G5

PROJECT DOCUMENTATION

Initial Requirements Model

Aerial Imagery Initiative

Index

Introduction

Spacial Services (SS) a department of Department of Customer Services (DCS) is seeking to develop a system that is able to extract flood boundaries from within captured aerial imagery.

During times of emergency SS/DCS receive requests through their Emergency Information Coordination Unit (EICO) requesting aerial photography of flooded regions. The requests received through EICO can come from any Emergency Service Organisation (ESO) including State Emergency Services (SES), Rural Fire Service (RFS), Infrastructure NSW .etc and are used for ongoing emergencies or recovery planning.

The aerial photography is captured upon request and usually coincides with the peak of the flood event. Once captured the imagerys is processed and released within 48hrs. The proposed system is set to create *“a repeatable process to use on imagery to extract flood extent boundaries”.* Currently requesting organisations and ESO’s are required to perform the extraction of the flood extent manually.

The proposed system will extract the flood extent boundaries and create a GIS compatible polygon automatically. This will allow DCS/SS to provide an accurate flood extent quickly allowing ESO resources to be utilised elsewhere. The generation of the flood extent polygon will also allow a consistent flood extent data to be used by all agencies and ESO’s thus reducing confusion or errors arising from the use of inconsistent data sets between agencies.

1. Functional Requirements

1.1 Business Rules

- 1.1.1 The system must be able to accept a batch of aerial images and map the extent of flooding within them.
- 1.1.2 The system will utilise a machine learning model, or some other image classification system, to classify areas of interest as either flooded or not flooded.
- 1.1.3 The system will use a process of image segmentation to identify areas of interest within the image for feature extraction.

- 1.1.4 The system will extract appropriate feature vectors from the images that will determine accurate classifications of flooded/un-flooded areas on new images.
- 1.1.5 The system will perform feature reduction and data cleaning operations as appropriate before passing the feature vectors into the machine learning model or other image classification system.
- 1.1.6 The system will be able to account for the effect of cloud cover, infrastructure, vegetation, and other objects present within the input images to produce an accurate flood extent map.
- 1.1.7 The output of the system will be a polygon of the flood extent. A subsequent process will be necessary to transform this output into a product that can be distributed to clients.
- 1.1.8 Generation of the polygon should have minimal impact on the current 48 hr turnaround time from capture of imagery to release.

1.2 External Interfaces

- 1.2.1 The system must utilise a machine learning mode or another method suitable for the task. The specific method (e.g. Convolutional Neural Network, Decision Tree Algorithm, KMeans clustering etc.) will be determined at a later stage in the design process. The SageMaker console itself utilises Jupyter Notebooks (Jupyter, 2021) for data pre-processing, model training, and hyper-parameter tuning.
- 1.2.2 The system must utilise a suitable tool for data labelling and pre-processing of input images. Possible candidates are AWS SageMaker GroundTruth (AWS, 2021b) and https://www.robots.ox.ac.uk/~vgg/software/via/via-1.0.6.html
- 1.2.3 The AWS CLI (command line interface) (AWS, 2021c) and AWS Web Console (AWS, 2021d) will be used for accessing AWS services.
- 1.2.4 The output of the system must be compatible, or at least transferable, to ArcGIS (Esri, 2021), a mapping and data analytics program used by NSW government departments for generating geo-spatial information. The Python third party library QGIS will be used for this (QGIS, 2021).
- 1.2.5 The system will utilise the AWS Simple Storage Service (S3) (AWS, 2021f) for data storage.

1.3 Authentication & Auditing

- 1.3.1 Use of the AWS services entails costs while code is executing or a service endpoint is up-and-running. The system and its development environment must only be accessed by authenticated developers to ensure only qualified users access resources.
- 1.3.2 The project has an allocated budget of $50 000. The system development and deployment environments must allow for auditing of resource use and cost.

1.4 Storage & Capacity

- 1.4.1 The output of this project is to serve as an historical record of flooding events. This requires long-term storage of the system's output.
- 1.4.2 If the project proves successful, the scope may be expanded so as to analyse up to ten years of past flooding events. The system must have the ability to scale its storage capacity.
- 1.4.3 The system must be able to handle raw input images that are approximately 10 GB in size. The pre-processing stage of the machine learning process will reduce the size of the machine learning model inputs, but the system must be able to handle large initial inputs.

1.5 Functional Requirement Priority Matrix

Functional Requirement	Priority	Justification
Business Rules	High	These form the core deliverables of the project. Given that the output of the project is a product to be used by multiple government agencies in their emergency response planning to flood events, it is imperative that all goals are met to a satisfactory standard. Substandard outcomes will nullify the business case for the project.

External Interfaces	High	The use of external interfaces will drastically decrease the project workload and assist in overcoming technical difficulties in the machine learning process which is a highly skilled and specialised area of software engineering and data science. It is important that these interfaces are incorporated into the project development and final outcome.
Authentication & Auditing	Low	The AWS services for system development will be accessed through DCS's AWS *ss-identity* account, so authentication for users interacting with the system will already follow best-practices. Furthermore, AWS CloudTrail (AWS, 2021e) allows authenticated administrators to access each API call from all AWS products for a given account. This provides a high level of auditing. Secure distribution of the final product is beyond the scope of this project.
Storage & Capacity	Low	AWS will host the infrastructure for the system, hence many of the details of storage and capacity will be accommodated by its scalable IaaS services.

2. Non-functional Requirements

2.1 Performance

- 2.1.1 The system is automating and consolidating a process that is performed by several government departments manually. Although no exact figure has been outlined by the stakeholders in terms of the model's classification accuracy, it must at least out-perform the current process.
- 2.1.2 Emergency response planning operations will use this rapid-response product as an authoritative source of information for multiple use cases such as identification of isolated communities and animal stock. Considering that the process of taking the aerial images and post-processing tasks takes 48 hours, it is important that the response time of the system is efficient.

2.2 Cost

- 2.2.1 The development of the system must not exceed the allocated $50 000 budget from DCS.

2.3 Security & Infrastructure

- 2.3.1 Data handled by any NSW Government agency must adhere to the NSW Data & Information Custodianship Policy (NSW Government, 2013). In relation to storage and archiving of data, the policy states the custodian of the data must "ensure appropriate records storage, maintenance, security, and archiving...even after active data collection and/or revision is complete" (NSW Government, 2013, pp. 10).
- 2.3.2 According to the NSW Government's Cloud Policy (NSW Government, 2020), when offering software as a service (Saas), a third party vendor is preferable for management of the underlying infrastructure.

2.4 Usability

- 2.4.1 Users will be skilled in aerial image analysis and familiar with command line interfaces, API calls in Python, and the AWS services particular to this project. As a result, a user interface with a conventional layout is not necessary. More emphasis will be placed on system performance and reliability.
- 2.4.2 One of the business goals of the system is to simplify and consolidate data processing tasks so they are not duplicated with variance amongst government agencies. It is important that all organisations in the wake of a flooding event have a single authoritative dataset to use for recovery planning and historical record keeping.

2.5 Reliability & Availability

- 2.5.1 Given the infrequency of flooding events, the system will be in sporadic use. But when it is needed the system must be able to perform uninterrupted and in a time-efficient manner as the results are to be used by multiple government departments and agencies for emergency response planning.
- 2.5.2 The time between taking aerial shots and their readiness for processing is forty eight hours. It is essential that the next steps of outputting a polygon layer in an ArcGIS environment does not add too much extra time to the process.
- 2.5.3 When the system is in use, it is imperative that the system does not crash. Loss of data half-way through model training, or generation of prediction set will require restarting the program. In other words, progress cannot be incremental.

2.6 Non-functional Requirement Priority Matrix

NFR	Priority	Justification
Performance	High	The business case for the system is premised on the efficiency and accuracy of an automated machine learning pipeline. The system must therefore achieve its goals with a high degree of efficiency and accuracy.
Cost	Moderate	The budget allocated for the project is likely to be sufficient. In particular, AWS Lambda offers pay-as-you-go pricing options in which customers are only charged for as long as their code is executing in a cloud environment.
Security	Low	Since the data to be used for the project will be stored in AWS S3 Buckets owned and managed under the DCS *ss-identity* account, it is assumed that any relevant laws, policies, or regulations pertaining to data confidentiality, integrity, and storage will be adhered to. G5 will only access the data through the AWS CLI environment which will leave a log of any access operations through AWS CloudTrail, thus providing a record of all user interactions that can be examined to ensure compliance. Furthermore, as DCS has outlined the use of AWS Services as a constraint, the project is in adherence to the NSW Government's Cloud Policy.
Usability	Low	The users for the system are a skilled set of employees at DCS and SS who are familiar with the processing of aerial images and geospatial mapping software. Development of an API has already been deemed

		unnecessary. As a result, little emphasis will be placed on making the system accessible to new users.
Reliability & Availability	Moderate	The system will be deployed on a continuous basis. Rather, it will be required in the wake of flooding events. However, it is important that the system does not fail while it is in use. Utilisation of AWS infrastructure services is likely to ensure reliability and availability.

3. Use Case Descriptions

3.1 Use Case - Flood Area Classification

3.1.1 Brief Description

When a DCS/SS employee,
Wants to generate a map of flooding extent, they access the AWS Sagemaker notebook to initiate a batch upload of the aerial images to the SageMaker endpoint to receive a classification of flooded areas,
So that they can distribute this information to other Emergency Service Organisations.

3.1.2 Actors

- DCS/SS Employee - An employee of one of the NSW Department of Customer Service and/or the Division of Spatial Services. They are responsible for the processing of aerial images of flood affected areas and providing products to other NSW government departments and agencies in their emergency response planning.
- ESO - End user of the identified flood extent data. This can be SES, RFS, Infrastructure NSW .etc

3.1.3 Pre-conditions

- The SageMaker endpoint is active.
- The user has a set of aerial images that have undergone pre-processing operations such as colour balancing, orthorectification, and stitching.
- Images are stored in a location accessible by the not Notebook (S3 Bucket)

3.1.4 Normal Flow

The use case begins when the ESO employee calls the AWS lambda function.

User	System
3.1.4.1 User passes batch of aerial images through AWS the image classification pipeline	3.1.4.2 The system performs pre-processing tasks on images for feature reduction and extraction.
	3.1.4.3 The system identifies segments of images as affected/unaffected by flood on a pixel by pixel basis.
	3.1.4.4 System returns its classification of flooded areas to the user.

3.1.5 Alternate Flow

If at 3.1.4.4 the system has the capability to transform its pixel-by-pixel analysis to an ArcGIS shapefile object.

User	System
	3.1.5.1 The system transforms the data into an ArcGIS shapefile object to return to the user.

3.1.6 Key Scenarios

3.1.5.1 Classification success

The SageMaker endpoint returns an accurate set of predictions of flood-affected areas at the pixel level of granularity.

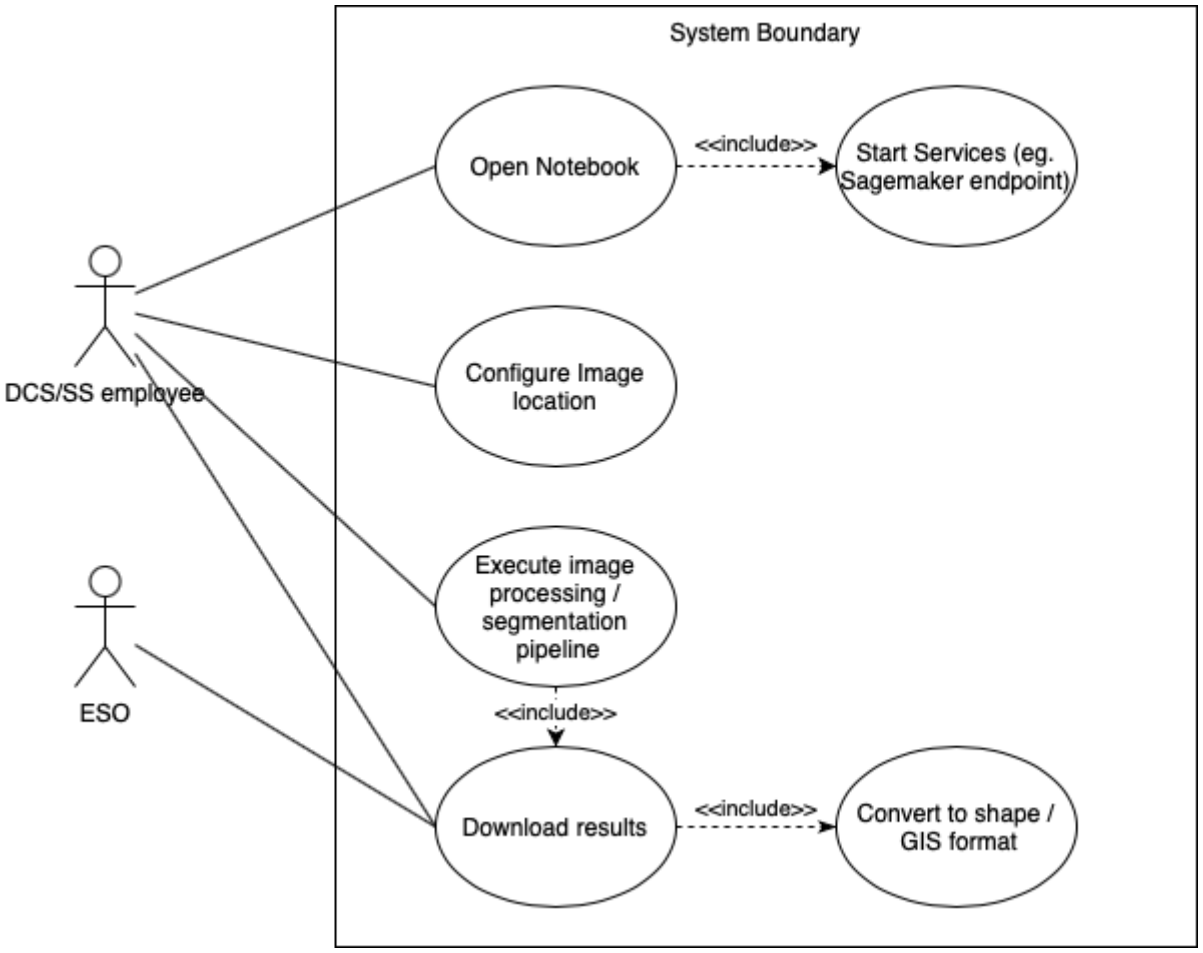
3.1.5.2 Classification success - Alternate scenario

The system returns an Esri shapefile object. This is a polygon overlay that will represent the flood extent map, usable by NSW ESOs.

3.1.6 Post-conditions

6.1 DCS/SS employees are able to provide a flood extent map to other Emergency Service Organisations.

3.1.7 Use Case Diagram



4. Domain Class Diagrams

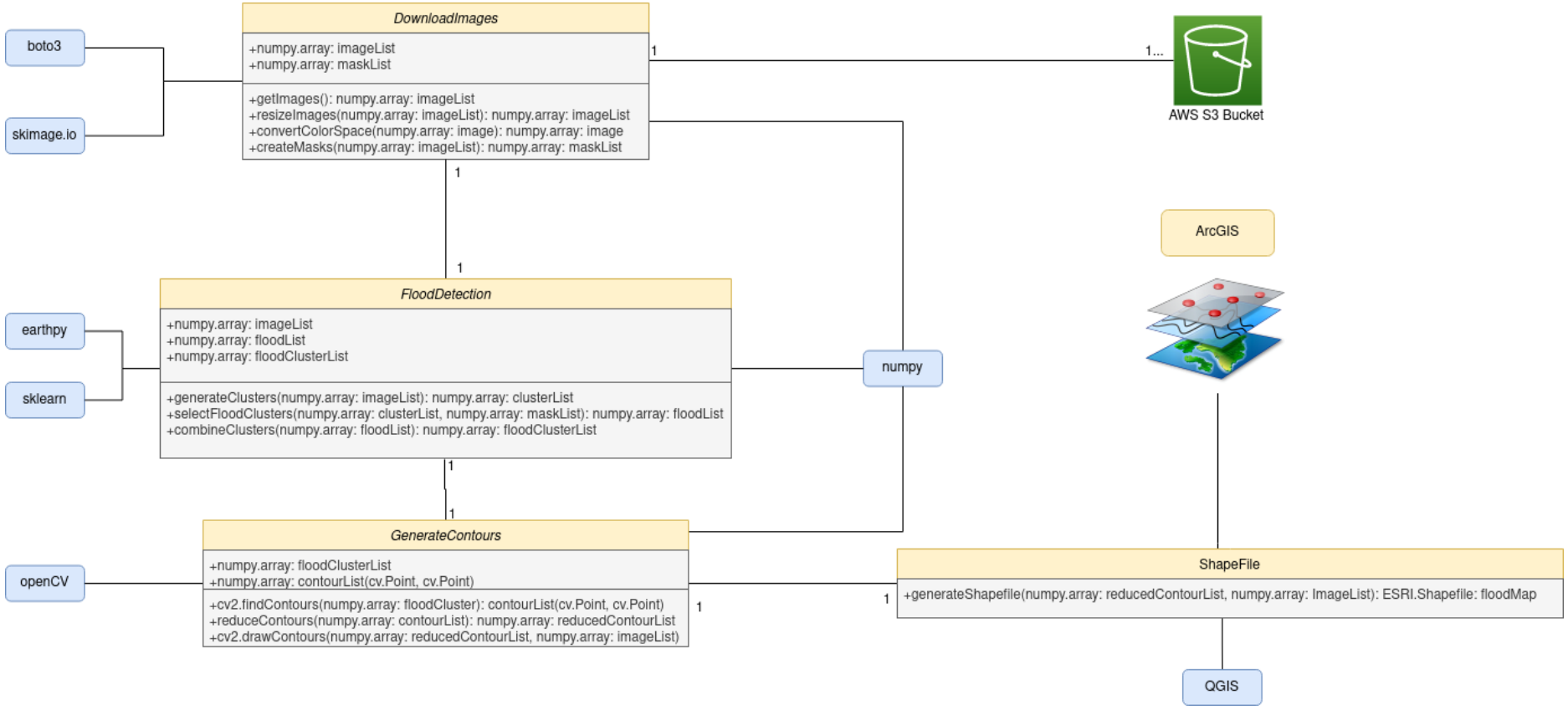


Figure 1 - Domain Class Diagram without Sagemaker Model

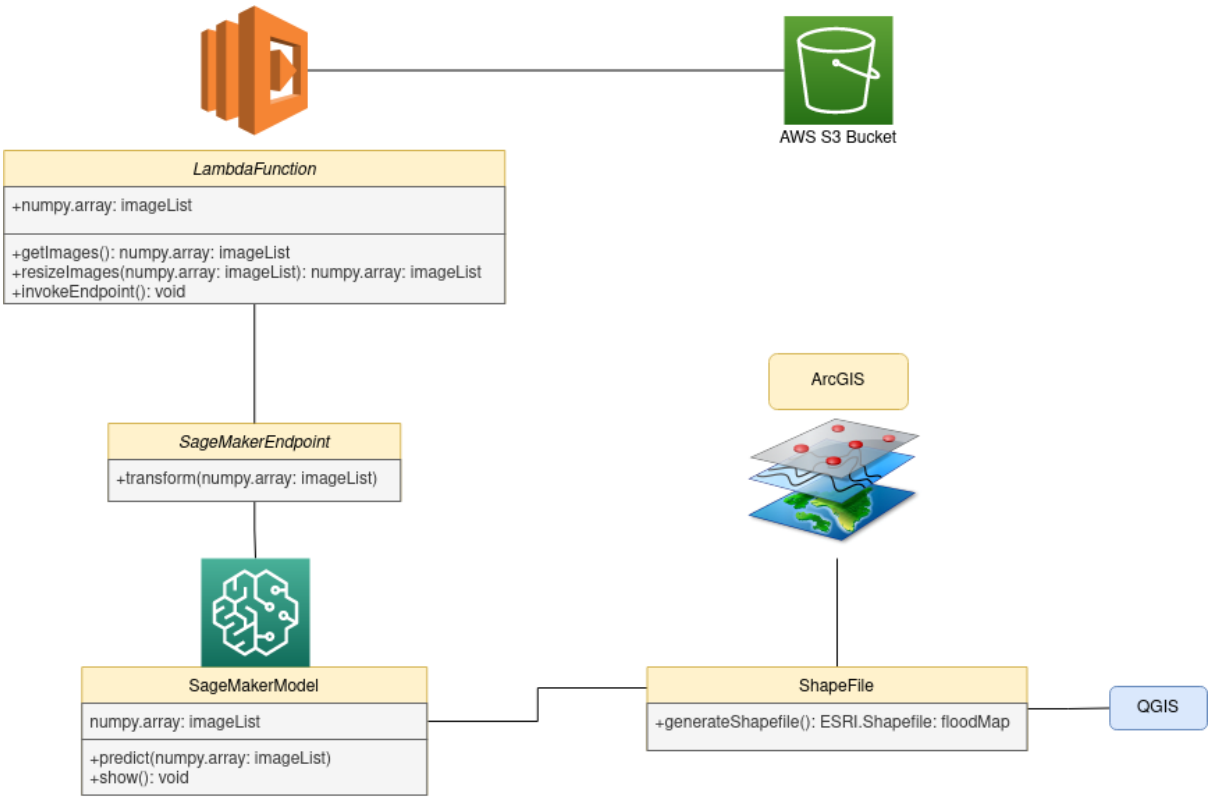


Figure 2 - Domain Class Diagram with SageMaker model[1]

[1] ~~Although domain class diagrams do not normally specify details of implementation such as the use of the AWS Lambda service, the constraints of the project have made such design implementations clear, hence they have been included in the above diagram.~~
The end product may or may not utilise a SageMaker machine learning model, depending on the relative accuracy and processing requirements between the alternative image segmentation and computer vision algorithms.

Change log

No.	Change
1	Added introduction to provide context for the document. Previously there was no introduction
2	Amendments to section 1.1 Business Rules. Initial document was very specific about using machine learning. Document now allows for use of traditional computer vision techniques.
3	Added point 1.1.7 to Business Rules. The time of turnaround is critical for the operation of the capture of aerial imagery. The usefulness of the images degrades with time. Old or stale images are less useful for ESO
4	Amendments to section 1.2.1 External interfaces. Documents now reflect potential use of CV of segmentation.
5	Amendments to section 1.2.2 External interfaces. Removed dependency of using Ground Truth for image labelling. Document now allows for any suitable tool for image labelling.
6	Amendments to section 1.2.3 External interfaces. Removed dependency of using AWS CLI for environment administration
7	Amendments to section 3.1.1 Use Case. User will access the Jupyter Notebook. Project no longer will rely on AWS Lamba functions.
8	Amendments to section 3.1.2 Use Case. Added ESO as an actor to reflect the end client receiving processed data.
9	Amendments to section 3.1.4 Use Case. Removed reference to Lambda function, replaced with “image classification pipeline”
10	Added use case diagram 3.1.7. Previously use case only used text to describe the use cas
11	Added point 1.1.6 to Business Rules to specify the system’s requirement to manage cloud cover and other variables that complicate the ability of image segmentation and clustering algorithms to delineate the flood area.
12	Added to section 1.2 the use of QGIS, a Python third party library for creating shapefile objects for use in GIS software.

13	Added to section 4 two new Domain Class Diagrams to represent two possible architectures.

References

AWS. (2021a). *AWS SageMaker*. https://aws.amazon.com/sagemaker/

AWS. (2021b). *AWS SageMaker Ground Truth*. https://aws.amazon.com/sagemaker/groundtruth/

AWS. (2021c). *AWS Command Line Interface*. https://aws.amazon.com/cli/

AWS. (2021d). *AWS Console*. https://aws.amazon.com/console/

AWS. (2021e). *AWS CloudTrail*. https://aws.amazon.com/cloudtrail/

AWS. (2021f). *AWS S3*. https://aws.amazon.com/s3/

Esri. (2021). *ArcGIS Overview*. https://www.esri.com/en-us/arcgis/about-arcgis/overview

Jupyter. (2021). *Project Jupyter*. https://jupyter.org/

NSW Government. (2013). *NSW Data & Information Custodianship Policy (v1.0)*, https://www.digital.nsw.gov.au/sites/default/files/NSW%20Data%20and%20Information%20Custodianship%20Policy%20v1-0.pdf

NSW Government. (2020). *Cloud Policy*. https://www.digital.nsw.gov.au/policy/cloud-strategy-and-policy/cloud-policy

QGIS. (2021). *A Free and Open Source Geographic Information System*. https://www.qgis.org/en/site/

Title page cover photo.

South Bank flooded. (2011). Retrieved from https://www.abc.net.au/news/2011-06-03/brisbane-floods3a-before-and-after-article/2742794?nw=0