# PROCESSING OCEANOGRAPHIC DATA BY PYTHON LIBRARIES NUMPY, SCIPY AND PANDAS

**Polina Lemenkova**

Ocean University of China, College of Marine Geo-Sciences, 238 Songling Road, Laoshan, 266100, Qingdao, Shandong Province, People's Republic of China

**ORCID IDs of the author(s):**
P.L. 0000-0002-5759-1089

**Correspondence:**
**Polina LEMENKOVA**
**E-mail:**
lemenkovapolina@stu.ouc.edu.cn

## ABSTRACT

The study area is located in western Pacific Ocean, Mariana Trench. The aim of the data analysis is to analyze the potential influence of how various geological and tectonic factors may affect the geomorphological shape of the Mariana Trench. Statistical analysis of the data set in marine geology and oceanography requires an adequate strategy on big data processing. In this context, current research proposes a combination of the Python-based methodology that couples GIS geospatial data analysis. The Quantum GIS part of the methodology produces an optimized representative sampling dataset consisting of 25 cross-section profiles having in total 12,590 bathymetric observation points. The sampling of the geospatial dataset are located across the Mariana Trench. The second part of the methodology consists of statistical data processing by means of high-level programming language Python. Current research uses libraries Pandas, NumPy and SciPy. The data processing also involves the subsampling of two auxiliary masked data frames from the initial large data set that only consists of the target variables: sediment thickness, slope angle degrees and bathymetric observation points across four tectonic plates: Pacific, Philippine, Mariana, and Caroline. Finally, the data were analyzed by several approaches: 1) Kernel Density Estimation (KDE) for analysis of the probability of data distribution; 2) stacked area chart for visualization of the data range across various segments of the trench; 3) spacial series of radar charts; 4) stacked bar plots showing the data distribution by tectonic plates; 5) stacked bar charts for correlation of sediment thickness by profiles, versus distance from the igneous volcanic areas; 6) circular pie plots visualizing data distribution by 25 profiles; 7) scatterplot matrices for correlation analysis between marine geologic variables. The results presented a distinct correlation between the geologic, tectonic and oceanographic variables. Six Python codes are provided in full for repeatability of this research.

**Keywords:** Mariana Trench, Pacific Ocean, Python, Programming language, SciPy, NumPy, Pandas, Statistics, Data analysis

## Introduction

There are various geodynamic processes that influence tectonic rift dynamics and structure as well as and rifted margin geomorphology. Currently, the interest towards the geodynamics, the drivers and consequences of these processes was implemented as key target goals of the oceanographic research in China (Cui et al., 2014; Cui & Wu, 2018). Knowing and proper understanding of the driving factors affecting the ocean ecosystems gives an understanding of the possible dynamics, accumulation, and location of the target ocean resources that are crucial for economic development.

Understanding the bathymetry of the ocean is crucial for the marina geological research. As noted by Dierssen & Theberge (2014), the distribution of elevations on the Earth or hypsography is highly uneven. Thus, the majority of the depths is occupied by deep basins (4–6.5 km) covered with abyssal plains and hills, while seafloor with ranges 2- 4 km depth mostly consists of oceanic ridges and in total cover about 30% of the total ocean seafloor. Finally, the shallow areas and continental margins with 2 km depth and shallower cover only the least amount of area, that is 15% of the seafloor (Litvin, V. M., 1987). Finally, the valley, seamounts and submarine canyons are only the minor features of the seafloor. Given the importance of the hadal areas, the study of the ocean trenches geomorphology and distribution of its features with regards to the bathymetry seems to be obvious.

There are many attempts undertaken to understand, to what extent and how do the geophysical movements in the subduction zones affect the trench geomorphology, deformation and migration (e.g. Doglioni, 2009; Fernandez & Marques, 2018; Gorbatov et al., 2001; Hubble et al., 2016; Lemoine et al., 2002). General concepts and understanding of the functioning and current problems in research directions of the marine hadal observations were implied in the current research. Active sedimentation on the bottom of the seafloor leads to the accumulated amount at rifted margins, particularly at the deltas of the large rivers. Sediments outflowing further to the ocean provide important geological bodies and resources. Besides, the natural hazards taking place in the ocean, strongly correlate with submarine earthquakes and volcanic eruptions during active rifting (Brune, 2016). Moreover, there is a certain correlation between the high oceanic features and thickness of the subduction channels and earthquake rupture segments, as shown with a case study of the trenches in the eastern Pacific Ocean by Contreras-Reyes and Carrizo. (2011). Ocean hadal trenches result from the complex geodynamic processes that continuously shape the surface of the seafloor (Bogolepov & Chikov, 1976). Nowadays, the ocean seafloor demonstrates 'footprints' of the many continuous steps of the seafloor evolution.
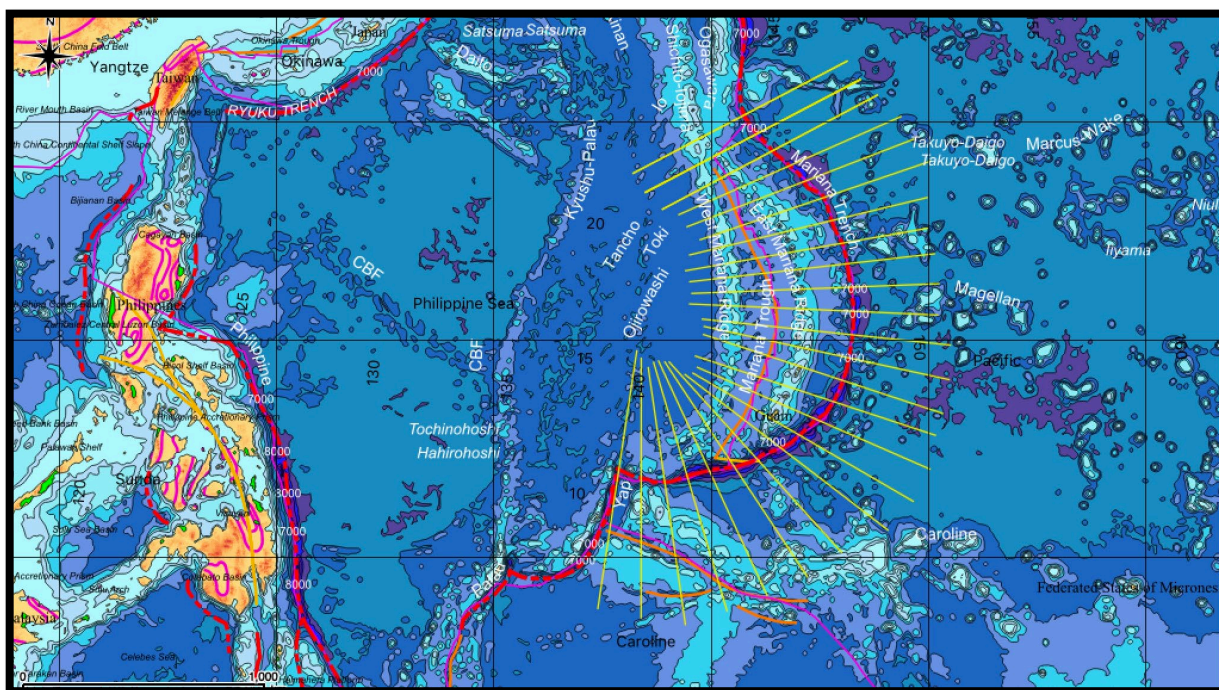


**Figure 1.** Study area visualizing 25 cross-section bathymetric profiles (yellow): QGIS map

Traditional methods of the marine geological modelling include using GIS based processing of the remote sensing images, such as for instance aerial photos, SPOT3, SPOT4 and ENVISAT data, or producing digital maps based on the data capture in the field (e.g., Bogdanov et al, 2011). On the contrary, the current paper makes an accent on using high-level programming language Python and its libraries Pandas, NumPy and SciPy for the processing of the large data frames imported from GIS. The effectiveness of the data computing and visualization by Python was the key factor for applying its functionality in this research. Some approaches of the scientific visualization and methods of the data analytics discussed previously by (Crameri, 2018) were considered in this research.

The actuality of the studies of marine natural hazards, such as submarine earthquakes and tsunamis cannot be underestimated. Recent progress in modelling earthquakes in Pacific ocean were proposed by (Kong et al., 2017). Using the global dataset of broadband and long-period seismograms, recorded as a time series ranging 2006-2014, from the Incorporated Research Institutions for Seismology (IRIS), it has been detected that there is a clearly descendance in the morphology of the Pacific plate, which becomes flatten at the base of the upper mantle and further goes westward towards a northern-central China (Dokht et al, 2016). Application of the geodynamic studies related to the tsunami, its possible reasons and consequences, are presented recently reporting that the shallowest reaches of plate boundary subductions host substantial slips that generate large and destructive tsunamis (Ikari et al. 2015). Attempts towards studies of the ocean geomorphology, dynamics, and intercorrelation between various factors affecting its functioning are given by various research (e.g. Mao et al., 2016; Masson, 1991;  Luo et al., 2018; Loher et al., 2018).

Nevertheless, the problem of the proper understanding of the hadal areas in the ocean lies in its unreachable location. As justly noted by Jamieson  (2018), understanding marine ecosystems for proper management and use of marine resources has a certain paradox, since there is a need to evaluate and protect the marine life and ocean ecosystems. However, the current knowledge on ocean functioning is relatively scarce. At the same time, modeling ocean and marine environment is a critical for the sustainable development of ocean resource usage. Recent studies only stress the strong correlation of the research with increasing ocean depths. However, the majority of the recent methods of ocean observations have overlooked the Python programming approach for statistical analysis where the large data sets are being processed by a set of the embedded mathematical algorithms. Here, the paper presents improvements on the oceanographic data processing and interpretation methods by applying Python 3.2.7 languages and using its most essential libraries: NumPy, SciPy, Pandas and Matplotlib for data visualization and analysis.

## Material and Methods

### *NumPy for Processing Arrays*

Using Python modules and libraries enables processing of the large oceanographic data more effective and significantly improves the computation algorithms. The general functioning of the Python followed the existing references and manuals (e.g., Oliphant, 2007; Pedregosa et al., 2011; Perez et al., 2007). Using libraries enable to create namespaces while working with modules. Python's modules contain packed classes, objects, functions and constants used in the work. The installation of the libraries was done using pip upon the installing NumPy and SciPy, its dependancies:

    $ python3 -m pip install numpy

    $ python3 -m pip install scipy

The sorting and selecting of read in data from the tables was performed using NumPy. NumPy creates a multidimensional array object from a given 'table.csv'. Using Python's syntax and semantics, it operates with matrices using logical, bitwise, functional operations with elements, and performs a series of routines for fast operations on arrays (NumPy community, 2019). Finally, NumPy enables various object-oriented approach, mathematical and logical manipulations with table using ndarray. The scripts, modules and codes were written using reference semantics and built-in functions of NumPy and Python. The saved script included the written codes that parse command lines and perform graphs plotting by executing functions and modules. The namespaces of NumPy were imported from numpy.core and numpy.lib by calling:

    >>> import numpy as np

The depending libraries have been installed as well. These include Jinja2, NumPy, Pillow, PyYAML, Six, Tornado. Manipulating with tabular data stored in csv files has been implemented in suitable Python library Pandas optimized for the high-level processing of tabular data. The Matplotlib library was installed as well and imported to customize plots.
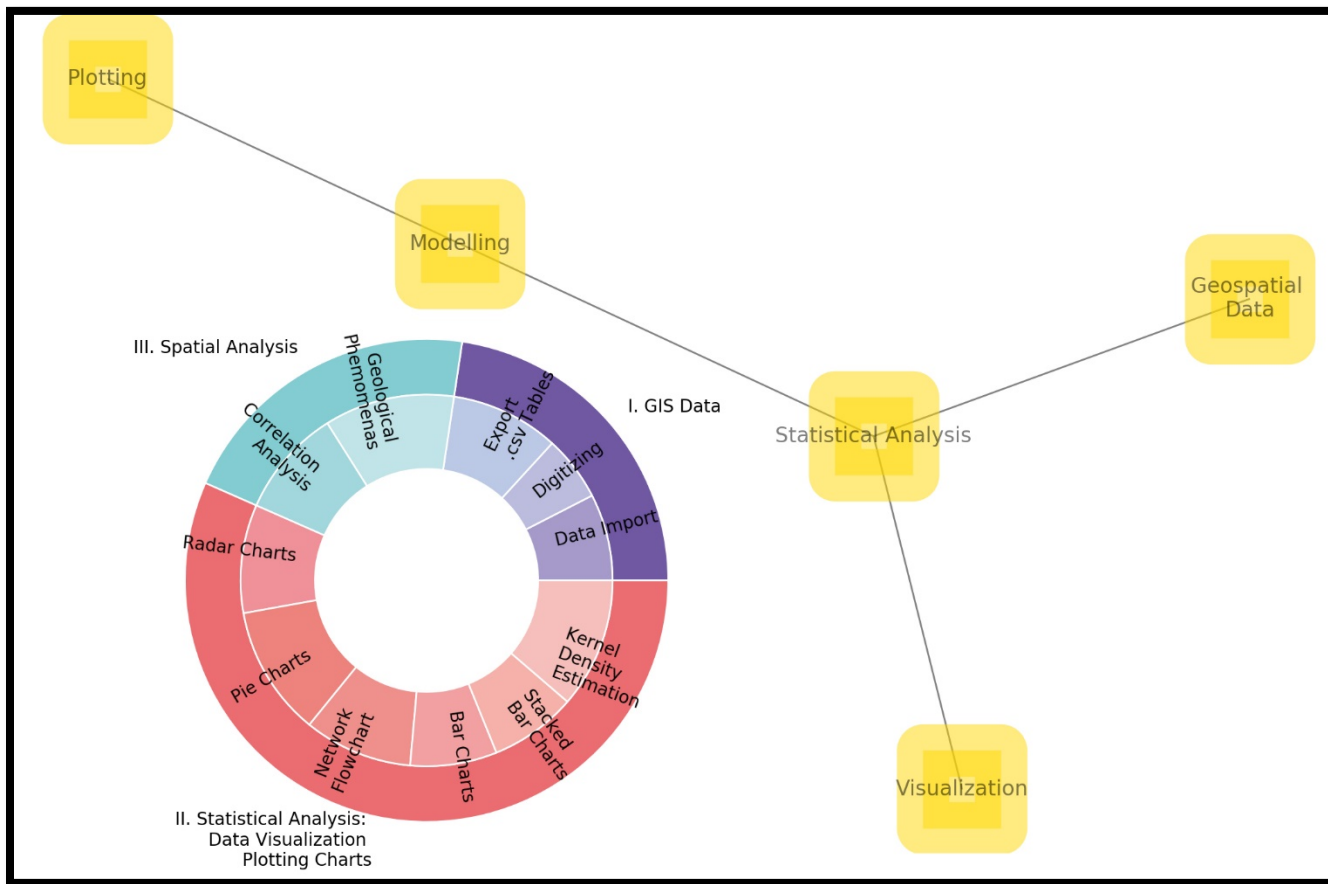
**Figure 2.** Methodological network

Various mathematical algorithms used in this work were applied from the statistical functionality provided by Python language (Beazley, 2009). The SciPy, an extension of NumPy, is another Python based package that was loaded for mathematical computations. The specific questions of usage SciPy were supported by large explanations of the SciPy principles and its usage in the statistical analysis (Jones et al., 2014).

### Methodological Network

The methodological flowchart includes three main stages (Figure 2) visualized as the logical parts of this research: first, GIS part using Quantum GIS (QGIS), second, statistical analysis on Python language; third, spacial analysis of the data similarities on R language.

First block consisted in processing oceanographic data using QGIS software: data import, digitizing profiles, data export into .csv tables for further processing in Python and R. The cross-section profiles were digitized and the attribute tables were created (Figure 1). The tables contained numerical in-

formation on geology, tectonics, oceanography and bathymetry by observation points along each profile. In total there was 25 profiles, each containing 518 observation points. Hence, the total data intakes consisted in pool of 12,590 points.

Second block contained in data interpretation and statistical analysis. The steps include following approaches of the statistical data analysis: 1) Kernel Density Estimation (KDE) for analysis of the probability of data distribution; 2) stacked area chart for visualization of the data range across various segments of the trench; 3) spacial series of radar charts; 4) stacked bar plots showing the percentage of data distribution by tectonic plates; 5) stacked bar charts for correlation of sediment thickness by profiles, versus distance from the igneous volcanic areas; 6) circular pie plots visualizing data distribution by 25 profiles.

Third block presents the geospatial analysis of the data correlation. This implies correlation analysis of the scatterplot matrices by visualizing geological and tectonic interplay be-

tween the phenomenas. The scatterplot matrices for correlation analysis between marine geologic variables were performed using R language.

### Probability of the Depths Distribution by Kernel Density Estimation Plots

In this part of the work, an implementation of the fundamental frequency estimation is presented. The algorithm of Kernel Density Estimation (KDE) is based on a frequency-domain approach (Figure 3). It was applied to visualize probability of the depth ranges and bathymetric patterns in various segments of the Mariana Trench. The method was implemented using the following code (Code 1):

**Code (1), Python: Kernel Density Estimation, example for the subplot on Figure 3 (F).**

```python
# step-1. Loading libraries
import seaborn as sns
from matplotlib import pyplot as plt
import pandas as pd
import os
os.chdir('/Users/pauline/Documents/Python')
df = pd.read_csv("Tab-Morph.csv")
sns.set_style('darkgrid')
# step-2. plotting 4 variables
ax=sns.kdeplot(df['Min'], shade=True, color="r")
ax=sns.kdeplot(df['Mean'], shade=True, color="#ffd900")
ax=sns.kdeplot(df['Max'], shade=True, color="b")
ax=sns.kdeplot(df['1stQ'], shade=True, color="#65318e")
ax=sns.kdeplot(df['3rdQ'], shade=True, color="#00a3af")
# step-3. Adding aesthetics and annotations
ax.set(xlabel='Depths, m', ylabel='KDE')
plt.title("Kernel Density Estimation: \nprobability of the statistical depth ranges, profiles 1-25")
ax.annotate('F', xy=(0.03, .90), xycoords="axes fraction", fontsize=18,
        bbox=dict(boxstyle='round, pad=0.3', fc='w', edgecolor='grey', linewidth=1, alpha=0.9))
plt.show()
```

Python libraries Pandas, Matplotlib, Seaborn and OS were used to process data by an embedded algorithms to obtain probability frequency. An open source Python code used for this plot is provided above (Code 1).

*Visualizing Bathymetric Pattern by Stacked Area Charts*

In marine geologic data sets, plotting stacked area charts is one of the key approaches to visualize the range of the bathymetric depths. In other words, we can answer the question of to what extent are the depths may reach in this or that particular segment of the trench? Apart from the visual clearance of the plot (Figure 4), showing the maximal abrupt depth by profiles 20 and 21 (that is, south-west of the Mariana Trench), there are other interesting particularities in this approach. Thus, on can investigate different phenomena of the oceanographic data sets by adding color ranges for stepwise visualization of the plot, sub-divided by statistical steps: minimal depths, third quartiles, mean depths, median values of the depths, first quartile, and finally, the shallowest parts of the geomorphology that is the minimal depths. In this way, one can understand the variability of the geomorphic patters by the segments that could reveal new insights into how the bathymetric data variability affects the complex geomorphology of the profile.

**Code (2) of Python: stacked area charts**.

```python
# Step-1. Loading libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
import os
# Step-2. Importing data
os.chdir('/Users/pauline/Documents/Python')
df = pd.read_csv("Tab-Morph.csv")
df.head(5)
# Step-3. Plotting the dataset
fig = plt.figure(figsize=(8, 6))
df = pd.DataFrame(data=df, columns=['Min', '1stQ', 'Median', 'Mean', '1stQ','Max'])
ax = df.plot.area(stacked=False, alpha=0.8, colormap='PuBu_r')
# Step-4. Adding aesthetics and annotations
plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
plt.title('Stacked area chart for the Mariana Trench bathymetry: \ndepths by 25 cross-section profiles', font-
size=12, fontfamily='serif')
ax.set_xlabel('Bathymetric profiles')
ax.set_ylabel('Depths, m')
plt.xticks(np.arange(1, 26, step=1), rotation=30)
plt.show()
```

The following Python libraries were used to plot stacked area charts: Pandas, NumPy, Matplotlib, Seaborn and OS. An open source code is provided above (Code 2).
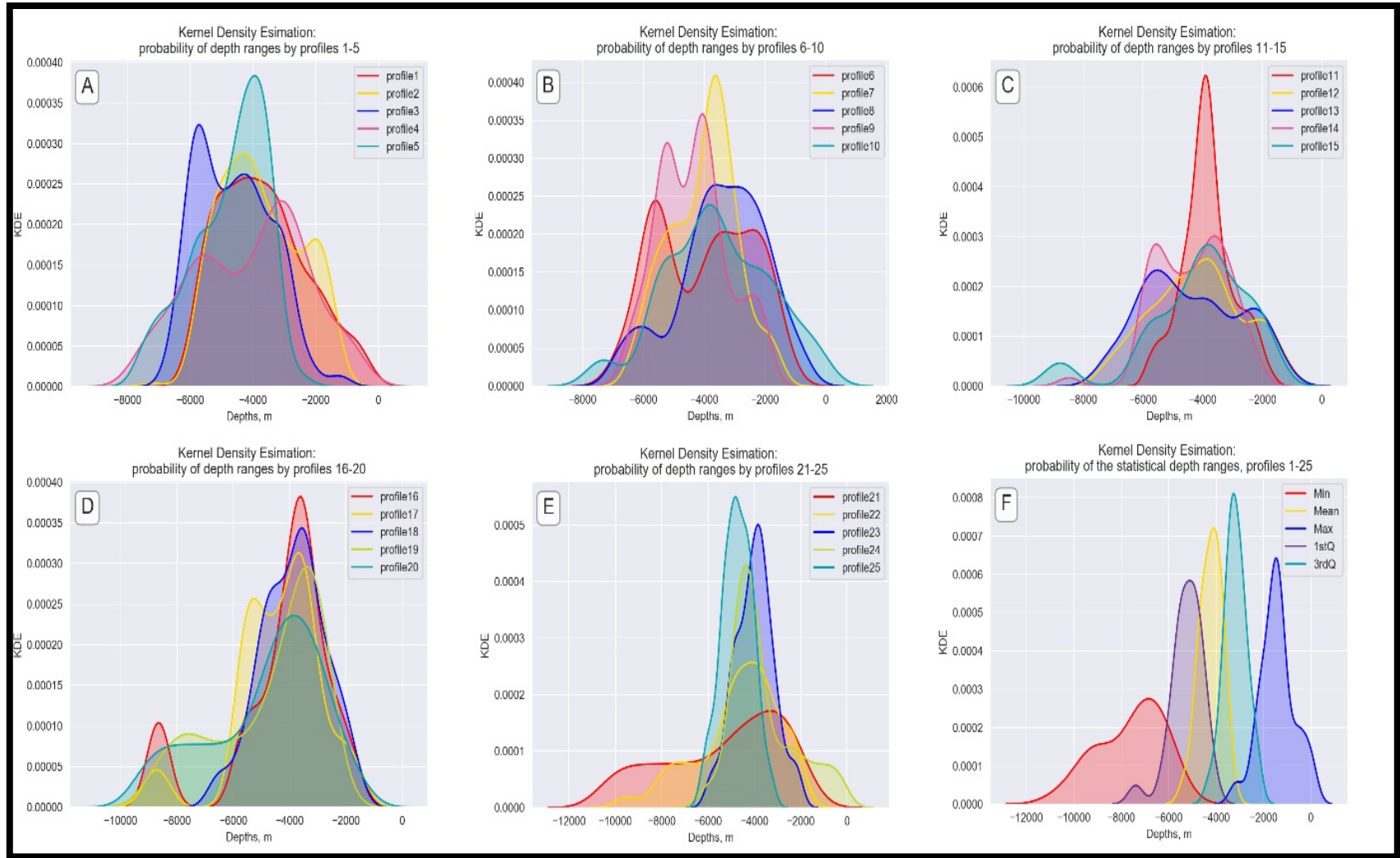
**Figure 3.** Kernel Density Estimation (KDE) for the bathymetry, profiles 1:25

**Figure 4.** Mariana Trench: bathymetric patterns visualized by stacked area charts

### *Statistical Distribution of the Bathymetric Values by Radar Charts*

Of particular interest is the case of radar charts. Recently, radar charts turned into a very interesting visualizing method in the data analysis. A radar chart is a graphical method of displaying multivariate data in the form of a two-dimensional circular chart of six quantitative variables represented on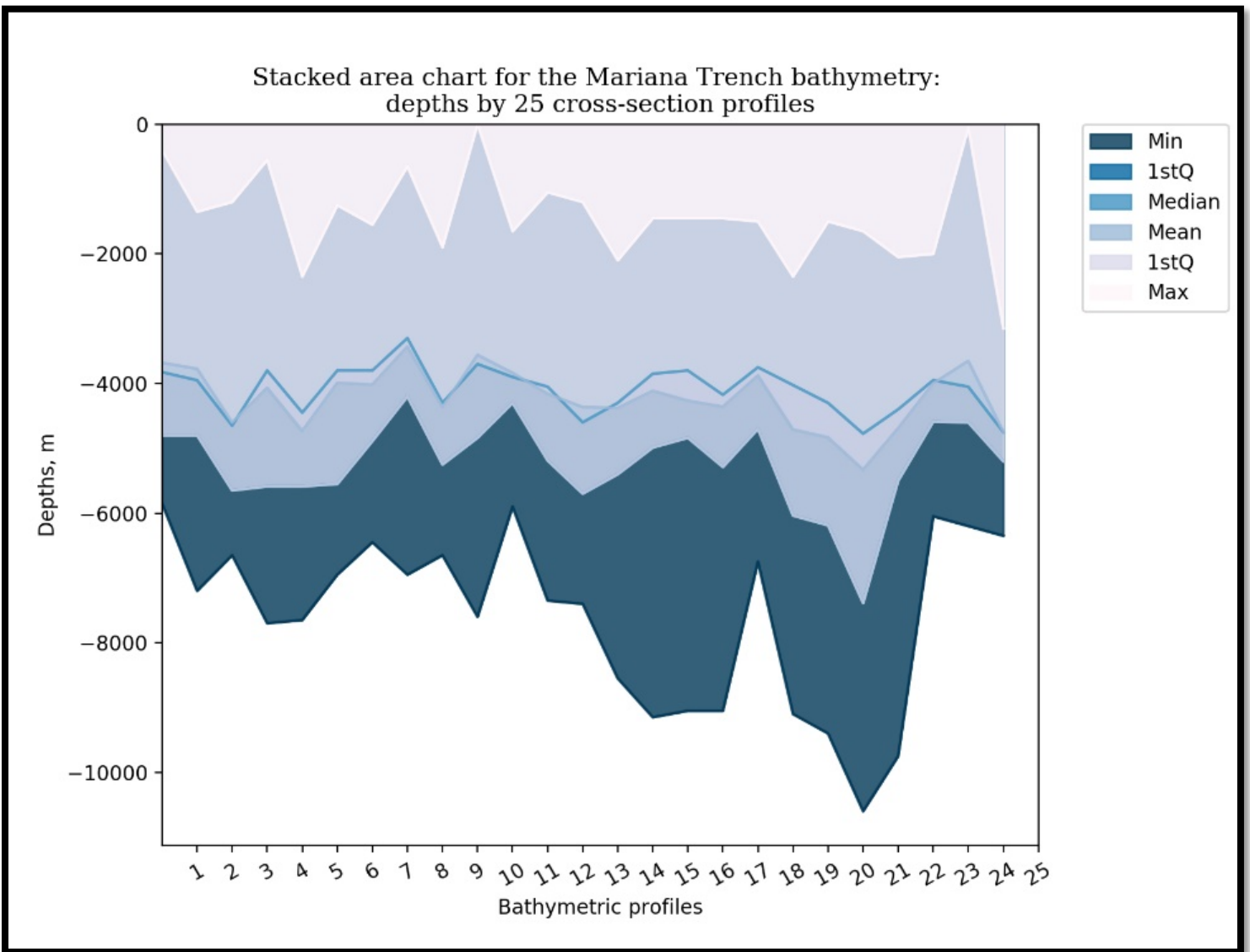 axes bathymetric depths and on the circular axes bathymetric values: maximal, $3^{rd}$ quartile, median, mean $3^{rd}$ quartile and minimal values.

The reason to choose the radar charts is that there are various statistical values that can be visualized by profiles, which requires a certain visualization technique for the facetted multi-plots (Figure 5). The series of the radar charts were plotted by libraries Math, Pandas, NumPy, Matplotlib, Seaborn and OS. An open source code is provided below (Code 3).

The algorithm for the radar charts was taken from the Matplotlib library of Python, well referenced by (Hunter, 2007).

**Code (3) of Python, for Radar charts in 4 steps (a case study for profile Nr. 1):**

```python
# Step-1. Loading libraries and data
import matplotlib.pyplot as plt
import pandas as pd
from math import pi
import seaborn as sns
import numpy as np
import os
os.chdir('/Users/pauline/Documents/Python')
df = pd.read_csv("Tab-Morph.csv")
#df.head(5)
# Step-2.  Show 6 different variables on our radar chart, so take them out and set as a np.array.
labels=np.array(['Median', 'Max', '1stQ', '3rdQ', 'Min', 'Mean'])
stats=df.loc[1, labels].values
# Step-3. close the plot
angles=np.linspace(0, 2*np.pi, len(labels), endpoint=False)
stats=np.concatenate((stats,[stats[0]]))
angles=np.concatenate((angles,[angles[0]]))
# Step-4
fig = plt.figure()
ax = fig.add_subplot(111, polar=True)
ax.plot(angles, stats, 'o-', linewidth=2)
ax.fill(angles, stats, c='g',alpha=0.2)
ax.set_thetagrids(angles * 180/np.pi, labels)
plt.setp(ax.get_xticklabels(), fontsize=10)
plt.setp(ax.get_yticklabels(), fontsize=8)
plt.title('Radar chart for the Mariana Trench \nStatistics on bathymetric profile (nr.1)',
          fontsize=12, fontfamily='sans-serif')
ax.grid(True)
ax.annotate('(A)', fontsize=18, xy=(1.02, .90), xycoords="axes fraction")
plt.show()
```
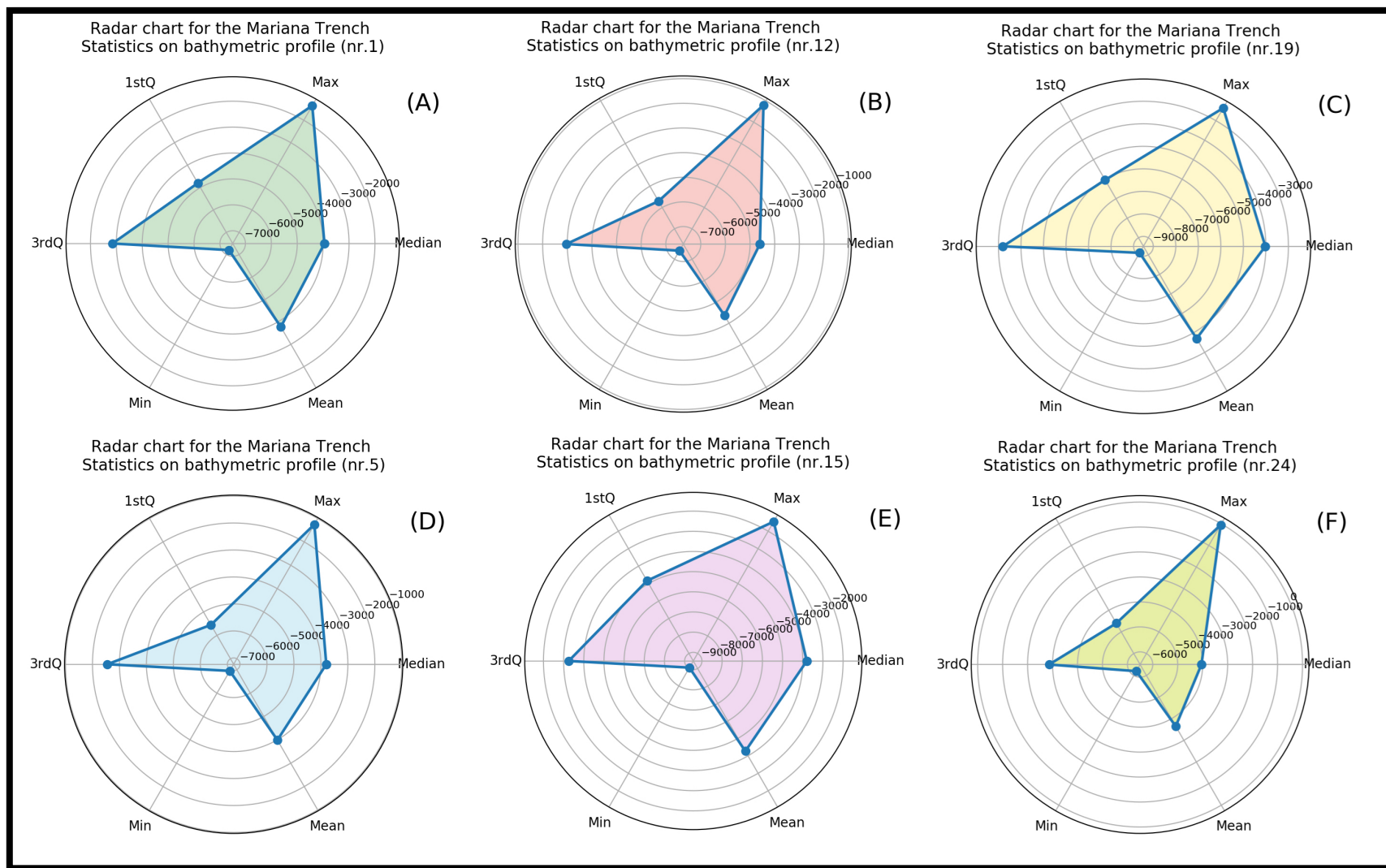
**Figure 5.** Series of the radar charts showing variation on the bathymetry by selected profiles

*Variation in the Distribution of the Bathymetric Data by Stacked Bar Plots*

Figure 6 shows the variation in the distribution of the bathymetric data by stacked bar plots

The following Python libraries were used to plot stacked area charts: NumPy, Pandas, Matplotlib and OS. An open source code is provided above (Code 4).

**Code (4) of Python, for distribution of the bathymetric data by stacked bar plots:**

```python
# Step-1. Loading libraries
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import rc
import pandas as pd
import os
# Step-2. Importing data
os.chdir('/Users/pauline/Documents/Python')
df = pd.read_csv("Tab-Morph.csv")
# Step-3. Setting up values of each group
bars1 = df.plate_phill
bars2 = df.plate_pacif
bars3 = df.plate_maria
bars4 = df.plate_carol
# Step-4. Defining positions
profiles = df.profile
# Step-5. Selecting the names of the group
names = ["profile1", "profile2", "profile3", "profile4", "profile5",
        "profile6", "profile7","profile8", "profile9", "profile10",
        "profile11", "profile12", "profile13", "profile14", "profile15",
        "profile16", "profile17", "profile18", "profile19", "profile20",
        "profile21", "profile22", "profile23", "profile24", "profile25"]
barWidth = 1
# Step-6.Creating bars
ax = plt.subplot(111)
plt.bar(profiles, bars1, color='#dbd0e6', edgecolor='white', width=barWidth, label='Philippine Plate')
plt.bar(profiles, bars2, bottom=(bars1), color='#a0d8ef', edgecolor='white', width=barWidth,
        label='Pacific Plate')
plt.bar(profiles, bars3, bottom=(bars1 + bars2), color='#eebbcb', edgecolor='white', width=barWidth,
       label='Mariana Plate')
plt.bar(profiles, bars4, bottom=(bars1 + bars2 + bars3), color='#c1d8ac', edgecolor='white', width=barWidth,
        label='Caroline Plate')
# Step-7. Adding aesthetics
plt.xticks(profiles, names, fontweight='normal', fontsize=7, rotation=30)
plt.legend()
ax.legend(loc='upper center', bbox_to_anchor=(0.5, -0.10), shadow=True,
          markerscale=2, ncol=4, fontsize=6, title=False)
plt.title('Mariana Trench: Stacked barplots for the distribution \nof the bathymetric observations across tectonic
plates',
          fontsize=10, fontfamily='sans-serif')
plt.show()
```
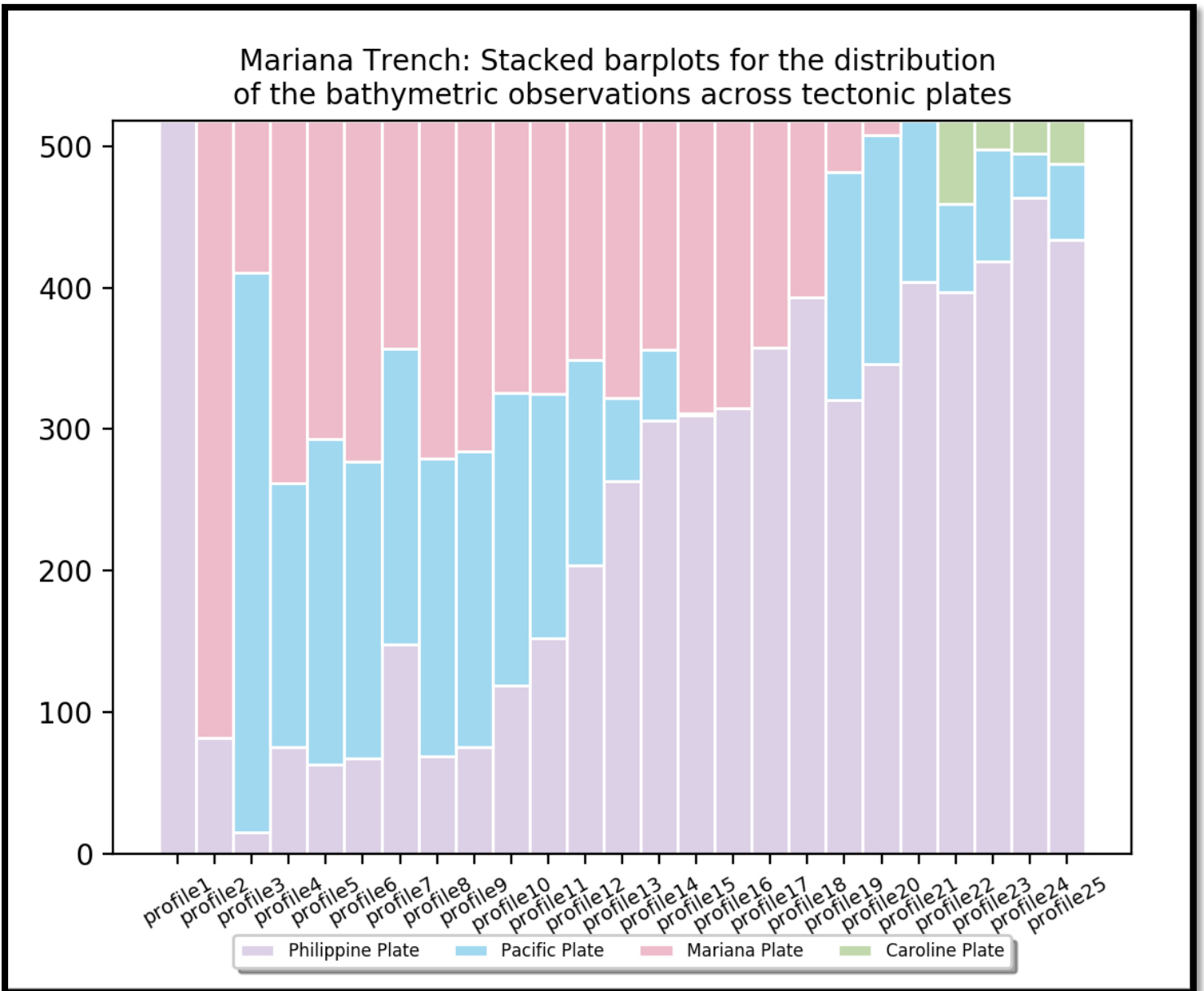
**Figure 6.** Stacked bar plots showing variation in the distribution of the bathymetric observations by the profiles, Mariana Trench

*Analyzing Distribution of the Sediment Thickness by Stacked Bar Charts*

Analysis of the distribution of the sediment thickness is visualized by the by stacked bar charts (Figure 7). The Python Code (5) in 7 steps provides an approach to visualize the sediment thickness by profiles and its correlation with closeness of the igneous volcanic areas as by distance. The code was written using calling following Python libraries: NumPy, Matplotlib, Pandas and OS.
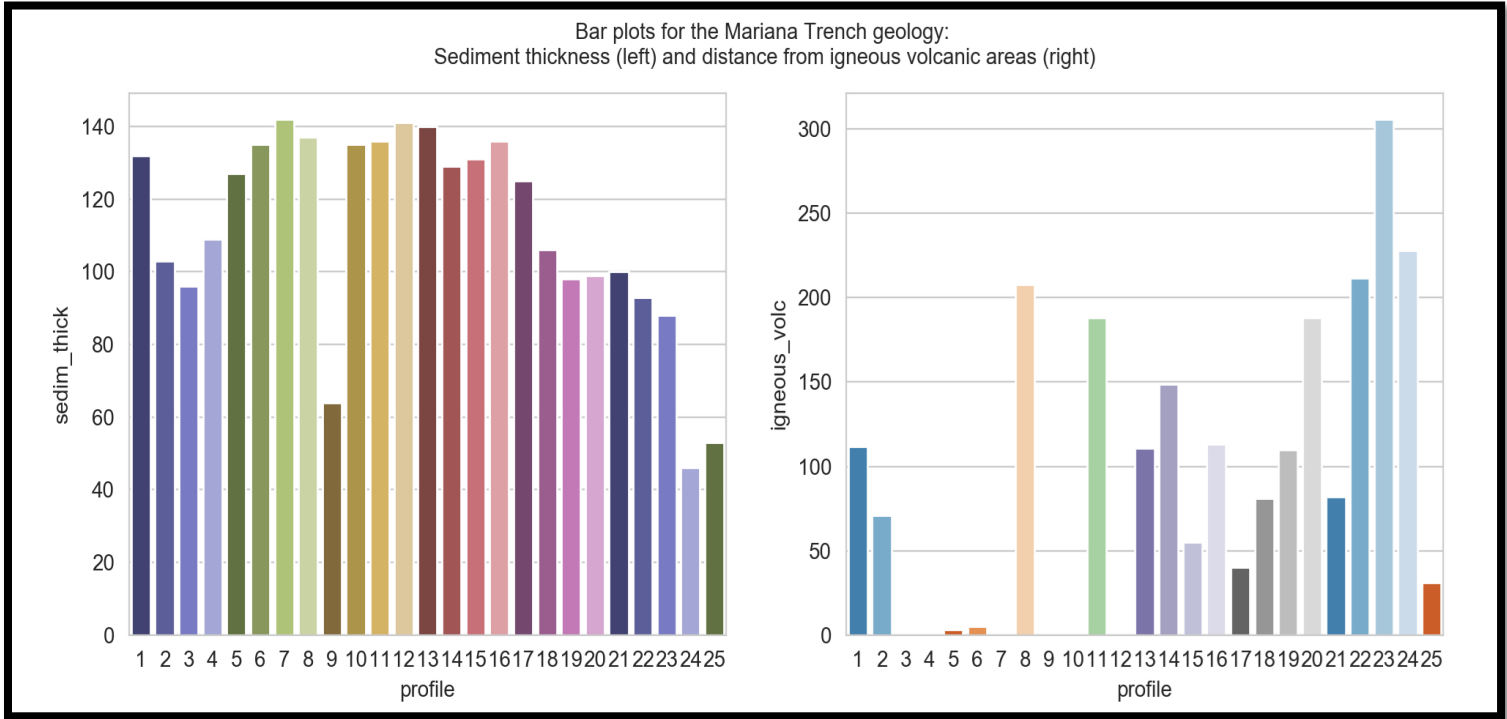


**Figure 7.**   Stacked bar charts showing distribution of values of sediment thickness by profiles (left), versus distance from the igneous volcanic areas (right), Mariana Trench.

**Code (5) of Python, for analysis of sediment thickness distribution, stacked bar charts:**

```
# Step-1. Loading libraries
import numpy as np

import matplotlib.pyplot as plt

from matplotlib import rc

import pandas as pd

import os

# Step-2. Importing aata

os.chdir('/Users/pauline/Documents/Python')

df = pd.read_csv("Tab-Morph.csv")

# Step-3. Defining values for each group

bars1 = df.plate_phill

bars2 = df.plate_pacif

bars3 = df.plate_maria

bars4 = df.plate_carol
```

```
# Step-4. Setting up position of the bars on the x-axis
profiles = df.profile
# Step-5. Selecting the names of the groups and bar width
names = ["profile1", "profile2", "profile3", "profile4", "profile5",
        "profile6", "profile7","profile8", "profile9", "profile10",
        "profile11", "profile12", "profile13", "profile14", "profile15",
        "profile16", "profile17", "profile18", "profile19", "profile20",
        "profile21", "profile22", "profile23", "profile24", "profile25"]
barWidth = 1
# Step-6. Plotting bars
ax = plt.subplot(111)
plt.bar(profiles, bars1, color='#dbd0e6', edgecolor='white', width=barWidth, label='Philippine Plate')
plt.bar(profiles, bars2, bottom=(bars1), color='#a0d8ef', edgecolor='white', width=barWidth,
        label='Pacific Plate')
plt.bar(profiles, bars3, bottom=(bars1 + bars2), color='#eebbcb', edgecolor='white', width=barWidth,
        label='Mariana Plate')
plt.bar(profiles, bars4, bottom=(bars1 + bars2 + bars3), color='#c1d8ac', edgecolor='white', width=barWidth,
        label='Caroline Plate')
# Step-7. Customizing aesthetics
plt.xticks(profiles, names, fontweight='normal', fontsize=7, rotation=30)
plt.legend()
ax.legend(loc='upper center', bbox_to_anchor=(0.5, -0.10), shadow=True,
          markerscale=2, ncol=4, fontsize=6, title=False)
plt.title('Mariana Trench: Stacked barplots for the distribution \nof the bathymetric observations across tectonic plates',
          fontsize=10, fontfamily='sans-serif')
plt.show()
```

### *Circular Visualization of the Bathymetry Versus Tectonic Plates by Pie Charts*

Regarding the categorial distribution methods, a pie chart plotting is based on the analysis of the bathymetric distribution of the values by four tectonic plates (Figure 8). Circular visualization of the bathymetry showing the relationship between tectonic plates and distribution of bathymetric data by pie charts was performed by Code (6) in 3 steps. It used the following Python libraries: Pandas, NumPy, Matplotlib and OS.

**Code (6) of Python, for visualizing bathymetry versus tectonic plates by pie charts:**

```
# Step-1. Loading libraries
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import os
os.chdir('/Users/pauline/Documents/Python')
dfM = pd.read_csv("Tab-Morph.csv")
# Step-2. Importing dataset
```

```
df = pd.DataFrame({'Pacific Plate':dfM.plate_pacif,

                  'Philippine Plate':dfM.plate_phill,

                  'Mariana Plate':dfM.plate_maria,

                  'Caroline Plate':dfM.plate_carol},

               index=dfM.profile)
# Step-3. Plotting chart
df.plot(kind='pie', subplots=True, figsize=(10, 10),  legend=False, table=False,
     fontsize=8, sort_columns=True, layout=(2, 2), colormap='tab20b',
     title='Mariana Trench: Pie Charts for the \nBathymetry distribution by tectonic plates')
plt.show()
```



**Figure 8.** Circular visualization of the data distribution: bathymetric observation points by tectonic plates. Visualization method: pie charts

**Figure 9.** Scatterplot matrices showing correlation between environmental factors

*Scatterplot Matrices of the Data Correlation: Geomorphology of the Mariana Trench*

In general, the algorithms of the plotting scatterplot matrices consist of taking the values of the variables to show correlation of their values with others across the range of the data set. It is expected to show similarity at diagonal view, with increasing dissimilarity as the differences in the geologic and tectonic values increases. Scatterplot matrices have the following correlation characteristic (Figure 9): the correlation function shows similarities between the characteristics of the four tectonic plates: Philippine, Mariana, Caroline and Pacific. When the time bathymetric values increases (Figure 9, right center), the correlation with sediment thickness increases to a maximum (Figure 9, right low).

As the observation points change their position from plates by 518 observation points, the correlation again increases back to a maximum, because the characteristics of the geology change by geographic location.

The colors stand for the following variables: green for Philippine Plate, purple triangle – for Mariana Plate, red square for Caroline Plate, and orange – for Pacific Plate. The notable peak in the correlation between the magmatism of the close igneous volcanic areas and slope angle degree indicates the dependencies between the geomorphology and tectonic properties. The method of the scatterplot matrices shows appropriate visualization for the large data set overstepping thousands of the observation points. The scatterplot matrices were plotted using functionality of R language (R Core Team, 2014).

*Visualized Concept of the Mariana Trench Project*

Fundamental principle of the usage of Python libraries in oceanography is still a novel topic There are many approaches to analyze big data sets in marine science, and many of them are based on the use of GIS in various contexts: e.g. marine biology, bathymetric mapping, navigation, marine geology for exploration, etc. The difficulty of the current research lies in the attempt to perform a multidisciplinary approach that combine Python coding using its various libraries with traditionally geoscience domain of marine geology. Moreover, most of the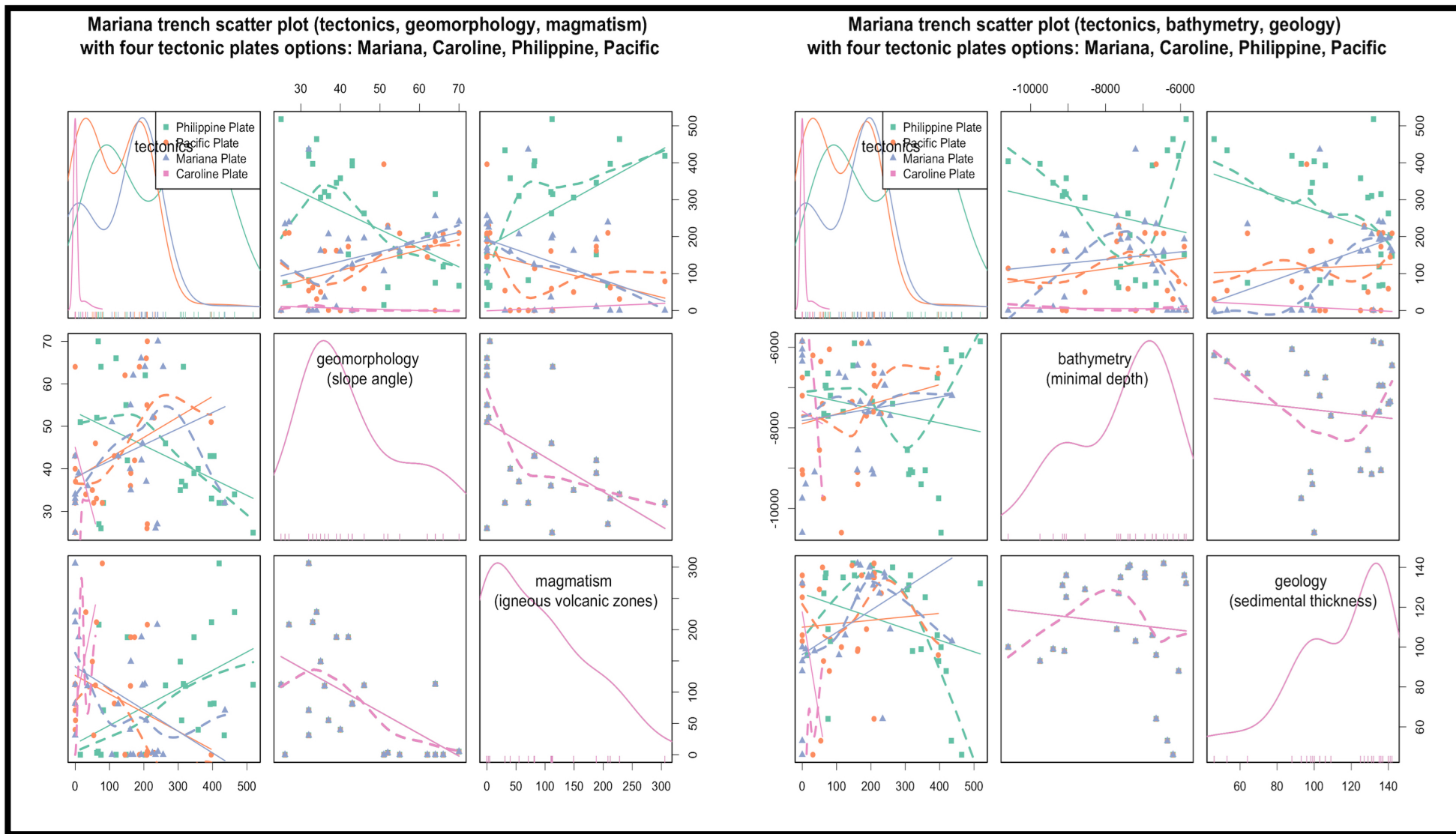 current GIS tools do not include the power functionality of Python for the statistical analysis of the large data sets in full. Certain plugins do not able to perform a Python-level data analysis with the same effectiveness and visualization.

In addition, Python powerful libraries, such as NumPy, SciPy, as well as their dependencies such as Matplotlib, developed for a particular application in the IT domain, such as computer data analysis or data science 'pure', can effectively

be applied in the domain of the marine geology. An example of a programming tool in open source code provided in this research has a main focus of the analysis of the submarine geomorphology and factors affecting its structure and variability across various segments of the hadal trench, rather than the analysis of the marine ecosystems. However, in view of the multidisciplinarity of the Python algorithms, the methods and codes demonstrated with a case study of the given data frame can effectively be applied in other aspects of the oceanographic data, including those focused on the marine biological analysis of the data taken from the cruise R/V observations.

The schematic view of the combination of various methods and key words, concepts and tools, are visualized as a word cloud (Figure 10).



**Figure 10.** Word cloud on the 'Mariana Trench' project.

## Results and Discussion

This paper, has presented a new approach for processing oceanographic data in big data sets on marine geology. The Python language and its libraries adequately perform statistical analysis and represent visualized graphs. The research implemented the proposed statistical approach using SciPy, NumPy and Pandas libraries of Python and embedded algorithms. Applied methods of Python programming defines the variability in the local geomorphological structure of the Mariana Trench, Pacific Ocean. Spatial correlation between certain environmental variables, such as sediment thickness, slope angle degree of the profiles and the geospatial location of the segments of the trench were detected.

Moreover, using Python language in the marina geology domain provides an appropriate base for the geospatial analysis of the environmental factors that may affect the morphology of the trench in its distinct parts across the crescent: south-west, central and north-west. Finally, processing a large set of data consisting of 518 observation points in 25 profiles, respectively, gives a set of 12,590 bathymetric points with variable numeric values: geomorphic, geologic and tectonic, as well as geometric values (degree of angle slope steepness by profiles).

## Conclusions

The novelty and perspectives of the proposed approach lies in its repeatability using provided codes. The proposed research methodology can guide similar research focused on the understanding marine geologic variables in other trenches, in the context of oceanographic studies and marine geologic spatial analysis. Six Python codes supported this research are provided in full in for repeatability of the methods in other case studies of the oceanography.

The Python scripts provided in this research are freely available for others and may be repeated in similar research for plotting graphs: e.g. KDE curves, radar charts, stacked area and bar plots, circular plots. All graphs in this research were made using Python, a free open source programming language, distributed from the official web site: **https://www.python.org/** A map on Figure 1 is done using open source software Quantum GIS: **https://www.qgis.org**.

**Compliance with Ethical Standard**

**Conflict of interests:** The authors declare that for this article they have no actual, potential or perceived conflict of interests.

## References

Beazley, D.M. (2009). *Python essential reference*. Addison-Wesley Professional. Available at ***www.python.org*** (accessed 18.03.2019)

Bogdanov, I., Huaman, D., Thovert, J.-F, Pierre, G., Adler, P.M. (2011). Tectonic stresses seaward of an aseismic ridge Trench collision zone. A remote sensing approach on the Loyalty Islands, SW Pacific. *Tectonophysics,* 499, 77-91.

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

Bogolepov, K. V., Chikov, B. M. *Geologiya dna okeanov (Geology of the ocean floor)*. Russian. Ed. by Saks, V. N., Fotiadi, E. E. Moscow: Nauka, 1976, 246.

Brune, S. (2016). *Plate Boundaries and Natural Hazards*. AGU Geophysical Monograph 219. ed. by Duarte, J. C. & Schellart, W. P. Sydney, Australia: AGU. Chap. Rifts and rifted margins: A review of geodynamic processes and natural hazards, 1-21.

Jones, E., Oliphant, T., Peterson, P. (2014). *SciPy: open source scientific tools for Python*. Available at www.scipy.org (accessed 23.03.2019)

NumPy community (2019). *NumPy Reference. Release 1.16.1*. 1372 p.

Contreras-Reyes, E., Carrizo, D. (2011). Control of high oceanic features and subduction channel on earthquake ruptures along the Chile-Peru subduction zone. *Physics of the Earth and Planetary Interiors* 186, 49-58.

Crameri, F. (2018). Geodynamic diagnostics, scientific visualisation and StagLab 3.0. Geoscientific Model Development, 11, 2541-2562.

Cui, W., Hu, Y., Guo, W., Pan, B., Wang, F. Reprint of a preliminary design of a movable laboratory for hadal trenches. *Methods in Oceanography,* 10(2014), 178-193.

Cui, W., X., Wu (2018). A Chinese strategy to construct a comprehensive investigation system for hadal trenches. *Deep-Sea Research Part II* 155, 27–33.

Dierssen, H.M., Theberge, A.E.J. (2014). Encyclopedia of Natural Resources. Taylor & Francis. Chap. Bathymetry: Features and Hypsography, 1–7. https://doi.org/10.1081/E-ENRW-120048589

Doglioni, C. (2009). Comment on 'The potential influence of subduction zone polarity on overriding plate deformation, trench migration and slab dip angle' by W.P. Schellart. *Tectonophysics,* 463, 208-213.

Dokht, R.M.H., Gu, Y.J., Sacchi, M.D. (2016). Waveform inversion of SS precursors: An investigation of the northwestern Pacific subduction zones and intraplate volcanoes in China. *Gondwana Research,* 40, 77-90.

Fernandez, M.O., Marques, A.C. (2018). Combining bathymetry, latitude, and phylogeny to understand the distribution of deep Atlantic hydroids (Cnidaria). *Deep-Sea Research Part I,* 133, 39-48.

Gorbatov, A., Fukao, Y., Widiyantoro, S., Gordeev, E. (2001). Seismic evidence for a mantle plume oceanwards of the Kamchatka Aleutian trench junction. *Geophysical Journal International,* 146, 282-288.

Hubble, T., Webster, J., Yu, P., Fletcher, M., Airey, D., Clarke, S., Mitchell, D., Voelker, D., Puga-Bernabeu, A., Howard, F., Gallagher, S., Martin, T. (2016). Submarine Mass Movements and their Consequences. Advances in Natural and Technological Hazards Research. ed. by G. Lamarche. Switzerland: Springer International Publishing. Chap. Chapter 12. Submarine Landslides and Incised Canyons of the Southeast Queensland Continental Margin, 125-134. https://doi.org/10. 1007/978-3-319-20979-1_12

Ikari, M.J., Kameda, J., Saffer, D.M., Kopf, A.J. (2015). Strength characteristics of Japan Trench borehole samples in the high-slip region of the 2011 Tohoku-Oki earthquake. *Earth and Planetary Science Letters,* 412, 35-41.

Jamieson, A.J. (2018). A contemporary perspective on hadal science. *Deep-Sea Research Part II,* 155, 4-10.

Kong, X., Li, S., Wang, Y., Suo, Y., Dai, L., Géli, L., Zhang, Y., Guo, L., Wang, P. (2017). Causes of earthquake spatial distribution beneath the Izu-Bonin-Mariana Arc. *Journal of Asian Earth Sciences*, 151, 90-100.

Lemoine, A., Madariaga, R., Campos, J. (2002). Slab-pull and slab-push earthquakes in the Mexican, Chilean and Peruvian subduction zones. *Physics of the Earth and Planetary Interiors,* 132, 157-175.

Litvin, V.M. (1987). *Morfostruktura dna okeanov (Morphostructure of the ocean floor)*. In Russian. Ed. by A. N. Lastochkin. Leningrad: Nedra. 275.

Loher, M., Marcon, Y., Pape, T., Römer, M., Wintersteller, P., Santos Ferreira, C. dos, Praeg, D., Torres, M., Sahling, H., Bohrmann, G. (2018). Seafloor sealing, doming, and collapse associated with gas seeps and authigenic carbonate structures at Venere mud volcano, Central Mediterranean. *Deep-Sea Research Part I,* 137, 76-96.

Luo, M., Algeo, T.J., Tong, H., Gieskes, J., Chen, L., Shi, X., Chen, D. (2018). More reducing bottom-water redox conditions during the Last Glacial Maximum in the southern Challenger Deep (Mariana Trench, western Pacific) driven by enhanced productivity. *DeepSea Research Part II,* 155, 70-82.

Mao, X., Zhang, B., Deng, H., Zou, Y., Chen, J. (2016). Three-dimensional morphological analysis method for geologic bodies and its parallel implementation. *Computers & Geosciences,* 96, 11-22.

Masson, D.G. (1991). Fault Patterns at Outer Trench Walls. *Marine Geophysical Researches,* 13, 209-225.

Oliphant, T.E. (2007). Python for scientific computing. *Computing in Science & Engineering* 9(3), 10-20.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. (2011). Scikit-Learn: machine learning in Python. *The Journal of Machine Learning Research,* 12, 2825-2830.

Perez, F., Granger, B.E. (2007). IPython: a system for interactive scientific computing. *Computing in Science & Engineering* 9(3), 21-29.

R Core Team. (2014). R: a language and environment for statistical computing. Vienna. Available at http://www.R-project.org (accessed 14.12.2018)