Prepare and analyze taxation data using the Python Pandas library

M Vagizov¹, A Potapov¹, K Konzhgoladze¹, S Stepanov² and I Martyn²

¹Department of information systems and technologies, Institute forest and nature management, St. Petersburg State Forest Technical University, 5 Institutskiy Lane, St. Petersburg 194021, Russian Federation
²Department of applied computer science, Russian State Hydrometeorological University, 79 Voronezhskaya Street, St. Petersburg 192007, Russian Federation

*Corresponding email: bars-tatarin@yandex.ru

Abstract. The article discusses the application of the PANDAS library as a method of machine learning for the processing, preparation and analysis of taxation data. Taxation data is a set of data obtained during forest inventory, collection of information about forest stands. One of the most common types of data in forest inventory is taxation data, on the basis of this data forest thematic maps are created, it is possible to predict economic activities in the forest. The large volume of accumulated forest inventory data allows us to study and analyze these data as "big data".

1. Introduction
Large volume of accumulated information of taxation descriptions is a good informative base for application of machine learning technologies in of forest plantations inventory. Mostly, various software packages are used for research in the forestry industry, such as MS Excel, Statistica, SPSS Statistics and others. Much less often, statistical research is carried out using programming languages, for example, Python or R. Programming languages, in contrast to software packages, are very promising and have the following advantages: they are more flexible in configuration and operation [1], more optimized for processing large data sets [2, 3], allow you to create ready-made applications for the needs of certain specialists in the forestry industry [3], and also allow you to develop machine learning models and neural networks for automated data analysis, in some cases even without human assistance. The development of machine learning methods in the forestry industry is increasingly being applied to various research tasks and the processing of collected forest data.

2. Methods and Materials
The institute of forest and natural resources management of SPbSFTU was given data as part of a curriculum. The main dataset was presented as a .txt file. It contained the valuation data of the Lisinsky experimental forestry enterprise in Leningrad region from 2005. It was necessary to mark up and group the data for further work with them. The research included the following tasks:
1) Transfer unlabeled data from the .txt format to a tabular human-readable form (in this case, the .xlsx format).
2) To group the data so that the characteristics of forest elements are taken into account within one section, while being correlated with other valuation indicators (for correct statistical analysis), for example, the species with its age, height, diameter, growth class.

3) To show some of the possibilities of visualizing and processing data using the Python programming language and its libraries: Pandas, Seaborn, Scikit-learn.

The work was carried out using the following software: Microsoft Excel for data preprocessing; development environment in Python - Jupyter Notebook, used libraries: Pandas, Seaborn, Scikit-Learn.

3. Results and Discussion
To separate data into columns we can use function of Microsoft Excel – Data – Text to Columns, or we can use function Pandas – read_csv(), with argument separator (sep = ‘ ‘). After this we can watching that (figure 1):

![Figure 1](image)

**Figure 1.** Data after separation to columns, original image in program.

We should clean data after separation: deletesuperfluous characters (Pandas – function replace()), and some calculated data from the initial set were removed in order to optimize further processing: age classes, age groups, stocks per stand, and management orders [4].

We should use the function pandas.DataFrame.isin() to set up a filter, which will allowed to show lines with necessary values. Use the function pandas.DataFrame.values to display value lists in necessary columns. General appearance of request will occur approximately so:

**[Species] + [Relative completeness values + Types of forest + Types of habitat]**

When the query is executed, columns with the values remain unchanged because taxation indicators (ages, heights, diameters, reserves, bonitet attached to species.

When the data are broken down by columns, text sentences (for example, description of undergrowth) are broken down by columns too. But when the query is executed, unnecessary text is deleted. And we can see table of data which is easy to work with.

Table 1 has a column (names “T”) which includes three important taxation indicators - density of forest, type of forest and type of site characteristics. This data to have been split for analyzing in future. Firstly, the column "T" duplicated three times. In the first copy - only values of the density of forest (0,3…1,0), second - type of forest, third - type of site characteristics (A2, A4, B2 etc.). After that in four columns we provide table data to the next view using function pandas.DataFrame.ffill(limit=6) and we get a table 2.

### Table 1. Dividing one column into three.

| N | Sp   | A  | H  | D  | B  | T          |
|---|------|----|----|----|----|------------|
| 4 | Spruce | 140 | 28 | 30 | 2  | 0.5        |
| 2 | Spruce | 90  | 24 | 24 | -  | OX         |
| 3 | Birch  | 90  | 27 | 26 | -  | B2         |
| 1 | Aspen  | 90  | 30 | 36 | -  | -          |

### Table 2. Readily available data.

| N | Sp   | A  | H  | D  | B  | P  | T | U   |
|---|------|----|----|----|----|----|----|-----|
| 4 | Spruce | 140 | 28 | 30 | 2  |    |    | 0.5 |
| 2 | Spruce | 90  | 24 | 24 | 2  |    |    | OX  |
| 3 | Birch  | 90  | 27 | 26 | 2  |    |    | B2  |
| 1 | Aspen  | 90  | 30 | 36 | 2  |    |    |     |
It will help us easily to address any line of dataset in future. It is so important for train models of machine learning and for analyzing the data, for example, correlation and regression analyses, when value "x" necessarily requires value "y".

To optimize work in the integrated development environment (IDE) for Python [1, 2], all Cyrillic was replaced by Latin. As a dataset, 100 forest blocks and 1925 forest patches of their taxation descriptions were investigated. After the data preparation was done and table 3 was created, we can start the data analysis.

Table 3. An example of processing and autocomplete a data table.

| Numb | Area | N  | Sp  | A  | H  | D  | M  | B  | T  | U  |
|------|------|----|-----|----|----|----|----|----|----|----|
| 1    | 0.8  | 5  | Spruce | 90.0 | 27 | 28 | 137.50 | 1 | 0.5 | KC | B2 |
| 1    | 0.8  | 3  | Aspen  | 100.0 | 32 | 40 | 87.50 | 1 | 0.5 | KC | B2 |
| 1    | 0.8  | 2  | Birch  | 100.0 | 27 | 28 | 62.0 | 1 | 0.5 | KC | B2 |

where, Numb - section number; Area - plot area, ha; N - units of the forest element; Sp - species; A - age of the forest species, years; H - height, m; D - diameter, cm; M - stock, m³/ha; B - growth class; P - relative completeness; T - type of forest; U - the type of habitat.

Let’s start analyzing of valuation data. You can group any information by tree species (or any other column) and represent as a table, using the pandas.DataFrame.groupby() method [4]. In this case, the medians are taken as the values presented in the table. As a measure of the central trend, you can use arithmetic mean values, weighted averages values, truncated means values.

Analyzing the table 4: the value of age which characterizes the birch is 70 years old, the height value is 23 meters, and the diameter value is 20 centimeters. The reserve is 62 cubic meters. Then we can compare indicators of central trends - the median and arithmetic averages for the same indicators.

Table 4. Median values of indicators grouped by forest species.

| Forest species | A, years | H, m | D, cm | M, m³/ha |
|----------------|----------|------|-------|----------|
| Birch          | 70.0     | 23.0 | 20.0  | 62.0     |
| Spruce         | 80.0     | 22.0 | 22.0  | 64.0     |
| Linden         | 55.0     | 20.0 | 17.0  | 27.0     |
| Black alder    | 60.0     | 21.0 | 20.0  | 3.04     |
| Gray alder     | 35.0     | 13.0 | 10.0  | 22.0     |
| Aspen          | 75.0     | 26.0 | 28.0  | 52.0     |
| Pine           | 90.0     | 24.0 | 26.0  | 7.03     |

The above-mentioned table 5 shows the average indicators for forestry, which can give a general idea of the research area. You can set various additional filters, for example, select the necessary forest species, their age, growth classes, and other indicators.

Table 5. Average values of indicators, grouped by forest species.

| Forest species   | A, years | H, m | D, cm | M, m³/ha |
|------------------|----------|------|-------|----------|
| Birch            | 74.183722| 20.752392| 19.059809| 81.705045|
| Spruce           | 83.067926| 19.826864| 20.178754| 97.986610 |
| Linden           | 55.0     | 20.0  | 17.0  | 27.0     |
| Alder black      | 60.0     | 21.0  | 20.0  | 3.04     |
| Alder gray       | 35.0     | 13.0  | 10.0  | 22.0     |
| Aspen            | 75.0     | 26.0  | 28.0  | 52.0     |
| Pine             | 90.0     | 24.0  | 26.0  | 7.03     |
Next, we are grouping the indicators by height (table 6). Above-mentioned part of the table with heights of forest species from 15 to 30 meters. A pattern can be traced: with increasing age, the height of the stands, its diameter, stock, and the class of growth class increases but the relative completeness is reduced. It is necessary to conduct a correlation analysis to confirm the above-mentioned theses.

**Table 6.** Average valuation indicators grouped by height (from 15 to 30 m).

| H, m | A, years | D, cm | M, m³/ha | B | P   |
|------|----------|-------|-----------|---|-----|
| 15   | 51.433333| 13.516667 | 65.744884 | 2.425000 | 0.667500 |
| 16   | 54.556701 | 13.958763 | 73.665454 | 2.371134 | 0.661458 |
| 17   | 55.222222 | 15.914530 | 81.566602 | 2.299145 | 0.705128 |
| 18   | 62.898058 | 16.815534 | 87.210918 | 2.344660 | 0.726341 |
| 19   | 66.29565  | 17.921739 | 91.376148 | 2.191304 | 0.736087 |
| 20   | 68.337553 | 19.308017 | 90.049195 | 2.105485 | 0.713502 |
| 21   | 77.071429 | 21.612403 | 100.735008| 2.119048 | 0.736087 |
| 22   | 85.682171 | 21.612403 | 84.950311 | 2.085271 | 0.680103 |
| 23   | 88.318786 | 23.028463 | 100.344759| 2.028463 | 0.671157 |
| 24   | 95.373057 | 24.314335 | 115.495580| 1.956822 | 0.648359 |
| 25   | 93.295707 | 25.456280 | 108.953670| 1.764706 | 0.648359 |
| 26   | 102.967936| 27.016000 | 105.281987| 1.744000 | 0.648359 |
| 27   | 111.339779| 30.453039 | 96.379573 | 1.798343 | 0.616022 |
| 28   | 108.993056| 31.805556 | 142.057245| 1.684028 | 0.620833 |
| 29   | 110.540541| 32.972973 | 117.175449| 1.589189 | 0.619459 |
| 30   | 112.068966| 35.275862 | 143.130056| 1.362069 | 0.625862 |

In table 7 we can see a strong positive relationship (0.91) between the diameter (D) and the height (H), which suggests that as the value of one variable increases, the value of the second increases. The same can be said about the age (A) and height (H) of the stands, but the relationship is less pronounced (correlation coefficient 0.69). There is a weak negative relationship between the growth class (B) and the height (H) (-0.41), which means that as the value of one variable increases, the value of the second variable decreases, and vice versa.

**Table 7.** Table of correlation coefficients.

| A     | H    | D    | M    | B    | P    |
|-------|------|------|------|------|------|
| A     | 1.000000 | 0.690563 | 0.761400 | 0.126099 | 0.074173 | -0.437450 |
| H     | 0.690563 | 1.000000 | 0.914046 | 0.303330 | -0.412574 | -0.195339 |
| D     | 0.761400 | 0.914046 | 1.000000 | 0.226055 | -0.243395 | -0.296259 |
| M     | 0.126099 | 0.303330 | 0.226055 | 1.000000 | -0.242914 | 0.232623 |
| B     | 0.074173 | -0.412574 | -0.243395 | -0.242914 | 1.000000 | -0.217650 |
| P     | -0.437450 | -0.195339 | -0.296259 | 0.232623 | -0.217650 | 1.000000 |

Python programming language libraries have a lot of functionality in the field of data visualization. Due to the very flexible data grouping settings that are not available in some other common software packages for data processing and analysis, you can create very simple and informative graphs.

First, we will build a histogram of the diameters and heights distribution for spruce and pine for example. Values of the total diameters and heights are 5741. The curve on the graph shows that the distribution has a bell-shaped appearance (figure 2).
Figure 2. Histogram of the distribution of spruce and pine by diameters (a) and heights (b).

Analysis of data using visualization allows fast and easy to estimate dataset, to understand which data we have to work with and information about them (distribution, structure, anomalies) [5]. Library seaborn is performing well with such challenges. It allows create graphs with group by specific indicators. For example, you can plot the distribution of heights by forest species and growth classes (figure 3).

Figure 3. Histogram of the distribution of heights by type and growth class (B – growth classes).

This graph displays average heights which group by species and classes of bonitet. Initially, bonitets related to heights, therefore, we can watch descending pattern. It concerns to all species except linden which has a lack of data and therefore there are violation of pattern.

Also we can build the graph of distribution and frequency of diameters by classes of bonitet for all species.

On figure 4 we can estimate what range of diameters is more often corresponds to each classes of bonitet based on the colour and area. If we decide analyze fifth class of bonitet, we can see - it is very rare because it has a smaller area then other classes. Minimal diameter of this class - 4 cm, maximal - 20 cm, usually fifth class of bonitet has a diameter - 12 cm.
To illustrate the presence of correlations, you can build an indicative heat map (figure 5). We can quickly find a couple of indicators using gradient (Warm-Cold), which have strong positive or negative correlations. It allows easily to analyze the dataset to determine the presence or not of correlations.

After this for analyzing of data we can apply methods machine learning using library of Python - scikit-learn. In this case we will use a model of linear regression which help us describe correlations different variable as an equation and use them. From the table of correlation coefficients, it can be seen that the greatest relationship exists in the age-diameter pair of indicators. Let's try to describe the relationship of these parameters using a linear regression model (figure 6) [6].
Figure 6. Graph of linear regression model, where $R^2$-coefficient of determination.

For description of interaction between these two indicators: Height - Diameter, without the species, we derived equations: Height = $0.66 \times$ Diameter + 7.17

Also we can create regression models for different species, for example (figure 7):

Figure 7. Graphs of linear regression models, where (a) - for spruce, (b) - for pine.

In the first case we have a regression model for all species, the coefficient of determination $R^2 = 0.81$, for models of spruce $R^2 = 0.83$ (figure 7a), and for pine $R^2 = 0.83$ (figure 7b). It means high quality of models and this results we can use in future researches. Also we can to predict the needed indicator using regression equations (in this case - diameter).

4. Conclusions
One of the proposed names for the directions of machine learning methods in relation to the processing of forest inventory data and forestry is called digital forest inventory [7]. The Python programming language and its libraries allow at the initial stage to effectively prepare data for further processing. Then you can analyze large amounts of data and visualize the results of the research. Particularly important for improving the technology of data processing, can be the integration
of additional data on the studied area of forest plantations, as such data can be remote sensing data [8, 9]. Similarly, we can analyse taxation data, large volumes, and analyse taxation data from several forest blocks. Also the advantage of data processing in Python is that all the calculations performed can be transferred to servers for remote data processing [10], which makes it possible to speed up the process of studying and using the collected material in the forest industry [11].

References
[1] Jake Vander Plas. 2016 Python Data Science Handbook: Essential Tools for Working with Data (1st. ed.). (US: O'Reilly Media)
[2] McKinney W. 2013 Python for Data Analysis: Data Wrangling with Pandas, NumPy and IPython. (US: O'Reilly Media)
[3] Hampton S E, Strasser C A, Tewksbury J J, Gram W K, Budden A E, Batcheller A L, Duke C S and Porter J H. 2013 Big data and the future of ecology. Frontiers in Ecology and the Environment 11(3) pp 156–162
[4] Vyichen V Yu and Yurik E A. 2020 Array data analysis using PANDAS library for Python. Scientific Review. Technical Sciences. No 4 pp 41-45
[5] Kizyanov A O. 2018 Using Seaborn palettes in plotting in Python programming language. Postula1 No 7(33) p 18
[6] Chibirova M E. 2019 Data analysis and regression modeling using Python and R programming languages Scientific notes of young researchers vol 7 No 2 pp 37-45
[7] Vagizov M R. 2018 On the Concept of Digital Forest Inventory. The issues of development of modern science and practice during the formation of the digital economy Materials of the International Scientific and Practical Conference ed V M Gedio et al (St.Petersburg: SPbGLTU) pp 25-28
[8] Reichman O J, Jones M B and Schildhauer M P. 2011 Challenges and opportunities of open data in ecology Science 331(6018) pp 703–705 DOI: 10.1126/science.1197962
[9] Vagizov M R et al. 2020 Technological aspects of the development of the automated method of air-photo interpretation of forest stand. IOP Conference Series: Earth and Environmental Science 574(1) 012023
[10] Marconi S et al. 2019 A data science challenge for converting airborne remote sensing data into ecological information PeerJ vol 2019 No 2 p e5843 DOI 10.7717/peerj.5843
[11] Kussul N, Lavreniuk M, Skakun S and Shelestov A. 2017 Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data IEEE Geoscience and Remote Sensing Letters, vol. 14 No 5, pp 778-782 DOI: 10.1109/LGRS.2017.2681128