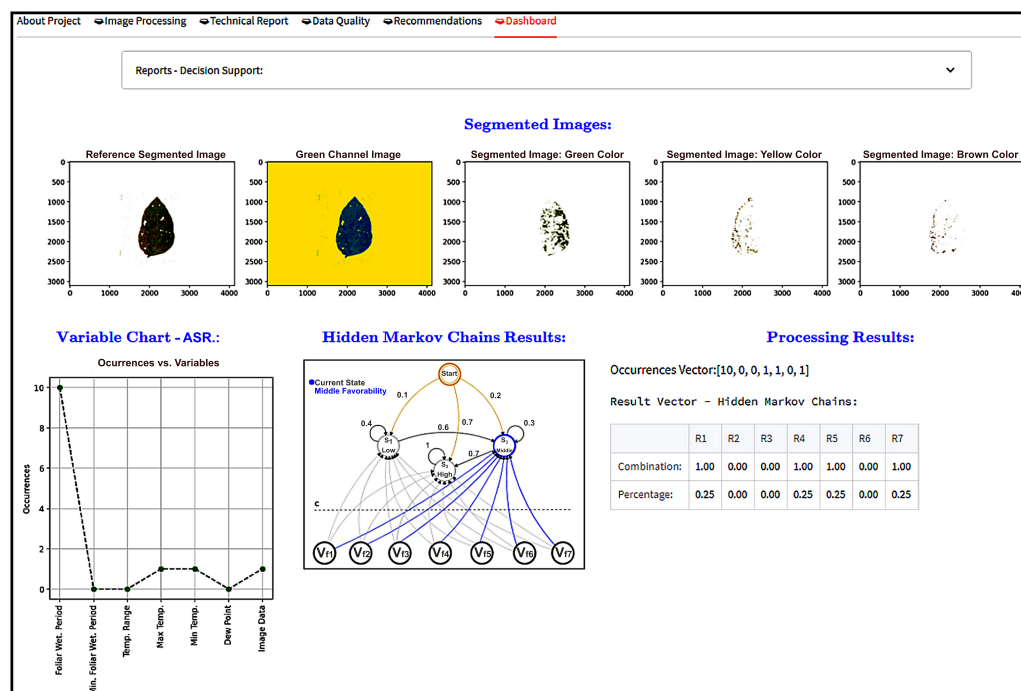

Figure 20. Final dashboard interface (when under processing).

3.6. Analytical Reports

The analytical reports represent another important aspect of the analyzed results. These reports were generated from OLAP tool queries based on the DW historical database, whose model was constructed according to the defined requirements.

A load was executed on the DW using an SQL script, and data of interest to the producer was collected. The script was created by combining the data tables of the transactional database, which responded to the queries of the developed requirements. These requirements involved (1) the influence of climatic variables on the favorability of ASR (Subject 1, Figure 21); (2) the accounting of low, median, and high favorability per year (Subject 2, Figure 22); and (3) the influence of the image of the soybean leaf on the favorability of ASR per year, in the planting and harvesting stages, primarily R5 and R6 as they are the most affected by the disease (Subject 3, Figure 23).
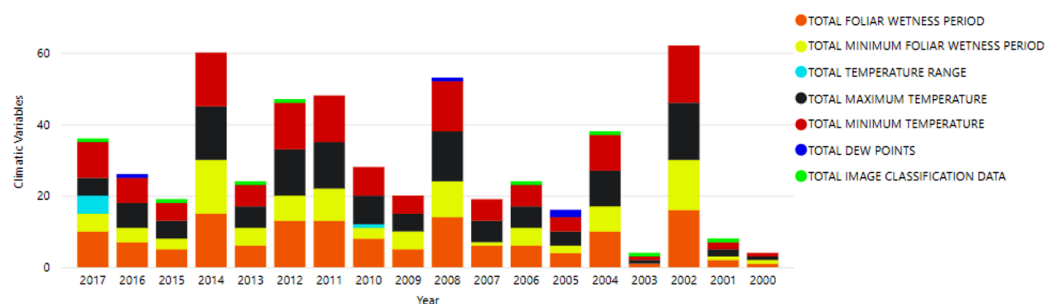

Figure 21. Data analysis analytical report (subject 1).

The analytical report (Subject 1) showed that the highest incidences of ASR favorability over time were associated with the leaf wetting period, maximum temperature, minimum temperature, minimum leaf wetting period, and soybean leaf image classification data.

Following this line of reasoning, the other most significant variables contributing to disease favorability were the minimum period of leaf wetness, followed by the dew point and temperature range. These findings highlight the variables and their respective relationships on an annual scale in the historical data series.

In the analytical report (Subject 2), it was possible to observe the evaluation of favorability accounting. No cases of low favorability were found during the one crop cycle period. However, both medium and high favorabilities were identified. Figure 22a illustrates the historical series overview for the high-favorability case, while Figure 22b shows the overview for the medium-favorability case.

The period encompassing the interval between soybean reproductive phenological stages R4, R5, and R6 was assessed in the analytical report (Subject 3). ASR was found to be predominant during stages R5 and R6, which corresponded to the period with the highest incidence of the disease. This 17-day interval (17 November to 4 December) was mapped between the 85th and 95th days (stages V_{f5} and V_{f6}). This result indicates a high degree of favorability for this specific period.

Regarding high favorability (Figure 23), in a few years of the time series, this level of favorability was not observed, even during the R5 and R6 stages of crop development. However, in the remaining years, disease occurrence was recorded. In one instance, during the R5 and R6 stages, only a single record of average favorability was found.
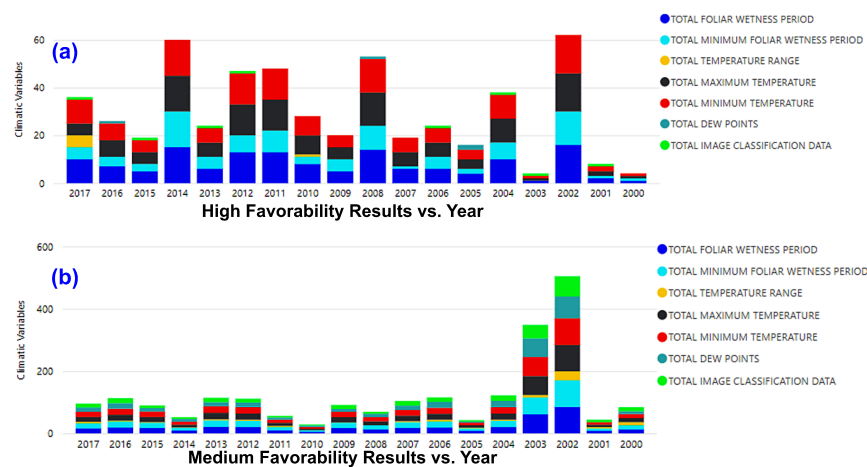

Figure 22. Example of an analytical report (subject 2). In (**a**), the example illustrates that the combinations of the rule base variables were between 66.7% and 100%, while in (**b**) these combinations were between 33.4% and 66.6%.
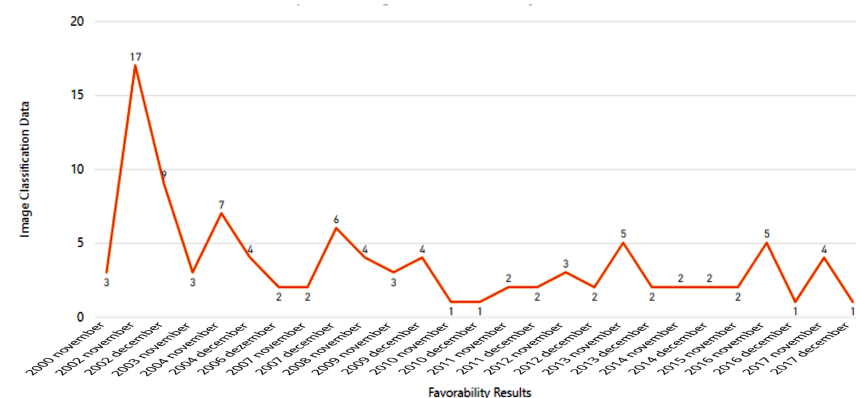

Figure 23. Data analysis analytical report (subject 3).

3.7. Computational Cost

To analyze the computational cost, both the CPU nuclei and memory were evaluated across the four-stage pipeline: (1) segmentation, (2) pattern recognition and PCA, (3) machine learning, and (4) variable fusion for the decision-support process. This evaluation was conducted from two perspectives. The first was a single-instance analysis, focusing on a specific time point from the climate data and its corresponding digital image from a soybean leaf (Figure 24). The second one was a full-dataset analysis, encompassing the processing of all the climate time series and the digital images for producing the result. The resulting percentage utilization of the processing units and memory is shown in Table 17.

The resource consumption dynamics during runtime are detailed in Figure 24, while a consolidated statistical summary is presented in Table 17. The machine learning and all seven variables for the data fusion stages exhibited intense and stable processing demands, with mean usages of 90.55% and 89.26%, respectively. In contrast, the feature extraction together with the PCA stage showed more variable behavior, with a mean usage of 27.51% and a high standard deviation (22.06), indicating that processing loads peaked at 75.10%. Memory consumption remained consistently less than 11.18%, and it was stable across all the stages, peaking at 11.60%, which demonstrates that memory was not a critical computational resource.
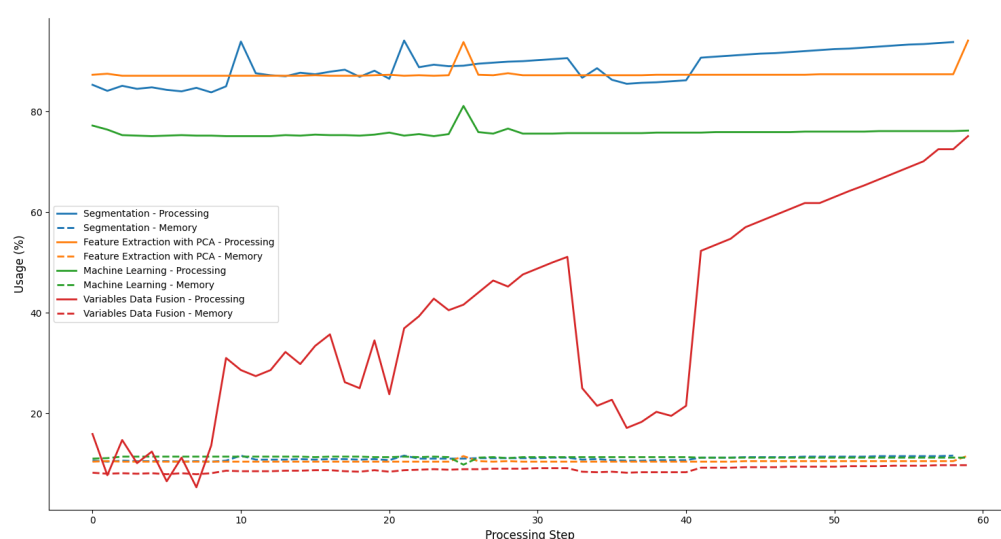

Figure 24. Computational cost analysis.

Table 17. Statistical summary of computational cost by process.

Process	Processing (%)				Memory (%)			
	Mean	Std. Dev.	Min.	Max.	Mean	Std. Dev.	Min.	Max.
Segmentation	76.24	2.10	75.10	81.10	11.18	0.44	9.80	11.40
Feature Extraction with PCA	27.51	22.06	5.30	75.10	8.52	0.65	7.90	9.70
Machine Learning	90.55	3.51	83.80	94.10	10.95	0.52	10.40	11.60
Variable Data Fusion	89.26	3.49	83.80	94.10	10.96	0.51	10.40	11.60

3.8. System Validation with Phytopathologists and Agronomists

The tests submitted to the specialists were also processed by the developed system. The responses of the specialists (phytopathologists and agronomists) were used as references, alongside the system's corresponding accuracy relative to those responses. Additionally,

to organize the responses into a unified dataset, normalization was applied based on the maximum and minimum values within the response set.

The system validation results demonstrate a strong correlation between the system and the specialists: the identification of Asian soybean rust presence yielded a coefficient of determination of $R^2 = 0.94$, while the estimation of severity levels reached $R^2 = 0.88$, as shown in Figure 25 and Figure 26, respectively.
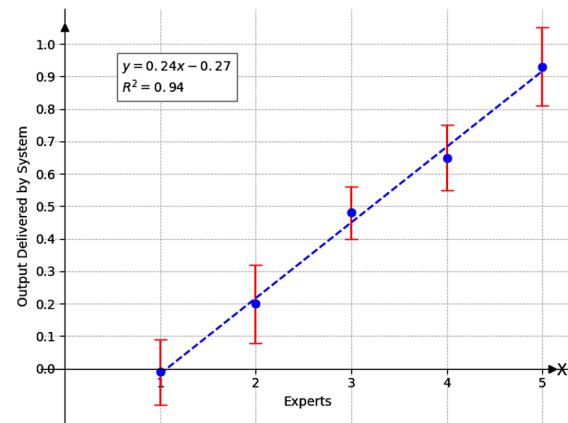

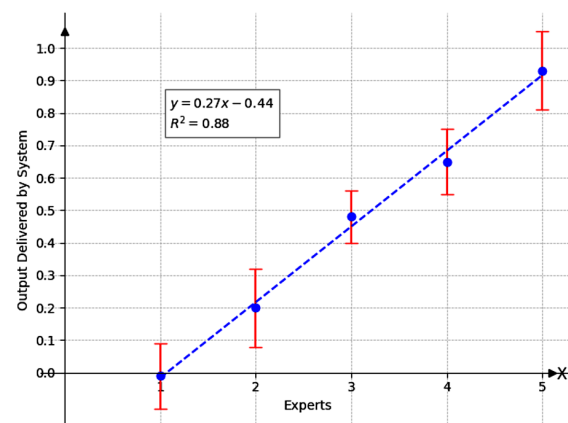**Figure 25.** Validation of the presence or absence of Asian soybean rust.

Figure 26. Validation of Asian soybean rust severity level.

These results express the amount of data variance explained using the linear model. Therefore, the high R^2 values found indicate that the developed system performed satisfactorily in relation to the responses provided by the consulted specialists.

It is important to observe in this developed method that it is not universal. In fact, it is customized for ASR risk management in soybean crops. However, it is based on the idea of being adaptive to other diseases that can occur in soybeans and other grain crops. In this case, such customization may become possible as long as these diseases encounter favorable situations involving climatic issues and symptoms expressed in the crop phenotype.

4. Conclusions

Crop diseases represent the main challenges faced by the agricultural sector. This paper presents a new method for assessing ASR in crops regarding an advanced intelligent and computational decision-making system based on cloud infrastructure. Early detection of ASR disease in crops is crucial to reduce not only its severity and spread in the field but also to minimize the use of fungicides. The decision model was implemented considering not only climatic time series of data but also digital images from soybean leaves, spatially

collected for the evaluation of changes in their phenotype. In relation to the climatic time series of data, the use of B-splines resulted in correlation coefficients (CCs) in the interval $0.63 \leq CC \leq 0.82$, which avoided missing data. The absence of data reduces statistical power, which refers to the probability that the test will reject the null hypothesis when it is false. Also, the lost data can cause bias in the estimation of parameters, and it can reduce the representativeness of the samples. For ASR's risk analysis, the processing employed a large data scale, incorporating both data lake and data warehouse systems, web-based operation, and integrated image feature extraction methods based on SIFT, HOG, and HU (invariant moments) for the pattern's recognition on leaves, and PCA for dimensionality reduction. Moreover, classification using an SVM with a polynomial kernel was used, which achieved an accuracy larger than 84% and AUC larger than 0.90, demonstrating adequate performance. In addition, the use of the metrics PSNR, MSE, and SSIM enabled demonstrating the robustness of such an arrangement, i.e., leading to values in the ranges of $14.00 \leq PSNR \leq 15.00$, $0.03 \leq MSE \leq 0.05$, and ≥ 0.91, respectively. For the data fusion of the variables, i.e., the climate ones and the classified image patterns, the model based on the hidden Markov chain was selected since it presented the best effectiveness, allowing 100% matching for the three different levels of possible risk occurrences. The development of the data quality framework allowed a comprehensive evaluation, supporting the reliability of the method. The quality indicators were also evaluated based on autocorrelation theory and estimation of expected values from the processed data. According to the results, these indicators showed adequate accuracy and precision, with cross-correlation validation by experts in phytopathology, achieving linear regression correlation factor values above 0.85, i.e., confirming the method's reliability. In conclusion, the results validated the developed method, demonstrating significant improvements over traditional climate- or even digital-image-only approaches through the integration of heterogeneous data fusion. Likewise, its practical viability has been shown for field implementation through an intuitive web interface, with the potential to reduce ASR-related losses through disease prevention, early detection, and rational use of fungicides. This development is of great relevance both for advancing knowledge in computer science techniques related to signal and digital image processing and reducing production risks in agriculture. The current method was specifically calibrated for soybeans and requires adaptation for other grain cultivars and geographic regions. For future work, one may aim to use convolutional networks and evaluate opportunities to enable unsupervised operations for agricultural plant disease assessments.

Author Contributions: This work was conducted collaboratively by both authors. Conceptualization, R.A.N. and P.E.C.; formal analysis, R.A.N. and P.E.C.; writing—original draft proposition, R.A.N.; writing—review and editing, P.E.C.; supervision, P.E.C. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The original data presented in the study are openly available in the repository 20240219_RAN_PEC on GitHub®, which can be accessed at https://github.com/ricardo-a-neves/20240219_RAN_PEC, available to be accessed since 15 May 2025.

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

Abbreviations

The following abbreviations are used in this manuscript:

Acronym	Meaning
ACS	Analytics Cloud Service
AD	Autonomous Database
AI	Artificial Intelligence
AMD	Advanced Micro Device
API	Application Programming Interface
ASR	Asian Soybean Rust
AUC	Area Under the Curve
CC	Correlation Coefficient
CNN	Convolutional Neural Network
COG	Center of Gravity
DL	Data Lake
DM	Data Mart
DSE	Data Science Environment
DW	Data Warehouse
Embrapa	Brazilian Agricultural Research Corporation
ETL	Extract, Transform, Load
Fapesp	São Paulo Research Foundation
FN	False Negative
FP	False Positive
FSIM	Feature Similarity Index
GB	Gigabyte
GIS	Geographic Information System
HOG	Histogram of Oriented Gradients
HU	Hu Moments
INMET	Instituto Nacional de Meteorologia
IoT	Internet of Things
KNN	K-Nearest Neighbor
MAPA	Ministry of Agriculture, Livestock and Food Supply (Brazil)
MHz	Megahertz
MSE	Mean Squared Error
OLAP	Online Analytical Processing
OS	Object Storage
PCA	Principal Component Analysis
PSNR	Peak Signal-to-Noise Ratio
RD	Relational Database
RF	Random Forest
RGB	Red, Green, Blue
RH	Relative Humidity
ROC	Receiver Operating Characteristic
ROI	Region of Interest
SIFT	Scale-Invariant Feature Transform
SQL	Structured Query Language
SSIM	Structural Similarity Index
SVM	Support Vector Machine
TFP	True False Positive
TN	True Negative
TNR	True Negative Rate
TP	True Positive
TPR	True Positive Rate

TVP	True Positive Rate
VCN	Virtual Cloud Network

References

1. Rinaldi, M.; Murino, T.; Gebennini, E.; Morea, D.; Bottani, E. A literature review on quantitative models for supply chain risk management: Can they be applied to pandemic disruptions? *Comput. Ind. Eng.* **2022**, *170*, 108329. [CrossRef] [PubMed]

2. García-Machado, J.J.; Greblikaitė, J.; Iranzo Llopis, C.E. Risk Management Tools in the Agriculture Sector: An Updated Bibliometric Mapping Analysis. *Studies in Risk and Sustainable Development*; University of Economics in Katowice: Katowice, Poland, 2024; pp. 1–26.

3. Hackfort, S.; Marquis, S.; Bronson, K. Harvesting value: Corporate strategies of data assetization in agriculture and their socio-ecological implications. *Big Data Soc.* **2024**, *11*, 20539517241234279. [CrossRef]

4. Ali, G.; Mijwil, M.M.; Buruga, B.A.; Abotaleb, M.; Adamopoulos, I. A survey on artificial intelligence in cybersecurity for smart agriculture: State-of-the-art, cyber threats, artificial intelligence applications, and ethical concerns. *Mesopotamian J. Comput. Sci.* **2024**, *2024*, 53–103. [CrossRef] [PubMed]

5. Sahu, A.; Acharya, B.; Sahoo, P.S. Agricultural farming decision support system using artificial intelligence: A comparative analysis. In *Optimizing Smart and Sustainable Agriculture for Sustainability*; CRC Press: Boca Raton, FL, USA, 2025; pp. 212–236.

6. Armstrong, M. The World's Leading Soybean Producers. Statista. 2023. Available online: https://www.statista.com/chart/19323/the-worlds-leading-soybean-producers/ (accessed on 22 June 2025).

7. Oerke, E.C.; Dehne, H.W. Safeguarding production—Losses in major crops and the role of crop protection. *Crop Prot.* **2004**, *23*, 275–285. [CrossRef]

8. U.S. Department of Agriculture, Foreign Agricultural Service. Foreign Agricultural Service. 2025. Available online: https://www.fas.usda.gov/ (accessed on 25 June 2025).

9. Godoy, C.V.; Seixas, C.D.S.; Soares, R.M.; Meyer, M.C.; Costamilan, L.M.; Adegás, F.S. *Best Practices for the Management of Asian Soybean Rust*; Technical Bulletin (Infoteca-E); Embrapa Soybean: Londrina, Brazil, 2017. Available online: http://www.infoteca.cnptia.embrapa.br/infoteca/handle/doc/1074899 (accessed on 10 October 2024). (In Portuguese)

10. Goellner, K.; Loehrer, M.; Langenbach, C.; Conrath, U.; Koch, E.; Schaffrath, U. Phakopsora pachyrhizi, the causal agent of Asian soybean rust. *Mol. Plant Pathol.* **2010**, *11*, 169–177. [CrossRef]

11. Beruski, G.C.; Gleason, M.L.; Sentelhas, P.C.; Pereira, A.B. Leaf wetness duration estimation and its influence on a soybean rust warning system. *Australas. Plant Pathol.* **2019**, *48*, 395–408. [CrossRef]

12. Bedin, E. Foliar Applications of Copper in the Management of Asian Soybean Rust. Ph.D. Thesis, University of Passo Fundo, Passo Fundo, RS, Brazil, 2018. (In Portuguese)

13. Nunes, C.D.M.; da Silva Martins, J.F.; Del Ponte, E.M. *Validation of a Model for Predicting Asian Soybean Rust Occurrence Based on Rainfall Data*; Technical Bulletin 1516-8832; Embrapa Clima Temperado: Pelotas, Brazil, 2018; INFOTECA-E. (In Portuguese)

14. Mila, A.; Yang, X.; Carriquiry, A. Bayesian logistic regression of Soybean Sclerotinia Stem Rot prevalence in the US North-central region: Accounting for uncertainty in Parameter Estimation. *Phytopathology* **2003**, *93*, 758–764. [CrossRef]

15. de Carvalho Alves, M.; Pozza, E.A.; do Bonfim Costa, J.d.C.; de Carvalho, L.G.; Alves, L.S. Adaptive neuro-fuzzy inference systems for epidemiological analysis of soybean rust. *Environ. Model. Softw.* **2011**, *26*, 1089–1096. [CrossRef]

16. Zagui, N.L.S.; Krindges, A.; Lotufo, A.D.P.; Minussi, C.R. Spatio-Temporal Modeling and Simulation of Asian Soybean Rust Based on Fuzzy System. *Sensors* **2022**, *22*, 668. [CrossRef]

17. Yu, M.; Ma, X.; Guan, H. Recognition method of soybean leaf diseases using residual neural network based on transfer learning. *Ecol. Inform.* **2023**, *76*, 102096. [CrossRef]

18. Ponte, E.M.D.; Godoy, C.V.; Li, X.; Yang, X.B. Models and applications for risk assessment and prediction of asian soybean rust epidemics. *Fitopatol. Bras.* **2006**, *31*, 533–544. [CrossRef]

19. Simionato, R.; Torres Neto, J.R.; Santos, C.J.d.; Ribeiro, B.S.; Araújo, F.C.B.d.; Paula, A.R.d.; Oliveira, P.A.d.L.; Fernandes, P.S.; Yi, J.H. Survey on connectivity and cloud computing technologies: State-of-the-art applied to Agriculture 4.0. *Rev. Ciênc. Agrôn.* **2021**, *51*, e20207755. [CrossRef]

20. de Oliveira, C.F.; Nanni, M.R.; Furuya, D.E.G.; de Souza, B.A.M.; Antunes, J.F.G. Detecting soybean rust in different phenological stages by vegetation indices from multi-satellite data. *Comput. Electron. Agric.* **2023**, *210*, 107923. [CrossRef]

21. González-Domínguez, E.; Caffi, T.; Rossi, V.; Salotti, I.; Fedele, G. Plant disease models and forecasting: Changes in principles and applications over the last 50 years. *Phytopathology®* **2023**, *113*, 678–693. [CrossRef]

22. Jeger, M.; Madden, L.; Van Den Bosch, F. Plant virus epidemiology: Applications and prospects for mathematical modeling and analysis to improve understanding and disease control. *Plant Dis.* **2018**, *102*, 837–854. [CrossRef] [PubMed]

23. Garin, G.; Fournier, C.; Andrieu, B.; Houlès, V.; Robert, C.; Pradal, C. A modelling framework to simulate foliar fungal epidemics using functional–structural plant models. *Ann. Bot.* **2014**, *114*, 795–812. [CrossRef] [PubMed]

24. Feng, J.; Zhang, S.; Zhai, Z.; Yu, H.; Xu, H. DC2Net: An Asian soybean rust detection model based on hyperspectral imaging and deep learning. *Plant Phenomics* **2024**, *6*, 0163. [CrossRef]

25. Khalili, E.; Kouchaki, S.; Ramazi, S.; Ghanati, F. Machine learning techniques for soybean charcoal rot disease prediction. *Front. Plant Sci.* **2020**, *11*, 590529. [CrossRef]

26. Li, W.; Guo, Y.; Yang, W.; Huang, L.; Zhang, J.; Peng, J.; Lan, Y. Severity Assessment of Cotton Canopy Verticillium Wilt by Machine Learning Based on Feature Selection and Optimization Algorithm Using UAV Hyperspectral Data. *Remote Sens.* **2024**, *16*, 4637. [CrossRef]

27. Godoy, C.V.; Seixas, C.D.S.; Soares, R.M.; Marcelino-Guimarães, F.C.; Meyer, M.C.; Costamilan, L.M. Asian soybean rust in Brazil: Past, present, and future. *Pesqui. Agropecu. Bras.* **2016**, *51*, 407–421. [CrossRef]

28. Neves, R.A.; Cruvinel, P.E. Application of Image Processing and Advanced Intelligent Computing for Determining Stage of Asian Rust in Soybean Plants. In Proceedings of the 2022 IEEE 16th International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA, 26–28 January 2022; pp. 280–286.

29. Abbas, A.; Zhang, Z.; Zheng, H.; Alami, M.M.; Alrefaei, A.F.; Abbas, Q.; Naqvi, S.A.H.; Rao, M.J.; Mosa, W.F.; Abbas, Q.; et al. Drones in plant disease assessment, efficient monitoring, and detection: A way forward to smart agriculture. *Agronomy* **2023**, *13*, 1524. [CrossRef]

30. Embrapa Soja. Digipathos Repository—Embrapa Soybean. 2021. Available online: https://www.digipathos-rep.cnptia.embrapa.br/ (accessed on 12 February 2021). (In Portuguese)

31. Instituto Nacional de Meteorologia (INMET). Meteorological Database for Teaching and Research. 2019. Available online: https://portal.inmet.gov.br (accessed on 3 July 2019). (In Portuguese)

32. Embrapa Soja Soy in Numbers (2019/20 Season). 2023. Available online: https://www.embrapa.br/web/portal/soja/cultivos/soja1/dados-economicos (accessed on 21 September 2023). (In Portuguese)

33. Barbedo, J.G.A.; Koenigkan, L.V.; Halfeld-Vieira, B.A.; Costa, R.V.; Nechet, K.L.; Godoy, C.V.; Junior, M.L.; Patricio, F.R.A.; Talamini, V.; Chitarra, L.G.; et al. Annotated Plant Pathology Databases for Image-Based Detection and Recognition of Diseases. *IEEE Lat. Am. Trans.* **2018**, *16*, 1749–1757. [CrossRef]

34. Yanowitz, S.D.; Bruckstein, A.M. A New Method for Image Segmentation. *Comput. Vision Graph. Image Process.* **1989**, *46*, 82–95. [CrossRef]

35. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing*; Pearson Education do Brasil: São Paulo, SP, Brazil, 2010. (In Portuguese)

36. Horé, A.; Ziou, D. Image Quality Metrics: PSNR vs. SSIM. In Proceedings of the 20th International Conference on Pattern Recognition (ICPR), Istanbul, Turkey, 23–26 August 2010; pp. 2366–2369. [CrossRef]

37. Preedanan, W.; Kondo, T.; Bunnun, P.; Kumazawa, I. A Comparative Study of Image Quality Assessment. In Proceedings of the 2018 International Workshop on Advanced Image Technology (IWAIT), Chiang Mai, Thailand, 7–9 January 2018; pp. 1–4. [CrossRef]

38. Sara, U.; Akter, M.; Uddin, M.S. Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. *J. Comput. Commun.* **2019**, *7*, 8–18. [CrossRef]

39. Lowe, D.G. Object Recognition from Local Scale-Invariant Features. In Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV), Kerkyra, Greece, 20–27 September 1999; Volume 2, pp. 1150–1157. [CrossRef]

40. Hu, M.K. Visual Pattern Recognition by Moment Invariants. *IRE Trans. Inf. Theory* **1962**, *8*, 179–187. [CrossRef]

41. Dalal, N.; Triggs, B. Histograms of Oriented Gradients for Human Detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893. [CrossRef]

42. Zhao, W.; Wang, J. Study of Feature Extraction Based Visual Invariance and Species Identification of Weed Seeds. In Proceedings of the 2010 Sixth International Conference on Natural Computation (ICNC), Yantai, China, 10–12 August 2010; Volume 2, pp. 631–635. [CrossRef]

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

44. Vapnik, V. *The Nature of Statistical Learning Theory*; Information Science and Statistics; Springer: New York, NY, USA, 1999.

45. Faceli, K.; Lorena, A.C.; Gama, J.; Carvalho, A.C.P.L.F. *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*; LTC: Rio de Janeiro, Brazil, 2011.

46. da Silva, G.; Ferreira, A.; Guilherme, D.; Grigolli, J.F.; Weber, V.; Pistori, H. Recognition of Soybean Diseases Using Machine Learning Techniques Based on Segmentation of Images Captured by UAVs. In Proceedings of the 16th Workshop on Computer Vision (WVC), Virtual Event, 7–10 November 2020; Brazilian Computer Society (SBC), 2020; pp. 12–17. . (In Portuguese) [CrossRef]

47. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; A comprehensive reference for the following metrics: Variance, standard deviation, precision, recall, F1-score, and ROC/AUC; MIT Press: Cambridge, MA, USA, 2012.

48. Karhunen, K. *Über Lineare Methoden in der Wahrscheinlichkeitsrechnung*; Annales Academiae Scientiarum Fennicae; Series A. I. Mathematica-Physica; Suomalainen Tiedeakatemia: Helsinki, Finland, 1947.

49. Hotelling, H. Analysis of a Complex of Statistical Variables into Principal Components. *J. Educ. Psychol.* **1933**, *24*, 417. [CrossRef]

50. Klema, V.; Laub, A. The Singular Value Decomposition: Its Computation and Some Applications. *IEEE Trans. Autom. Control* **1980**, *25*, 164–176. [CrossRef]

51. Greville, T.N.E. *Theory and Applications of Spline Functions*; Army Mathematics Research Center: Madison, WI, USA; Academic Press: New York, NY, USA, 1969.

52. Boudaren, M.E.Y.; Pieczynski, W. Dempster–Shafer Fusion of Evidential Pairwise Markov Chains. *IEEE Trans. Fuzzy Syst.* **2016**, *24*, 1598–1610. [CrossRef]

53. Li, Y.; Jha, D.K.; Ray, A.; Wettergren, T.A. Information-Theoretic Performance Analysis of Sensor Networks via Markov Modeling of Time Series Data. *IEEE Trans. Cybern.* **2017**, *48*, 1898–1909. [CrossRef]

54. Neves, R.A. Cloud-Based Computer Vision and Intelligence System for Asian Rust Risk Management in Soybean Crops. PhD Thesis, Federal University of São Carlos, São Carlos, SP, Brazil, 2024. (In Portuguese)

55. Zadeh, L.A. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [CrossRef]

56. Jang, J.S.R.; Sun, C.T.; Mizutani, E. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*; Prentice Hall: Upper Saddle River, NJ, USA, 1997.

57. Pedrycz, W. Why triangular membership functions? *Fuzzy Sets Syst.* **1994**, *64*, 21–30. [CrossRef]

58. Prokopowicz, P.; Czerniak, J.; Mikołajewski, D.; Apiecionek, Ł.; Ślęzak, D. (Eds.) *Theory and Applications of Ordered Fuzzy Numbers: A Tribute to Professor Witold Kosiński*; Studies in Fuzziness and Soft Computing; Springer International Publishing: Cham, Switzerland, 2017; Volume 355. [CrossRef]

59. Mamdani, E.; Assilian, S. An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *Int. J.-Hum.-Comput. Stud.* **1999**, *51*, 135–147. [CrossRef]

60. Scikit-Fuzzy. Version 0.4.2. Python Software. 2019. Available online:. (accessed on 21 June 2025). [CrossRef]

61. Baum, L.E.; Petrie, T. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *Ann. Math. Stat.* **1966**, *37*, 1554–1563. [CrossRef]

62. Baum, L.E.; Eagon, J.A. An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes and to a Model for Ecology. *Bull. Am. Math. Soc.* **1967**, *73*, 360–363. [CrossRef]

63. Markov, A. Extension of the Limit Theorems of Probability Theory to a Sum of Variables Connected in a Chain. In *Dynamical Probabilistic Systems*; Iosifescu, M., Ed.; English translation of the original 1906 Russian publication; John Wiley & Sons: Hoboken, NJ, USA, 1971; Volume 1, p. 552.

64. Ching, W.K.; Ng, M.K. *Markov Chains: Models, Algorithms and Applications*; International Series in Operations Research & Management Science; Springer: New York, NY, USA, 2006. [CrossRef]

65. Berg, B.A. *Markov Chain Monte Carlo Simulations and Their Statistical Analysis: With Web-Based Fortran Code*; World Scientific Publishing Company: Singapore, 2004.