ADVANCES IN FOREST FIRE RESEARCH

2022

Edited by
DOMINGOS XAVIER VIEGAS
LUÍS MÁRIO RIBEIRO

A Platform for Large Scale Application of Remote Sensed Data to Wildland Fire Management

Fredrick Bunt*[1]; Jesse V. Johnson[1]; John Hogland[2]

[1]The University of Montana Department of Computer Science. 32 Campus Drive, Missoula MT 59812, USA, {fredrick.bunt, jesse.johnson}@umontana.edu
[2] Rocky Mountain Research Station, U.S. Forest Service, Missoula, MT 59801, USA, {john.s.hogland@usda.gov}

**Corresponding author*

Keywords

Geospatial processing, remote sensing, fire mitigation, big data, scalable

Abstract

Spatial modelling and machine learning are powerful techniques that can be used to identify patterns within data and build complex relationships between response and predictor variables. While powerful, many of these techniques are computationally intensive and are not designed to fully leverage high performance computing resources, especially when used within a geospatial context. To fully leverage system resources, while facilitating various spatial, machine-learning, and statistical modelling workflows, we developed a Python-based processing library called *raster-tools*. The *raster-tools* library automates delayed reading and parallel processing using Dask and integrates seamlessly into popular spatial, machine learning, and visualisation libraries such as geopandas, rasterio, xarray, scikit-learn, xgboost, pygeos, shapely, matplotlib, plotly, folium, and many more. Combined, these open-source libraries provide users with free and powerful analytical capabilities that can be used at scale and can dynamically display textual, tabular, spatial and graphical data. In this paper, we will provide a brief overview of the *raster-tools* library and demonstrate how the described open-source stack can be used to perform GIS analyses in both a web and desktop environment.

1. Introduction

Our environment is constantly being monitored. Today, satellite and airborne sensors on programs and platforms such as MODIS (MODIS, n.d.), Landsat (USGS, n.d.), Sentinel (EOS, n.d.), and NAIP (NAIP, n.d.) are acquiring data at spatial, spectral, and temporal resolutions that were, until recently, hard to imagine. Similarly, with advancements in drone technology and sensor hardware, the amount of remotely sensed data that is constantly being acquired and used to quantify aspects of natural resources is staggering. The recognition that large volumes of data are not being fully leveraged to inform decision making has led to an increased awareness in the fields of data (Gibert et al, 2018) and decision (Elshawi et al, 2018) science and the potential of what has become known as “Big Data” (Markwo et al, 2017). While there is great potential and promise attributed to “Big Data”, the practical use of data to drive decision making within the natural resource community has not been fully realised (Gibert et al, 2018).

In large part the discrepancy between the potential and use of “Big Data" to aid in natural resource decision making stems from two primary deficiencies: 1) a lack of analysts and technicians trained in the tenets of data science within the natural resources community and 2) software libraries that fully leverage computer resources and integrate tabular, geospatial, and machine learning (ML) domains. Within the first deficiency, common obstacles to implementation include: a lack of education and skills associated with integrating the various mathematical, statistical, ML techniques, computer programming languages, data formats, and the size of the data (Gibert et al, 2018). Less understood issues include leveraging data processing results (e.g. modelled outputs) for efficient decisions, the impact of applying models to new domains, the propagation of errors, and model misspecification. Issues of scale, domain, error, and relevance can have additional meanings within a complex natural resource setting. These issues often prevent studies that convert data into pertinent forest and fire related information from being used to their potential for planning and management decisions.

Within the second identified deficiency (integrated software libraries), geospatial analysis is core to natural resource management and planning. To facilitate geospatial analyses, geographic information systems (GISs) and remote sensing software such as ESRI's software suite (ESRI, n.d.), ERDAS (Hexagon, n.d.), ENVI (L3Harris, n.d.), IDRISI (Clark Labs, n.d.), QGIS (QGIS Dev. Team, n.d.), GRASS (GRASS Dev. Team, n.d.), and Whitebox (Whitebox Geospatial Inc., n.d.) have been developed to support spatial analytics and visualisation. However, commercial software platforms are expensive, typically have only a subset of commonly used routines, have a proprietary code base, do not necessarily integrate well with other processing libraries, and are only partially designed to fully leverage computer hardware, making it challenging to use those systems within a Big Data context. Open-source projects such as QGIS, GRASS, and Whitebox address cost issues but also tend to be plagued by issues similar to their commercial counterparts and typically are less intuitive to use, have stability issues, and often lack documentation.

These obstacles have led some to develop open-source data processing libraries such as gdal (GDAL Contributors., 2022), geopandas (Jordahl et al, 2020), rasterio (Gillies et al, 2013), xarray (Hoyer & Harmon, 2017), scikit-learn (Pedregosa et al, 2011), xgboost (Chen & Guestrin, 2016), shapely (Gillies, 2007), matplotlib (Hunter, 2007), plotly (Plotly Tech. Inc., 2015), and folium (python-visualization, 2020) that build upon common processing frameworks such as numpy (Harris et al, 2020), scipy (Virtman et al, 2020), and pandas (McKinneyl, 2010). However, these libraries alone do not natively address issues of parallel processing, memory management, or excessive use of input and output (Hogland & Anderson, 2017). To address these issues, Dask (Dask Dev. Team, 2016) has built a newer processing framework that builds upon the core processing frameworks of numpy and pandas that can be leveraged to facilitate and integrate tabular, geospatial, and ML analyses through lazy processing and parallelization. Two relatively recent coding projects that have successfully leveraged Dask to facilitate lazy processing, parallelization, and geospatial analyses from a vector and raster perspective include dask-geopandas (Geopandas Dev. Team, n.d.) and xarray-spatial (Makepath, n.d.), respectively. To further address the need for geospatial libraries that fully leverage computer resources and integrate tabular, geospatial, and ML analyses we have developed a new open-source project called *raster-tools*.

Our open-source package leverages the extensive data science, data processing and geospatial ecosystems of Python to provide a platform for developing data-driven decision making tools. Through the use of Python's Dask library (Dask Dev. Team, 2016), *raster-tools* allows users to easily scale their workflow from small laptops up to servers or high-performance computing (HPC) clusters, while fully utilising available resources. It contains a subset of the processing functions offered by ESRI software but can be used to implement many others. *raster-tools* also integrates easily with popular spatial, ML, and visualisation libraries such as geopandas, xarray, scikit-learn, xgboost, shapely, matplotlib, jupyter-lab, folium, and more.

Here, we highlight the use of our *raster_tools* package, in conjunction with an open-source stack, to inform decision making at scale through multiple use cases. Moreover, we demonstrate how this newly developed technology can be easily integrated with other spatial and statistical modelling workflows to realise the potential of Big Data. Finally we discuss the benefits of using *raster-tools* to perform geospatial analyses.

2. Methods

The *raster-tools* project is roughly based on the Rocky Mountain Research Station (RMRS) Raster Utility project (Hogland & Anderson, 2017). While similar in concept, *raster-tools* furthers the intent of the RMRS Raster Utilities project by improving processing efficiencies, increasing the size of datasets that could be processed, expanding possible compute platforms, and providing an open-source set of efficient geospatial, remote sensing, and ML procedures. The *raster-tools* package provides the same lazy processing and execution functionality as RMRS Raster Utility but can easily scale to larger datasets and hardware configurations, run on a wider range of platforms, and directly integrates with Python's wider ecosystem, making it a significant improvement over RMRS Raster Utility project.

The *raster-tools* package is built on the open-source python library Dask (Dask Dev. Team, 2016). Dask is a general-purpose library that embraces lazy operations and data partitioning for parallel data processing. At its core, Dask breaks data into smaller chunks of data, similar to pixel blocks within ESRI's ArcObjects [34], and allows users to apply lazy operations to those data subsets. An operation on the whole dataset is applied as lazy

tasks on the constituent chunks and only takes place when requested by the user. In this way, Dask allows for extensive, lazy data pipelines to be built. For execution, Dask provides robust task scheduling that can distribute per-chunk tasks across available compute resources. This approach to processing automates the parallel aspect of Dask procedures and allows for easy scaling from a single CPU core on a small laptop to distributed HPC clusters. Moreover, processing is performed "out-of-core", making it possible to process large datasets that exceed available memory by only loading chunks into memory at any given time. When chunks are kept small, computation can occur in memory constrained environments, making it feasible to process extremely large datasets quickly and efficiently in a parallel fashion.

To illustrate the benefits of chunking and lazy processing within a geospatial context, we use the *raster-tools* package (Raster-Tools Dev. Team, n.d.) and highlight common and not so common functionality through two use-cases. These are 21st century planning for fire resilient landscapes (Hogland et al, 2021) and burn severity prediction. These use vector and raster based datasets to demonstrate data acquisition, spatial and ML modelling, and visualisation. The rest of the article is separated into Use Case, Discussion, and Conclusion sections that describe the use cases, discuss how *raster-tools* facilitated the analyses within the use cases, and provide concluding remarks, respectively.

3. Use Cases.

3.1. 21st century planning for fire resilient landscapes

The 21st century planning for fire resilient landscapes use case, uses *raster_tools* to perform various spatial analyses to aid in planning forest treatments and quantify the costs and impact of those treatments at scale across a broad landscape. Spatial analyses used in this example include: arithmetic, logical, data format transformations, surface distance, surface allocation, and surface traceback, region grouping, and zonal summaries. Spatial data outputs created in this example include: multiple raster surfaces that quantify desired future conditions (DFCs), actual tonne removal, potential and actual cost, revenue, and profit of treatment implementation for 1.2 million ha in north central Oregon, USA at spatial resolution of 30 m. Primary datasets used within the use case include basal area per ha (BAH: m^2 ha^{-1}), and a most likely classification raster surfaces (Hogland et al, 2021), potential operational delineations (Dunn et al, 2020), United States Census Bureau Tiger/Line files (USCBl, n.d.), the National Hydrography Dataset (NHD) Flowline and Waterbody line and polygon features (NHD, n.d.), and the location of the Malheur Lumber company. All datasets used within the analyses are available for download at Hogland, n.d.a and are explained in further detail in (Hogland et al, 2021).

We provide a Jupyter notebook (Hogland, n.d.b) with an in-depth, dynamic example of the analyses performed within (Hogland et al, 2021) using *raster-tools*. The notebook is meant as a companion piece with (Hogland et al, 2021) and demonstrates everything from installation to final analysis. It is free to download and can be used with Google's Colab for research and educational purposes (Google Research Colab, n.d.).

Key results from the analyses performed in the notebook include spatial surfaces describing estimated treatment cost, revenue, and profit, and the amount of material removed to meet DFCs at 30 m resolution, along with summary reports based on POD boundaries and management priority (Figure 1). Moreover, outputs can be displayed as an interactive map and saved as a HTML file and further embedded within any website. From start to finish the analyses performed within the notebook takes approximately 16 minutes to complete using Colab and represent a substantial improvement in both processing time (minutes vs hours) and storage space (no intermittent datasets were created) over the delivered cost processing technique described in (Hogland et al, 2021).

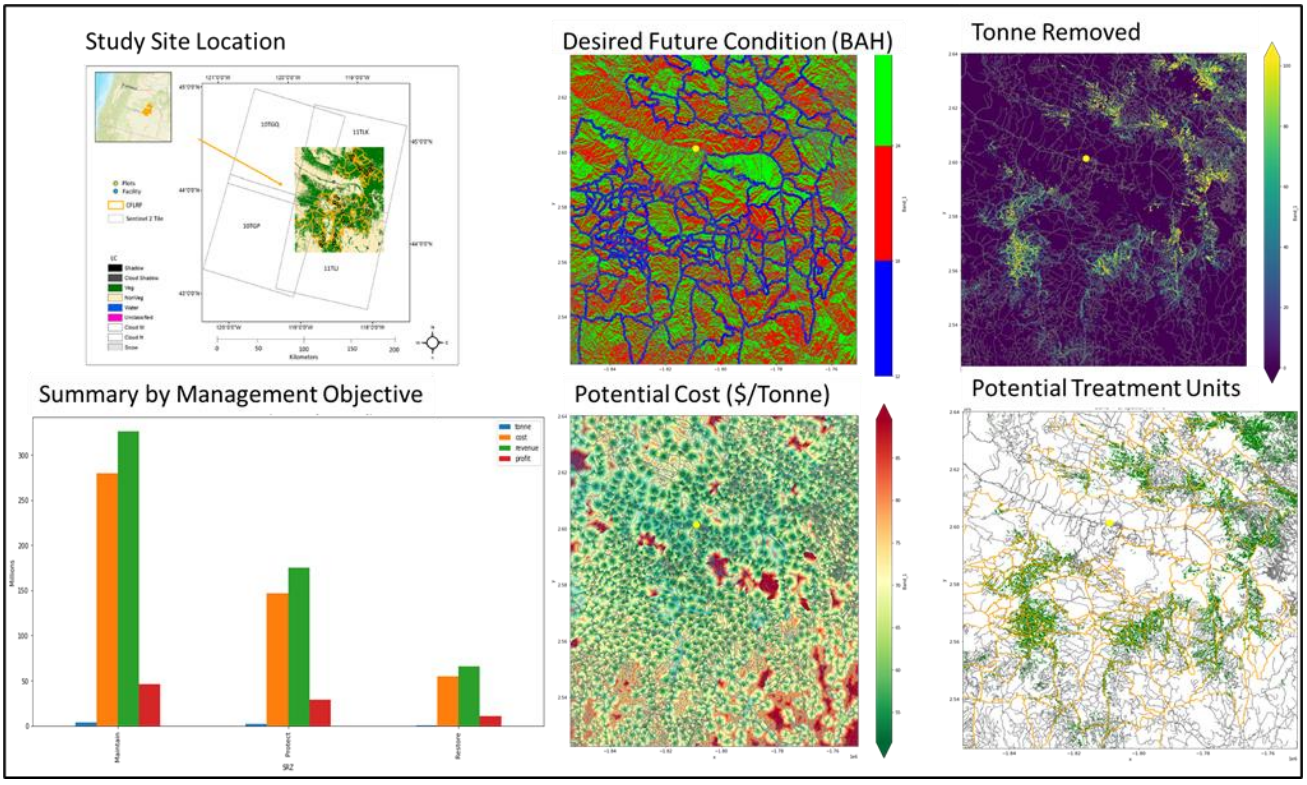

Figure 1- Study site location, Desired Future Condition (DFC), Potential Cost, Removals, Potential Treatment Units, and summarised tonne removed (blue), cost (orange), revenue (green), and profit (red) derived from the 21st century planning notebook.

3.2. Burn Severity Prediction

Our second use case was the development of a burn severity classifier similar to (Parks et al, 2018). For this, we created a training dataset consisting of 29.4 million burn severity labels and predictor values for the state of Montana from 1984–2020. Like (Parks et al, 2018), we used MTBS (Eidenshink et al, 2007) for the severity labels, but increased the number of severity classes used. For the predictors, we used 30 m CONUS EDNA elevation and derivative products (USGS, 2005) and 4 km CONUS gridMET reanalysis products (Abatzoglou, 2013). The training dataset was assembled using *raster-tools* to take MTBS data and pull the corresponding collocated data values from the predictor datasets. Using *raster-tools* allowed us to assemble the training dataset efficiently and in parallel in only a few hours on a desktop computer with limited memory. Figure 2 shows results for a single fire.

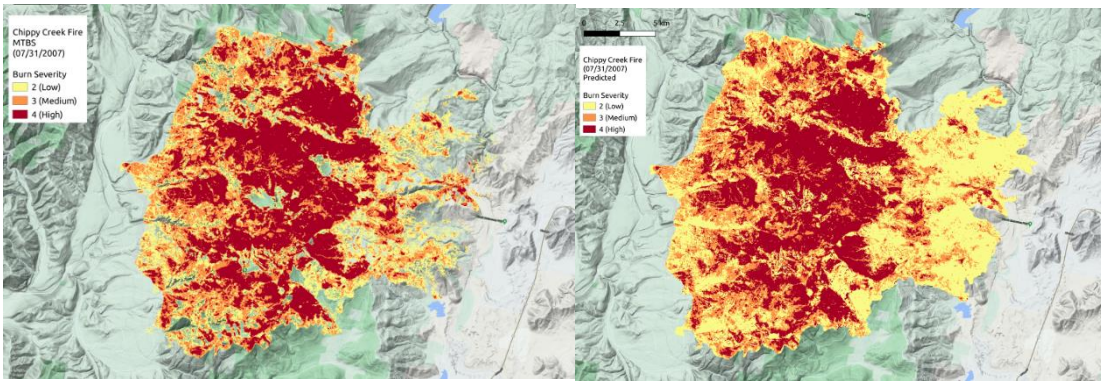

Figure 2- Comparison of MTBS (left) and model predicted (right) burn severity for the 2007 Chippy Creek fire in Montana.

4. Discussion

In both of the above use cases, large, high-resolution rasters and vectors were used as input data. Traditional methods for handling the processing of such data (e.g. ESRI) requires significant investment in computing resources, licensing, and time. With *raster-tools*, we were able to get meaningful, actionable results quickly using relatively small compute platforms. In the first use case, *raster-tools* allowed us to work with large rasters and vectors seamlessly and to carry out a very large number of computations quickly. It eliminated the need to write intermediate results to disk and also allowed us to carry out data analysis on the results with Python's wider data ecosystem.

In the second use case, *raster-tools* allowed us to pull together and work together with nearly 2000 large, high-resolution rasters, simultaneously to produce 29.4 million data points. This would not normally be possible without large investments in computing power, time, and extensive optimization work. Because *raster-tools* uses Python, we were also able to automate this task so that, in the future, we can apply the same processing pipelines to the other parts of the U.S. with only minor changes.

5. Conclusions

Our *raster-tools* package is a free and open-source tool for processing geospatial data. It provides the ability to build automatically scaling processing pipelines that can be run across compute platforms. We think that it can be used as a platform for building data driven applications that produce timely and actionable results at scale.

6. References

MODIS. TERRA The EOS Flagship, Available online: https://terra.nasa.gov/about, (accessed 31 of October, 2019)

United States Geological Survey [USGS]. Landsat 8. Available online: https://www.usgs.gov/land-resources/nli/landsat/landsat-8?qt-science_support_page_related_con=0#qt-science_support_page_related_con (accessed on 23 of October, 2019).

Earth Observing System [EOS]. Sentinel-2. Available online: https://eos.com/sentinel-2/ (accessed on 23 of October 2019).

National Agriculture Imagery Program [NAIP]. National Agriculture Imagery Program (NAIP) Information Sheet. Available online: http://www.fsa.usda.gov/Internet/FSA_File/ naip_info_sheet_2013.pdf (accessed on 14 May 2014).

Gibert, K.; Horsburgh, J.S.; Athanasiadis, I.N.; Holmes, G. Environmental Data Science. Environmental Modelling & Software. 2018, 106, 4-12, doi: 10.1016/j.envsoft.2018.04.005.

Elshawi, R.; Sakr, S.; Talia, D.; Trunfio, P. Big Data Systems Meet Machine Learning Challenges: Towards Big Data Science as a Service, Big Data Research. 2018, 1, 1-11.

Markwo, S.; Braganza, S.; Taska, B. The Quant Crunch - How the Demand for Data Science Skills is Disrupting the Job Market, Burning Glass Technologies Technical Report, 2017, online: https://www.burning-glass.com/wp-content/uploads/The_Quant_Crunch.pdf

Hogland, J.; Dunn, C.J.; Johnston, J.D. 21st Century Planning Techniques for Creating Fire-Resilient Forests in the American West. Forests 2021, 12, 1084. https://doi.org/10.3390/f12081084

Environment Systems Research Institute (ESRI), About ESRI | The Science of Where, Available online: https://www.esri.com/en-us/about/about-esri/overview (accessed on 2 July 2022).

Hexagon, ERDAS Imagine, available online: https://www.hexagongeospatial.com/products/power-portfolio/erdas-imagine, (accessed on 2 July 2022).

L3Harris, Image Processing & Analysis Software | Geospatial Image Analysis Software | ENVI®, available online: https://www.l3harrisgeospatial.com/Software-Technology/ENVI, (accessed on 2 July 2022).

Clark Labs, IDRISI GIS Analysis | Clark Labs, available online: https://clarklabs.org/terrset/idrisi-gis/ (accessed on 2 July 2022).

QGIS Development Team, Welcome to the QGIS project!, https://www.qgis.org/en/site/index.html (accessed 2 July 2022)

GRASS Development Team. GRASS GIS, available online: https://grass.osgeo.org/ (accessed on 2 July 2022).

Whitebox Geospatial Inc. The Whitebox Platform, available online: https://www.whiteboxgeo.com/, (accessed on 2 July 2022).

GDAL/OGR contributors (2022). GDAL/OGR Geospatial Data Abstraction software Library. Open Source Geospatial Foundation. URL https://gdal.org DOI: 10.5281/zenodo.5884351

Jordahl, K. et al. (2020, July 15). geopandas/geopandas: v0.8.1 (Version v0.8.1). Zenodo. http://doi.org/10.5281/zenodo.3946761

Gillies, S. et al.. (2013-). Rasterio: geospatial raster I/O for Python programmers. URL https://github.com/rasterio/rasterio

Hoyer, S. & Hamman, J., (2017). xarray: N-D labeled Arrays and Datasets in Python. Journal of Open Research Software. 5(1), p.10. DOI: https://doi.org/10.5334/jors.148

Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). ACM.

Gillies, S., & others. (2007–). Shapely: manipulation and analysis of geometric objects.

Hunter, J. (2007). Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3), 90–95.

Plotly Technologies Inc. Collaborative data science. Montréal, QC, 2015. https://plot.ly.

python-visualization. (2020). Folium. Retrieved from https://python-visualization.github.io/folium/

Harris, C.R., Millman, K.J., van der Walt, S.J. et al. Array programming with NumPy. Nature 585, 357–362 (2020). DOI: 10.1038/s41586-020-2649-2

Virtanen, P. et al. & SciPy 1.0 Contributors (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17, 261–272.

McKinney, W. (2010). Data Structures for Statistical Computing in Python . In Proceedings of the 9th Python in Science Conference (pp. 56 - 61).

Hogland, J.; Anderson, N. Function Modeling Improves the Efficiency of Spatial Modeling Using Big Data from Remote Sensing. Big Data Cogn. Comput. 2017, 1, 3. https://doi.org/10.3390/bdcc1010003

Dask Development Team (2016). Dask: Library for dynamic task scheduling https://dask.org

Geopandas Development Team, dask-geopandas, available online: https://dask-geopandas.readthedocs.io/en/latest/index.html, (accessed on 2 July 2022).

Makepath, xarray-spatial, https://github.com/makepath/xarray-spatial, (accessed on 2 July 2022).

Raster-tools Development Team. raster-tools, Available online: https://github.com/UM-RMRS/ raster_tools (accessed on 2 July 2022)

ESRI. ArcObjects SDK 10 Microsoft .Net Framework—ArcObjects Library Reference (Spatial Analyst), Available online: http://help.arcgis.com/en/sdk/10.0/arcobjects_net/componenthelp/ index.html#/PrincipalComponents_Method/00400000010q000000/ (accessed on 3 March 2017).

Dunn, C.J.; O'Connor, C.D.; Abrams, J.; Thompson, M.P.; Calkin, D.E.; Johnston, J.D.; Stratton, R., Gilbertson-Day, J. Wildfire risk science facilitates adaptation of fire-prone social-ecological systems to the new fire reality. *Environ. Res. Lett.* 2020, *15*, 1–13, doi:10.1088/1748-9326/ab6498.

USCB. TIGER/Line Shapefiles [Machine-Readable Data Files]. Available online: https://www2.census.gov/geo/tiger/TGRGDB20/ (accessed on 12 May 2021).

National Hydrography Dataset [NHD]. Available online: http://prd-tnm.s3-website-us-west-2.amazonaws.com/?prefix=StagedProducts/Hydrography/NHD/State/HighResolution/GDB/ (accessed on 12 May 2021).

Hogland, J. BMFPNotebookdata.zip. Available online: https://drive.google.com/file/d/1zNYZRTgNEX4mNAtU1I3-7RD8VSo8-ATH/view?usp=sharing (accessed on 7/1/2022).

Hogland, J., 21st Century Planning Techniques for Creating Fire-Resilient Forests in the American West: Notebook, available online: https://github.com/jshogland/SpatialModelingTutorials/blob/main/Notebooks/PODs_Integration.ipynb, (accessed on 7/1/2022).

Google Research. Colab, available online: https://colab.research.google.com/, (accessed 7/1/2022)

Parks, S.; Holsinger, L; Panunto, M; Jolly, W Matt; Dobrowski, Solomon; Dillon, Gregory. High-severity fire: evaluating its key drivers and mapping its probability across western US forests, 2018, Environmental Research Letters Vol. 13 No. 4.

Eidenshink, J., Schwind, B., Brewer, K. et al. A Project for Monitoring Trends in Burn Severity. fire ecol 3, 3–21 (2007). https://doi.org/10.4996/fireecology.0301003.

U.S. Geological Survey, 2005, Elevation derivatives for national applications: U.S. Geological Survey Fact Sheet 2005–3049, 2 p., https://doi.org/10.3133/fs20053049

Abatzoglou, J. T. (2013), Development of gridded surface meteorological data for ecological applications and modelling. Int. J. Climatol., 33: 121–131.