Taylor & Francis
Taylor & Francis Group

Journal of Maps

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tjom20

3D Visualization of geological structures using Python: the case study of the Palomeque sheets (SE, Spain)

Manuel Bullejos & Manuel Martín-Martín

To cite this article: Manuel Bullejos & Manuel Martín-Martín (2023) 3D Visualization of geological structures using Python: the case study of the Palomeque sheets (SE, Spain), Journal of Maps, 19:1, 2282593, DOI: 10.1080/17445647.2023.2282593

To link to this article: https://doi.org/10.1080/17445647.2023.2282593

JOURNAL OF MAPS
2023, VOL. 19, NO. 1, 2282593
https://doi.org/10.1080/17445647.2023.2282593

OPEN ACCESS

3D Visualization of geological structures using Python: the case study of the Palomeque sheets (SE, Spain)

Manuel Bullejos [a] and Manuel Martín-Martín [b]

[a]Departamento de Álgebra, University of Granada, Granada, Spain; [b]Departamento de Ciencias de la Tierra y Medio Ambiente, University of Alicante, Alicante, Spain

ABSTRACT

The goal of this paper is the construction of computerized 3D visualization of geological structures. Several Python applications have been used to adapt the paper map-based geological classical information to numerical geological maps represented in HTML files. The models include a map with the stratigraphic and structural contacts and symbols, five serial vertical sections, and a geological block diagram, all with real topography. This block diagram made with 2D figures allows a 3D visualization. Palomeque area (Murcia region, southeastern Spain) has been used as a key-case. This area consists of a deformed Upper Cretaceous to Oligocene succession belonging to the Internal Zone Malaguide Complex. The main structure consists of two thrust-fold sheets forming an imbricate system, also affected by a set of strike-slip faults with a sinistral regime. The constructed maps show a good agreement with the published classical geological maps and cross-sections demonstrating the benefits of using these Python applications.

ARTICLE HISTORY
Received 7 September 2023
Revised 1 November 2023
Accepted 6 November 2023

KEYWORDS
Python applications; geological map and vertical sections; geological block diagram; thrust-fold sheets; Internal Betic Zone; Malaguide Complex

1. Introduction

3D geological modeling has increasingly developed over the last thirty years through the use of computerized tools mainly developed for applied geology. These tools were implemented with large sums of money specifically focused on those applied geological scenarios. In particular, CAD and GIS tools were customized to address geological problems, but this often led to a complicated solution with the use of multiple software, which became difficult for geologists to use. To solve this, many geological organizations began to develop software systems and methodologies (free access or commercial) to facilitate the migration from 2D paper map-based models to 3D digital geological models through 2+1/2D or pseudo-3D models (Jackson, 2005). It is quite frequent nowadays to make use of geostatistical methods to study numerical variables distributed in space and apply these methods to mapping, modeling, and visualizing geological structures and sedimentary frameworks. Geospatial and geostatistical techniques usually require large amounts of data to produce stochastic or deterministic maps and models. Usually, these techniques allow the analysis to validate or calibrate the predictions providing degrees of certainty. Moreover, computerized tools also allow measuring the precision of the studied variables in the 3D mapping, modeling, and visualizations (Ford et al., 2010; Jessell, 2001; Rohmer et al., 2020; Wycisk et al., 2009). Computerized 3D models also allow the incorporation of new data into representations along the creation process, being very interesting for decision-making in applied geology such as hydrogeology, civil engineering, and/or mining. There are many 3D applications for visualizing geological elements performed with different softwares. These can be grouped into: (1) open source: Gempy, OSGeo, Qgis; (2) and commercial: MOVE, 3D Geomodeller, Gocad, Autocad, VOXI, ArcGis, Geoscene 3D, PETREL, among others. The great advantage of commercial tools is that those are easier to use and usually have a technical help service. The negative part is the elevate cost of acquisition. In contrast, open source tools are more difficult to use and do not have technical help service but the absence of acquisition cost and adaptability are the most positive aspects of open tools. In our case we do not have a large amount of data, the data with which our models are built comes from classical field work (field geological mapping, cross-sections, and stratigraphic column measurements) without the use of drilling or geophysics studies. In contrast, a statistical study usually requires a lot of data, in most cases coming from drilling, which is resource-intensive. Commercial applications usually require a large amount of data to

CONTACT Manuel Martín-Martín manuel.martin.m3@gmail.com Departamento de Ciencias de la Tierra y Medio Ambiente, University of Alicante, 03080 Alicante, Spain

Supplemental data for this article can be accessed online at https://doi.org/10.1080/17445647.2023.2282593.

bring the study of an area into agreement with expectations. Several Python tools are used in pure and applied geology as GeoPandas, GemPy, or Albion and posts listing libraries as GISgeography and Parpoil. Recently, many papers show Python tools applied to geological contexts (Evangelidis et al., 2018; Hobona et al., 2006; Husillos, 2022; Miao et al., 2017; Pyrcz, 2023; Semmo et al., 2015). In recent years, our team has gained experience applying Python tools to geological problems, developing HTML-rendered 3D maps and models using boreholes data (Bullejos et al., 2022a, 2022b, 2023; Martín-Martín et al., 2023). In this work, we go further and we have applied the same methodology to paper map-based geological classical information from surface without drilling or geophysics studies. For details on the procedures see the GitHub repository with Jupyter notebooks which show the Python codes and the used methodology (https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain).

HTML maps and models, as mentioned above, require only an Internet browser to work with them. It is possible to download plots as png files and to rotate and/or zoom in and out the model. In addition, there is a side legend that allows the user to toggle on and off the visibility of different elements. The Palomeque sheets consist of a duplex system (Martín-Martín & Martín-Algarra, 2002; Martín-Martín et al., 2006b; Martin-Rojas et al., 2007) in the Mula-Pliego Depression from Sierra Espuña area (Martín-Martín et al., 2006a; Martín-Martín & Robles-Marín, 2020; Moliner-Aznar et al., 2021; Perri et al., 2017; Tent-Manclu et al., 2001) in the Murcia region (southeastern Spain: Figure 1). This area belongs to the Internal Betic Cordillera where the Malaguide Complex cropsout (Martín-Martín et al., 2006a, 2006b; Martín-Martín & Martín-Algarra, 2002; Martín-Martín & Robles-Marín, 2020; Martin-Rojas et al., 2007; Moliner-Aznar et al., 2021; Perri et al., 2013, 2017; Tent-Manclu et al., 2001). This cordillera is the continuation of other Alpine chains as the Rif-Tell in north Africa, to the south, and the Iberian Range, Pyrenees, and Alps in Iberia-Europe, to the north. All these chanins are derived from subduction and plates collisions in the westernmost Tethys during Cenozoic times (Belayouni et al., 2023; Guerrera et al., 2021; Jabaloy-Sánchez et al., 2019; Martín-Martín et al., 2020a, 2020b). In the study area (Figure 1C), Malaguide Upper Cretaceous to Oligocene stratigraphic formations crop out (Martín-Martín et al., 2020c, 2021; Martín-Martín & Robles-Marín, 2020; Moliner-Aznar et al., 2021; Perri et al., 2017). The entire area is developed in (Figure 1D) in the Mula-Pliego Depression (a synclinorium with minor structural culminations related to fold-thrust), to the north, followed to the south by the antiformal stack of Sierra Espuña s.s. (Martín-Martín & Martín-Algarra, 2002; Martín-Martín et al., 2006b; Martin-Rojas et al., 2007). In the synclinorium two thrust-fold sheets systems can be found (Martín-Martín & Martín-Algarra, 2002; Martín-Martín et al., 2006b; Martin-Rojas et al., 2007): the Mula sheets to the northeast, and the Palomeque ones to the southwest (Figure 1C).

This area has been largely studied by one of us in the past (Martín-Martín, 1996; Martín-Martín et al., 1997, 1998, 2006a, 2006b; Martín-Martín & Martín-Algarra, 2002; Martin-Rojas et al., 2007) and in recent times (Martín-Martín & Robles-Marín, 2020, Martín-Martín et al., 2021; Moliner-Aznar et al., 2021; Perri et al., 2017) so that it has been selected as case study to run the application.

The aim of this work is the 3D visualization of maps and models of the Palomeque sheets structures (Figure 2). These sheets are made of two thrusting elements (Calvillo and Palomeque heights from N to S) affected later by several strike-slip faults. In these structures, a deformed Cretaceous to lower Miocene Succession is observed (Figure 2). For our representation we used the stratigraphic nomenclature available in literature (Figure 2) according to Martín-Martín and Martín-Algarra (2002), Martín-Martín et al. (2006b), Martin-Rojas et al. (2007), and Martín-Martín et al. (2020c, 2021): (1) the Upper Cretaceous Capas Blancas Fm consisting in white marls and marly-limestones; (2) The Mula Fm (Thanetian in age in this area) made of microcodium-rich greyish sandy marls and calcarenites; (3) the Espuña Fm (Ypresian-Lower Lutetian) made of cream calcarenites and limestones with small size Nummulites and Alveolines; (4) the lower Malvariche Fm (Middle-Upper Lutetian) made of pinkish marls and intercalate beds of calcarenites with large size larger benthic foraminifera (Nummulites, Assilines, Discocyclines, etc.); (5) the upper Malvariche Fm (Bartonian) made of cream calcarenites with large size larger benthic foraminifera (Nummulites, Assilines, Discocyclines, etc.); (6) the Bosque Fm (Chattian) made cream marls in the area; and (7) the Río Pliego Fm (Aquitanian in the area) made of pinkish marls and pelites with turbidite beds (sandstones and conglomerates) intercalations.

2. Methodology and overview of the software

Recently, many works using Python have been published in different fields of geology including: groundwater (Matott et al. 2011; Memari and Clement, 2021), seismic studies (Bueno et al., 2020; Cavalcanti-Bezerra-Guedes et al., 2022), volcanic hazard (Tonini et al., 2015), paleontological studies (Casanova-Arenillas et al., 2020), boreholes analysis (Fuentes et al., 2020), mineralogical studies (Scharf et al., 2022), geophysical exploration (Krieger and Peacock, 2014), landscape analysis and engineering projects

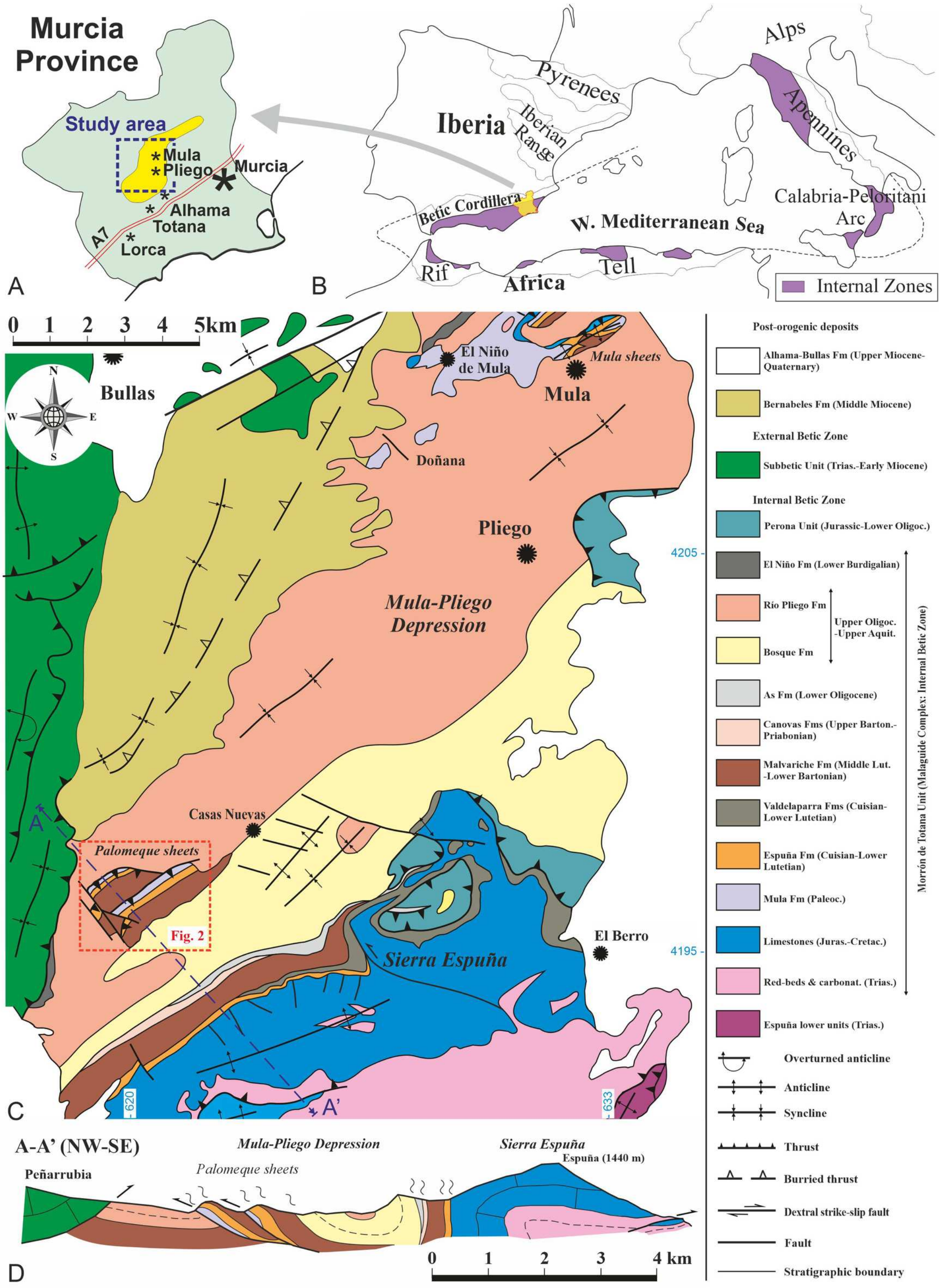

Figure 1. Maps of the study area (Mula-Pliego Depression and Sierra Espuña). (A) Geographic map of the Murcia province (south-eastern Spain) with indication of the study area represented in Figure 1C; (B) Geographic location in the framework of the western Mediterranean alpine chains.; (C) Geological map of the study area (Mula-Pliego Depression and Sierra Espuña) with the position of the Palomeque sheets represented in Figure 2; (D) Geological cross-section of the study area (Mula-Pliego Depression and Sierra Espuña) located in Figure 1C.

(Adamczyk and Tiede, 2017; Rivillas-Ospina et al., 2022), among others. Most of these studies seek automatizations and statistical analyses from a large amount of data obtained from different sources as boreholes (Casanova-Arenillas et al., 2020; Fuentes et al., 2020), geophysical (Krieger and Peacock, 2014; Tonini et al., 2015; Bueno et al., 2020; Cavalcanti-Bezerra-Guedes et al., 2022) or direct measures of physical properties of elements (Matott et al. 2011; Adamczyk and Tiede, 2017; Memari and Clement, 2021; Rivillas-Ospina et al., 2022). In this work, we have used Python to build 3D visualizations of

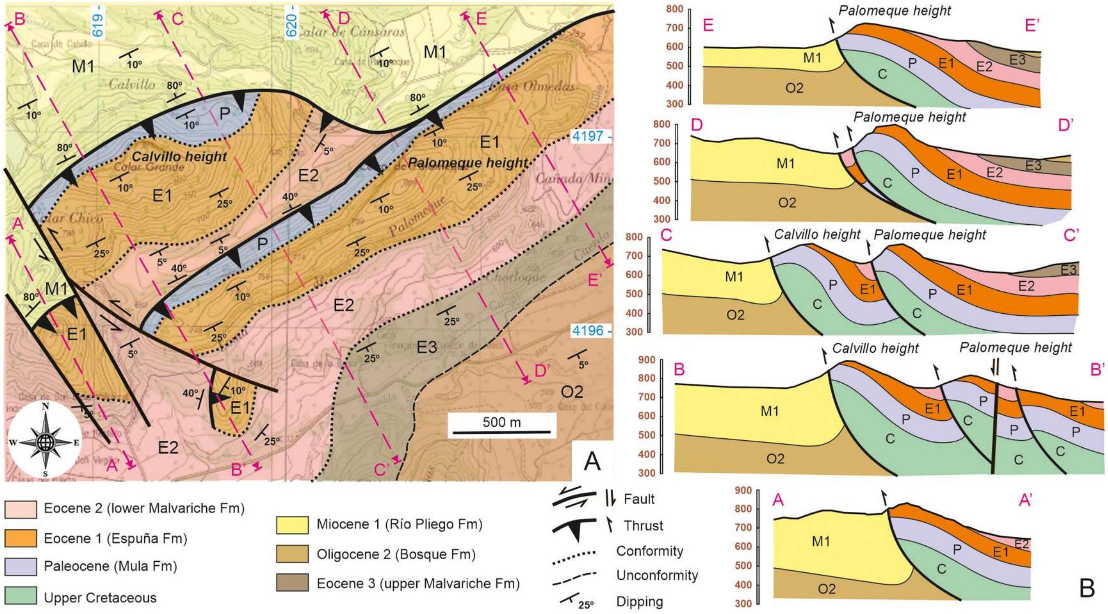

Figure 2. Geological map and vertical cross-sections from the Palomeque thrust-fold structures (based on maps and sections from Martín-Martín, 1996; Martín-Martín & Martín-Algarra, 2002; Tent-Manclús et al, 2001; Martín-Martín et al., 2006a; Martín-Martín et al., 2006b; Martin-Rojas et al., 2007; Perri et al., 2013, 2017; Martín-Martín & Robles-Marín, 2020; Moliner-Aznar et al., 2021). (A) geological map of the Palomeque structures with the position of vertical sections; (B) a set of five serial vertical geological sections (A-A′, B-B′, C-C′, D-D′ and E-E′) oriented in the normal position to the main axis of the structures.

geological maps and models and the entire creation process has been compiled in four Jupyter notebooks that are located in the GitHub repository (https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain). In this repository, users can also find the necessary instructions to execute the code, as well as the necessary python modules that we have used. The HTML files with the models are also hosted in the repository, but they are larger than what GitHub supports, so they need to be downloaded and opened with a browser locally. In the repository, there is a readme file where the necessary instructions to run the code are specified. During the process of creation of the models, the supervision of a geologist is fundamental in order to be faithful to the geological reality. Apart from Python, we have used other tools listed in Table 1.

We also list the main Python modules and libraries that we have used in Table 1. The proposed methodology mainly uses the Python graphic libraries Plotly and Matplotlib (https://plotly.com/python/ and https://pypi.org/project/matplotlib/). These are very extended comprehensive libraries with many users, and even with some active forums where it is possible to obtain help for any technical question (https://community.plotly.com/). The python modules we use are very standard and easy to download. Many of them come built into standard Python installations. In other cases, the user can install these applications using pip (https://docs.python.org/3/installing/index.html) or conda (https://docs.conda.io/projects/conda/en/latest/user-guide/concepts/installing-with-conda.html). A few custom functions were used to plot data and geometric objects (Table 1).

It would be worthwhile to make the process we present automatic so that one only has to enter the data to get the model, but, at this moment, we do not have the necessary resources to create a python library with the entire process automated. On the contrary, we present a system of representing and viewing geological structures where the necessary data to perform the models (maps, vertical sections, and stratigraphic logs) are obtained in a traditional way in the field by geological researchers (paper map based). We start from a geological map and several cross-sections like those appearing in Figure 2 (Martín-Martín, 1996), but we need to digitize them and have all the geological data computerized. Using the map and the sections from Figure 2 we will construct png figures (in a plane) and html figures (in the space).

To digitalize the map in Figure 2 we can follow a standard procedure: first scan the map, then georeference the raster image and digitize the lines with a GIS program like QGIS and finally export the data to cvs files. Nevertheless, the basic python modules UTM and Copernicus can be used to do the same operation directly without using any commercial or open-source GIS programs. Those programs usually make it difficult to extract data from the software (they try to keep people working within their environment). We decided to use Google Earth Pro (https://maps.google.com/intl/es/earth/download/gep/agree.html) to create the tracks that represent the geological map in Figure 2A. Therefore, the geological map is scanned

Table 1. General tools, Python modules and libraries, and Python custom functions used.

General tools	
Google Earth pro	Free software that allows visualize and create geospatial data.
GPSVisualizer	Free online utility that can be used to handle geographic data.
Anaconda	Open source platform very useful for beginner to install and handle Python environments.
Jupyter	Web application for creating and sharing computational documents
Python modules and librarieswe	
Pandas	Python package for data analysis
NumPy	Python library for working with numerical data
SymPy	Python library for symbolic mathematic
Plotly	Python library for data visualization
Matplotlib	Python graphic library
Pillow	Python library for image processing
Copernicus and UTM	Python libraries for geo-spatial data
Bezier.py	Python module to create Bezier curves
Custom functions	
surface	Function that interpolate from a grid
contact_tr	Function that prepare data from a cvs files to be used for plotting
contact_dat	Function that uses matplotlib function Scatter3d to obtain the plot data that allows to visualize marks or lines
eq and tr	Two functions used to compute triangles that are uses as geological symbols
falls	Function that prepare data to visualize faults
plane_3d	Function that uses matplotlib command Mesh3d and prepare the data to visualize a plane
plane, plane_y, impl_line, para_line, inter_line_plane	Functions that get different mathematical equations of lines, planes and its intersections.
bz, bz2, bz3	Functions that uses Bezier functions to compute Bezier curves from a point, a direction and lists of intervals and heights
fil_pol, fil_pol1,fil_pol2, fil_pol3	Functions that uses the points_in poly function of skimage.measure to select points inside a polygon
grid, grid,grid2	Functions to create different grids

and imported into Google Earth Pro so we can draw directly on the image to create the tracks. The selected area shows clear geological contacts, clearly visible in the orthoimage since they generate changes in the relief and vegetation. Moreover, it must be remembered that the geologist on this team has worked in the area for more than 30 years and has a great knowledge of its geology. Once the tracks are created, they are exported as kml files and the online resource gpsvisualizer (https://www.gpsvisualizer.com/) is used to convert kml files to cvs files.

Once we have all the geological data as cvs files, we use the Pandas Python module to import them as data frames, then we use the UTM Python module to convert geographic coordinates to UTM coordinates and the Copernico module to extract elevations from the DEM file of the area eu_dem_v11_E30N10. We download this DEM from the Copernicus DEM – Global and European Digital Elevation Model (https://spacedata.copernicus.eu/es/collections/copernicus-digital-elevation-model). The complete process is given in the Jupyter notebook Copernicus.ipynb which is hosted in the Github repository (https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain). At the end of the process, we will have several cvs files with UTM coordinates and elevations. The corresponding points will be used to represent the geological elements of the area on the geological map (map.png also hosted in the Github repository) in Figure 3A. In this figure all the geological elements are included (stratigraphic and structural contacts, and structural elements as thrusts and faults) and also the straight lines representing the position of the geological cross-sections (traces).

On the other hand, for each vertical section in Figure 2B, there are lines (curves) that represent stratigraphic or tectonic contacts that delimit the different stratigraphic units and that we want to represent mathematically. From the possible kinds of curves (periodic, polynomial, Bezier, etc.), we have chosen Bezier curves since these are the ones that best adapt to the geological reality. Therefore, for each contact of each vertical section, we have to find the mathematical expression of the Bezier curve that represents it. To do this using Python's Bezier module, we need to give a list of control points (Figure 4). Our objective then will be to choose the control points appropriately so that the Bezier curve that they define, represents the contacts as closely as possible. We have developed a strategy that will allow us to choose these points appropriately, so that uniformity is obtained in the way of choosing control points for different contacts. This strategy is carried out in the definition of a custom function that we call bz that depends on certain parameters that determine the choice of the control points and thus the Bezier curve (see the Geological cross-sections.ipynb Jupyter notebook in the Github repository where we explain the process). The result is a png figure for each cross-section like that of Figure 5A representing the A-A' section. Note that in the figures that represent the cross-sections, lines appear that represent the real topology. To obtain these lines we simply select points on a horizontal line drawn on the plane that represents the section and using copernicus module we calculate the elevations of these points. Then we represent the points that we had selected on the horizontal line but with the real height. The methodology employed is synthetized in the flow chart from Figure 6.

To obtain the colored figures both on the geological map and in the cross-sections we need to define polygons that enclose each stratigraphic unit. These polygons are determined by their vertices, which will be chosen from the points of a track, a Bezier curve or the cutting line with the topography, and the choice is laborious. The next step is the filling of polygons with colors representing the different stratigraphic units. We face this task in two different contexts: in

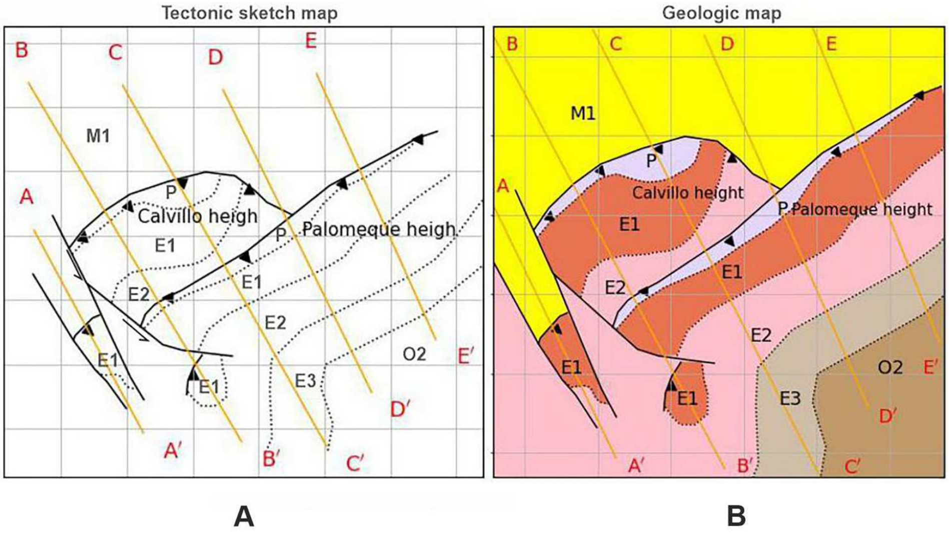

Figure 3. Methodology of the geological maps performed in black and white (tectonic sketch map) and with colors (geological map).

a plane (for png figures) or in space (for HTML figures). The matplotlib Python package has a specific function to fill polygons in a plane. This function is fill_between so that if we have a well-defined polygon in a plane we can fill the interior simply using this function. It has been used to construct the colored map and cross-sections for png figures (Figures 3B and 5B). It can be seen how this function is used in the notebook Geological cross-sections.ipynb of the Github repository.

Filling the interior of a polygon in space is somewhat less straightforward. The method we have used for this is the following (Figure 6): first, we use the Numpy Pythom package (https://numpy.org) to define a grid of points that covers the area of the polygon. Then, we use the Scipy Pythom package (https://scipy.org) to select the points of the grid that are inside the polygon. Finally, we color the points inside (if the grid density is sufficient and the size of the points is adequate, it will seem like a continuously color). We have defined the custom function fil_pol to perform this task. This technique has been used to construct the HTML figures (Figure 7) and it is explained in the Model_color.ipynb and Bolck.ipynb notebooks in the repository (https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain).

The final part of the procedure is adding to the HTML models the structural symbols as well as the labels with the toponymy and the stratigraphic units. This is also performed with pythom packges. These symbols and labels are integral part of the models, so that if the viewing perspective is changed these symbols change position or rotate. To perform the triangles linked to the thrusts, a few mathematical computations are required and we defined ad hoc customs functions for this task. Texts indicating places

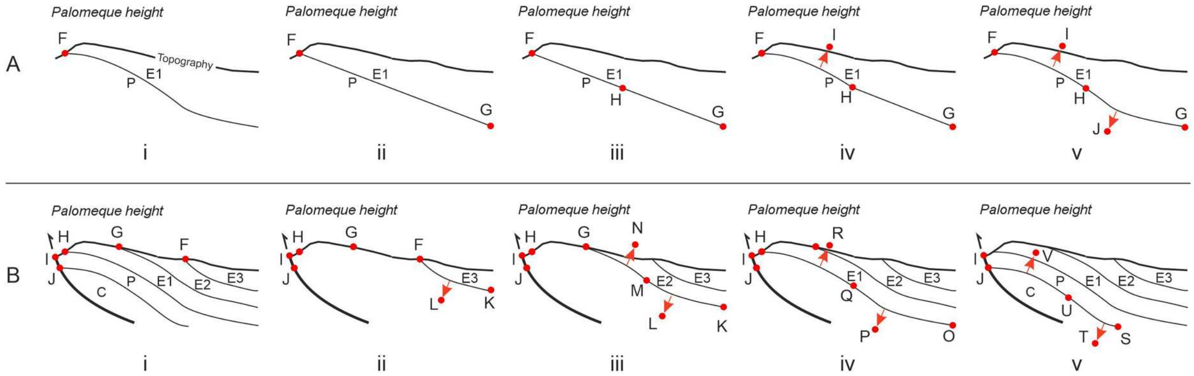

Figure 4. Bézier curves by Python. (A) a single contact: (i) original curve as example in Figure 2B with a curve from Figure 2B (Mula Fm-Espuña Fm contact and topography from vertical section E-E′), (ii) example of Bézier curve made using two control points [F, G], (iii) example of Bézier curve using three control points [F, G, H], (iv) example of Bézier curve using four control points [F, G, H, I], (v) example of Bézier curve using five control points [F, G, H, I, J]; (B) example of a folded succession (vertical section E-E′ from Figure 2B): (i) full section E-E′, (ii) example of determining the lower-upper Malvariche Fm stratigraphic contact with the control points [F, L, K], (iii) example of determining the Espuña Fm-lower Malvariche Fm stratigraphic contact with the control points [G, N, M, L, K], (iv) example of determining the Mula Fm-Espuña Fm stratigraphic contact with the control points [H, R, Q, P, O], (v) example of determining the Capas Blancas-Mula Fm stratigraphic contact with the control points [J, V, U, T, S].

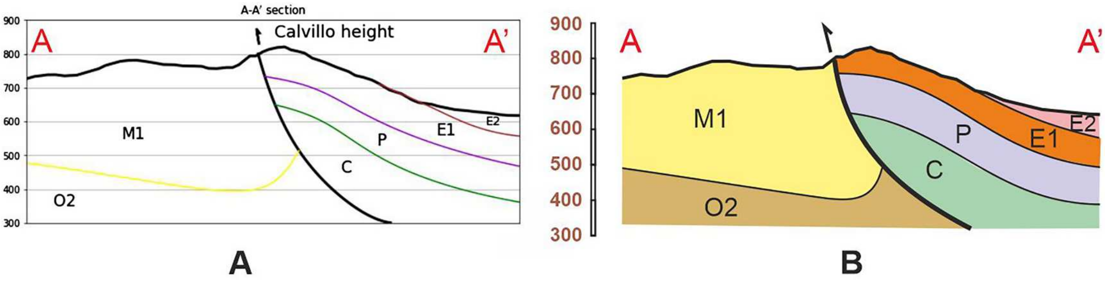

Figure 5. Methodology of performing the geological cross-sections with real topography and colors.

and stratigraphic units, and arrows for the strike-slip faults were also inserted in precise locations so that the geological maps acquire the 3D visualization with all the geological elements and the real topography.

3. Results

3.1. Geological map and vertical sections in png files

With the above-mentioned procedure, a geological map andfive vertical serial cross-sections (A-A', B-B', C-C', D-D' and E-E') were developed and they will be used to construct the geologic model (Figure 8). In these vertical sections, the defined stratigraphic formations were represented by conserving the proportions (same horizontal-vertical scale) and dippings.

The geological map and each vertical section were saved as a png file and all of them are hosted in the directory 'figures' of the GitHub repository (https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain). This mapincludes the location of the cross-sections.

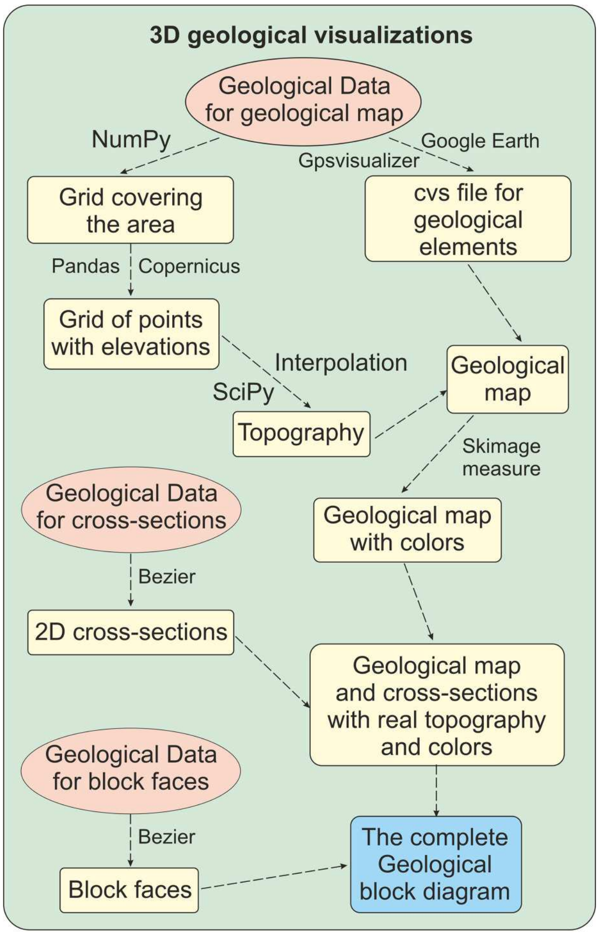
3D geological visualizations
Geological Data for geological map
NumPy
Google Earth
Gpsvisualizer
Grid covering the area
cvs file for geological elements
Pandas
Copernicus
Grid of points with elevations
Interpolation
SciPy
Geological map
Topography
Skimage measure
Geological Data for cross-sections
Geological map with colors
Bezier
2D cross-sections
Geological map and cross-sections with real topography and colors
Geological Data for block faces
Bezier
Block faces
The complete Geological block diagram

Figure 6. Diagram chart showing the methodological steps for the performing of the map, cross-sections, and geological block diagram.

3.2. Geological map with vertical sections and real topography

The data computation work carried out on the geological maps given in Figure 2 is essential for the final objective of this work. We have to find a procedure to integrate all the data that make up the vertical cross-sections into space, so that we obtain HTML figures with all the plane figures obtained in the previous section integrated in space. An essential object of these HTML figures will be the topographic surface of the area. To create this surface, we define again a grid that covers the area and use the Copernico package to extract the elevations from the DEM. We will thus have a mesh of points in the space that covers the area and then we use linear interpolation to obtain a surface that represents the real topology of the study area.

The Python packages Numpy (https://numpy.org), Simpy (https://www.sympy.org/en/index.html), and Scipy (https://scipy.org) and the plot command Surface (see Bullejos & Martín-Martín, 2023 and also the notebook Geological cross_section.ipynb in Github the repository for details). Once we have a model with real topography, we proceed to add the rest of the geological elements (thrusts, faults, and stratigraphic contacts) to the topographic model. We already have computed lists of points (with UTM_X and UTM_Y coordinates) that represent those geological elements so we only have to plot those points into the model. We also want to add geological symbols to the model such as the triangles linked to the thrusts, the arrows for the strike-slip faults, as well as the texts indicating places and stratigraphic units

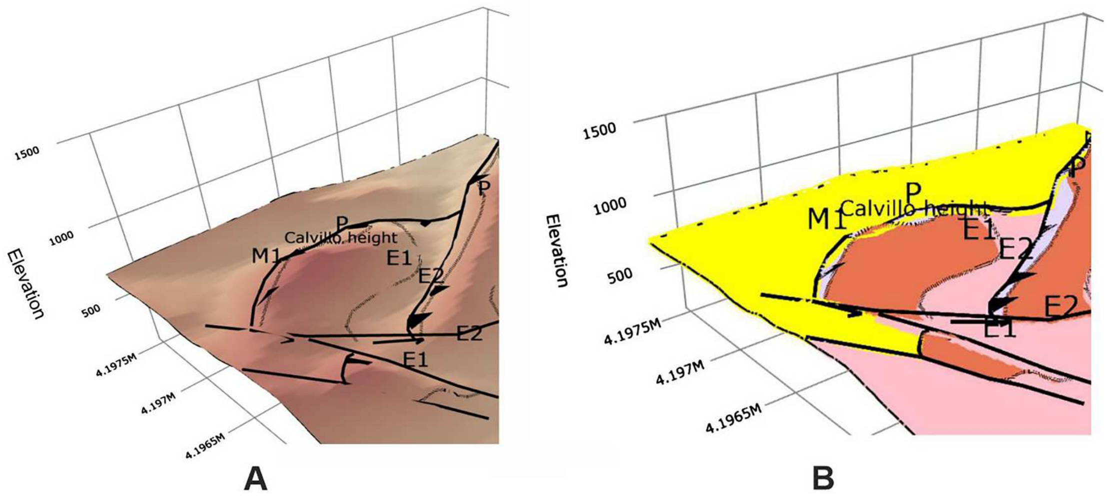

Figure 7. Methodology of performing the tectonic sketch map (A) and the geological map with colors (B), both with real topography.

and we want these symbols to be integrated into the model, so that if the viewing perspective is changed these symbols rotate and change position. We will define a few mathematical functions of our own to do this. Finally, we proceed to integrate the cross sections into the model. Again the computational work is done, since each cross-section is integrated into a plane with its corresponding spatial coordinates and each curve in each cross-section (whether corresponding to the topography or the stratigraphic-

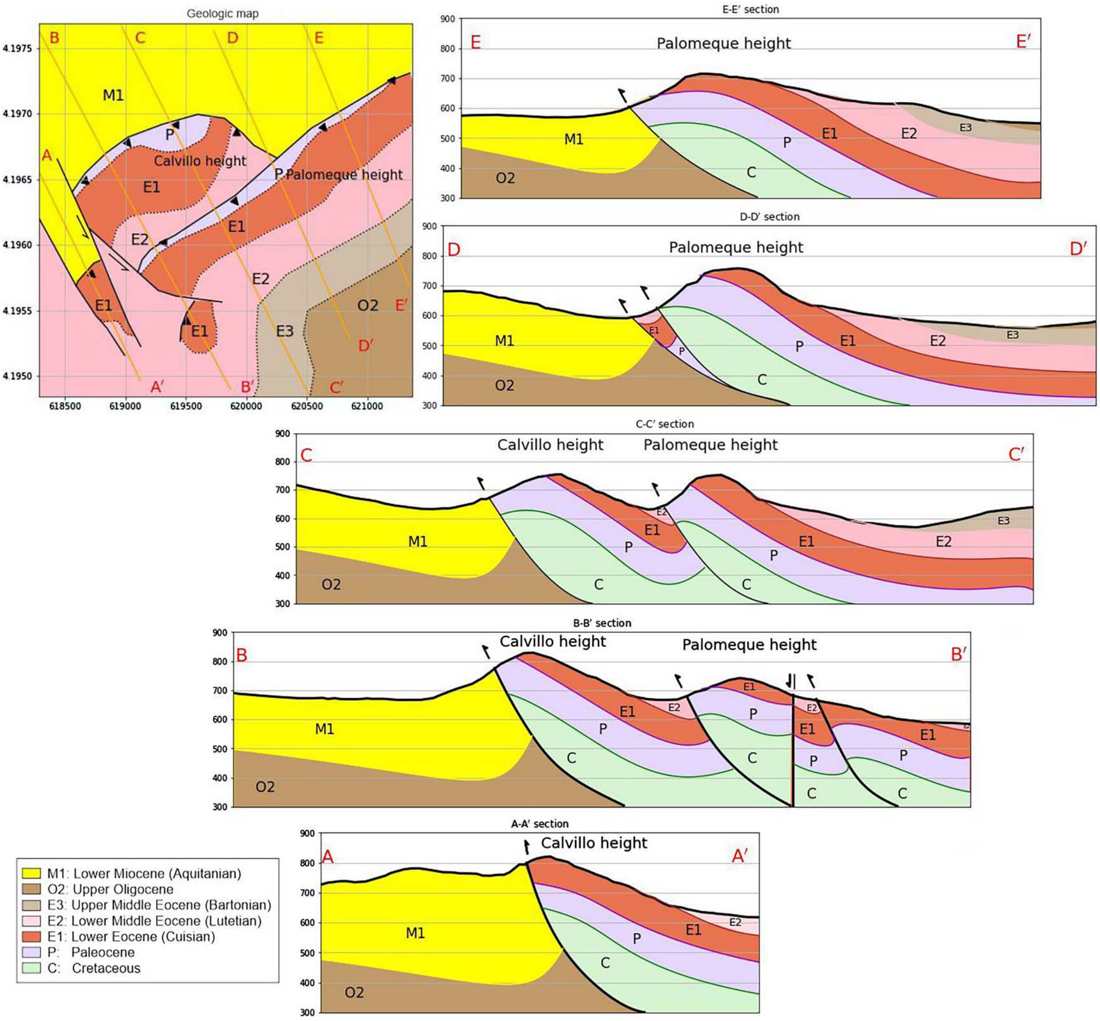

Figure 8. 2D Geological map and vertical cross-sections performed using Python. The serial vertical sections are signaled in the index map of the left-upper corner: A-A′, B-B′, C-C′, D-D′ and E-E′. The defined stratigraphic formations are represented in all cases.

tectonic contacts), and we know their equations. All we have to do is place these data in space and plot then to the model. The complete process is given in the notebook Geological cross_section.ipynb of the Github repository. Two HTML versions of this model were created: one with a frame of topography and a second with colors representing the stratigraphic units. In the first case, the geological map exposed in Figure 2A was performed as an interactive HTML map (Palom_map.html available as supplementary material). Four views from this model are reproduced in Figure 9. This figure illustrates the thrust-fold structure affected by two strike-slip faults to the west (all with real topography) using the defined stratigraphic formations but with a progressive frame representing the changes in topography.

In the second case, the geological map and the cross-sections were integrated into a HTML interactive model (Palom_map_sections.html available as supplementary material) but now with colors according to the stratigraphic units. Five views of this model are exposed in Figure 10.

3.3. Geological block diagram with the stratigraphic and structural elements

The next objective was to build a geological block diagram with the essential geological elements. For this, we do something analogous to what we have done to build the cross-sections. The complete process of constructing the block model is explained in the jupyter notebook Bolck.ipynb of the Github repository. Note that, as we have done when calculating the Bezier curves in the previous cross-sections, the first and last control points in each of the lists that determine the Bezier curves for geologic contacts on the block faces are mathematically calculated as intersections of lines. The criteria of the geologist will be the one that determines the degree of complexity of those curves and will be essential for choosing the rest of the control points for each curve. Finally, as in the calculation of the cross-sections, we will use the lists of points that determine the Bezier curves in each one of the faces of the block to define polygons that enclose the different materials and we will color the interiors of these polygons.

The geological map area was therefore completed in spaceby adding four side faces and the surface at the bottom to complete an HTML three-dimensional geological block diagram. The technique to complete these faces is the same used to represent the cross-sections. Six views of this model are exposed in Figure 11. The file Palom_block.html (available as supplementary material) is the output for the 3D geological block diagram. This representation allows a 3D visualization which consists of a pseudo-3D or 2+1/2D model composed of several 2D images (upper map view, the bottom face, and the lateral faces).

4. Discussion and final remarks

The 2D (Figures 9 and 10) and pseudo-3D (Figure 11) models presented before and performed by Python derived from the HTML files (Palom_map.html and Palom_map_sections.html available as supplementary material). These HTML files provide 3D visualizations which consist of pseudo-3D or 2+1/2D models composed of several 2D images. These models faithfully reproduce the geological reality of the maps published in the last decades in the area (Figure 2).

The first published maps in the area belong to the IGME (Geological and Mining Institute of Spain) Plan MAGNA (National Geological Map) in the seventies of the XX century. In concrete, the Palomeque area belongs to the Murcia (number 79) geological map (First Serie: scale 1:200.000) made by Fontboté (1970) and the Coy (number 932) geological map (Second Serie: scale 1:50.000) made by Kampschuur et al. (1972). Both series are fully free to access on the IGME website (https://info.igme.es/cartografiadigital/geologica/Default.aspx?language=en). IGME has developed in recent times (in the XXI century) georeferenced versions of both series of maps in Spain and are available in ArcGis files. The georeferenced 1:200.000 maps appear by compilation of the 1:50.000 maps and are available for a few areas of Spain. In the case of the Murcia map, this version is not still performed. The Third Serie of geological maps are 1:50.000 georeferenced maps are also only available in a few areas of the Spanish territory but it is neither produced the Coy map. The most complete and detailed geological map in the area belongs to the Ph. Thesis from Martín-Martín (1996) covering the Sierra Espuña and the Mula-Pliego depression at 1:25.000 scale. This map was created with Autocad but it was not georeferenced. Later, several partial geological maps from the area have been published in different research articles (Martín-Martín & Martín-Algarra, 2002; Martín-Martín et al., 2006a, 2006b; Martín-Martín & Robles-Marín, 2020, 2021; Martin-Rojas et al., 2007; Moliner-Aznar et al., 2021; Perri et al., 2017) performed with Corel Draw and also without georeferentiation. In the area are also available free-access orthoimages from Google Earth pro (https://earth.google.com/intl/earth/) and Google maps (https://www.google.com/maps/) but also from other national or regional sites as the National Geographic Institute (https://www.ign.es/web/seccion-ortoimagenes) and GeaMap (https://www.geamap.com/es/murcia).

The proposed models allow interactive views of the geological structures and the stratigraphic units with real topography. Moreover, the vertical geological cross-sections trustworthy reproduce the location of

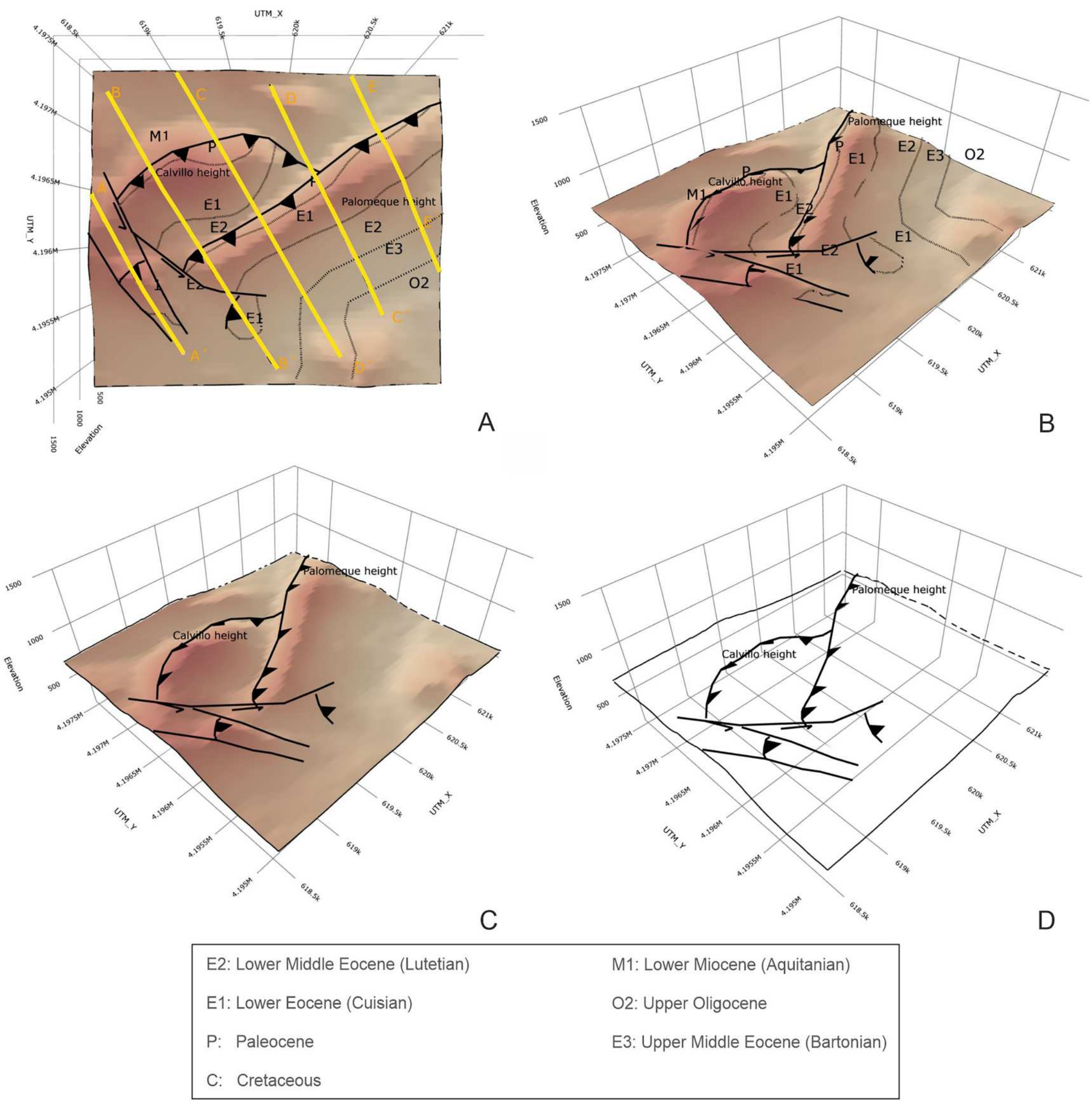
UTM_X
UTM_Y
Elevation
M1
P
Calvillo height
Palomeque height
E1
E2
E3
O2
A
B
C
D
E2: Lower Middle Eocene (Lutetian)
M1: Lower Miocene (Aquitanian)
E1: Lower Eocene (Cuisian)
O2: Upper Oligocene
P: Paleocene
E3: Upper Middle Eocene (Bartonian)
C: Cretaceous

Figure 9. Different views of the interactive HTML geological map with real topography were performed using Python. The HTML files are available as Supplementary Materials (Palom_map.html). The defined stratigraphic formations, and the structural and stratigraphic contacts are shown. (A) upper view; (B) intermediate upper-lateral view from the SW only with the contacts and texts turned on; (C) intermediate upper-lateral view from the SW only with location texts and tectonic contacts turned on; (D) intermediate upper-lateral view from the SW only with location texts and tectonic contacts turned on.

the structural and stratigraphic contacts with respect the real topography and other geological surfaces since these are mathematically calculated. Finally, the HTML geological block diagram (Figure 11: Palom_block.html available as supplementary material) whose lateral sides also represent geological cross sections, allows visualizing the structures and stratigraphic framework of the study area. All HTML models can be run with a simple internet browser (not needing additional software). It is allowed to see any views, zoom in and out on detail and export to jpg any part of the model at user choice. Concluding, all the exposed formerly indicates that Python tools (Numpy, Matplotlib and Sympy, among others) are easy to run tools and ergonomic applications for mapping and modeling of geological structures in a deformed area such the Palomeque key-case. The proposed models are interactive and allow measuring proportions or properties at choice of users while representing, what is very interesting for applied geology such geotechnical, and mining (and fluid reservoirs) research. There is a lot of commercial software available for 3D visualizing and modelling. Contrarily, free software is much less available. In the literature, until now, papers aimed to modeling of geological structures without geological information coming from drilling or geophysics studies (only with classical mapping geological information) are very scarces. All presented here indicates that Python tools and packages could be very useful in classical geological modeling. Several Jupyter notebooks were generated (Copernicus.ipynb, Geological cross-sections.ipynb, Model_color.ipynb, and Bolck.ipynb) and uploaded at https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain for interested future users. In this repository, the

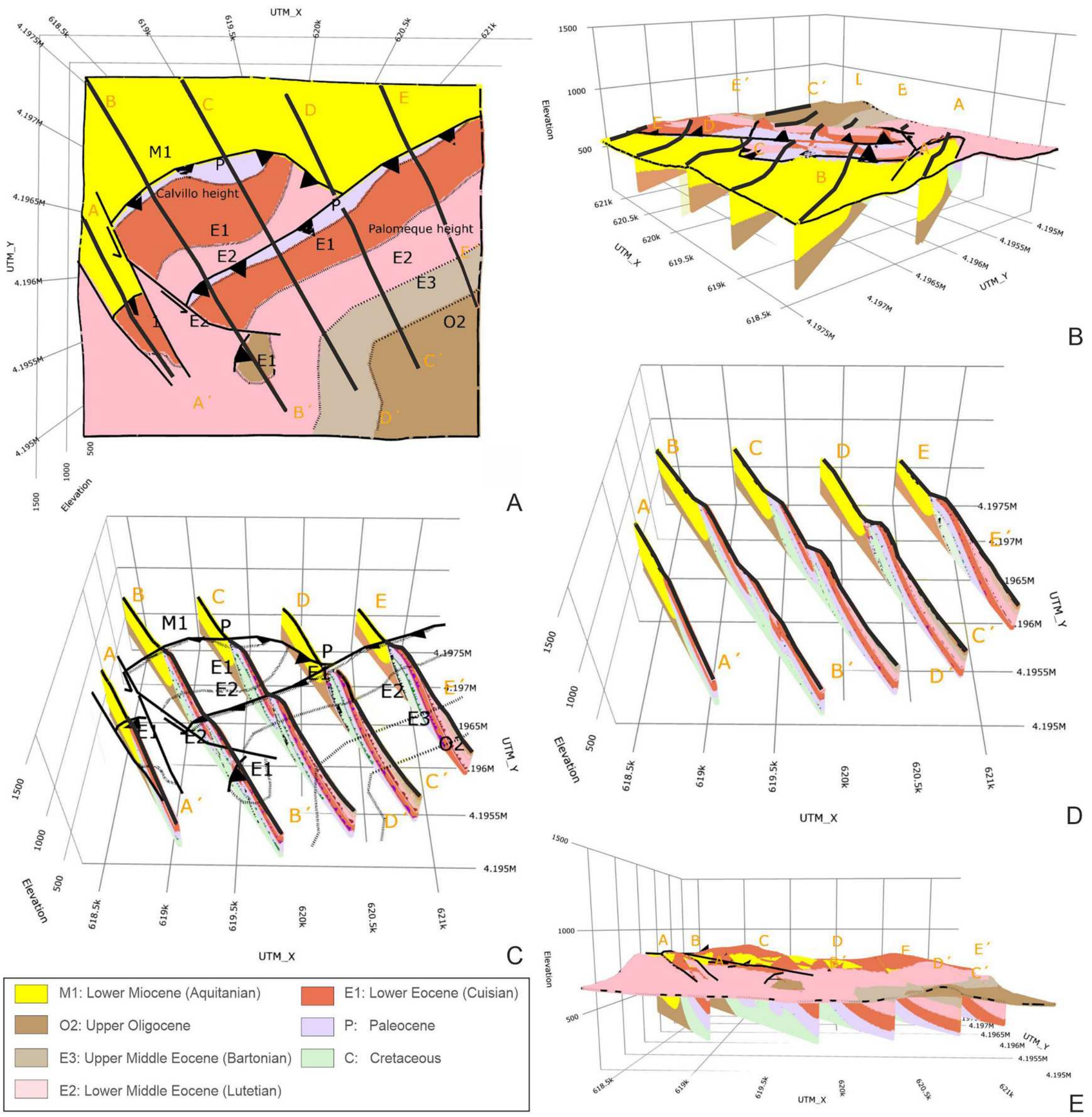
A
B
C
D
E
UTM_X
UTM_Y
Elevation
M1
P
E1
E2
E3
O2
Calvillo height
Palomeque height
M1: Lower Miocene (Aquitanian)
O2: Upper Oligocene
E3: Upper Middle Eocene (Bartonian)
E2: Lower Middle Eocene (Lutetian)
E1: Lower Eocene (Cuisian)
P: Paleocene
C: Cretaceous

Figure 10. Different views of the HTML geological map and vertical sections with colors and real topography (Palom_map_sections.html). (A) upper view with all elements turned on; (B) intermediate upper-lateral view from the NW with all elements turned on; (C) intermediate upper-lateral view from the NW only with tectonic and stratigraphic contacts and vertical sections turned on; (D) intermediate upper-lateral view from the NW only with geological vertical section turned on; (E) lateral view from the S with all elements turned on.

geological data that we have used and also a readme file with the necessary instructions to run the notebooks can be found. In the Copernicus.ipynb notebook we explain how to treat kml files to get UTM coordinates and elevations. We also show how to define a grid that covers the area, with points in UTM coordinates, and then how to get latitude and longitude coordinates and elevations. In the Geological_cross-sections.jpynb notebook, we explain how to calculate the intersections of the geological elements with the planes that define the geological cross-sections. Then we give a strategy to define the control points of the Bezier curves that represent the geological contacts in each cross section. Finally, we use the curves associated with the geological contacts to define polygons that enclose the different materials that are located in each cross-section and thus we can color the different zones to obtain the computerized versions. In the Model_color.ipynb notebook we used the calculations done in Geological_cross-sections.jpynb to integrate the cross-sections in a figure which model the geological elements in the study area. Finally, in the Block.jpynb notebook we follow the given methodology to build the cross-sections to obtain new sections following South-North and West-East directions that will represent the faces of a 300-meter-deep block of the area. We also covered this block at the bottom to make a compact block that can be viewed from any perspective. Although data processing in this work is burdensome, we have achieved a geological map and 3D visualizations which consist of pseudo-3D or 2+1/2D models composed of several 2D images in a standard HTML format. This format does not require complex tools to be visualized and that is obtained in the absence of large databases covering the area. These

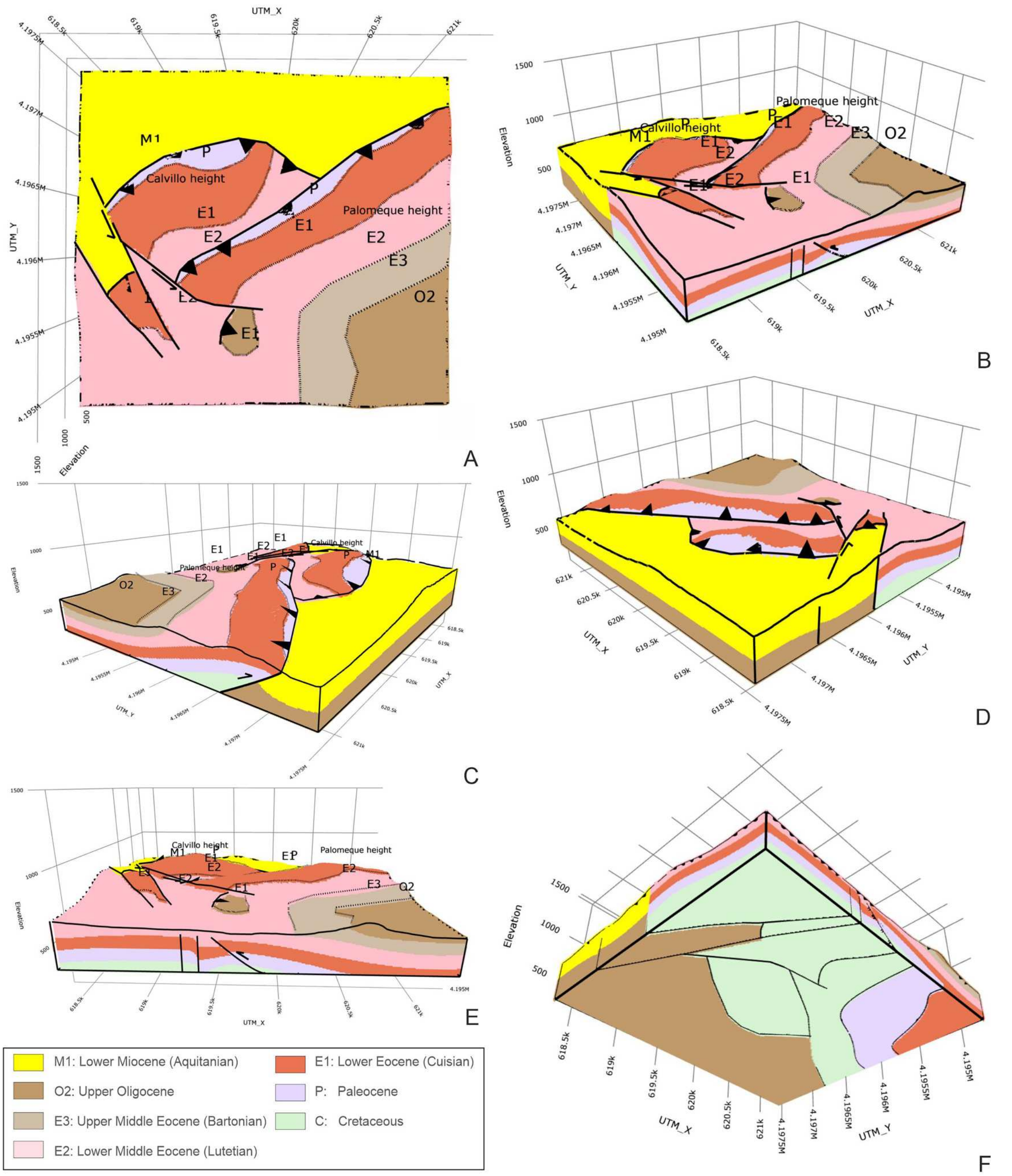
A
B
C
D
E
F
UTM_X
UTM_Y
Elevation
Calvillo height
Palomeque height
M1
P
E1
E2
E3
O2
M1: Lower Miocene (Aquitanian)
O2: Upper Oligocene
E3: Upper Middle Eocene (Bartonian)
E2: Lower Middle Eocene (Lutetian)
E1: Lower Eocene (Cuisian)
P: Paleocene
C: Cretaceous

Figure 11. Different views of the HTML geological block diagram with colors and real topography (Palom_block.html) with all the elements turned on. (A) upper view; (B) intermediate upper-lateral view from the NW; (C) intermediate upper-lateral view from the SW; (D) intermediate upper-lateral view from the SW; (E) intermediate upper-lateral view from the S; (F) lateral view from the S.

models have a real topography and the key points that allow the representation have been mathematically calculated, which makes the model more reflective of reality.

Software

The models have been obtained using Python programming language. The Jupyter Notebook web application is the interactive development environment in which we write and run Python code. The main Python packages we have used are: Plotly and Matplotlib as graphic libraries, Pandas for data analysis, and Numpy and Simpy as mathematics libraries. Apart from Python utilities, we use Google Earth pro and GPSVisualizer. The html supplementary materials can be opened with any web browser. The texts and the figures in the manuscript have been performed respectively with Microsoft Word and Corel Draw.

Code availability

Name of the code: A-3D-model-of-a-geological-structure-in-SE-Spain.

Contact: bullejos@ugr.es, +34 958 243375.

Year first available: 2023.

Hardware requirements: Any computer that can run Python 3.7.

Program language: Python.

Software required: Python 3.7 distribution (Anaconda distribution is recommended). Jupyter is the recommended IDE to use the notebooks.

Program size: 864 KB for the Python scripts and Jupyter notebooks and 6.5 MB for the data file.

The source codes are available for downloading at the link: https://github.com/bullejos/A-model-of-a-geological-structure-in-SE-Spain.

Acknowledgments

The authors would like to acknowledge the Spanish Ministry of Science and Innovation Research Project PID2020-114381GB-100, semigrupos.ugr.es from University of Granada and Research Groups FQM-343 of the Junta de Andalucía, and Research Groups and Projects of the Generalitat Valenciana-University of Alicante, for financial support. The work done by the two reviewers is greatly appreciated.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by Ministerio de Ciencia e Innovación: [Grant Number PID2020-114381GB-100].

Authorship contribution

Manuel Bullejos: creation of HTML models using Python programing language. Manuel Martín-Martín: geological information and figures in this article. Both authors: redaction of the manuscript.

ORCID

Manuel Bullejos http://orcid.org/0000-0001-6914-8052
Manuel Martín-Martín http://orcid.org/0000-0002-5797-9892

References

Adamczyk, J., & Tiede, D. (2017). ZonalMetrics - a Python toolbox for zonal landscape structure analysis. *Computers & Geosciences 99*, 91–99.

Belayouni, H., Guerrera, F., Martin-Martin, M., Le Breton, E., & Tramontana, M. (2023). The Numidian formation and its lateral successions (Central-Western Mediterranean): A review. *International Geology Review*, 1–33. https://doi.org/10.1080/00206814.2023.2199429

Bueno, A., Zuccarello, L., Díaz-Moreno, A., Woollam, J., Titos, M., Benítez, C., Álvarez, I., Prudencio, J., & De Angelis, S. (2020). PICOSS: Python Interface for the Classification of Seismic Signals. *Computers & Geosciences 142*, 104531.

Bullejos, M., Cabezas, D., Martín-Martín, M., & Alcalá, F. J. (2022a). A Python application for visualizing the 3D stratigraphic architecture of the onshore Llobregat River Delta in NE Spain. *Water*, *14*(12), 1882. https://www.mdpi.com/2073-4441/14/12/1882

Bullejos, M., Cabezas, D., Martín-Martín, M., & Alcalá, F. J. (2022b). A k-nearest neighbors algorithm in Python for visualizing the 3D stratigraphic architecture of the Llobregat River Delta in NE Spain. *Journal of Marine Science and Engineering*, *10*(7), 986. https://doi.org/10.3390/jmse10070986

Bullejos, M., Cabezas, D., Martín-Martín, M., & Alcalá, F. J. (2023). Confidence of a k-nearest neighbors Python algorithm for the 3D visualization of sedimentary porous media. *Journal of Marine Science and Engineering*, *11*(1), 60. https://doi.org/10.3390/jmse11010060

Bullejos, M., & Martín-Martín, M. (2023). A Python Application for Visualizing an Imbricate Thrust System: Palomeque Duplex (SE, Spain). *Geosciences*, *13*, 207.

Casanova-Arenillas, S., Rodríguez-Tovar, F. J., & Martínez-Ruiz, F. (2020). Applied ichnology in sedimentary geology: Python scripts as a method toautomatize ichnofabric analysis in marine core images. *Computers & Geosciences 136*, 104407.

Cavalcanti Bezerra Guedes, V. J., Ramalho Maciel, S. T., & Rocha, M. P. (2022). Refrapy: A Python program for seismic refraction data analysis. *Computers & Geosciences 159*, 105020.

Evangelidis, K., Papadopoulos, T., Papatheodorou, K., Mastorokostas, P., & Hilas, C. (2018). 3D geospatial visualizations: Animation and motion effects on spatial objects. *Computers & Geosciences*, *111*, 200–212. https://doi.org/10.1016/j.cageo.2017.11.007

Fontboté, J. M. (1970). Hoja 79 de Murcia (Serie Primera: escala 1:200.000). Plan MAGNA, IGME.

Ford, J., Mathers, S., Royse, K., Aldiss, D., & Morgan, D. J. R. (2010). Geological 3D modelling: Scientific discovery and enhanced understanding of the subsurface, with examples from the UK. *Zeitschrift der Deutschen Gesellschaft für Geowissenschaften*, *161*(2), 205–218. https://doi.org/10.1127/1860-1804/2010/0161-0205

Fuentes, I., Padarian, J., Iwanaga, T., & Willem Vervoort, R. (2020). 3D lithological mapping of borehole descriptions using word embeddings. *Computers & Geosciences 141*, 104516.

Guerrera, F., Martín-Martín, M., & Tramontana, M. (2021). Evolutionary geological models of the central-western peri-Mediterranean chains: A review. *International Geology Review*, *63*(1), 65–86. https://doi.org/10.1080/00206814.2019.1706056

Hobona, G., James, P., & Fairbairn, D. (2006). Web-based visualization of 3D geospatial data using Java3D. *IEEE Computer Graphics and Applications*, *26*(4), 28–33. https://ieeexplore.ieee.org/document/1652923

Husillos, C. (2022). cesarhusrod/sarai_piezo_precip: Mejora de la documentación (v1.0.1). Zenodo. https://doi.org/10.5281/zenodo.7197288

Jabaloy-Sánchez, A., Martín-Algarra, A., Padrón-Navarta, J. A., Martín-Martín, M., Gómez-Pugnaire, M. T., Sánchez-Vizcaíno, V. L., & Garrido, C. J. (2019). Lithological successions of the internal zones and flysch trough units of the betic chain. In C. Quesada & J. T. Oliveira (Eds.), *The geology of Iberia: A geodynamic approach, regional geology reviews* (pp. 377–432). Springer Nature.

Jackson, I. (2005). Addressing the real needs of all the users of geological information: The opportunities, issues and problems. In S. Ostaficzuk (Ed.), *The current role of geological mapping in geosciences. Proceedings of the NATO advanced research workshop on innovative applications* *of GIS in geological cartography, Kazimierz Dolny, Poland, November 2003* (pp. 59–68). Springer.

Jessell, M. (2001). Three-dimensional geological modelling of potential-field data. *Computers & Geosciences*, *27*(4), 455–465. https://doi.org/10.1016/S0098-3004(00)00142-4

Kampschuur, W., Langenberg, C. W., Baena, J., Velando, F., García-Monzón, G., Paquet, J., & Rondeel, H. E. (1972). Hoja 932 de Coy (Serie Segunda: escala 1:50.000). Plan MAGNA, IGME.

Krieger, L., & Peacock, J. R. (2014). MTpy: A Python tool box for magnetotellurics. *Computers & Geosciences 72*, 167–175.

Martín-Martín, M. (1996). *El Terciario del Dominio Maláguide en Sierra Espuña (Cordillera Bética oriental, SE de España): estratigrafía y evolución paleogeográfica* [Doctoral thesis]. Universidad de Granada. (299 pp.).

Martín-Martín, M., Bullejos, M., Cabezas, D., & Alcalá, F. J. (2023). Using python libraries and k-nearest neighbors algorithms to delineate syn-sedimentary faults in sedimentary porous media. *Marine and Petroleum Geology*, *153*, 106283. https://doi.org/10.1016/j.marpetgeo.2023.106283

Martín-Martín, M., El mamoune, B., Martín-Algarra, A., Martín-Pérez, J. A., & Serra-Kiel, J. (1997). Timing of deformation in the malaguide complex of the Sierra Espuña (SE Spain). geodynamic evolution of the internal betic zone. *Geologie en Mijnbow*, *75*, 309–316.

Martín-Martín, M., Guerrera, F., Miclăuş, C., & Tramontana, M. (2020a). Similar oligo-miocene tectono-sedimentary evolution of the paratethyan branches represented by the moldavidian basin and maghrebian flysch basin. *Sedimentary Geology*, *396*, 105548. https://doi.org/10.1016/j.sedgeo.2019.105548

Martín-Martín, M., Guerrera, F., Tosquella, J., & Tramontana, M. (2020c). Paleocene-Lower eocene carbonate platforms of westernmost tethys. *Sedimentary Geology*, *404*, 105674. https://doi.org/10.1016/j.sedgeo.2020.105674

Martín-Martín, M., Guerrera, F., Tosquella, J., & Tramontana, M. (2021). Middle eocene carbonate platforms of the westernmost tethys. *Sedimentary Geology*, *415*, 105861. https://doi.org/10.1016/j.sedgeo.2021.105861

Martín-Martín, M., Guerrera, M., & Tramontana, M. (2020b). Geodynamic implications of the latest Chattian-Langhian central-western peri-Mediterranean volcano-sedimentary event: A review. *The Journal of Geology*, *128*(1), 29–43. https://doi.org/10.1086/706262

Martín-Martín, M., & Martín-Algarra, A. (2002). Thrust sequence and syntectonic sedimentation in a piggy-back basin: The oligo-aquitanian mula-pliego basin (internal betic zone, SE Spain). *Comptes Rendus Geoscience*, *334*(5), 363–370. https://doi.org/10.1016/S1631-0713(02)01757-1

Martín-Martín, M., Martín-Rojas, I., Caracuel, J. E., Estévez-Rubio, A., Martín-Algarra, A., & Sandoval, J. (2006a). Tectonic framework and extensional pattern of the malaguide complex from sierra espuña (internal betic zone) during jurassic-cretaceous: Implications for the westernmost tethys geodynamic evolution. *International Journal of Earth Sciences*, *95*, 815–826.

Martín-Martín, M., & Robles-Marín, P. (2020). Alternative methods for calculating compaction in sedimentary basins. *Marine and Petroleum Geology*, *113*, 104132. https://doi.org/10.1016/j.marpetgeo.2019.104132

Martín-Martín, M., Sanz de Galdeano, C., García-Tortosa, F. J., & Martín-Rojas, I. (2006b). Tectonic units from the Sierra Espuña - Mula area (SE Spain): implication on the triassic paleogeography and the geodynamic evolution for the Betic-Rif internal zone. *Geodinamica Acta*, *19*(1), 1–15. https://doi.org/10.3166/ga.19.1-15

Martín-Martín, M., Serra-Kiel, J., El Mamoune, B., Martín-Algarra, A., & Serrano, F. (1998). The Paleocene of the eastern malaguides (Betic Cordillera, Spain): stratigraphy and paleogeography. *Comptes Rendus Geoscience*, *326*, 35–41.

Martin-Rojas, I., Sanz de Galdeano, C., Martín-Martín, M., & García-Tortosa, F. J. (2007). Geometry and kinematics of an antiformal stack deduced from brittle structures. Example of the Internal Betic Zone in the Sierra Espuña (province of Murcia, Spain). *Comptes Rendus Geoscience*, *339*(7), 506–515. https://doi.org/10.1016/j.crte.2007.04.005

Matott L. Shawn, Kenny Leung, & Junyoung Sim. (2011). Application of MATLAB and Python optimizer stotwo-case studies involving groundwater flow and contaminant transport modeling. *Computers & Geosciences 37*, 1894–1899.

Memari S. Sama, & Clement T. Prabhakar. (2021). PySWR-A Python code for fitting soil water retention functions. *Computers & Geosciences 156*, 104897.

Miao, R., Song, J., & Zhu, Y. (2017). 3D geographic scenes visualization based on WebGL. In *6th international conference on agro-geoinformatics* (Vol. 1, pp. 1–6). IEEE. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8046999

Moliner-Aznar, S., Martín-Martín, M., Rodríguez-Estrella, T., & Romero-Sánchez, G. (2021). The cenozoic malaguide basin from Sierra Espuña (Murcia, S Spain): An example of geological heritage. *Geosciences*, *11*(34), 11010034. https://doi.org/10.3390/geosciences11010034

Perri, F., Critelli, S., Martín-Algarra, A., Martín-Martín, M., Perrone, V., Mongelli, G., & Zattin, M. (2013). Triassic redbeds in the malaguide complex (Betic Cordillera — Spain): petrography, geochemistry and geodynamic implications. *Earth-Science Reviews*, *117*, 1–28. https://doi.org/10.1016/j.earscirev.2012.11.002

Perri, F., Critelli, S., Martín-Martín, M., Montone, S., & Amendola, U. (2017). Unravelling hinterland and offshore palaeogeography from pre-to-syn-orogenic clastic sequences of the Betic Cordillera (Sierra Espuña), Spain. *Palaeogeography, Palaeoclimatology, Palaeoecology*, *468*, 52–69. https://doi.org/10.1016/j.palaeo.2016.11.049

Pyrcz, M. (2023). GeostatsGuy lectures. https://www.youtube.com/c/GeostatsGuyLectures

Rivillas-Ospina, German, Diego Casas, Mauro Antonio Maza-Chamorro, Marianella Bolívar, Gabriel Ruiz, Roberto Guerrero, José M. Horrillo-Caraballo, Milton Guerrero, Karina Díaz, Roberto del Rio, & Erick Campos. (2022). APPMAR 1.0: A Python application for downloading and analyzing of WAVEWATCH III® wave and wind data. *Computers & Geosciences 162*, 105098.

Rohmer, O., Bertrand, E., Mercerat, E. D., Régnier, J., Pernoud, M., Langlaude, P., & Alvarez, M. (2020). Combining borehole log-stratigraphies and ambient vibration data to build a 3D model of the lower Var valley, nice (France). *Engineering Geology*, *270*, 105588. https://doi.org/10.1016/j.enggeo.2020.105588

Scharf, T., Kirkland, C. L., Daggitt, M. L., Barham, M., & Puzyrev, V. (2022). AnalyZr: A Python application for zircon grain image segmentation and shape analysis. *Computers & Geosciences 162*, 105057.

Semmo, A., Trapp, M., Jobst, M., & Döellner, J. (2015). Cartography-oriented design of 3D geospatial information visualization–overview and techniques. *The Cartographic Journal*, *52*(2), 95–106. https://doi.org/10.1080/00087041.2015.1119462

Tent-Manclu, J. E., Martín-Martín, M., Martín-Pérez, J. A., & Serrano, F. (2001). Structural evolution of the early miocene in the eastern betic internal-external zone boundary (SE Spain). *Bulletin de la Société Géologique de France*, *172*(1), 41–47. https://doi.org/10.2113/172.1.41

Tonini, R., Sandri, L., & Thompson, M. A. (2015). PyBetVH: A Python tool for probabilistic volcanic hazard assessment and for generation of Bayesian hazard curves and maps. *Computers & Geosciences 79*, 38–46.

Wycisk, P., Hubert, T., Gossel, W., & Neumann, C. (2009). High-resolution 3D spatial modelling of complex geological structures for an environmental risk assessment of abundant mining and industrial megasites. *Computers & Geosciences*, *35*(1), 165–182. https://doi.org/10.1016/j.cageo.2007.09.001

Cited websites

3D Geomodeller. https://www.intrepid-geophysics.com/

Albion. https://gitlab.com/Oslandia/albion

ArcGis. https://www.esri.es/es-es/arcgis/productos/ind

Autocad Civil. https://www.autodesk.com/

Gempy. https://www.gempy.org

GeoPandas. https://geopandas.org/en/stable

Geoscene 3D. https://geoscene3d.com/

GISgeography. https://gisgeography.com/python-libraries-gis-mapping

Gocad. https://www.emerson.com/es-es/automation/roxar

MOVE. https://www.petex.com/products/move-suite/move/

OSGeo. https://www.osgeo.org/

Parpoil. https://oslandia.com/en/2020/07/09/geologie-open-source

PETREL. https://www.software.slb.com/products/petrel/petrel-geology-and-modeling

Qgis. https://qgis.org/en/site/

VOXI. https://www.seequent.com/products-solutions/geosoft-oasis-montaj/voxi-earth-modelling-extension/