

# Tutorials for the GRASS geocomputation engine

Brendan A. Harmon  <sup>1</sup>, Veronica Andreo  <sup>2</sup>, Anna Petrasova  <sup>3</sup>, Vaclav Petras  <sup>3</sup>, Caitlin Haedrich  <sup>3</sup>, and Corey White  <sup>3</sup>

<sup>1</sup> Louisiana State University, United States  <sup>2</sup> Instituto Gulich, Argentina  <sup>3</sup> North Carolina State University, United States 

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

## Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Submitted: 01 January 1970

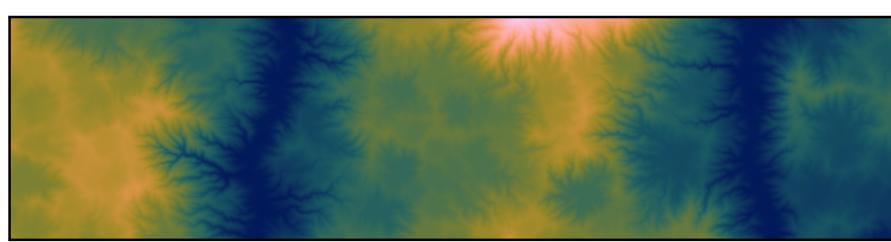
Published: unpublished

## License

Authors of papers retain copyright<sup>12</sup> and release the work under a<sup>13</sup> Creative Commons Attribution 4.0<sup>14</sup> International License ([CC BY 4.0](#))<sup>15</sup>

## Summary

This collection of tutorials is an introduction to the GRASS geospatial processing engine.<sup>7</sup> GRASS is an open source computational engine for spatiotemporal data management,<sup>8</sup> analysis, modeling, and simulation ([GRASS Development Team et al., 2025](#); [Neteler & Mitášová, 2008](#)). As an engine that can be integrated in data science pipelines with shell scripting, Python, R, Jupyter, and Colab, there are many ways to use GRASS. While GRASS already had extensive documentation of individual processing tools, tutorials were needed to introduce the many ways to interface with the tools and combine them into computational workflows ([Figure 1](#)). These open education tutorials - which cover integrations, core features, and disciplinary applications - were developed as part of an effort to grow the GRASS community. The tutorials are built with Quarto and are deployed as webpages paired with Jupyter computational notebooks. The tutorials are available at <https://grass-tutorials.osgeo.org> under both the GNU Free Documentation License v1.2 or later and the Creative Commons Attribution-ShareAlike 4.0 International License.<sup>16</sup>



**Figure 1:** An example of fractal terrain generated with GRASS and eroded with LandLab from a [tutorial](#) on scientific modeling. This tutorial demonstrates a seamless workflow for scientific modeling in Python, showing how gridded data can be passed as arrays between GRASS, NumPy ([Harris et al., 2020](#)), and Landlab ([Barnhart et al., 2020](#)).

## Statement of Need

As GRASS has grown from its roots as a geographic information system ([Westervelt, 2004](#)), it has evolved into a geocomputational engine with many interfaces. As an engine, it can be integrated in geospatial data science pipelines using shell scripting, application programming interfaces, tangible interfaces ([Petrášová et al., 2018](#)), computational notebooks ([Haedrich et al., 2023](#)), cloud computing environments ([Neteler et al., 2019](#); [White et al., 2023](#)), or high performance computing environments. While GRASS is well documented with books, a user manual, developer manuals, and a wiki, it lacked official tutorials. Over the years, the community developed many tutorials across different platforms, but as these

29 are independently maintained, many have become outdated and obsolete. The current  
30 roadmap for GRASS – established in 2024 – calls for official new tutorials to encourage  
31 community growth and demonstrate integrations in data science pipelines.

32 The design and implementation of the new official tutorials for GRASS was based on  
33 experience teaching university courses and conference workshops using open educational  
34 resources. Over the last decade, the GRASS community has developed many open  
35 educational resources, experimenting with delivery via web documents, computational  
36 notebooks, and cloud computing services. Online tutorials for GRASS have been built  
37 from source in HTML (Petráš et al., 2015), built from Markdown with a static site  
38 generator (Harmon, 2020), included Jupyter notebooks (Haedrich et al., 2023), and used  
39 cloud computing services such as Binder (Petrášová & Petráš, 2019), Whole Tale (Andreо,  
40 2023b), and Google Colab (Andreо, 2023a). Petráš et al. (2015) used a modular structure  
41 with tabs to teach the core interfaces for GRASS – the GUI, CLI, and Python API –  
42 separating explanatory text introducing geospatial concepts from software specific text  
43 for each interface. This scaffolding helps learners to focus on concepts, while building  
44 their skills with increasingly complex interfaces. Haedrich et al. (2023) developed the  
45 GRASS–Jupyter integration to incorporate more scripting into a graduate-level course on  
46 geospatial computing and simulation. The package extends the existing GRASS Python  
47 APIs with data visualization and management tools for the Jupyter environment. The  
48 new course materials include Jupyter Notebooks that combine tutorials and assignments,  
49 allowing students to write and modify code, interact with examples, and explain their  
50 reasoning in markdown, all within a single document. Based on these experiences, our  
51 design principles for the new tutorials include teaching geospatial concepts discretely  
52 from software specifics to encourage spatial thinking, supporting live coding to encourage  
53 computational thinking, and using an open source publishing system to build documents  
54 from plain text tracked with version control.

## 55 **Description**

### 56 **Learning Objectives**

57 This collection of tutorials was designed to teach geocomputational thinking using the  
58 GRASS geoprocessing engine. To introduce computational approaches to thinking (National  
59 Research Council, 2010; Weintrop et al., 2016) about space and time, the tutorials  
60 cover the fundamentals of geoprocessing with GRASS, integrations of GRASS into data  
61 science pipelines, and disciplinary applications of GRASS. The tutorials were designed for  
62 self-study by learners of all levels, integration into courses, and deployment in workshops.

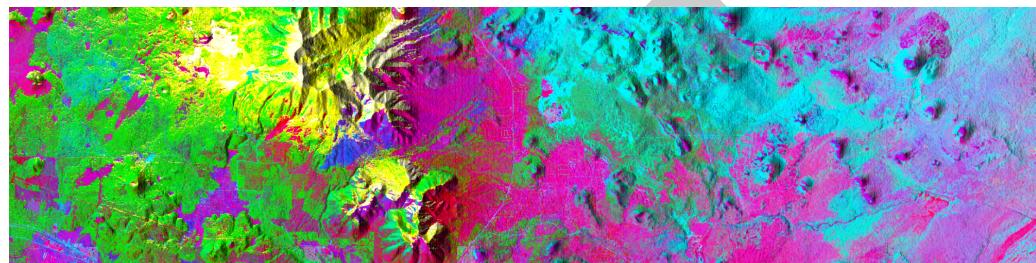
### 63 **Instructional Design**

64 In order to teach a computational approach to thinking about spatiotemporal phenomena  
65 through different interfaces to the GRASS engine, the tutorials were designed:

- 66 • as modules for reuse and remixing,
- 67 • as worked examples to reduce cognitive load,
- 68 • as interactive lessons for active learning and engagement,
- 69 • as scaffolded prose and code to structure learning,
- 70 • and as computable content to teach computational thinking.

71 Drawing on the education benefits of computational notebooks (Barba et al., 2022), the  
72 tutorials introduce geocomputational concepts through worked examples that synthesize  
73 prose explanations, graphics, and executable code. The tutorials, which range from  
74 introductory to advanced, have a modular design for reuse and remixing so that learners  
75 can choose their own course of study and teachers can select modules for their lesson  
76 plans. The tutorials build in complexity from introductory to core to disciplinary modules.

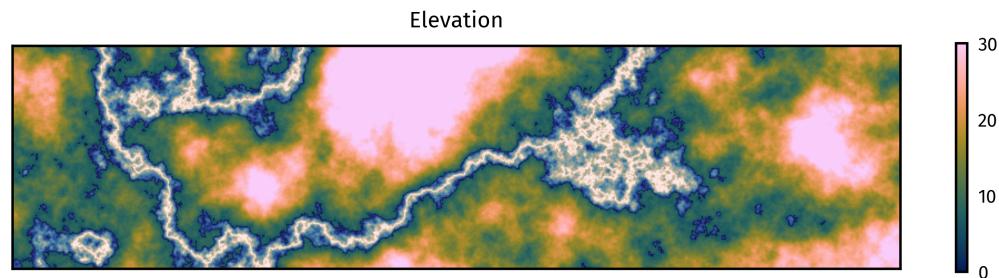
77 A set of getting-started tutorials introduce different ways to interface with the GRASS  
78 engine. The core tutorials cover important concepts such as geovisualization ([Figure 2](#)),  
79 map algebra ([Figure 3](#)), geospatial modeling, and the temporal framework. Disciplinary  
80 tutorials demonstrate applications for GRASS in domains such as climatology, ecology,  
81 hydrology, geomorphology. The disciplinary tutorials build engagement by working through  
82 applications in the learner's domain and thus motivate further exploration. Throughout  
83 the tutorials, different ways to interface with GRASS are presented as tabs in code blocks,  
84 so that learners can work their way through the same tutorial repeatedly using increasingly  
85 challenging interfaces – building proficiency first with the graphical user interface (GUI),  
86 then with the command line interface (CLI), and finally with the Python or R application  
87 programming interfaces (API).



**Figure 2:** An example of image fusion of principal components analysis of multi-band images of the San Francisco volcanic field from a [tutorial](#) introducing the basics of remote sensing in GRASS. This tutorial demonstrates how to process and visualize multi-band remote sensing imagery.

## 88 Implementation

89 This collection of tutorials was published as web documents paired with computational  
90 notebooks. To reach a broad audience, the tutorials are published as web documents for  
91 immediate, easy access via web browsers. When appropriate, web documents are accompa-  
92 nied by a downloadable computational notebook, encouraging interactivity, engagement,  
93 and geocomputational thinking. The tutorials – which are built and deployed using the  
94 Quarto scientific publishing system ([Allaire et al., 2025](#)) – are written in Markdown with  
95 YAML frontmatter. Tutorials are composed in Markdown for human-readable source code,  
96 efficient version control, executable code blocks for different interfaces, rendering in multi-  
97 ple formats, and reproducibility. As this open education project aims to teach different  
98 ways to interface with GRASS, executable code for multiple relevant interfaces such as the  
99 GUI, CLI, Python, or R can be included in tutorials as tabs. Once tutorials have been  
100 written, they are reviewed by the GRASS Development Team, rendered as web documents  
101 and Jupyter notebooks, and deployed to an Open Source Geospatial Foundation website.  
102 The source code for the tutorials is built in the GitHub repository <https://github.com/OS-Geo/grass-tutorials> and deployed to the website <https://grass-tutorials.osgeo.org> using  
103 GitHub Actions.  
104



**Figure 3:** An example of synthetic terrain generated using map algebra from a [tutorial](#) introducing the basics of map algebra in GRASS. This tutorial demonstrates local algebraic operations using the raster map calculator, focal operations using nearest neighbors analysis, zonal operations using conditional statements with the raster map calculator, and global operations using raster metadata.

## 105 Content

106 This official collection of tutorials is maintained by the GRASS Development Team as part  
 107 of the documentation for the GRASS geocomputational engine ([Table 1](#)). This ensures  
 108 that tutorials are standardized, undergo rigorous review, and are maintained and updated  
 109 as GRASS evolves. The website also includes a curated collection of external tutorials  
 110 that are hosted and maintained by their creators ([Table 2](#)).

**Table 1:** Official GRASS tutorials

Modules	Tutorials	Level	Lang.
Core	Get started with GRASS GUI	Beginner	En
	Get started with GRASS & Python in Jupyter Notebooks	Beginner	En
	Get started with GRASS in Google Colab	Beginner	En
	Get started with GRASS in Jupyter Notebooks on Windows	Beginner	En
	Quick comparison: R and Python GRASS interfaces	Intermediate	En
	Get started with GRASS & R: the rgrass package	Advanced	En
	Basics of map algebra	Beginner	En
	Making plots with GRASS	Beginner	En
	Visualizing and modeling terrain from DEMs in GRASS	Beginner	En & Pt
	Introduction to remote sensing with GRASS	Beginner	En
Disciplinary	Making thematic maps	Beginner	En
	Introduction to time series in GRASS	Intermediate	En
	Temporal subset, import and export	Intermediate	En
	Temporal aggregations	Advanced	En
	Temporal algebra	Advanced	En
	Temporal accumulation	Advanced	En
	Temporal gap-filling	Advanced	En
	Temporal query with vector data	Advanced	En
	Modeling movement in GRASS	Advanced	En & Pt
	Basic earthworks	Beginner	En
	Gully modeling	Beginner	En

Modules	Tutorials	Level	Lang.
	Coastal infrastructure	Beginner	En
	Terrain synthesis	Intermediate	En
	Procedural noise	Intermediate	En
	Hydro-flattening a digital elevation model	Intermediate	En
	Using GRASS, NumPy, and Landlab for scientific modeling	Intermediate	En
	fasterRaster: faster raster processing in R Using GRASS	Intermediate	En
	Estimating wind fetch	Advanced	En
	Parallelization of overland flow simulation	Advanced	En

Table 2: External tutorials

Modules	Tutorials	No.	Level	Lang.
Integrations	Unleash the power of GRASS	5	Beginner - Advanced	En
	GRASS for remote sensing data processing with Jupyter Notebooks	1	Advanced	En
Core	NCSU geospatial modeling and analysis course	13	Beginner - Intermediate	En
	Geoprocessamento com GRASS	1	Beginner - Intermediate	Pt
Disciplinary	Tutoriales de GRASS en grasswiki	4	Beginner - Intermediate	Es
	GISMentors	30	Beginner - Advanced	En & Cs
Disciplinary	Deforestation study using GRASS	1	Beginner	En
	Teledetección, OBIA y series de tiempo	5	Beginner - Advanced	Es
Disciplinary	GIS for designers	12	Beginner - Intermediate	En
	GRASS for environmental monitoring and disease ecology	2	Beginner - Intermediate	En
Disciplinary	Processing lidar and UAV point clouds	1	Beginner - Intermediate	En
	Physically-based hydrologic modeling using GRASS r.topmodel	1	Intermediate	En
Disciplinary	Spatio-temporal data handling and visualization	1	Intermediate	En
	Ecodiv.earth tutorials	16	Beginner - Advanced	En
Disciplinary	Urban growth modeling with FUTURES	1	Advanced	En

### <sup>111</sup> Acknowledgements

<sup>112</sup> The development of the tutorial platform and the majority of the tutorials presented in  
<sup>113</sup> this collection were supported by the U.S. National Science Foundation under Award

<sup>114</sup> No. [2303651](#). This grant directly funded the development of tutorials and also funded  
<sup>115</sup> mentor support for community contributors working on tutorial content. Two tutorials  
<sup>116</sup> received additional partial support from NSF Award No. [2322073](#) granted to Natrx,  
<sup>117</sup> Inc. Another tutorial was developed with combined support from NSF Award No. 2322073  
<sup>118</sup> and USDA NRCS Award No. NR233A750023C043. We thank the GRASS development  
<sup>119</sup> team and the broader GRASS community for their continued support and contributions  
<sup>120</sup> to this educational effort.

## <sup>121</sup> References

- <sup>122</sup> Allaire, J. J., Teague, C., Scheidegger, C., Xie, Y., Dervieux, C., & Woodhull, G. (2025).  
<sup>123</sup> *Quarto* (Version 1.8). <https://doi.org/10.5281/zenodo.5960048>
- <sup>124</sup> Andreo, V. (2023a). *Procesamiento y análisis de series temporales con GRASS GIS*.  
<sup>125</sup> <https://veroandreo.github.io/cursograssgis/>
- <sup>126</sup> Andreo, V. (2023b). *Using satellite data for species distribution modeling with GRASS GIS and r*.  
<sup>127</sup> [https://veroandreo.github.io/grass\\_ncsu\\_2023/studio\\_index.html](https://veroandreo.github.io/grass_ncsu_2023/studio_index.html)
- <sup>128</sup> Barba, L. A., Barker, L. J., Blank, D. S., Brown, J., Downey, A., George, T., Heagy,  
<sup>129</sup> L. J., Mandli, K., Moore, J. K., Lippert, D., Niemeyer, K., Watkins, R., West, R.,  
<sup>130</sup> Wickes, E., Willling, C., & Zingale, M. (2022). *Teaching and Learning with Jupyter*.  
<sup>131</sup> <https://doi.org/10.6084/m9.figshare.19608801.v1>
- <sup>132</sup> Barnhart, K. R., Hutton, E. W., Tucker, G. E., Gasparini, N. M., Istanbulluoglu, E., Hobley,  
<sup>133</sup> D. E., Lyons, N. J., Mouchene, M., Nudurupati, S. S., Adams, J. M., & Bandaragoda,  
<sup>134</sup> C. (2020). Landlab v2. 0: A software package for earth surface dynamics. *Earth  
<sup>135</sup> Surface Dynamics*, 8(2), 379–397. <https://doi.org/10.5194/esurf-8-379-2020>
- <sup>136</sup> GRASS Development Team, Landa, M., Neteler, M., Metz, M., Petrášová, A., Petrás, V.,  
<sup>137</sup> Clements, G., Zigo, T., Larsson, N., Kladivová, L., Haedrich, C., Blumentrath, S.,  
<sup>138</sup> Andreo, V., Cho, H., Gebbert, S., Nartišs, M., Kudrnovsky, H., Delucchi, L., Zambelli,  
<sup>139</sup> P., ... Bowman, H. (2025). *GRASS GIS* (Version 8.4.0). <https://doi.org/10.5281/zenodo.4621728>
- <sup>141</sup> Haedrich, C., Petrás, V., Petrášová, A., Blumentrath, S., & Mitášová, H. (2023). Integrating GRASS GIS and jupyter notebooks to facilitate advanced geospatial modeling education. *Transactions in GIS*, 27(3), 686–702. <https://doi.org/10.1111/tgis.13031>
- <sup>144</sup> Harmon, B. (2020). *GIS for designers*. <https://baharmon.github.io/gis-for-designers>
- <sup>145</sup> Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau,  
<sup>146</sup> D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S.,  
<sup>147</sup> Kerkwijk, M. H. van, Brett, M., Haldane, A., Río, J. F. del, Wiebe, M., Peterson, P., ...  
<sup>148</sup> Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362.  
<sup>149</sup> <https://doi.org/10.1038/s41586-020-2649-2>
- <sup>150</sup> National Research Council. (2010). *Report of a workshop on the scope and nature of computational thinking*. The National Academies Press. <https://doi.org/10.17226/12840>
- <sup>153</sup> Neteler, M., Gebbert, S., Tawalika, C., Bettge, A., Benelcadi, H., Löw, F., Adams, T.,  
<sup>154</sup> & Paulsen, H. (2019). Actinia: Cloud based geoprocessing. *Proceedings of the 2019 Conference on Big Data from Space (BiDS'2019)*, 41–44. <https://doi.org/10.5281/zenodo.2631917>
- <sup>157</sup> Neteler, M., & Mitášová, H. (2008). *Open source GIS: A GRASS GIS approach*. Springer.  
<sup>158</sup> <https://doi.org/10.1007/978-0-387-68574-8>
- <sup>159</sup> Petrás, V., Petrášová, A., Harmon, B. A., Meentemeyer, R. K., & Mitášová, H. (2015).  
<sup>160</sup> Integrating free and open source solutions into geospatial science education. *ISPRS*

- 161        *International Journal of Geo-Information*, 4(2), 942–956. <https://doi.org/10.3390/ijgi4020942>
- 162
- 163        Petrášová, A., Harmon, B., Petráš, V., Tabrizian, P., & Mitášová, H. (2018). *Tangible Modeling with Open Source GIS* (2nd ed.). Springer. <https://doi.org/10.1007/978-3-319-89303-7>
- 164
- 165
- 166        Petrášová, A., & Petráš, V. (2019). *FUTURES model introduction using jupyter notebook*. <https://github.com/ncsu-landscape-dynamics/futures-model-intro-notebook>
- 167
- 168        Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, 25(1), 127–147.
- 169
- 170
- 171        Westervelt, J. (2004, July). GRASS roots. *Proceedings of the FOSS/GRASS User Conference*. [https://grass.osgeo.org/files/westervelt2004\\_GRASS\\_roots.pdf](https://grass.osgeo.org/files/westervelt2004_GRASS_roots.pdf)
- 172
- 173        White, C. T., Petrášová, A., Petráš, V., Tateosian, L. G., Vukomanovic, J., Mitášová, H., & Meentemeyer, R. K. (2023). An open-source platform for geospatial participatory modeling in the cloud. *Environmental Modelling & Software*, 167, 105767. <https://doi.org/https://doi.org/10.1016/j.envsoft.2023.105767>
- 174
- 175
- 176