University of South Brittany, Faculty of Science, Department of Geoinformatics

Paris Lodron University Salzburg, Faculty of Natural Sciences, Department of Geoinformatics

HOW RELIABLE ARE SENTINEL-2 CLOUD DETECTION ALGORITHMS?: GLOBAL UNCERTAINTY ESTIMATION WITH GAUSSIAN PROCESSES.

Diploma thesis

Author
Cesar Luis Aybar Camacho

Supervisor (University of South Brittany)
Prof. Francois Septier

Co-supervisor (Paris Lodron University Salzburg)
Prof. Dirk Tiede

Erasmus Mundus Joint Master Degree Programme
Copernicus Master in Digital Earth

Specialization track GeoData Science
Vannes, France, 2022

COPERNICUS MASTER IN DIGITAL EARTH

With the support of the Erasmus+ Programme of the European Union

To Prof. Bram Leo Willems, who gifted me my first computer ten years ago. This document encapsulates everything I have learned since then.

Abstract

Cloud detection (CD) is one of the most critical metadata filters for searching, selecting, and accessing imagery in Earth Observation (EO) platforms. In recent years, the extensive archive of EO datasets has boosted the use of data-driven algorithms to improve cloud and cloud-shadow detection. However, data-driven algorithms require large manually annotated datasets, which are expensive and time-consuming to collect. The first chapter of this diploma thesis introduce CloudSEN12, a new multi-temporal global dataset created exploiting different EO datasets offered by the Copernicus program. CloudSEN12 has 49,400 image patches, including (1) Sentinel-2 level-1C and level-2A multi-spectral data, (2) Sentinel-1 synthetic aperture radar data, (3) auxiliary remote sensing products, (4) different hand-crafted annotations to label the presence of clouds and cloud shadows, and (5) the results from eight state-of-the-art cloud detection algorithms. At present, CloudSEN12 exceeds all previous efforts in terms of annotation richness, scene variability, metadata complexity, control quality, data distribution and size. In the second chapter, cloudSEN12 is used to analize how reliable is Sen2COR Sentinel-2 2A cloud mask. The results shows that ...

KEYWORDS

cloud detection, deep learning, U-Net, gaussian process, non-stationary.
Number of pages: 56
Number of appendices: 5

Declaration

This thesis has been composed by Cesar Luis Aybar Camacho for the Erasmus Mundus Joint Master's Degree Program in Copernicus Master in Digital Earth for the academic year 2021/2022 at the Department of Geoinformatics, Faculty of Natural Sciences, Paris Lodron University Salzburg, and Department of Geoinformatics, Faculty of Science, Southern Brittany University.

Hereby, I declare that this piece of work is entirely my own, the references cited have been acknowledged and the thesis has not been previously submitted to the fulfillment of the higher degree.

Cesar Aybar
Southern Brittany, France
28 March 2022

Acknowledgements

I would like to express my gratitude to my supervisors, Prof. Francois Septier and Prof. Dirk Tiede, for their constant help and encouragement. From the start of the project to the end, your guidance, mentoring, and support cleared the way for me to successfully complete this dissertation. The computational requirements for this research were partially covered by the Google Cloud Credits Research Grant Program. Besides, the Radiant Earth Foundation provides us with a space to store the dataset. This work was also partially supported by the Spanish Ministry of Science and Innovation (project PID2019-109026RB-I00, ERDF), the Austrian Space Applications Programme within the SemantiX project (#878939, ASAP 16), and the Linux Foundation Grant projects (project 21-ISC-1-1). The following R and Python packages were used in the course of this investigation and I would like to acknowledge their developers: rgee (Aybar et al. 2020), sf (Pebesma 2018), raster (Hijmans et al. 2015), stars (Pebesma 2020), numpy (Harris et al. 2020), lubridate (Grolemund and Wickham 2011), reticulate (Ushey et al. 2020), dplyr (Wickham et al. 2014), tmap (Tennekes 2018), magick(Ooms 2020), rgeos (Bivand et al. 2017) and ggplot2(Wickham 2011). Finally, I would like to thank B.S. Joselyn Inga and Wendy Espinoza for their work reporting manual labeling errors in the quality control phase of the dataset.

Cesar Aybar
Southern Brittany, France
28 March 2022

Contents

List of Figures

List of Tables

List of Abbreviations

1-D, 2-D . . . . One- or two-dimensional, referring **in this thesis** to spatial dimensions in an image.

Otter . . . . . One of the finest of water mammals.

Hedgehog . . . Quite a nice prickly friend.

What am I in the eyes of most people - a nonentity, an eccentric or an unpleasant person - somebody who has no position in society and never will have, in short, the lowest of the low. All right, then - even if that were absolutely true, then I should one day like to show by my work what such an eccentric, such a nobody, has in his heart.

— Vincent Van Gogh - 1882

1 Uncertainty estimation

Contents

1.1 Introduction

Cloud masking is an essential pre-processing for any application of optical remote sensing imagery. Various coarse-resolution cloud cover datasets (Sassen and Wang 2008; Winker et al. 2010; Wilson and Jetz 2016) estimate the worldwide multi-annual cloud occurrence percentage to be around 0.6 ± 0.2, with hotspots in tropical and subtropical forests. Assuming these products are indicative of Sentinel-2 cloud conditions, we can anticipate that more than half of the pixels

will need to be removed, i.e. masked out, in order to avoid distortions in further analyses. Given that cloud masking can be interpreted as a statistical classification problem, the confusion matrix can be used to distinguish between two distinct types of errors. On the one hand, cloud omission errors (cloud as non-cloud) can lead to inconsistencies in time series of surface reflectance pixels, whereas cloud commission (clear as non-cloud) reduces the number of valid observations and, as a result, the frequency of cloud-free data (Skakun et al. 2022). Cloud masking techniques are aimed to have a balance between commission and omission errors. Over the last three decades, a plethora of cloud masking methods have been presented (Hagolle et al. 2017; Domnich et al. 2021; Louis et al. 2016; Qiu et al. 2019; Richter and Schläpfer 2019; Wevers et al. 2021; López-Puigdollers et al. 2021; Frantz 2019). These methods can be classified into two main categories: knowledge-driven (KD) and data-driven (DD). While KD emphasizes the use of physical rules formulated on spectral and contextual features, DD is subjected to the exigency of large pixel-level annotation and costly computational requirements to distinguish cloud versus non-cloud regions.

Only a few studies have attempted to compare the various Sentinel-2 cloud masking methods. For instance, Cilli et al. 2020 compare DD with KD methods by analyzing 135 Sentinel-2 images distributed worldwide. They concluded that DD methods outperform KD methods. According to their experiments, 10^4 manually labeled pixels are sufficient for train machine learning algorithms to operate accurately cloud masking. Nonetheless, it is well established that DD models are highly dependent on the training dataset (López-Puigdollers et al. 2021). As a result, the comparison could be unfair, especially if the KD methods have not been as well calibrated to the dataset. Zekoll et al. 2021 compare three KD threshold-based: FMask, ATCOR, and Sen2Cor using a sample-based dataset. The results show that Sen2Cor outperforms the other methods. However, human-made datasets, especially those created by sampling, can be positively skewed if we consider that humans tend to overlook unpleasant information such as cloud borders (ostrich-effect, Valdez et al. 2017). Using four different datasets, the Cloud Mask Intercomparison eXercise (CMIX) recently compared ten cloud detection algorithms. They suggested that no single algorithm performed better

than the others (Skakun et al. 2022). Similar conclusions are found in Tarrio et al. 2020 by analyzing 28 images over six Sentinel-2 tiles in Africa and Europe.

In this chapter, we propose to use a novel approach for exploiting the hand-crafted cloud and cloud shadow masking information freely available in cloud-SEN12. Unlike previous studies, we intend not only to characterize but also to predict cloud masking uncertainty worldwide. Inspired by deep kernel learning Wilson, Hu, et al. 2016, we explore the use of Gradient boosting Gaussian Processes (GBGP) which combines the structural properties of machine learning architectures with the non-parametric flexibility of kernel methods.

1.2 Data

The basis for Cmask is to predict what the reflectance would be for individual pixels in a Landsat 8 image if cirrus clouds were not present at the time the observations were collected. The basis for those pre- dictions is past observations and the atmospheric conditions, i.e. the water vapor content, at the time of image acquisition. Therefore, two major inputs were included in this analysis: Integrated Water Vapor (IWV) provided by the second Modern-Era Retrospective analysis for Research and Applications (MERRA-2) and cirrus band time series provided by Landsat 8.

1.2.1 cloudSEN12

CloudSEN12 is a globally spatio-temporal distributed dataset for cloud and cloud shadow semantic understanding that consists of 49,400 image patches (IP) that are evenly spread throughout all continents except Antarctica (Figure X). Each IP has an average size of 5090 x 5090 meters and contains data from Sentinel-2 optical levels 1C and 2A, Sentinel-1 Synthetic Aperture Radar (SAR), digital elevation model, surface water occurrence, land cover classes, cloud masking results from eight different algorithms and hand-crafted labeling data created using an active learning system Francis et al. 2020. cloudSEN12 offers three different manual annotation types in order to support different deep learning strategies: (i) 10,000 IPs with high-quality pixel-level annotation, (ii) 10,000 IPs with scribble annotation, and (iii) 29,250 unlabeled IPs. Only high-quality

IPs are used in this study because of the risk of bias due scene incompleteness in scribble annotation. A detailed description of the CloudSEN12 dataset is given in chapter one.

1.2.2 Sen2Cor

Sentinel 2 Correction (Louis et al. 2016) is a mono-temporal image processor designed for scene classification and atmospheric correction of Sentinel-2 Level 1C input data. Sen2Cor version 2.8 is the version used in the Sentinel-2 ground segment with L2A processing baseline version 02.12. This processing baseline is the most recurrent version in cloudSEN12 (90% IPs). Sen2Cor (Figure X) uses a series of spectral reflectance thresholds, ratios, and indices based on bands 1–5, 8, and 10–12 to compute cloud and snow probabilities for each pixel. Besides, it includes a cloud shadow and cirrus detection algorithm. The cloud shadows are estimated by multiplying two probability layers: (1) a geometric probability layer constructed from the final thick cloud mask, sun position, and cloud height distribution, and (2) radiometric probability layer created from a Kohonen map to detect dark areas. On the other hand, cirrus probabilities are calculated simply by threshold band 10 (1.375 μm). Finally, a series of additional steps to improve the quality of the classification are automatically triggered using a priori information: digital elevation model (DEM) information, ESA CCI Water Bodies Map v4.0 (Lamarche et al., 2017), ESA CCI Land Cover Map v.2.0.7 (2015) and a snow climatology. In this study, SCL classes 8, 9 and 10 were used for cloud, class 3 for cloud shadow and the remaining SCL classes for non-cloud.

1.2.3 Cloud type ocurrence

The cloud type ocurrence (CTO) are derived from the CloudSat cloud profiling radar (CPR). This sensor permited for the first time to see cloud vertical structure (Stephens et al. 2008). ClouSat is part from Afternoon Constellation, or A-Train, at a frequency of 94 GHz and an altitude of about 700 km. Since 2011, it only produce daytime-only measurements due to battery malfunction, and in February 2018, the spacecraft systems moved to a lower orbit (C-train) to reduce the risk of collision with the other A-train spacecraft. The track of the satellite overpasses the

same location every 16 days and provides a resolution of 1.4 km in cross-track and 1.8 km in along-track. Using the vertical profiles of clouds and precipitation, create an algorithms () to classify clouds according to the Cloud Climatology Project (ISCCP) approach: cumulus (Cu), stratocumulus (Sc), stratus (St), alto- cumulus (Ac), altostratus (As), nimbostratus (Ns), cirrus/ cirrostratus, or deep convective clouds. This information is storage in the 2B-CLDCLASS product. Particularly, in this study, we make use of the 3S-RMCP product of CloudSat level 3 observations. This product collocated 2B-CLDCLASS measurements in a 2.5°x2.5° grid. Based on this dataset, we constructed cloud type occurrence at 1° spatial resolution. The TPR data used corresponds to the 2007–2016 period, the cloud type ocurrence were smoothed from 2.5° to 1° using cubic spline interpolation. We aggregathed the ISCCP classification into three classes accordin to the table x.

1.2.4 MODIS

1.3 Methodology

This study aims to create a model that predicts cloud masking error in a 1° x 1° worldwide grid system y^* from a set of predictors x^*. The regression model f is trained using a dataset, $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$ with x_i and $y_i \in \mathbb{R}$. The segment x_i represents a vector of $1 \times m$ predictors (Table X), y_i is the metric error values (see section 1.3.4) determined from comparing cloudSEN12 and Sen2Cor cloud masking results, and N the number of observations set as 10000. Only high-quality IPs are used in this study because of the risk of bias due to scene incompleteness in scribble annotation. A detailed description of the CloudSEN12 dataset is given in chapter one. We propose the use of a deep kernel learning (DKL) regression. It combines deep neural networks (DNN) with standard gaussian process regression (GP). While ANN captures the non-stationary and hierarchical structure, GP permits the estimation of their uncertainty considering the local autocorrelation of the observations. Rasmussen 2003 provide a detail description of GP. The standard GP, DKL, and the approach used to create worldwide cloud error predictions are briefly reviewed in the following sections.

Dataset	Used for	Spatial coverage	Data Type	Description
Sen2Cor SLC - snow	T	Focal	B	Percentage of snow cover in a cloudSEN12 IP. If at least 5% of snow cover exists, 1; otherwise, 0.
CloudSEN12 - cloud type	T	Focal	N	Metadata available in cloudSEN12, derived from human-photo interpretation. There are five classes: cumulus, stratus, cirrus, haze, and contrails. A single IP might have multiple classes.
CloudSEN12 - cloud extent	T	Focal	N	Metadata available in cloudSEN12, derived from human-photo interpretation. There are two classes: isolated, and extended.
CloudSAT 3S-RMCP - cloud ocurrence	P	Global	C	CloudSAT level 3 product used to generalize CloudSEN12 - cloud type in prediction time.
CloudSEN12 - Mean solar zenith angle	TP	Focal	C	The angle formed by the sun's beams with respect to the vertical axis. In training phase values are obtained from cloudSEN12 metadata. In prediction phase is obtained by averaging image properties from 2018 to 2020.
MODIS - MCD43A4.006 (HOT and NDVI)	TP	Global	C	Spectral bands obtained from MODIS composite from 2018 to 2020. In training phase, the values are resampled considering each specific cloudSEN12 IP geotransform. Then, the pixels are spatially reduced by the mean.
MERIT Hydro - Elevation	TP	Global	C	Elevation map available in cloudSEN12.
Inter-annual Cloud frecuency	TP	Global	C	Obtained from global 1-km cloud dataset.
Cloud intra-anual variability	TP	Global	C	
Latitude	TP	Global	C	In training phase are obtained from the cloudSEN12 IP centroid.
Longitude	TP	Global	C	

1.3.1 Gaussian process regresion

Standard Gaussian Process Regression (GP) models is a expressive probabilistic model in which both training and testing data points are regarded samples of a joint multivariate normal distribution (Williams and Rasmussen 2006). As other regression models, a GP model is formed by noisy variables of the true underlying function f that projects the vector space X into real-valued targets y, i.e. $y = f(\mathbf{x}) + \epsilon$. The element ϵ represents the noise variables with $\mathcal{N}(0, \sigma^2)$. In a GP model we assume that all the finite dimensional distributions $f(\mathbf{x})$ are normally distributed with μ as a mean and $K_{XX}|\gamma$ as the prior covariance matrix. The covariance (kernel) matrix regulates the smoothness of GPs and its values are implicitly dependent on the kernel hyperparameters γ. In this specific case,

the estimation of $f(\mathbf{x})$ can be expressed given by:

$$\mathbf{f} = f(\mathbf{x}) = [f(x_1), ..., f(x_m)]^\top \sim \mathcal{GP}\left(\mu, K_{XX}|\gamma\right) \tag{1.1}$$

Conditioning the joint normal distribution by the output values at the training points (X and $\mathbf{y}$), the posterior distribution of the output values $f(\mathbf{x}_*)$ at the test data point X_* can be inferred as:

$$\begin{aligned} f(\mathbf{x}_*) \mid X_*, X, \mathbf{y}, \boldsymbol{\gamma}, \sigma^2 &\sim \mathcal{N}(\mu^*, \Sigma^*), \\ \mu^* &= \mu_{X_*} + K_{X_*X}\widehat{K}_{XX}^{-1}\mathbf{y}, \\ \Sigma^* &= K_{X_*X_*} - K_{X_*X}\widehat{K}_{XX}^{-1}K_{XX_*} \end{aligned} \tag{1.2}$$

A hat denotes an added diagonal, i.e. $\widehat{K}_{XX} = K_{XX} + +\sigma^2 I$. μ^* and Σ^* are the posterior mean and covariance matrix respectively. The matrices of the form $K_{X_iX_j}$ denote cross-covariances between the train (X) and test (X_*) vector spaces. The hyperparameters λ of the kernel are usually learned directly by minimizing the negative log marginal likelihood $\mathcal{L}(\theta)$ with respect to training observations:

$$\begin{aligned} \mathcal{L} &= -\log p(\mathbf{y} \mid \gamma, X) \propto \mathbf{y}^\top \widehat{K}_{XX}^{-1}\mathbf{y} + \log\left|\widehat{K}_{XX}\right|, \\ \frac{\partial \mathcal{L}}{\partial \theta} &= \mathbf{y}^\top \widehat{K}_{XX}\frac{\partial \widehat{K}_{XX}^{-1}}{\partial \theta}\widehat{K}_{XX}\mathbf{y} - \operatorname{tr}\left\{\widehat{K}_{XX}^{-1}\frac{\partial \widehat{K}_{XX}}{\partial \theta}\right\} \end{aligned} \tag{1.3}$$

The main bottleneck for kernel learning is solve the linear system $\widehat{K}_{XX}^{-1}y$ in equation 1.3. The standard approach is to compute the Cholesky decomposition of the matrix $\widehat{K}_{XX}^{-1}$. The Cholesky decomposition's core algorithm uses a divide-and-conquer approach that is inefficient on GPU acceleration (Krishnamoorthy and Menon 2013). Furthermore, it requires $\mathcal{O}(n^3)$ computation and $\mathcal{O}(n2)$ storage for GP inference and kernel learning (Rasmussen 2003). To address the above challenges, several approaches to scaling up GP inference have been proposed (Gardner et al. 2018; Cunningham et al. 2008; Dong et al. 2017; Bach 2013; Wilson, Dann, et al. 2015). In this paper, we address the GP inference issue by using the Blackbox Matrix-Matrix multiplication inference (BBMM, Gardner et al. 2018). BBMM use preconditioned batched conjugate gradients to solve linear systems, reducing the asymptotic time complexity of GP inference from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$. Besides, it overcomes memory constraints by divvying the kernel matrix to perform matrix-vector multiplication (MVM, Demmel 1997) without having

to explicitly construct the kernel matrix, reducing the memory requirement to $\mathcal{O}(n)$. Finally, BBMM parallelize partitioned MVMs across multiple core, enabling a better use of GPU hardware in comparison to the Cholesky factorization.

1.3.2 Deep kernel learning

DKL is a probabilistic deep network that simultaneously learns a feature extractor and a Gaussian process on the feature space (cite). The network's structure is depicted in Figure X. The deep non-linear feature extractor $\mathbf{h}(\mathbf{x}, \mathbf{w})$, parametrized by weights $\mathbf{w}$, is applied to the observed input variable $\mathbf{x}$. Next, the DNN outputs are modeled using $\mathcal{GP}$ by:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{h}_\mathbf{w}(\mathbf{x})), k_\gamma(\mathbf{h}_\mathbf{w}(\mathbf{x}), \mathbf{h}_\mathbf{w}(\mathbf{x}'))) \tag{1.4}$$

All parameters of the model, including neural network weights $\mathbf{w}$ and kernel parameters γ, are optimized end-to-end via backpropagation to minimize the negative log marginal likelihood (equation 1.3).

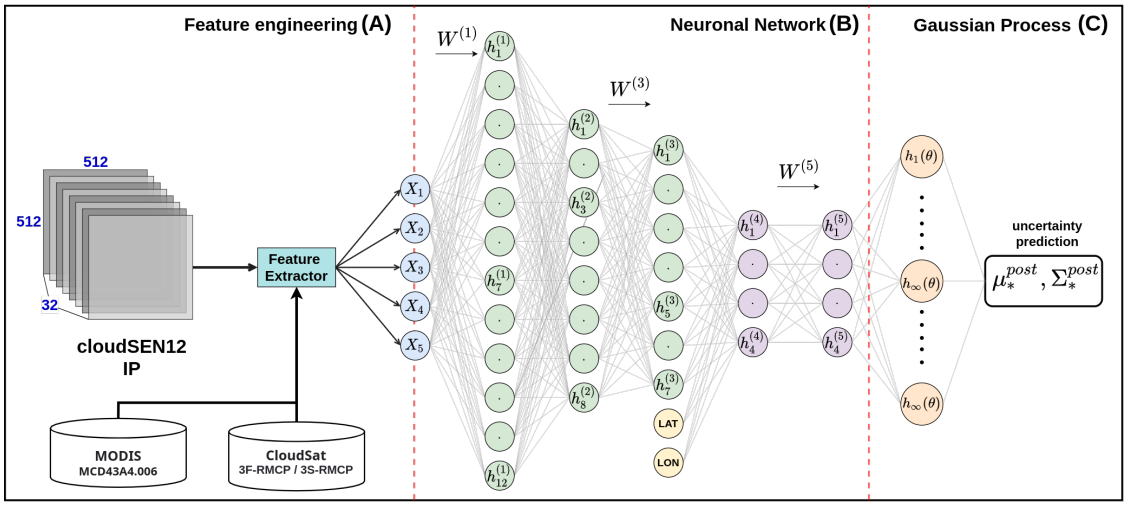

Figure 1.1: Number of hand-crafted pixel annotations between different cloud detection datasets. All the labeled pixels in the CloudSEN12 no-annotation group come from cloud-free IPs.

1.3.3 Model set-up and training

For our deep kernel learning model, we used deep neural networks which produce C-dimensional top-level features. Here C is the number of classes. We place a Gaussian process on each dimension of these features. We used RBF base

kernels. The additive GP layer is then followed by a linear mixing layer A R C×C . We initialized A to be an identity matrix, and optimized in the joint learning procedure to recover cross-dimension correlations from data. We first train a deep neural network using SGD with the softmax loss objective, and rectified linear activation functions. After the neural network has been pre-trained, we fit an additive KISS-GP layer, followed by a linear mixing layer, using the top-level features of the deep network as inputs. Using this pre-training initialization, our joint SV-DKL model of section 3 is then trained through the stochastic variational method of section 4 which jointly optimizes all the hyperparameters of the deep kernel (including all network weights), as well as the variational parameters, by backpropagating derivatives through the proposed marginal likelihood lower bound of the additive Gaussian process in section 4. In all experiments, we use a relatively large mini-batch size (specified according to the full data size), enabled by the proposed structure exploiting variational inference procedures. We achieve good performance setting the number of samples T = 1 in Eq. 4 for expectation estimation in variational inference, which provides additional confirmation for a similar observation in [14].

1.3.4 Model metrics

1.3.5 Model predictions

1.4 Experimental results

1.5 Discussion

dsada

1.6 Conclusion

dsada

I may not have lived long, but I am certain of one thing. If there is a type of person capable of changing something, it is someone who is willing to sacrifice what he values most!. He is the type of person who, in order to confront a monster, is capable of losing his own humanity. A person who is unable to make a sacrifice may be unable to change anything!

— Armin Arlelt

Conclusion

If we don't want Conclusion to have a chapter number next to it, we can add the `{-}` attribute.

More info

And here's some other random info: the first paragraph after a chapter title or section head *shouldn't be* indented, because indents are to tell the reader that you're starting a new paragraph. Since that's obvious after a chapter or section title, proper typesetting doesn't add an indent there.

This paragraph, by contrast, *will* be indented as it should because it is not the first one after the 'More info' heading. All hail LaTeX. (If you're reading the HTML version, you won't see any indentation - have a look at the PDF version to understand what in the earth this section is babbling on about).

Appendices

A
Appendix - Code

This first appendix includes the non-stationary gaussian processes in Gpytorch:

In 02-rmd-basics-code.Rmd

And here's another one from the same chapter, i.e. Chapter ??:

B

Appendix - Figures

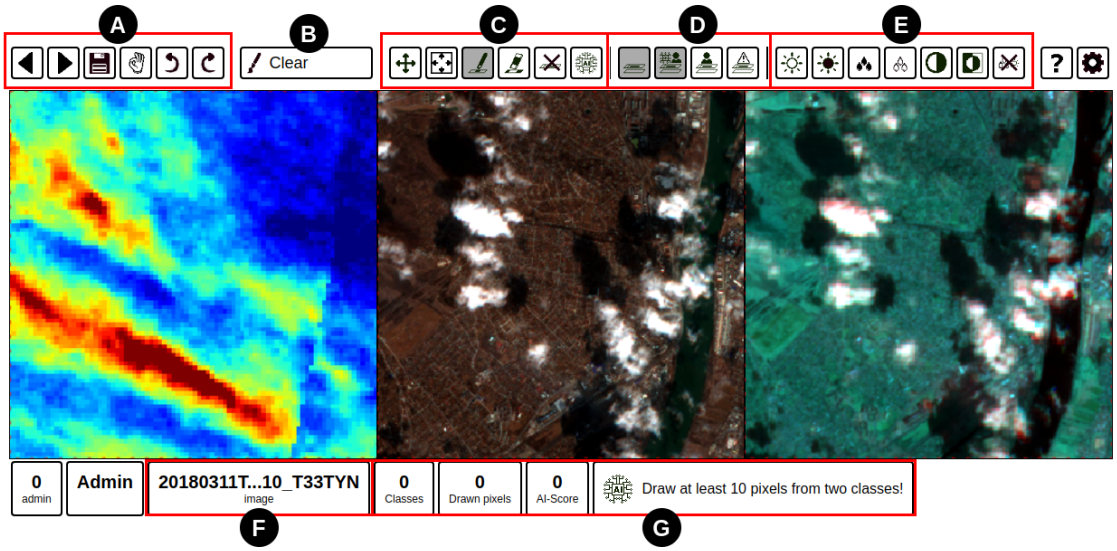

Figure S1: XXXX

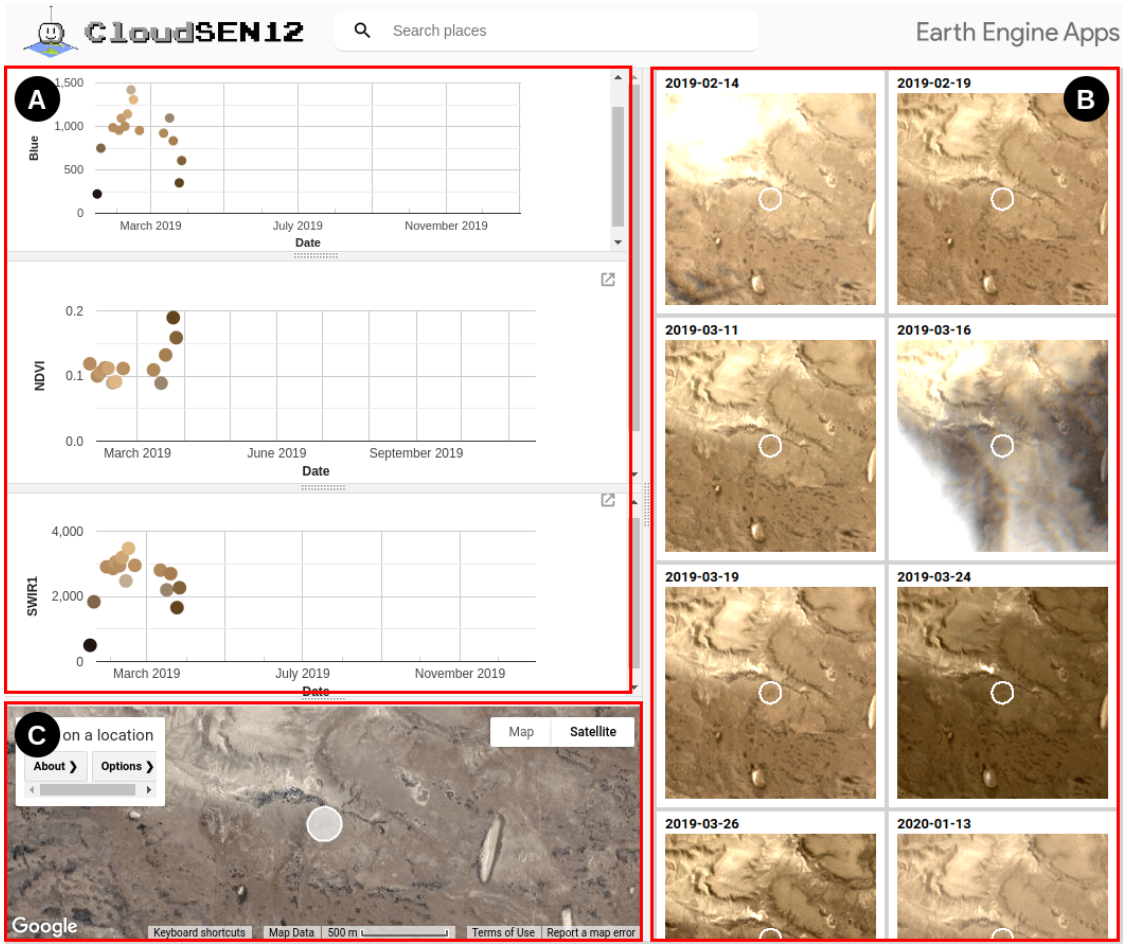

Figure S2: XXXX

Bibliography

Aybar, Cesar et al. (2020). "rgee: An R package for interacting with Google Earth Engine". In: *Journal of Open Source Software* 5.51, p. 2272. DOI: 10.21105/joss.02272.

Bach, Francis (2013). "Sharp analysis of low-rank kernel matrix approximations". In: *Conference on Learning Theory*. PMLR, pp. 185–209.

Bivand, Roger et al. (2017). "Package 'rgeos'". In: *The Comprehensive R Archive Network (CRAN)*.

Cilli, Roberto et al. (2020). "Machine Learning for Cloud Detection of Globally Distributed Sentinel-2 Images". In: ii, pp. 1–17. DOI: 10.3390/rs12152355.

Cunningham, John P, Krishna V Shenoy, and Maneesh Sahani (2008). "Fast Gaussian process methods for point process intensity estimation". In: *Proceedings of the 25th international conference on Machine learning*, pp. 192–199.

Demmel, James W (1997). *Applied numerical linear algebra*. SIAM.

Domnich, Marharyta et al. (2021). "KappaMask: Ai-based cloudmask processor for sentinel-2". In: *Remote Sensing* 13.20. DOI: 10.3390/rs13204100.

Dong, Kun et al. (2017). "Scalable log determinants for Gaussian process kernel learning". In: *Advances in Neural Information Processing Systems* 30.

Francis, Alistair et al. (Nov. 2020). *Sentinel-2 Cloud Mask Catalogue*. Zenodo. DOI: 10.5281/zenodo.4172871. URL: https://doi.org/10.5281/zenodo.4172871.

Frantz, David (2019). "FORCE—Landsat+ Sentinel-2 analysis ready data and beyond". In: *Remote Sensing* 11.9, p. 1124.

Gardner, Jacob et al. (2018). "Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration". In: *Advances in neural information processing systems* 31.

Grolemund, Garrett and Hadley Wickham (2011). "Dates and times made easy with lubridate". In: *Journal of statistical software* 40, pp. 1–25.

Hagolle, O. et al. (2017). "MAJA ATBD Algorithm Theoretical Basis Document". In: *MAJA-TN-WP2-030 V1.0 2017/Dec/07* 18639, pp. 0–39. URL: http://www.cesbio.ups-tlse.fr/multitemp/?p=12432.

Harris, Charles R et al. (2020). "Array programming with NumPy". In: *Nature* 585.7825, pp. 357–362.

Hijmans, Robert J et al. (2015). "Package 'raster'". In: *R package* 734.

Krishnamoorthy, Aravindh and Deepak Menon (2013). "Matrix inversion using Cholesky decomposition". In: *2013 signal processing: Algorithms, architectures, arrangements, and applications (SPA)*. IEEE, pp. 70–72.

López-Puigdollers, Dan, Gonzalo Mateo-García, and Luis Gómez-Chova (2021). "Benchmarking deep learning models for cloud detection in landsat-8 and sentinel-2 images". In: *Remote Sensing* 13.5, pp. 1–20. DOI: 10.3390/rs13050992.

Louis, Jérôme et al. (2016). "Sentinel-2 SEN2COR: L2A processor for users". In: *European Space Agency, (Special Publication) ESA SP* SP-740.May, pp. 9–13.

Ooms, Jeroen (2020). "magick: Advanced graphics and image-processing in R". In: *R package version* 2.1.

Pebesma, Edzer (2018). "Simple features for R: Standardized support for spatial vector data". In: *R Journal* 10.1, pp. 439–446. DOI: 10.32614/rj-2018-009.

Pebesma, Edzer (2020). “stars: Spatiotemporal arrays, raster and vector data cubes”. In: *R package version 0.4–1 ed2020 https://CRAN. R-project. org/package= stars*.

Qiu, Shi, Zhe Zhu, and Binbin He (2019). “Remote Sensing of Environment Fmask 4 . 0 : Improved cloud and cloud shadow detection in Landsats 4 – 8 and Sentinel-2 imagery”. In: *Remote Sensing of Environment* 231.May, p. 111205. DOI: 10.1016/j.rse.2019.05.024. URL: https://doi.org/10.1016/j.rse.2019.05.024.

Rasmussen, Carl Edward (2003). “Gaussian processes in machine learning”. In: *Summer school on machine learning*. Springer, pp. 63–71.

Richter, R and D Schläpfer (2019). “Atmospheric and topographic correction (ATCOR theoretical background document)”. In: *DLR IB*, pp. 564–03.

Sassen, Kenneth and Zhien Wang (2008). “Classifying clouds around the globe with the CloudSat radar: 1-year of results”. In: *Geophysical research letters* 35.4.

Skakun, Sergii et al. (2022). “Cloud Mask Intercomparison eXercise (CMIX): An evaluation of cloud masking algorithms for Landsat 8 and Sentinel-2”. In: *Remote Sensing of Environment* 274, p. 112990.

Tarrio, Katelyn et al. (2020). “Comparison of cloud detection algorithms for Sentinel-2 imagery”. In: *Science of Remote Sensing* 2, p. 100010.

Tennekes, Martijn (2018). “tmap: Thematic Maps in R”. In: *Journal of Statistical Software* 84, pp. 1–39.

Ushey, Kevin et al. (2020). “reticulate: Interface to Python”. In: *R package version* 1, p. 16.

Valdez, Calero, Martina Ziefle, and Michael Sedlmair (2017). “A Framework for Studying Biases in Visualization Research”. In: *VIS 2017: Dealing with Cognitive Biases in Visualisations*. URL: http://eprints.cs.univie.ac.at/5258/1/calero-valdez2017framework.pdf.

Wevers, Jan et al. (Dec. 2021). *IdePix for Sentinel-2 MSI Algorithm Theoretical Basis Document*. Version Version 1.0. DOI: 10.5281/zenodo.5788067. URL: https://doi.org/10.5281/zenodo.5788067.

Wickham, H et al. (2014). “Dplyr: A fast, consistent tool for working with data frame like objects, both in memory and out of memory”. In: *R package version 0.7* 6.

Wickham, Hadley (2011). “ggplot2”. In: *Wiley interdisciplinary reviews: computational statistics* 3.2, pp. 180–185.

Williams, Christopher K and Carl Edward Rasmussen (2006). *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA.

Wilson, Adam M. and Walter Jetz (2016). “Remotely Sensed High-Resolution Global Cloud Dynamics for Predicting Ecosystem and Biodiversity Distributions”. In: *PLoS Biology* 14.3, pp. 1–20. DOI: 10.1371/journal.pbio.1002415.

Wilson, Andrew Gordon, Christoph Dann, and Hannes Nickisch (2015). “Thoughts on massively scalable Gaussian processes”. In: *arXiv preprint arXiv:1511.01870*.

Wilson, Andrew Gordon, Zhiting Hu, et al. (2016). “Deep kernel learning”. In: *Artificial intelligence and statistics*. PMLR, pp. 370–378.

Winker, DM et al. (2010). “The CALIPSO mission: A global 3D view of aerosols and clouds”. In: *Bulletin of the American Meteorological Society* 91.9, pp. 1211–1230.

Zekoll, Viktoria et al. (2021). “Comparison of masking algorithms for sentinel-2 imagery”. In: *Remote Sensing* 13.1, pp. 1–21. DOI: 10.3390/rs13010137.