Informatica & Economie

A Spatiotemporal Analysis of Artefacts with Automated Methane Plume Detection using Decision Trees

Marnix Romeijn s2485133

Supervisors:
Mitra Baratchi & Julia Wąsala

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl 20/06/2024

Abstract

This thesis discusses the challenge of automated methane plume detection, focusing on the “retrieval artifact” occurring during this process. By utilizing data from the TROPOMI instrument on the Sentinel-5P satellite and using the Pearson Correlation Coefficients (PCC) between sensor channels to train a decision tree classifier, the study investigates if it can describe spatiotemporal patterns between artifacts and plumes. The results describe how decision trees can classify artifacts and plumes with reasonable accuracy, but that the approach suffers from instability, especially with the limited data used. The findings suggest more exploration is necessary in the field of artifacts occurring, and how the approach might be improved to help better understand and solve artifacts.

acknowledgment

This bachelor thesis has been made at Leiden Institute of Advanced Computer Science (LIACS), under the guidance and supervision of Julia Wąsala and Mitra Baratchi. I want to thank Julia Wąsala and Mitra Baratchi for the feedback and support during this challenging, but highly educational experience.

Contents

1 Introduction

As awareness of the impact of global warming grows, people of various organizations and nations are actively seeking new strategies to mitigate, or even stagnate, the rise of global temperatures. With carbon dioxide (CO_2) being the largest contributor to climate warming [NAS23], it is one of the potent greenhouse gases to consider when trying to reach that goal. When looking for short-term impacts and considering that molecules of CO_2 can persist for centuries, meaning that mitigating emissions of CO_2 has a very delayed effect on global warming, it is reasonable to also consider other options. Methane (CH_4) is another powerful greenhouse gas. CH_4 being the second-largest contributor to climate warming, and with a lifespan of about seven to twelve years in the atmosphere, is a potent option to achieve this goal [NAS23]. Besides CH_4 having a much shorter lifespan than CO_2, CH_4 also has an 86 times stronger warming impact than CO_2 per unit of mass over a period of 20 years [Cli23].

Over the last 200 years, the atmospheric CH_4 concentration has more than doubled, from roughly 800 parts per billion (p.p.b.) to an estimated 1932 p.p.b. measured in December 2023 [NAS23]. A reduction of CH_4 emissions can already significantly diminish its impact on the greenhouse effect within a decade [SMB+23]. One such way of achieving this is by finding sources of CH_4 emission. Using satellites equipped with specific instruments, like spectrometers, these sources can be identified. Especially finding so-called "super-emitters" has been a big focus. These super emitters consist, among other things, of oil and gas facilities, coal mines, and landfills [SMB+23]. According to the Dutch space research institute SRON [SRO23], we can significantly reduce the emissions from super-emitters with relatively simple measures. Using the Dutch TROPOMI instrument, these super-emitters can be automatically discovered. Using machine learning, together with these global maps, Schuit et al. [SMB+23] have now developed an algorithm that detects the methane plumes automatically. After getting a list of automatically detected methane plumes, these detected plumes are checked to make sure the detections are really plumes. Using other satellites with a higher resolution, persistent leaks are located. This information can be used by international and state actors, to take measures to diminish the emissions [SRO23].

One problem coming with this new technology is that supposedly a plume is detected, which after further investigation appears to be no real methane plume. These false plumes detected during the automatic detection are so-called "retrieval artifacts". Another way of referring to artifacts is often done by calling them "false positives". Other problems like clouds obscuring the view of the satellite, or not being able to detect plumes above water are also present. However, for the scope of this thesis, only solving artifacts will be discussed.

This bachelor thesis has been made at Leiden Institute of Advanced Computer Science (LIACS), supervised by Julia Wąsala and Mitra Baratchi. The aim of this thesis is to find temporal patterns that can be used to make it easier to distinguish real methane plumes from artifacts. Using spatial information, potential ways to find these temporal patterns are explored and reviewed. This all has been done with the guiding question:

"*Considering all relevant retrieval parameters, is there a significant difference between the spatiotemporal patterns when comparing real methane plumes with artifacts, taking correlations between* *methane concentrations and its supporting retrieval parameters for points of interest as a measure?*".

To answer this question, the following steps were taken.

- The collection of a dataset containing spatial information about locations marked as having frequent plumes, artifacts, or being empty, over a year.
- Filtering the data to ensure the quality of the data.
- Filtering of the data to sets of consecutive scenes retrieved from the dataset.
- Calculating the Pearson Correlation Coefficient (PCC) between the methane variable and supporting variable for every scene.
- Training decision tree classifiers on the consecutive scenes with PCCs calculated in the previous step.
- Validating the decision tree classifier by using 5x5 cross-validation, and taking the accuracy and standard deviation.
- The average cross-validation and standard deviation are used to describe if a classifier, if not overfitted, is reliable to use to describe spatiotemporal patterns within the data.

Section 2 discusses related works, like the automated plume detection discussed in Section 1, but also other relevant topics regarding the automated detection of greenhouse gases and artifacts. It also discusses a work related to the use of the Pearson Correlation Coefficient in image processing and comparison. Section 3 discusses details about the data retrieval, contents, and preprocessing. Section 5 discusses the experiments performed implementing the methods from Section 4. In Section 6 the results are discussed. Finally, in Section 7 a summary of conclusions and the thesis are given.

2 Related Work

This section describes related works. The work discussed is directly linked to this thesis, by either being the source of the problem, or an attempt to prevent or attenuate artifacts.

Automated methane detection: The source of artifacts comes from the automated detection of methane plumes. Schuit et al. [SMB+23] describes a study to automatically detect anthropogenic methane emissions by targeting super-emitters. These super-emitters are responsible for a large fraction of the total global methane emissions. By utilizing data from TROPOMI, a monitoring instrument on board the Sentinel-5P satellite, a two-step machine learning model was constructed. This machine learning model combined a CNN for plume-like structure detection and a Support Vector Classifier to distinguish real plumes from artifacts. The article describes that the machine learning model is very accurate and consistent, but still has some occasional artifacts. The paper is relevant to this research since the problem described in the paper is also explored in this research. This research aims to follow up on the artifact occurrences and describe patterns that lead to artifacts. The locations from which the data was collected for this research were also selected by using the map of plumes and data of artifact locations from this paper. Sánchez-García et al. [SGGnIL+22] describes how other satellites with much higher resolution can be used to detect methane plumes and allow for the pinpointing accurate locations of methane emitters. It is also mentioned that surface features with spectral signatures similar to methane plumes can complicate the detection of methane plumes.

Artifact attenuation: A work on the attenuation of artifacts has already been published by Roger et al. [RILG+23]. Artifact attenuation is the adjustment and filtering of certain data variables, causing a diminished or preventive effect of artifact within automated methane plume detection. This is a different method than Schuit et al. [SMB+23], which first filters out possible locations by using a CNN, and then checks the output for plumes using an SVC. Roger et al. [RILG+23] also describe the occurrence of retrieval artifacts when detecting methane plumes using data from the EnMAP and PRISMA missions. A Matched-Filter method is used to attenuate and sometimes practically remove retrieval artifacts. The method is tested on artifacts and actual plumes and shows that the method used causes fewer artifacts to be visible, making detection easier and more accurate. The method still allows for the detection of actual methane plumes. The difference between Roger et al. [RILG+23] and this project, is the main aim of the research. Roger et al. aim to decrease the effects of artifacts, while this research aims to find patterns within the data that can explain why artifacts occur. This research also aims to provide insight into artifacts to help efforts to reduce or prevent artifact occurrences in automated methane detection.

3 Data

This section describes the data and data processing. This includes the origin, structure, and processing of the data. Before using the data, it needs to be processed according to filters for the image size, percentage of methane pixels, and consecutive sets. Also, the Pearson Correlation Coefficient needs to be calculated. Finally, to potentially improve the decision tree classifier, two extra variables are added to every data point in the dataframe. The process has been visualized in Figure 3.

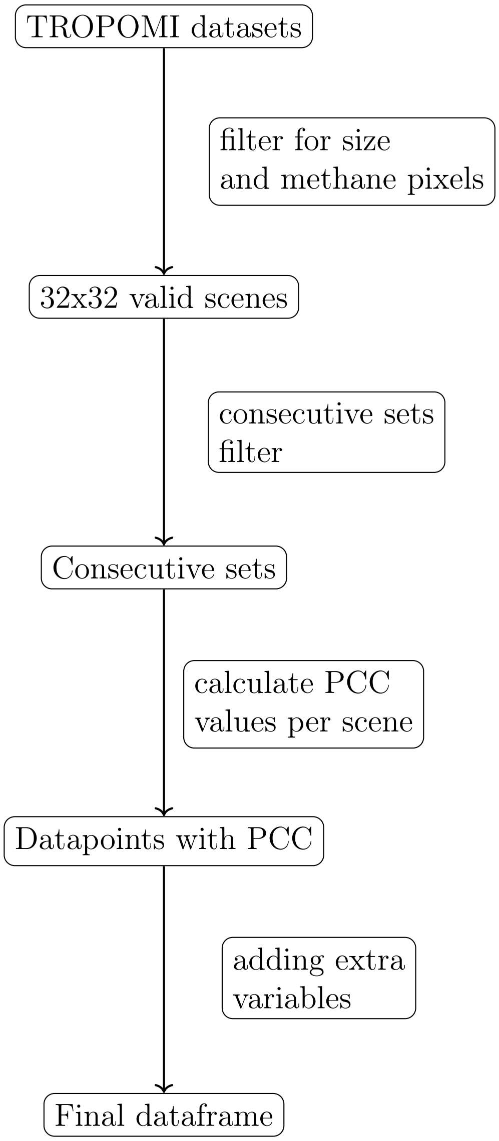

Figure 1: visualization of the steps taken to process data into a dataframe containing the PCCs of sets of consecutive scenes to train a decision tree classifier with.

Attribute	Description
Instrument	Spectrometer TROPOMI
Mission	Sentinel-5P
Data Level	Level 2 data products
Product Version	1.5.0
Spatial Resolution	5.5x7.0 km^2
Data Source	Available for download through the Copernicus website [Cop24]
Data destriping algorithm	Destriped using the algorithm described by Hasenkamp et al. [HLH+22]
Quality Control	Filtered based on methods described by Schuit et al. [SMB+23]

Figure 2: Metadata summary of the data used for the experiments performed in this research.

3.1 About the Data

To allow for a better understanding of the preprocessing and how the experiments are performed, a description of the data is given in this subsection.

3.1.1 Data Origin

The data used for this thesis has been captured by a spectrometer on the Sentinel-5P satellite, called TROPOMI [Age24], containing Level 2 data products and product version 1.5.0. The data contains images with a 5.5x7.0 km^2 spatial resolution. The data can also be downloaded through the Copernicus website [Cop24]. In addition, the data has been destriped using the algorithm described by Hasenkamp et al. [HLH+22]. The selection of plume, artifact, and empty locations was based on the detection map shown by Schuit et al. [SMB+23]. To ensure the quality of the data, the data retrieved has gone through a filtering process also described by Schuit et al. [SMB+23].

3.1.2 Data Structure

The data contains information about 30 different locations, 10 for each type of location (artifact location, empty location, plume location). The data included daily captures, between 1 January 2021, to December 31, 2021. Every item in the dataset contained a dataset with various coordinates, variables, and attributes. Table 1 shows the variables used in the dataset. Additional information about the variables can be found in Appendix Table 3 and in the technical report written by Arnoud et al. [APS+23].

From all shown variables, the "methane mixing ratio" variable is the reference variable. All other variables are used as the correlate variables. For extra clarification on the meaning of the variables, see Table 3 in the appendix.

Variable
methane mixing ratio
QA value
surface pressure
aerosol optical thickness measured in the SWIR spectrum
aerosol optical thickness measured in the NIR spectrum
surface albedo measured in the SWIR spectrum
surface albedo measured in the NIR spectrum

Table 1: Attributes of the specified variables in the dataset. For a description of the attributes see Appendix Table. 3

3.2 Preprocessing

This subsection discusses the preprocessing of the data, a required prerequisite of the experiments.

3.2.1 Data Filtering

To ensure the quality of the data, two main criteria were used to determine if a capture of a scene was considered a valid source of data. These criteria are based on the same criteria employed as discussed in Section 2.1 TROPOMI by Schuit et al. [SMB+23]. The first criterion the data was filtered on was if the data consisted of at least a 32x32 pixel capture, also considered as at least a 32x32 data array. If the scene has a smaller than 32x32 pixel capture, the scene is discarded. Afterward, a check for the second criterion was done after going through the regridding process. The second criterion was based on the ratio of valid pixels for the methane variable. If the ratio of valid pixels was 20% or lower, the scene was also discarded. Reasons for missing pixels include cloud cover, water, and discarded pixels due to quality measures. Only if a scene complied with both criteria, the scene was considered for regridding and concatenation with other valid scenes for that location.

3.2.2 Regridding

To be able to compare the different time captures of a location with each other, the images of the scenes had to go through a process of realignment, also known as regridding. The regridding of scenes involved creating a baseline grid. This baseline grid was constructed using the function `xe.util.cf_grid_2d` from the library `xesmf` [Z+24]. The baseline grid required the extremes for the available coordinates of every location, and the maximal amount of pixels for a location available on both the longitude and latitude axis. It also required the average size for every pixel to be described, by dividing the range of the longitudes and latitudes by the number of pixels on the x- and y-axis. This implies that the baseline grid exists out of a data array with its size based on the scenes with the highest longitude and latitude values. Therefore, every scene available for a location should fit within the grid based on their latitude and longitude values.

By using the Regridder from the function `xe.Regridder` from the `xesmf` library [Z+24], every scene from a location was fit into the baseline grid. Because the scenes have all been regridded based on the same baseline grid, it is possible to select the 32x32 grid around the center coordinates of the picked location, covering the exact same area for every scene. To offer a better understanding of the input and output of this process, an example of a regridded scene with some variables is shown in Figure 3. This example contains a coastline, offering a great indication of how the image looks after regridding. Figure 4 shows how all regridded images have the same x- and y-axis and view data from the same perspective.

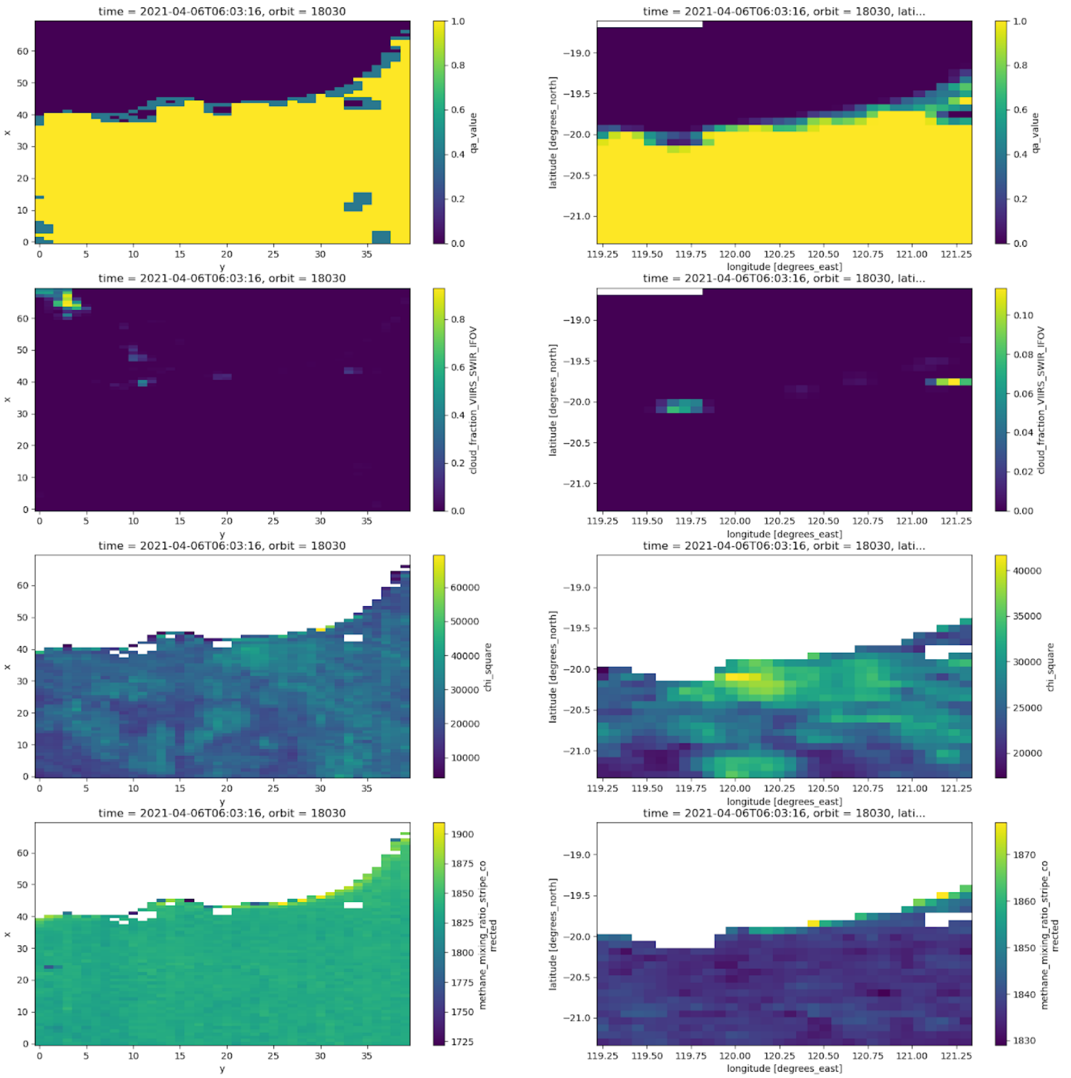

Figure 3: Example showing the plotted original data (left) and product of the regridding process (right) for a scene containing a coastline. The raw data for the scene has much more pixels than a 32x32 pixel area. The Regridder allows for focusing on a specific part of a raw scene and bases this on a standardized grid, giving all processed scenes the same output grid.

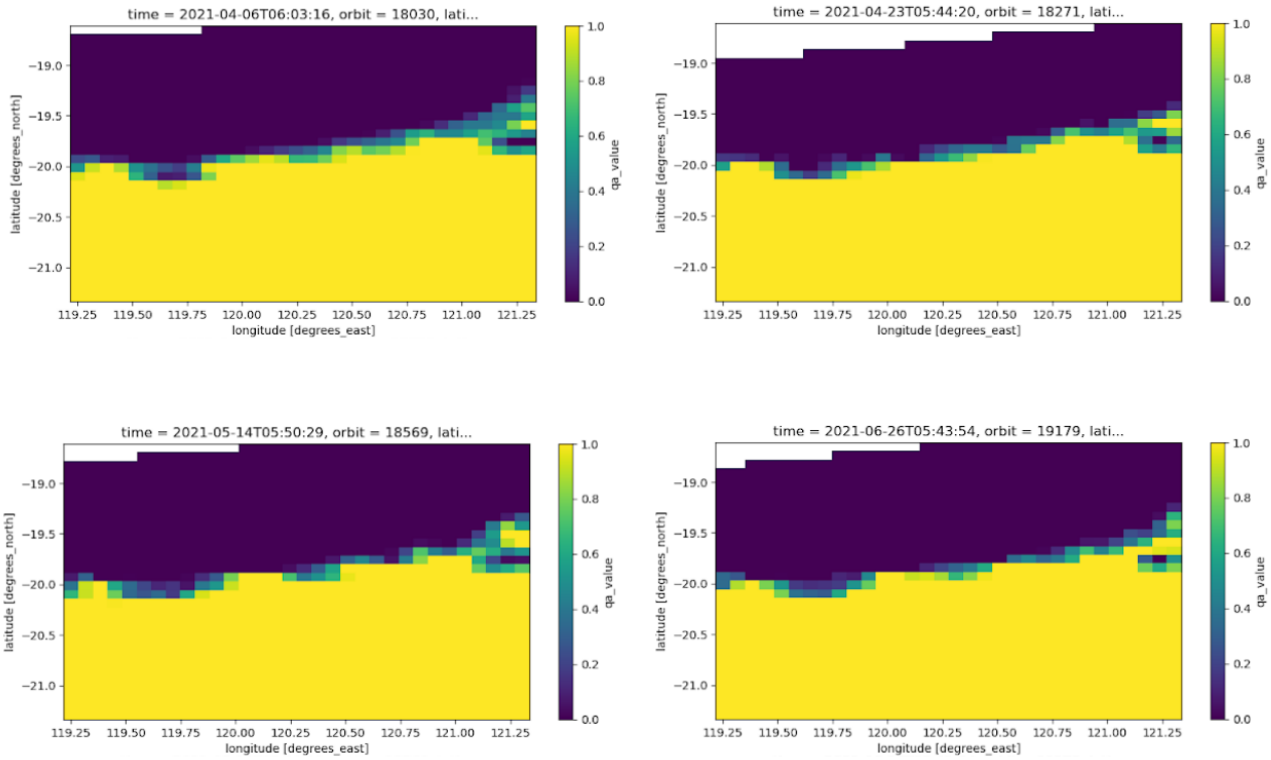

Figure 4: Plots for different times showing the product of the Regridder.

3.2.3 Sets of Consecutive Scenes

For the main experiments, sets of consecutive scenes were required. When exploring the data, it was visible that sometimes plumes and artifacts were occurring for multiple consecutive days. Therefore, consecutive sets of scenes are used to explore if it is possible to find spatiotemporal patterns in these consecutive sets. Every set of consecutive scenes contains n scenes, with the set being one single data point in the dataset used for the experiments. Two variants of consecutive sets were filtered out of the data. The first variant is sets of three consecutive scenes, which have been captured consecutively, meaning there were less than 24 hours between every capture within the set. The second variant of sets is sets of two consecutive scenes, which have been captured consecutively. For more clarification on what the set of scenes contains, a graph showing the structure of such consecutive sets is shown in Figure 5.

The reason for using the sets of consecutive scenes is because this way some temporal characteristics will be contained within every data point, while also having the spatial characteristics. The reason only two or three consecutive scenes were considered, is because considering more than three consecutive scenes would yield in a very small dataset of sets. Having too few data points would mean it would not be possible to reliably train and test a classifier.

Before selecting the sets, the availability of regridded scenes per location was checked. After checking the consistency of available scenes, and the total number of scenes available for a location, some

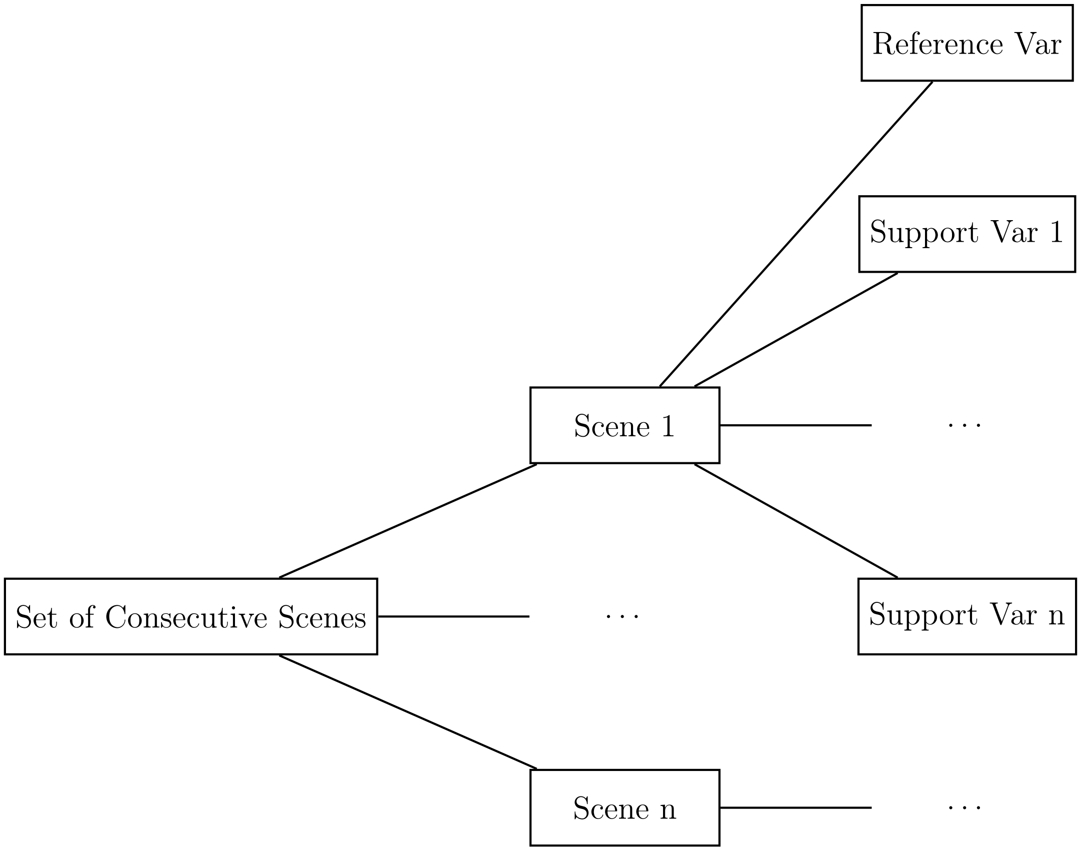

Figure 5: Illustration of the structure of one set of consecutive scenes.

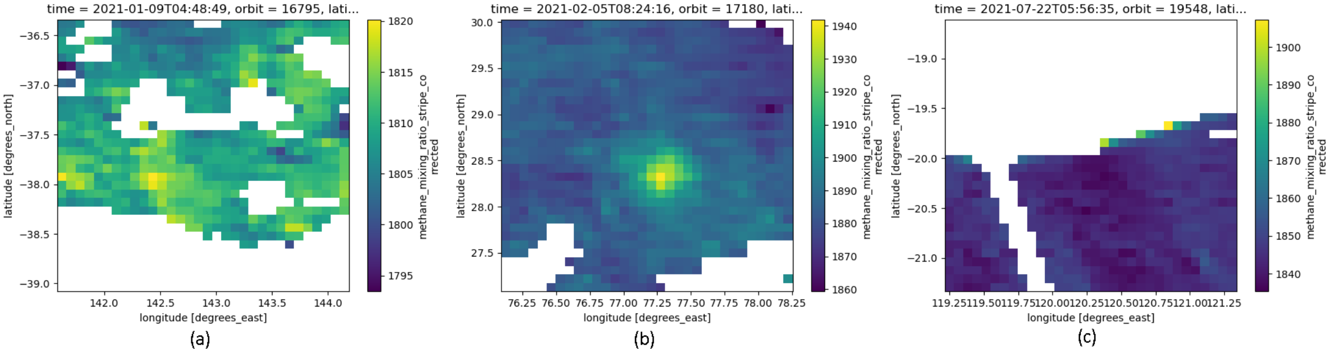

Figure 6: (a) A plot of the methane variable of an "empty" location in Australia. (b) A plot of the methane variable of a visible "plume" in Algeria. (c) A plot of the methane variable of a coastal "artifact" in Australia. As seen in Figure (c), it looks like there is a plume along the coast. In reality, this is not the case.

locations were dropped. The scenes within these sets were then manually labelled for further use.

3.2.4 Manual Labelling

As shown in Appendix section C.1, locations are referred to by an ID being an "artifact", "empty" or "plume". This does not indicate that all collected data for that particular location complies with that label. Therefore, after filtering out scenes to construct sets, the scenes had to be labelled. Labeling was mainly done using the plots for the methane variables. For plumes, very clear patterns were often visible in this data. For artifacts, mainly plots of the methane variable, northward wind variable, and eastward wind variable were used to identify artifacts, as well as the knowledge of artifacts always looking very similar. When unsure of being a plume or artifact, the scene was discarded. To provide some insight on how the different labels may look like, an example is shown in Figure 6.

After all scenes were labelled, the previously constructed sets were checked again. If a set consisted out of more than one label, the set was discarded. After this check, there were 244 sets of 3 consecutive scenes and 492 sets of 2 consecutive scenes. These sets are further used in the experiments. Before the experiments can be performed, the correlation between the dependent variable and all the support variables has to be performed. To do this, the data product of all preprocessing mentioned before will be used. This results in a time series of correlations for every location.

3.2.5 Preparing Variables for Calculating Correlation

The reason for trying to find similarity between variables of one scene is to explore if there is a distinct difference or pattern between certain sensor channels for plumes and artifacts, which allows for filtering out artifacts from detected plumes. One way of finding similarity between images is using the Pearson Correlation Coefficient (PCC). When calculating the correlation between the dependent variable and its supporting variables with the package `xr.corr` from the library `xarray` [HH20], normally all valid pixels will be used for each data grid, but since there are missing pixels, every image needed to be checked. Because the dependent variable often has missing pixels where other variables still have valid pixels, every pair of dependent variable and support variable needs to be corrected for this difference since you cannot calculate a correlation over NaN-values. This is done because the calculation of the correlation gets impacted by the complete data arrays. If the first data array has a NaN value and the second data array has a valid pixel, the valid pixel will influence the outcome of the correlation calculation, while the NaN pixel will not. Calculating the correlation will be further discussed in Section 4.

3.3 Final Dataframe Contents

After performing all preprocessing steps, the final dataset contained data points with correlations between the reference variable and support variables. For the sets of two consecutive scenes, the correlations of two scenes are included within a single data point, for sets of three consecutive scenes the correlations for three scenes are included within a single data point. Besides the correlations, the range of the methane variable and the maximal methane value of every scene within a set of consecutive scenes were included.

4 Methods

This section discusses all methods used in the experiments. The goal of the research is to find if it is possible to use a decision tree classifier to find spatiotemporal patterns, by training and testing the classifier with data containing the PCC of the mentioned variables in 1. Here, a decision tree classifier is used because it is possible to plot, and allows for easy insight into the decisions made by the classifier on the test set. If the decision tree classifier can reliably point out artifacts from plumes, the methods might turn out to provide valuable insight into the reasons why artifacts sometimes occur. First, the Pearson Correlation Coefficient is discussed. After that, the decision tree classifier is discussed. All these methods were used for the sets of two consecutive scenes and three consecutive scenes.

4.1 Pearson Correlation Coefficient

For the experiments, the data was used to calculate the Pearson Correlation Coefficient (PCC), denoted as r, of two data arrays (the data array of the dependent variable and a supporting variable) from the same scene. The PCC method is used for statistical analysis, pattern recognition, and image processing [MNVF+13]. The PCC used will be related to image processing, by functioning as a disparity measure. For all sets and every scene within each set, the PCC will be calculated between the dependent variable and each supporting variable, after preprocessing each pair of data arrays as discussed in Section 3.2.5. The PCC is given by:

$$(PCC) = r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$$

where:

- x_i and y_i are the pixel value at the same location i in data arrays x and y,
- $\bar{x}$ and $\bar{y}$ are the mean pixel value of data arrays x and y,
- n is the total number of pixels in each image.

The value of the PCC (r) ranges from -1 to 1. A value of:

- $r = 1$ indicates a perfect positive linear relationship between x and y,
- $r = -1$ indicates a perfect negative linear relationship, between x and y,
- $r = 0$ indicates that there is no linear relationship between x and y.

To calculate the PCC for every pair of data arrays, the function `xr.corr` from the library `xarray` is used. This is done by using the code:
`corr = cov / (da_a_std * da_b_std)`,

which translates to:

$$r = \frac{\text{cov}(array_a, array_b)}{\text{std}(array_a) \cdot \text{std}(array_b)}$$

This is the same as the method used by Neto et al [MNVF+13]. By using the PCC, various locations with plume and artifact detections will be compared and used to explore the possibilities of finding spatiotemporal patterns that allow for the distinction between artifacts and plumes.

After calculating the PCC for a pair of data arrays, the value was added to a set of other PCCs related to the set of consecutive scenes. Each value was given the variable name of the supporting variable and whether it was calculated from the first, second, or third scene from a set.

4.2 Decision Tree Classifier

To find potential patterns using the PCC, a decision tree classifier was used. The reason a decision tree was used, is its interpretability. A stable classifier, even with moderate accuracy, could be used to find spatiotemporal patterns, even if this is only applicable to a fraction of the artifacts. By visualizing the decision tree and analyzing it together with the dataset used to train and test the classifier, decisions leading to nearly pure leaves can be described as a pattern. The interpretability is important because the main goal is not to find a classifier to improve artifact detection, it is used to automate the detection of spatiotemporal patterns. The sets of PCC values were each used as a data point to train and test the decision tree model. For the implementation, the library `scikit-learn` in Python was used, which supports the use, training, and testing of a decision tree classifier model. `Scikit-learn` implements the CART algorithm to build decision tree classifiers [sci24a]. The CART decision tree model is an algorithm acting greedily at every node, working as follows [BFOS84]:

1. **Binary Splitting**: At each node, the data is split into two child nodes based on a threshold value of a feature, aimed at maximizing the quality of the target variable within each child node.

2. **Recursive Partitioning**: This splitting process is applied recursively, creating a binary tree structure. Each internal node represents a decision based on a feature, and each terminal node (leaf) represents a class label.

3. **Impurity Measures**: For classification trees, the impurity of a node can be measured using "Gini impurity" or entropy as a measure. The algorithm chooses splits that minimize the impurity in the resulting child nodes.

In addition, the `scikit-learn` library adds the following feature to the CART algorithm [sci24b]:

1. **Pruning**: the constructed tree may be pruned by removing nodes that provide little predictive power. This is often done using the technique "cost-complexity pruning". This method of pruning balances tree complexity with prediction accuracy.

To test the validity of the decision tree, 5x5-fold Cross-Validation (CV) was performed over the test set. To find the optimal parameters, the function `GridSearchCV` from the library `scikit-learn` was used.

The aim is to use a decision tree for finding spatiotemporal patterns for artifacts and plumes which could provide insight in preventing the occurrence of artifacts. It is also important to note that any decision tree able to accurately classify between artifacts and plumes is considered better than any decision tree which classifies between "empty" and artifacts. The reason for this is that any artifact classified as "empty" should be an acceptable (but not favorable) error since artifacts and plumes were originally both classified as plumes. However, plumes classified as artifacts are not acceptable, since the aim of the classifier is to filter out as many artifacts to be able to show any potential patterns. Plumes classified as artifacts would lead to an unreliable decision tree, which is therefore also unreliable for discussing any patterns. If the classifier classifies artifacts as plumes a lot, it indicates that the problem of artifacts in plume recognition is much more complicated and requires other, more advanced methods. When the decision tree is not accurate enough, or not stable enough, no reliable conclusions can be made using the decision tree.

5 Experiments

This section discusses the experiments performed to find if it is possible to find any patterns using the PCC. All experiments are performed using an 80-20 train-test split. Besides being a standard practice in computer science, an 80-20 train-test split was chosen because this would train on a relatively big amount of data points, while still maintaining enough data points to verify the performance and consistency of the classifiers. If the classifiers perform well, additional data might be needed to test the classifier again, to make sure the results are not biased. Every experiment is performed on both the dataset with two and three scenes in a single data point. The difference between every experiment is the labelled data in both the training and test sets. Before training a "cross-validation grid search" is performed for finding the most suitable parameters for the decision tree classifier. Also, a 5x5-fold cross-validation is performed before training. This is done to grant better insight into the accuracy and consistency of the tried experiment. 5-fold cross-validation was chosen because of the low number of data points. The 5-fold cross-validation is performed 5 times with a different random state, providing a more accurate cross-validation score and standard deviation. This means 25 runs of cross-validation have been done, to be able to picture a better understanding of the performance of the classifiers, while dealing with a low number of data. The Github repository containing the code used for this experiment can be found in the link shared in Appendix For insight into the code used to perform these experiments,

5.1 Testing and Training with All Labels

This experiment features no adjustments to the training and/or test set. Ideally, the classifier predicts all labels accurately, however, it does not matter much if the classification of artifacts and empty locations gets mixed up by the classifier. This means that an "empty" data point gets labelled as an artifact, which can be considered acceptable because data points as "empty" and artifact are both supposed to be "empty" data points. This also works the other way around, for artifacts being misclassified as "empty". On the contrary, this would indicate that there is much similarity in correlations between artifacts and empties, and therefore it should be possible to retrieve patterns from the decision tree.

Parameter	Value
Criterion	Gini
Max Depth	None
Min Samples Leaf	4
Min Samples Split	2

Figure 7: Parameters for the decision tree classifier for the experiments with all labels (both for sets of two and sets of three consecutive scenes).

5.2 No Empties in Data

This experiment features a dataset with no data points labelled as “empty”. By doing this, training and testing is only done on artifacts and plumes. This was done for several reasons. It could provide insight into how many data points labelled as “empty” are useful or can be considered unnecessary for the classifier. Also, since the labels and artifacts are all detected as plumes, training and testing on only plumes and artifacts might yield better results compared to the training and testing on the dataset which also contains data points labelled as “empty”. The reason for this is that the real problem is the distinguishment between plumes and artifacts. Therefore, the “empty” data points could be considered as noise or might interfere with solving this problem. This experiment is to test the performance when “empty” data points are removed. The reason for trying this is because the real problem is the distinguishment between plume and artifact. Comparing this experiment with the other experiments could also provide more insight into when you want to use the classifier.

Parameter	Value
Criterion	Gini
Max Depth	None
Min Samples Leaf	1
Min Samples Split	2

Figure 8: Parameters for the decision tree classifier for the experiments with no “empty” labels (both for sets of two and sets of three consecutive scenes).

6 Results

This section discusses the results of the performed experiments. The results of the best-performing experiment will be discussed on a more in-depth level, considering that it will be the best opportunity to find any patterns in the correlation between artifacts and plumes.

Dataset	Case	# data points	$\mu \pm \sigma$	Extra Vars
Two Consecutive Scenes	All Labels	492	0.636 ± 0.045	✓
			0.568 ± 0.047	□
	No Empties	451	0.829 ± 0.043	✓
			0.809 ± 0.059	□
Three Consecutive Scenes	All Labels	244	0.699 ± 0.078	✓
			0.604 ± 0.082	□
	No Empties	209	0.793 ± 0.087	✓
			0.795 ± 0.073	□

Table 2: Combined Cross-Validation results of all experiments. μ indicates the average accuracy over the 5x5 cross-validation, while σ indicates the standard deviation. The “Extra Vars” column indicates whether the data points included the range and maximum value of the methane variable as a variable per scene, as described in Section 3.3.

6.1 Testing and Training with All Labels

This section discusses the results of the experiments performed with the datasets with all labels. Considering Table 2, experiments using the datasets including the extra variables for the range and maximum of the methane ratio (see Section 3.3) show better results than when only using the PCC for classification. Therefore all discussed results will be about the results using the datasets including these extra variables.

6.1.1 Sets of Three Consecutive Scenes

For the classifier trained and tested on sets of three consecutive scenes, the CV test results show a mean accuracy of 70% and a standard deviation of 7.8%. This indicates that the decision trees constructed by the classifier can differentiate between the different labels in quite some situations. It should be noted that the standard deviation of 7.8% is quite high. Since a decision tree classifier is used, it is necessary to have a more stable model, meaning a much lower standard deviation, since describing the patterns found in one decision tree would need a stable classifier. If we were to use the unstable decision tree classifier, it would lead to unreliable prediction.

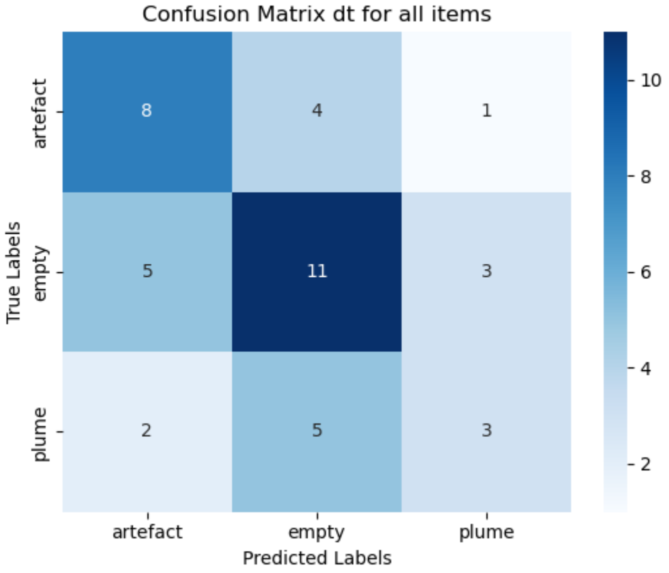

Figure 9: Confusion Matrix for testing and training with all labels with sets of three consecutive scenes. The matrix represents the results on the test set, with an accuracy of 52.3%, which is much lower than the mean CV accuracy score for this case. When artifacts and empties are considered the same, the accuracy is 73.8%.

The instability of the model can also be seen in the confusion matrix of the model on the test set in Figure 9. The figure also shows that taking the same measure as taken in the cross-validation, the mean accuracy is very low, and about the same as the mean accuracy when considering empties and artifacts as the same label. This can however be explained by the high standard deviation. The confusion matrix shown could also be considered to not be as reliable as required. The standard deviation from the CV tests shows that the outcome of testing the test set is influenced significantly by the test and training split. Considering everything mentioned in this section, the decision trees constructed in this case are deemed unreliable, and will not be used to describe spatiotemporal patterns. The reason a “better” decision tree model is sought in this case, is because this is considered overfitting, while the goal is to have a reliable, and suitable on a global scale, decision tree that allows for the discussion of spatiotemporal patterns, which could also be applicable on other locations not included in the training and test data.

6.1.2 Sets of Two Consecutive Scenes

The CV test results of this experiment show a mean accuracy of 63.6% and a standard deviation of 4.5%. This shows that the decision tree classifier in this case is performing worse than in the previous case with three consecutive scenes. A reason for this could be the lower amount of data available in the case of three consecutive scenes, leading to overfitting. The stability can be considered as an improvement in this case. A reason for an improvement in stability could be the fact that the number of data points in the training and test sets are more than double compared to the amount of data points in the case with three consecutive scenes.

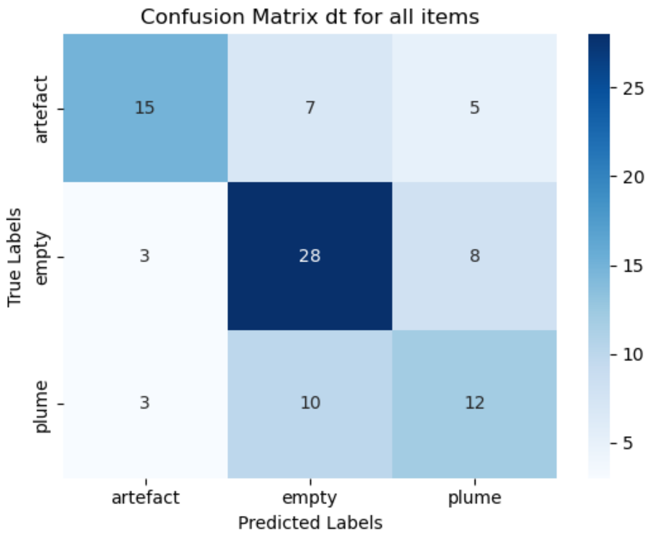

Figure 10: Confusion Matrix for testing and training with all labels with sets of two consecutive scenes. The matrix represents the results on the test set, with an accuracy of 60.4%, which is relatively close to the mean CV accuracy score for this case. When artifacts and empties are considered the same, the accuracy is considered to be 71.4%.

Figure 10 shows the confusion matrix for the case with all data and two consecutive sets. While showing an accuracy closer to the mean accuracy of the CV tests, it is quite low. Also, even though the standard deviation of the CV tests is relatively low compared to the case with three consecutive sets, it still indicates that the model is quite unstable. Yet, it may still be considered a valid option for finding spatiotemporal patterns, since an increase in data points yields an increase in stability. Another option could be the removal of noise in the data. Since the problem of artifacts is only between plumes and artifacts, "empty" locations could be considered as noise. Therefore, in the next experiment, the data points with the "empty" label for each case are removed. this does however reduce the number of data points but might yield a much more stable decision tree. In this case, the decision tree might not contain all spatiotemporal patterns, but if some leaves are near pure in the decision tree, the path to these pure leaves could be used to identify at least some spatiotemporal patterns.

6.2 No Empties in Data

Training a decision tree classifier on data containing all the labels shows potential but still lacks the stability to reliably show spatiotemporal patterns. One reason for this may be the fact that there is a lot of noisy data in the datasets. In this experiment "empty" data points are considered as noise. Since the problem of artifacts only occurs in scenes where there is a potential plume, it seems prudent to have the experiments performed without training and testing on data that also contains data with "empty" labels. This experiment aims to test if removing the "empty" data points yields significantly more stable decision tree models, allowing the use of those models for describing spatiotemporal patterns by analyzing the path of (nearly) pure leaves. To be able to find these spatiotemporal patterns, it is required to be much more stable than the decision tree models in the previous experiment. Besides stability, considering both Figure 9 and Figure 10 The removal of the "empty" data points also improves the accuracy of the models. Therefore testing and training a classifier might combine a more stable model with a more accurate model.

6.2.1 Sets of Three Consecutive Scenes

The CV test results of this case show a mean accuracy of 79.3% and a standard deviation of 8.7%. Compared to the case in the previous experiment, this is a much higher mean accuracy, showing improvement in the classification capabilities of the decision tree classifier. It should however be noted that the standard deviation also increased.

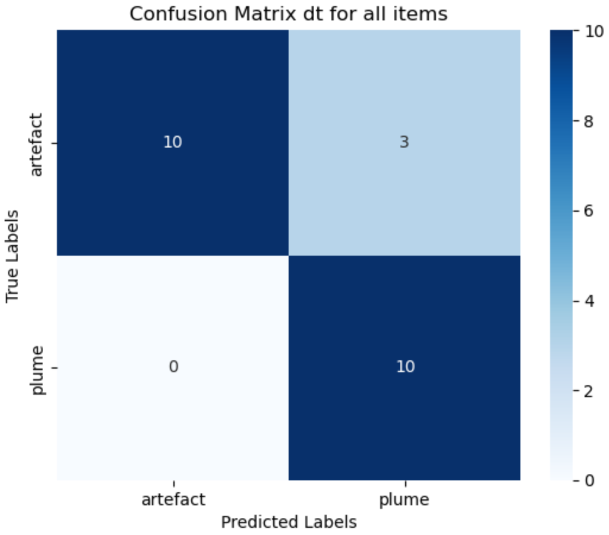

Figure 11: Confusion Matrix for testing and training with no empties in the data of sets of two consecutive scenes. The matrix represents the results on the test set, with an accuracy of 87.0%, which is above the mean CV accuracy score for this case.

Considering the confusion matrix in Figure 11, together with the CV scores, it may be said that the model is capable of classifying artifacts and plumes. In the confusion matrix it shows to always classify plumes, and only misclassifies artifacts. It should however be noted that because of the high standard deviation and the low number of data points in the dataset, the confusion matrix shown is probably biased. Since the aim was to find a more stable decision tree model, it can be said that while improving overall classification probabilities, the stability of the classifier decreased, because the standard deviation also increased. A reason for the instability may be caused by the lack of data, as seen in the previous experiment, where the case with three consecutive scenes had a much higher standard deviation while having fewer data points to train and test on than the case with two consecutive scenes. Therefore, it can be considered sensible the experiment is performed on the case with two consecutive scenes.

6.2.2 Sets of Two Consecutive Scenes

The last experiment, performed on the dataset with sets of two consecutive scenes, yields the most promising results The CV test results show a mean accuracy of 82.9% and a standard deviation of 4.3%. This means there is a big increase in accuracy for the case with two consecutive scenes when removing the “empty” data points.

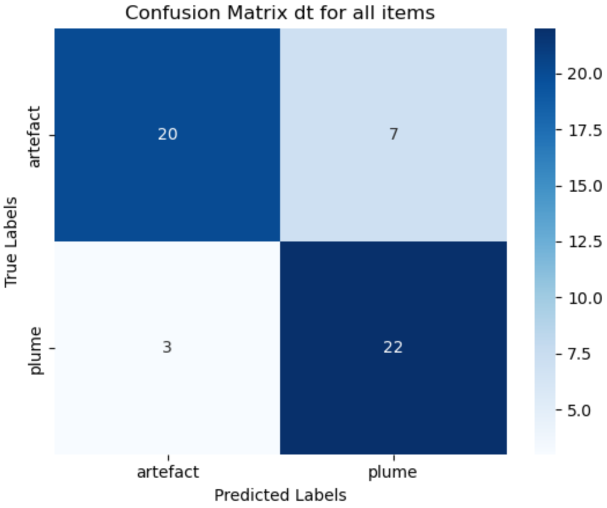

Figure 12: Confusion Matrix for testing and training with no empties in the data of sets of two consecutive scenes. The matrix represents the results on the test set, with an accuracy of 80.7%, which is slightly under the mean CV accuracy score for this case.

Considering the confusion matrix in Figure 12, it may be said that the classifier is capable of classifying artifacts and plumes, with only a few misclassifications for both plumes and artifacts. The standard deviation, regarding the standard deviation of the other experiments, is also the lowest. It should however be noted that the standard deviation still shows that the decision tree models in this case are unstable. Because of this, and since there is still a low number of data points available to train and test on, the decision tree models are still not reliable enough to be used to describe any spatiotemporal patterns.

6.3 Discussion

The results show a promising result in a decision tree being able to classify artifacts and plumes when using the PCC between various sensor channels and the methane ratio of sets of consecutive scenes as data points. The case using sets of two consecutive scenes, without data points labelled as “empty”, yielded the best results, both in mean accuracy (82.9%) and stability described by the standard deviation (4.3%). It should be noted that a pattern is visible after performing the experiments. Less scenes used in a data point to construct variables seem to lead to a higher stability in classifier performance. However, this also influenced the amount of data points, which may be the biggest reason for the instability of the classifiers. Therefore the results indicate that the use of decision trees with PCCs to describe any spatiotemporal patterns yields inconsistent and unfavorable decision trees when using too little data to train and test on. Here, an unfavorable decision tree is not necessarily a decision tree with a relatively low accuracy, but a decision tree constructed by a classifier that is deemed unstable. Another reason for this instability, besides the low number of data points available, is that decision trees are easy to be overfitted [HTF09]. Besides this, changes in the training set can lead to much different performing decision trees [BFOS84], as is shown in the high standard deviations of the experiments.

7 Conclusions and Further Research

Researching the distinguishment of real methane plumes with artifacts using the Pearson Correlation Coefficient (PCC) and multiple scenes within one data point for training a decision tree classifier has shown several insights. While the methods used in this thesis show that it is possible to classify plumes and artifacts, it also shows how having the PCC between various sensor channels from multiple scenes within one data point has a negative impact on the accuracy and stability of the classifiers. Reasons for this are the fact that it is a high dimensionality problem, limited data points were used, and the fact that decision trees are sensitive to changes in the training set. It could be possible that using a decision tree for finding spatiotemporal patterns is not favorable, since the decision tree picks a PCC between sensor channels from either the first, second, or third scene in this case. There is no regard for the actual temporal sequence in the data points. Trying to find spatial patterns combined with temporal characteristics is a high-dimensional problem. This may indicate the use of other, more advanced methods are needed, which can consider several rules, to take both temporal and spatial dimensions into account when classifying. These methods allow for much more accurate and stable classification, like the method described by Roger et al. [RILG+23]. The problem with these methods is that the decision-making often tends to be less interpretative and does not easily allow for insight into decision-making, besides which classification has been given. It is also that spatiotemporal analysis should be performed by not using a classification model, but with a more in-depth analysis of spatial features of locations, their impact on being classified as an artifact at certain timestamps, and what other effects that get captured by the TROPOMI satellite lead to classification.

References

[Age24] European Space Agency. Sentinel-5p mission, 2024. Accessed: 2024-05-24.

[APS+23] Arnoud Apituley, Mattia Pedergnana, Maarten Sneep, J. Pepijn Veefkind, Diego Loyola, Otto Hasekamp, Alba Lorente Delgado, and Tobias Borsdorff. Sentinel-5 precursor/tropomi level 2 product user manual methane. Technical report, SRON Netherlands Institute for Space Research, September 2023. Version 2.6.0.

[BFOS84] Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone. *Classification and Regression Trees*. Wadsworth International Group, 1984.

[Cli23] Climate and Clean Air Coalition. Methane. https://www.ccacoalition.org/short-lived-climate-pollutants/methane, 2023. Accessed: 2024-05-17.

[Cop24] Copernicus Sentinel-5P. Copernicus Sentinel-5P Data Collection. Available online, Accessed 2024.

[HH20] S. Hoyer and J. Hamman. *Xarray: N-D labeled arrays and datasets in Python*. Xarray Development Team, 2020.

[HLH+22] O. Hasekamp, A. Lorente, H. Hu, A. Butz, J. Aan de Brugh, and J. Landgraf. Algorithm Theoretical Baseline Document for Sentinel-5 Precursor methane Retrieva, SRON The Netherlands Institute for Space Research, Leiden, the Netherlands, 2022.

[HTF09] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, 2009.

[MNVF+13] Arthur Miranda Neto, Alessandro Corrêa Victorino, Isabelle Fantoni, Douglas Eduardo Zampieri, Janito Vaqueiro Ferreira, et al. Image processing using pearson's correlation coefficient: Applications on autonomous robotics. In *13th International Conference on Mobile Robots and Competitions (Robotica 2013)*, pages 14–19, Lisbon, Portugal, April 2013.

[NAS23] NASA. Methane. https://climate.nasa.gov/vital-signs/methane/?intent=121, 2023. Accessed: 2024-05-17.

[RILG+23] Javier Roger, Itziar Irakulis-Loitxate, Javier Gorroño, Adriana Valverde, and Luis Guanter. The improvement of methane plume detection with high-resolution satellite-based imaging spectrometers. *Environmental Sciences Proceedings*, 28(1), 2023.

[sci24a] scikit-learn . *1.10. Decision Trees*, 2024. Accessed: 2024-05-27.

[sci24b] scikit-learn. 1.10. decision trees — scikit-learn 1.5.0 documentation, 2024. https://scikit-learn.org/stable/modules/tree.html#minimal-cost-complexity-pruning.

[SGGnIL+22] E. Sánchez-García, J. Gorroño, I. Irakulis-Loitxate, D. J. Varon, and L. Guanter. Mapping methane plumes at very high spatial resolution with the worldview-3 satellite. *Atmospheric Measurement Techniques*, 15(6):1657–1674, 2022.

[SMB+23] Berend J Schuit, Joannes D Maasakkers, Pieter Bijl, Gourav Mahapatra, Anne-Wil Van den Berg, Sudhanshu Pandey, Alba Lorente, Tobias Borsdorff, Sander Houweling, Daniel J Varon, et al. Automated detection and monitoring of methane super-emitters using satellite data. *Atmospheric Chemistry and Physics*, 23(16):9071–9098, 2023.

[SRO23] SRON Netherlands Institute for Space Research. Methane super-emitters revealed weekly using satellites and machine learning. https://earth.sron.nl/nieuws/methane-super-emitters-revealed-weekly-using-satellites-and-machine-learning/, 2023. Accessed: 2024-05-24.

[Z+24] Jiawei Zhuang et al. xesmf: Universal regridder for geospatial data, 2024. Accessed: 2024-06-07.

A Tables

A.1 Variable Descriptions

Variable	Description
methane mixing ratio	Probability density function of the CH_4 dry air mixing ratio, stripe corrected
qa value	Continuous quality descriptor, varying between 0 (no data) and 1 (full quality data)
surface pressure	Pressure at surface elevation of S5P SWIR pixel.
aerosol optical thickness SWIR	Retrieved aerosol optical thickness in the SWIR band
aerosol optical thickness NIR	Retrieved aerosol optical thickness in the NIR band
surface albedo SWIR	Retrieved surface albedo in the SWIR band
surface albedo NIR	Retrieved surface albedo in the NIR band

Table 3: Variable descriptions, based on [APS+23].

B Github Repository

Here you can find the link to the Github repository containing all code used for the experiments:

https://github.com/maximilionis/bsc-experiments

C Additional Materials

C.1 CSV File: locations.csv

The locations file contains the following:

```
id,type,lat,lon,note,subtype
e0,empty,-37.650806,142.908026,australia,
e1,empty,-9.888889,18.835594,angola,
e2,empty,11.516510,-7.180029,mali,
e3,empty,26.861925,100.220530,yunnan,
e4,empty,53.546162,-66.277310,newfoundland,
e5,empty,51.769221,5.272029,NL,
e6,empty,69.983779,151.697808,russia,
e7,empty,-6.097547,-60.348648,amazone,
e8,empty,-37.892741,-72.477553,chile,
e9,empty,-45.087279,169.834380,new zealand,
p0,plume,-34.603722,-58.381592,google maps cords locatie door Schuit gegeven,
p1,plume,40.30,-3.64,paper table b4,
p2,plume,33.48,-7.54,paper table b4,
p3,plume,28.88,20.93,paper table b4,
p4,plume,-21.91,148.06,paper table b4,
p5,plume,-26.2, 29.2,given by Berend,
p6,plume,31.66,6.06,given by Berend,
p7,plume,15.56, 45.80,given by Berend,
p8,plume,28.6139,77.2090,google maps cords locatie door Schuit gegeven,
p9,plume,39.754,-80.224,cluster in paper Schuit et al.,
a0,artefact,-19.89,120.36,,artefact_coast
a1,artefact,-20.07,119.63,,artefact_coast
a2,artefact,-19.9,120.32,,artefact_coast
a3,artefact,-19.91,120.29,,artefact_coast
a4,artefact,71.15,98.83,,artefact_cloud
a5,artefact,-24.59,-54.51,,artefact_cloud
a6,artefact,22.26,-63.32,,artefact_albedo
a7,artefact,-37.35,-58.09,,artefact_cloud
a8,artefact,75.47,103.08,,artefact_albedo
a9,artefact,18.43,56.66,,artefact_coast_albedo
```