International Journal of Computer Trends and Technology
ISSN: 2231-2803 / https://doi.org/10.14445/22312803/IJCTT-V68I9P101

Volume 68 Issue 9, 1-10, September 2020

Original Article

Determination of the Psychotypes of Drivers Using Machine Learning Algorithms Using Two-Stage Cluster Analysis

Chantieva Milana[1], Gematudinov Rinat[2], Dzhabrailov Khizar[3], Gorodnichev Mikhail[4]

[1]*Moscow technical university of communications and informatics, 111024, Moscow, Aviamotornaya str., 8a, Russia*
[2]*Moscow Automobile and Road Construction State Technical University, 125167, Moscow, Leningradsky* av.,*64 Russia*

Received Date: 04 August 2020
Revised Date: 02 September 2020
Accepted Date: 06 September 2020

Abstract *- This article examines the literature on mathematical methods and the theory of traffic flows in the 1930s. These problems, both complex socio-technical systems, particularly, and in general, still attract the attention of researchers studying traffic in megacities. A number of researchers, based on practical data, prove that the results of the hydrodynamic approach to solving traffic problems were more modest than in the original environment, that is, in a liquid. This has previously been noticed by researchers who have noticed the limited usefulness of the entered relationships. Despite numerous attempts to reduce the model to mathematical physics equations, their authors rarely examined the proposed approaches for applicability. Often there are no strict definitions and sufficient conditions for their operation. It is noted that the problem of micro-vibrations in liquids - and especially cavitation - is rather complicated and remains open for further research. As a socio-technical system, generally speaking, the flow of cars has no relation of speed to density in a wide range of partially connected states: an increase in the number of vehicles per unit of carriageway does not have much effect on speed.*

The work considered the problem of recognizing the psychotypes of drivers - a very urgent task Today. The solution to such a problem as car accidents and fatal accidents is in the interest of many, if not everyone. There are already braking systems that can detect objects around the car road tracking systems that follow the markings using video cameras.

RFBR, project number 19-29-06036, funded the reported study.

Keywords *- Machine learning model, Deep learning, Neural network, Probabilistic modelling, Input processing, Data analysis, Clustering algorithms, Computer software, Speed mode, Cluster, and Recognition of drivers' Psychotypes.*

I. INTRODUCTION

In the 90s, computers fell sharply in price and began to appear everywhere, and users gradually realized their capabilities. A computer can store your photos, simplify business processes, and communicate with people from another continent. Today it is difficult to find a university that does not train programming specialists; it is difficult to find a company that does not employ at least one programmer. Now the same is happening with data. The need for data analysis has gone far beyond technological and internet companies [1].

II. PURPOSE AND SOLUTION OF A TASK

Supervised learning is characterized by the presence of information that indicates which datasets are satisfactory for learning purposes. An example is software that recognizes whether a given image is a face image: to study the program, we will need to provide different ideas and determine whether they are faces. However, in unsupervised learning, the program has no data determining what information is satisfactory. The primary purpose of these programs is usually to find patterns that allow you to divide and classify data into different groups according to their attributes. Following the previous example, unsupervised learning software cannot tell us if a given image is a face or not but can, for example, classify images between those that contain human faces, animals or those that do not. Information obtained using an unsupervised learning algorithm must later be interpreted by a human to be useful. Machine learning methods are increasingly used in the optimization technological processes or transport routes, with their help, new drugs or cars without a driver are created. The 21st century can be safely called the age of data. The popularization of data has led to the formation of the information society. The need for data analytics arises in all industries, from medicine to the oil industry. In our time, the indicator of socio-technical systems (road communications) has grown, and with them, the need for information analysis has evolved to simplify management. There is also deep

learning. To understand the difference between deep and machine learning, you first need to understand what the machine learning algorithms do directly. Machine learning requires three building blocks:

- Control input data - data can be taken, for example, pictures in a classification task.
- Examples of expected results - these can be tags in an image classification task, such as "dog" or "cat."
- A way of evaluating the quality of the algorithm's work is a need to understand how far the results deviate from the expected ones. A machine learning model transforms raw data into human-useful products based on well-known examples of inputs and outputs. That is, the main task of machine learning is a meaningful transformation of data, training in the representation of input data, which brings us closer to the expected result. Technically, machine learning is the search for a weighty representation of any data in a predefined space of possibilities using feedback signals. This simple idea allows you to solve a number of complex problems, from speech recognition to automatic control of vehicles. Deep learning is a special section of machine learning. This is an entirely new approach to data retrieval. Advanced learning focuses on analyzing all levels of representations to display the most accurate results. Machine learning depth is not the literal depth of representation but width due to the layering of representations. Data depth refers to the number of layers into which the data model is divided. Modern deep learning often involves tens or even hundreds of sequential levels of presentation. Other machine learning techniques focus on learning 1 or 2 levels of data representation. Layered representations are learned using models called neural networks. These models are structured in the form of superimposed layers [2].

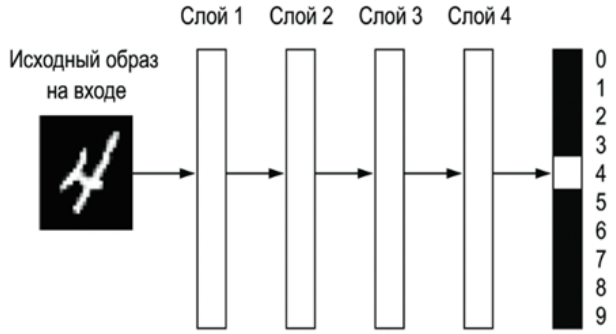


Fig. 1 Deep neural network for digit classification

Deep learning is part of a broader set of machine learning techniques represented by data representations. An observation (e.g., an image) can be expressed in pictures of figures (e.g., a pixel vector); Based on examples and research in this area. Several deep learning architectures such as deep neural networks, deep convolutional neural networks, and deep persuasion networks have been applied to areas such as computer vision, automatic speech recognition, and audio and music signal recognition. They have been shown to produce cutting-edge results in various tasks. There is no single definition of deep learning. In general, it is a class of algorithms designed for machine learning. Based on this broad perspective, different publications focus on other characteristics, for example:

- Use of cascading layers with nonlinear processors to extract and transform variables. Each layer uses the output of the previous layer as input. Algorithms can use supervised learning or unsupervised learning, and applications include data modelling and pattern recognition.
- Based on the study of multiple levels of characteristics or representations of data. Higher-level characteristics are derived from lower-level characteristics to form a hierarchical representation.
- Exploring multiple levels of presentation that correspond to different levels of abstraction. These levels form a hierarchy of concepts.
  These ways to define deep learning have in common: multilevel nonlinear processing; and supervised or unsupervised learning of feature representations in each layer. Layers form a hierarchy of characteristics from a lower level of abstraction to a higher one. Deep learning algorithms contrast with shallow learning algorithms by the number of transformations applied to a signal as it propagates from the input layer to the output layer. Each of these transforms parameters can be trained as weights and thresholds. There is no de facto standard for the number of transforms (or levels) that make an algorithm deep. Still, most researchers in the field believe that deep learning involves more than two intermediate transformations.

One of the main characteristics of deep learning is the use of an estimate of the distance between the network's prediction and the true value. For this, the loss function is used, or as it is also called the "objective function" see fig. 2.

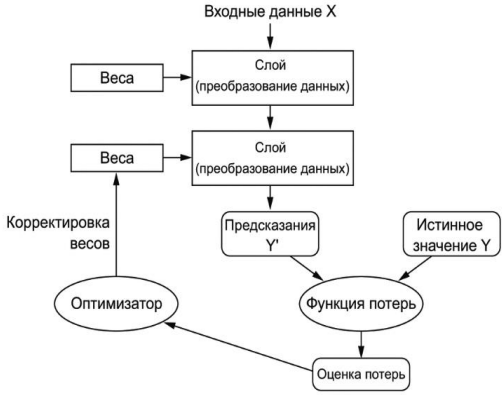


Fig. 2 The loss function evaluates the quality of the results produced by the neural network

Earlier, each level determines what to do with weights, which are a set of numbers. What the level does with the input is determined by its weights. Weights are also called layer parameters. The transformations produced by the layer are parameterized by their weights. In this context, learning means searching for a set of weights. A deep neural network can contain tens of millions of different parameters, so determining the required weights is a very difficult task, see Fig 3. Changing one parameter can affect all other parameters.

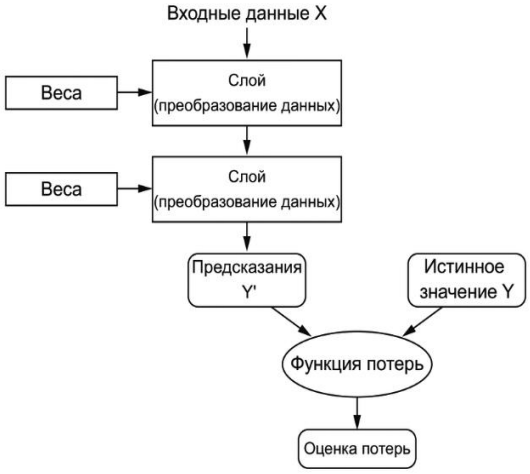


Fig. 3 The neural network is parameterized by its weights

One of the main tricks of deep learning is to use an estimate of the distance between the expected result and the true one to adjust the weights to minimize losses, see fig. 4.

Fig.4 Loss estimates are used as feedback for weight adjustments

This correction is the optimizer's task, which implements the so-called backpropagation algorithm: the central machine learning algorithm. Initially, the entire network is assigned random values. That is, the network executes a sequence of random transformations.

With each example processed by the networks, the weight is optimized and adjusted in the desired direction, the estimate of losses becomes smaller.Deep learning has not reached a level of public attention, and investment in the industry has never been seen before in AI history, but it is not the first successful form of machine learning. It is safe to say that most of the machine learning algorithms used in industry Today are not deep learning algorithms. Deep learning isn't always the right tool for the job - sometimes there isn't enough data for deep learning, and sometimes the problem is better solved with a different algorithm. Probabilistic modelling is the application of statistical principles to data analysis. It was one of the earliest forms of machine learning and is still widely used to this day. One of the most famous algorithms in this category is the Naive Bayes algorithm. A naive Bayesian type is a machine learning classifier based on the application of Bayes' theorem while at the same time assuming that all inputs are independent (strong or "naive" assumption, hence the name). This form of data analysis predated computers and was applied manually for decades before the first computer. Implementation (most likely dating back to the 1950s). Bayesian Theorem and Foundations of Statistics date back to the eighteenth century, and that's all you need to start using naive Bayesian classifiers. A closely related model is logistic regression (abbreviated as log reg), which is sometimes considered the "hello world" of modern machine learning. Don't be fooled by your name - log reg is a classification algorithm, not a regression algorithm. Like naive Bayes, log reg predated computation much earlier but is still useful Today due to its simple and versatile nature. It is often the first data scientist to try on a dataset to understand the classification task at hand. Early versions of neural networks have been entirely supplanted by modern options. Although the basic ideas of neural networks have been explored in toy forms as far back as the 1950s, this approach took decades. For a long time, the missing piece has been an effective way to train large neural networks. This changed in the mid-1980s when several people independently re-discovered the backpropagation algorithm.

A form of training chains of parametric operations using gradient descent optimization (we will define these concepts precisely later in the book) and started applying it to neural networks. The first successful practical applications of neural networks came in 1989 from Bell Labs when Yann LeCoon combined earlier ideas of convolutional neural networks and backpropagation and applied them to classifying handwritten numbers. The resulting network, dubbed LeNet, was used by the United States Postal Service in the 1990s to automate postal codes' reading on mail envelopes. The main reason deep learning started so quickly is that it gave the best performance on many problems. But this is not the only reason. Deep learning also makes problem-solving a lot easier as it completely automates what used to be The most

important step in the machine learning process: feature design.

Previous machine learning methods - surface learning - only involved transforming input data into one or two sequential presentation spaces, usually through simple transformations such as multidimensional nonlinear projections (SVMs) or decision trees. But the sophisticated insights required by complex problems usually cannot be achieved by such methods. Thus, people had to go to great lengths to make the original input data more usable with these methods: they had to manually design good presentation layers. This is called a technique feature. On the other hand, deep learning completely automates this step: with deep learning, you learn all the functions in one pass rather than engineer them yourself. This has dramatically simplified machine learning workflows, often replacing complex multi-stage pipelines with one simple, end-to-end, deep learning model. There is a rapid decrease in the return on successive applications of surface learning methods in practice since the optimal first level of representation in a three-layer model is not the optimal first level in a single-layer or two-layer model. What's deep about deep learning is that it allows the model to learn all levels of representation together, at the same time, rather than sequentially (greedily, as it's called). When a model adjusts one of its internal functions with collaborative learning of functions, all other functions that depend on it automatically adapt to the change without requiring human intervention. Everything is controlled by one feedback signal: every change in The model serves as the ultimate goal. These are two main characteristics of how deep learning learns from data: an incremental, incremental way of developing increasingly complex views. Fact that these intermediate incremental views are learned together, each level is updated to meet both the higher-level representative needs and needs. Layers below. Together, these two properties made deep learning significantly more successful than previous approaches to machine learning. A great way to understand the current state of machine learning algorithms And the tool is to look at machine learning contests on Kaggle. Because of its highly competitive landscape (some contests have thousands of entries and million-dollar prizes) and its wide range of machine learning challenges, Kaggle offers a realistic way to evaluate what works and what doesn't. In 2016 and 2017, two approaches dominated Kaggle: machine gradient boosting and deep learning. In particular, gradient enhancement is used for problems where structured data is available, whereas deep learning is used for perceptual difficulties such as image classification. Practitioners almost always use the excellent XGBoost library that offers support for the two most popular Data Science languages: Python and R. Meanwhile, most Kaggle deep learning contributors use the Keras library due to its ease of use, flexibility, and Python support. These are the two techniques you should be most familiar with. Today, he successfully applied machine learning: machines with gradient acceleration for small learning tasks; and deep learning for perception problems. Technically, this means you should be familiar with XGBoost and Keras - the two libraries currently dominated by the Kaggle competition. One of the key factors contributing to this influx of new faces in deep learning has been the toolboxes' democratization in this field. I did deep learning in the early days required significant knowledge of C ++ and CUDA, which few possessed. At present, necessary Python scripting skills are sufficient for deep learning. It was. This is primarily due to the development of Thea no. Then Tensor Flow, two symbolic Tensor manipulation frameworks for Python that support automatic differentiation, greatly simplify the implementation of new models and create user-friendly libraries. Like Keras, which makes deep learning as easy as manipulating LEGO bricks. Released in early 2015, Keras quickly became an in-depth learning solution for a large number of new startups, graduate students, and researchers working in the field.

The work aims to use algorithms for the analysis of the main characteristics of traffic flows to simplify the control of moving particles. The tasks are - studying the existing machine learning algorithms for solving clustering problems and choosing the most optimal machine learning algorithm.

III. INPUT PROCESSING

The data collection method is based on the terminal device's operation - Smart Sensor HD (SSHD) radar, manufactured by Wavetronix, USA. The device's purpose is to track and monitor the characteristics and indicators of moving particles in traffic flows [3].

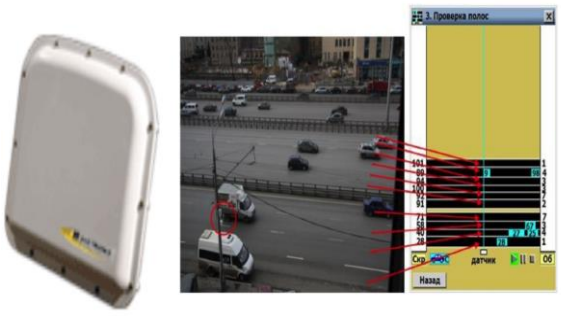
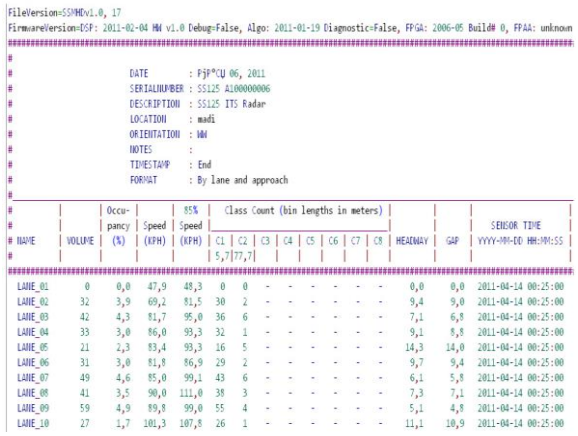


Fig. 5 Smart Sensor HD

The radars were installed in 2011 by the Moscow Department of Transport. The device reads the number of moving particles (cars) passing through the radar line, speed, distance between vehicles, occupancy, traffic load[4]. Speed determination error - + - 6 km / h. The characteristics of the transport streams are stored in log files. Data in log files are stored as follows; see fig. 6.



FileVersion=SSMMDv1.0, 17
FirmwareVersion=DSP: 2011-02-04 HW v1.0 Debug=False, Algo: 2011-01-19 Diagnostic=False, FPGA: 2006-05 Build# 0, FPAA: unknown

DATE : Pjp°CU 06, 2011
SERIALNUMBER : SS125 A100000006
DESCRIPTION : SS125 ITS Radar
LOCATION : madi
ORIENTATION : NW
NOTES :
TIMESTAMP : End
FORMAT : By lane and approach

# NAME	VOLUME	Occupancy (%)	Speed (KPH)	85% Speed (KPH)	Class Count (bin lengths in meters) C1	C2	C3	C4	C5	C6	C7	C8	HEADWAY	GAP	SENSOR TIME YYYY-MM-DD HH:MM:SS
					5,7	7,7									
LANE_01	0	0,0	47,9	48,3	0	0	-	-	-	-	-	-	0,0	0,0	2011-04-14 00:25:00
LANE_02	32	3,9	69,2	81,5	30	2	-	-	-	-	-	-	9,4	9,0	2011-04-14 00:25:00
LANE_03	42	4,3	81,7	95,0	36	6	-	-	-	-	-	-	7,1	6,8	2011-04-14 00:25:00
LANE_04	33	3,0	86,0	93,3	32	1	-	-	-	-	-	-	9,1	8,8	2011-04-14 00:25:00
LANE_05	21	2,3	83,4	93,3	16	5	-	-	-	-	-	-	14,3	14,0	2011-04-14 00:25:00
LANE_06	31	3,0	81,8	86,9	29	2	-	-	-	-	-	-	9,7	9,4	2011-04-14 00:25:00
LANE_07	49	4,6	85,0	99,1	43	6	-	-	-	-	-	-	6,1	5,8	2011-04-14 00:25:00
LANE_08	41	3,5	90,0	111,0	38	3	-	-	-	-	-	-	7,3	7,1	2011-04-14 00:25:00
LANE_09	59	4,9	89,8	99,0	55	4	-	-	-	-	-	-	5,1	4,8	2011-04-14 00:25:00
LANE_10	27	1,7	101,3	107,8	26	1	-	-	-	-	-	-	11,1	10,9	2011-04-14 00:25:00

Fig.6 Logfile content

In fig.8: Name - strip number;
Volume - number of cars (pcs);
Occupancy - load (%)
Speed - speed (km / h); C1-8 - car type;
Headway - distance from the driven front bumper to front bumper lead
Gap distance from the front bumper
Driven to rear bumper leading

Fig. 7 The sequence of data processing

Data processing was carried out using Python 2.7 in the shell. Anaconda. Data is stored in log files in an unusable format.
The files were processed in several stages. See fig. 7.

```
import pandas as pd
import re


# Ссоздание листа со списком наименований для циклической обработки файлов
df_lst = ['name','volume','occupancy','speed','speed_85','c1',
          'c2','c3','c4','c5','c6','c7','c8','gap','sensor_time',
          'interval','sp1','sp2','sp3','sp4',
          'sp5','sp6','sp7','sp8','sp9','sp10','sp11',
          'sp12','sp13','sp14','sp15','correct','wrong']
```

Fig. 8 Code snippet

For this, the function of separating the cap was developed, see fig.9. Fragment of data processing code with creating variables [5].
df_main = pd.DataFrame (df_lst) # create main dataframe from sheet
df_main = df_main.transpose () # dataframe transposition
new_header = df_main.iloc [0] # assign the first line to the variable
df_main = df_main [1:]
df_main.columns = new_header # take 1 row as the header of the dataframe

```
df_main = pd.DataFrame(df_lst) # создание основного датафрейма из листа
df_main = df_main.transpose() #транспонирование датафрейма
new_header = df_main.iloc[0] # присвоение переменной первой строки
df_main = df_main[1:]
df_main.columns = new_header #принимаем 1 строку за шапку датафрейма
```

Fig.9 Variable declaration

Below is a code snippet that specifies the path to the file that you want to process.
log_file_path = r "C: \\ Users \\ Desktop \\ logdet13-08-11 2.log" # path to file
regex = '(<property name = "(. *?)"> (. *?) <\ / property>)'
read_line = True

```
log_file_path = r"C:\\Users\\i.zugumov\\Desktop\\диплом\\logdet13-08-11 2.log" # путь к файлу
regex = '(<property name="(.*?)">(.*?)<\/property>)'
read_line = True


# цикл первичной обработки файла для очистки от шапки
with open(log_file_path, "r") as file:
    match_list = []
    if read_line == True:
        for line in file:
            for match in re.finditer(regex, line, re.S):
                match_text = match.group()
                match_list.append(match_text)
                print (match_text)
    else:
        data = f.read()
        for match in re.finditer(regex, data, re.S):
            match_text = match.group()
            match_list.append(match_text)
file.close()
```

Fig. 10 Separating the header of the log file

Since the data is still unsuitable for use and analysis (data in text format), they need to be brought into the desired form. In this regard, a code has been developed for line-by-line parsing of lines of a text file and turning them into an intermediate data frame, with their subsequent attachment to the main data frame.
log_file_path = r "C: \\ Users \\ Desktop \\ logdet14-10-11p.log"
output = r "C: \\ Users \\ Desktop \\ output \\ xl \\"
with open (log_file_path, "r") as inf:
number = 0
for line in inf: # start line loop
number + = 1
a = line.split (") # create a list from line items
cnt = 0 # counter
for i in a: # start of the cycle of cleaning list elements

if a [cnt] == '': # remove empty elements
a.remove (i)
cnt = cnt + 1 # counter
if len (a) == 34: # work only with full spims
name = a [0]
volume = a [1]
occupancy = a [2]
speed = a [3]
speed_85 = a [4]
df_main = df_main.drop_duplicates () # remove duplicates
df_main.to_csv ('logdet14-10-11p.csv', ';', encod0ing = 'ansi')
uploading data to a folder

```
with open(log_file_path, "r") as inf:
    number = 0
    for line in inf: # начало построчного цикла
        number +=1
        a = line.split('   ') # создание списка из элемнтов строки
        cnt = 0 # счетчик
        for i in a: # старт цикла чистки элементов списка
            if a[cnt] == '': # удаление пустых элементов
                a.remove(i)
            cnt = cnt + 1 # счётчик
        if len(a) == 34: # работа только с полными спимсками
            name = a[0]
            volume = a[1]
            occupancy = a[2]
            speed = a[3]
            speed_85 = a[4]
            c1 = a[5]
            c2 = a[6]
            c3 = a[7]
            c4 = a[8]
```

Fig. 11 Line-by-line processing of log files

```
df_main = df_main.drop_duplicates()
df_main.to_csv('logdet14-10-11p.csv', ';', encod0ing='ansi')
```

Fig. 12 Uploading the finished file to a folder

Machine learning is a collection of modern AI methods. AI began in the 1950s when budding computer science enthusiasts wondered if computers could "think." This task area can be referred to as "automating tasks performed by people. AI is a field that spans machine learning and deep learning. For machine learning, the dawn was in the 90s, becoming the most popular section of AI. With the advent of more powerful equipment, this trend was consolidated. Machine learning deals with large and complex datasets. Machine learning algorithms are automatically searching for data transformations into a ready-to-use form. Technically, machine learning is finding a meaningful representation of input data in a predefined space of possibilities. Machine learning is at the intersection of mathematical statistics, methods Optimization and classical mathematical disciplines have their specificity associated with computational efficiency problems and overfitting. Many inductive learning methods have been developed as an alternative to classical statistical approaches. Many methods are closely related to information extraction and data mining. Machine learning was born out of the search for artificial intelligence. In the early days of AI as an academic discipline, some researchers were interested in making machines learn. They tried to solve this problem with various symbolic methods and what they called "neural networks," which were generally percentages, and other models, mostly based on generalized linear models known in statistics.

IV. SUPERVISED LEARNING METHODS

These methods are the easiest to accomplish. They are based on a priori knowledge. Using some training data, the goal is to derive a function that makes the best possible mapping between inputs and outputs. The training data consists of tuples (X, Y), where X variables predict a specific output Y. The predicted variable Y can be quantitative (as in the case of regression problems) or qualitative (as in the case of classification problems).
The goal of supervised learning is to obtain an unknown function f given several tuples (X, Y) = (X, f (X)). The estimate (f ^) of the function f is obtained as a function that minimizes the empirical risk on the training set [6].
The empirical risk function has the following formula:

$$Remp\ (w) = 1n\sum L\ (yi, f\ ^\ (xi, w))$$

Where L is a cost function, w is a set of parameters, and xi is the data that predicts the variable yi. The function f is approximated by f ^, so:

$$f\ ^\ (x, w) = argminw\in\gamma Remp\ (w)$$

An example of a cost function in a regression problem:

$$L\ (y, f\ (x, w)) = (y - f\ (x, w))\ 2$$

V. UNSUPERVISED LEARNING METHOD

Unlike supervised learning, there is no prior knowledge in this case. Here you don't have tuples (X, Y). You just have X. The goal of unsupervised learning is to simulate the structure or distribution of the data to learn more about them. It serves both to understand and summarize a dataset. This is called uncontrollable because, unlike controlled, it tends to be more subjective because it does not have the right answers. Algorithms are used to discover and represent interesting structures in data [7].

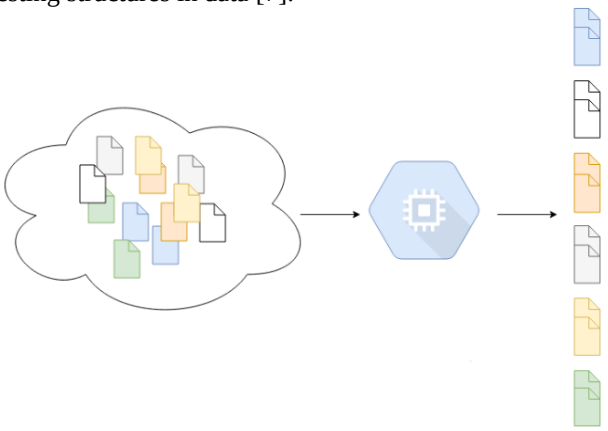


Fig. 13 Clustering scheme

In general terms, they can be grouped into clustering algorithms and association algorithms.

VI. DATA CLUSTERING TASK

The clustering challenge is a machine learning challenge that involves unsupervised learning. A clustering task gathers individual pieces of data into clusters with similar characteristics. Clustering can also find relationships in data that cannot be difficult to trace logically by merely observing the data. The clustering inputs and outputs depend on the chosen method. The clustering task refers to the unsupervised machine learning method. This is one of the learning methods in which the system itself spontaneously learns to perform the assigned task without human intervention.

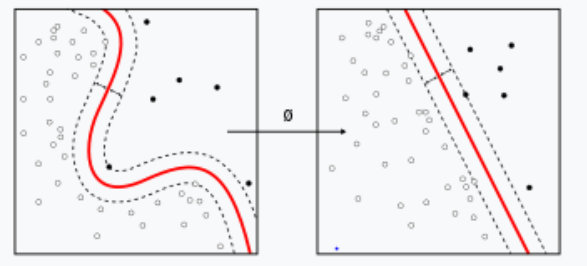


Fig. 14 Formal statement of the clustering problem

- There is no single best criterion for the quality of clustering. A number of heuristic criteria are known, as well as a number of algorithms that do not have a clearly defined criterion but carry out a fairly reasonable clustering "by construction." They can all give different results.
- The number of clusters is generally not known in advance and is set according to some subjective criterion.
- The clustering result essentially depends on the metric, the choice of which, as a rule, is also subjective and determined by an expert.

VII. APPLYING ALGORITHMS TO DATA

To work with machine learning algorithms, the IBM SPSS Statistics software (Statistical Package for the Social Sciences) was chosen - computer software used for data processing and visualization. This software has such advantages as a user-friendly interactive interface, ease of use of algorithms, and social sciences orientation. The disadvantages include the high cost of the license and the lack of flexibility in reports. SPSS can take advantage of popular machine learning algorithms such as K-Means, Decision Trees, Nearest Neighbors, etc. This chapter aims to process the available data from log files with SPSS tools to obtain high-quality results in the form of clusters - psychotypes of drivers of moving particles in traffic. Using the software is as follows: To open a file from the main menu, go to file -> Import data -> Excel / CSV

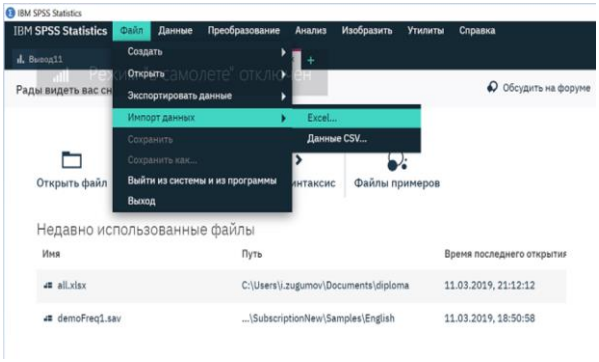


Fig. 15 Importing data into SPSS

Next, you need to select the algorithm with which you plan to process the data. You can choose an algorithm as follows: Analysis tab -> selection of the type of analysis and algorithm, see fig.16

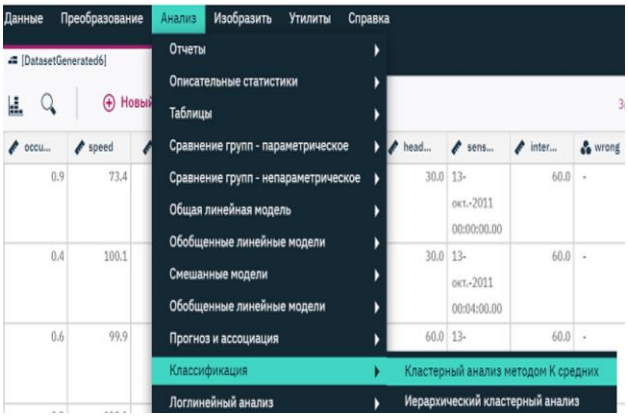


Fig. 16 Choosing an Algorithm in SPSS

When processing the data by the K-means method, the parameter of the number of clusters equal to 10 was chosen (it is assumed that there are 10-13 psychotypes of drivers of moving particles of traffic flows). The following results were obtained:

	Кластер									
	1	2	3	4	5	6	7	8	9	10
speed	123,0	18,1	40,6	45,8	102,1	141,4	63,6	80,0	135,5	120,3
gap	47,0	31,3	24,8	47,6	38,0	42,3	39,8	45,4	27,6	28,5
Наблюдения	26	10	8	14	262	2	12	171	9	61

Fig. 17 result of the cluster analysis algorithm

The result shows that there are different behavioural groups of moving particles of traffic flows. The following describes the behaviour in certain clusters: Behavioral groups of 1 and 10 clusters tend to drive at the speed limit, but if, in the case of the first cluster, a sufficient distance to the vehicle

in front is observed to allow timely response to manoeuvres, then in the case of cluster 10, the car neglects the safe distance, which is quite expected in case of non-observance of the speed limit or with borderline violation of driving. 2 It is more difficult to distinguish a cluster into a psychotypes since the driving speed is abnormally low for a high-speed lane, but in the vastness of traffic flows there are always cars moving along the lanes closest to the roadside for one reason or another.

Clusters 3 and 4 can be distinguished as 2 groups of very safe moving cars, presumably along the extreme right lane with one or another distance between vehicles. The two most common clusters are 5 and 8. Cluster 5 unites a group of drivers who observe the speed limit and measure the safe distance between their own and the front vehicle. In cluster 8, drivers are even more careful. They allow a very large distance between their own and the car in front while moving at a speed slightly below the permitted speed. This is one of the safest methods of driving in the fast lane. Clusters 6 and 9 bring together the worst offenders. In the case of cluster 6, the speed limit is violated by an average of 40 km per hour, while this distance to the car in front is safe enough, while in the case of cluster 9, the drivers not only maliciously violate the speed limit but also move at a reduced distance to the car in front, which is extremely unsafe. 7 The cluster may have united drivers driving at the maximum safe speed of about 60 kilometres per hour (the speed of passing crash tests), observing a considerable distance to the car in front. Below is a graph of the dependence of the distance between moving particles on the speed of movement, see fig. 18 [8].

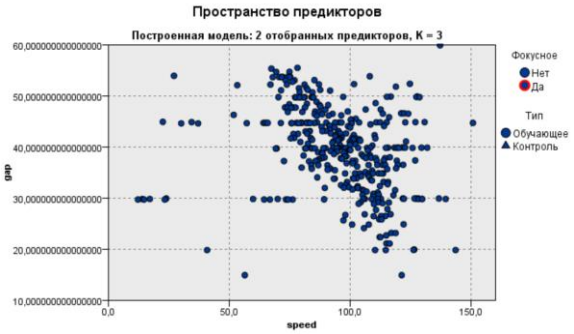


Fig. 18 Graph of speed versus distance between vehicles

Two-step cluster analysis is a tool for exploratory research and identification of groups of data, their division into clusters,Difficult to detect without the use of special tools.Categorical variables are normally distributed. This analysis can automatically search for the number of clusters. The algorithm allows you to work with large amounts of data while not requiring vast amounts of power. The best class number is determined automatically or specified manually - the procedure allows you to set the number.The work is carried out with continuous and categorical variables [9].

Based on the result of the two-stage cluster analysis algorithm, we can conclude that the K-means algorithm

	Кластеры									
	1	2	3	4	5	6	7	8	9	10
gap	40,3	50,08	35,99	42,35	30,6	47,16	23,31	45,9	27,48	29,77
speed	95,11	79,71	108,86	80,89	116,85	116,32	115,51	48,18	67,9	19,93
Набл.	118	98	89	71	69	46	38	19	11	10

Fig. 19 The result of the two-stage cluster analysis

Fig. 20 Percent of clusters formed

worked better since the K-means algorithm did not cut the "outliers" that occur as separate groups of drivers. In contrast, the two-step cluster analysis algorithm averaged the values of these groups. For example, we can give 6 clusters from the result of the k-means algorithm. A group, albeit small in number, of drivers moving at a very high speed and disregarding the rules of the road was highlighted. It is very important to single out such groups since later, this data can be used for deployment in a system that allows you to help drivers drive a car in one way or another. Otherwise, the result of the two-step cluster analysis is not so bad. Let's analyze the resulting cluster by cluster. The first cluster united a group of drivers strictly observing the speed limit and the distance between cars. In the second cluster, there is a similar group, but the participants' movement is a little more careful - the speed is less, the distance is greater.A very safe type of traffic. A not very good result can be seen on the example of the third and fifth clusters - the data only slightly differ in the scale of movement along the highway, but at the same time, it is separated. I would combine these 2 clusters into one group and call it a group of drivers close to breaking the traffic rules, which do not measure a sufficient distance between cars. The distance is not very small, but it would be much safer to move 10 meters further from the vehicle in front [10].

VIII. CONCLUSION

Machine learning opens up a completely unlimited space for humanity to improve the quality of life, optimize the business of any level, work with data, data analytics. What humanity has achieved now simply boggles the minds? The person does not take part in the selection of employees, calling a taxi. There is no need to lay routes and calculate travel times, and the wear of parts in production can always be prevented. Soon in some countries, the profession of a taxi driver may disappear because cars are already being tested that can drive autonomously, without human intervention. The future is not far off. The work considered the problem of recognizing the psychotypes of drivers - a very urgent task Today. The solution to such a situation as car accidents and fatal accidents is in the interest of many, if not everyone. There are already braking systems that can detect objects around the car road tracking systems that follow the markings using video cameras. Psychotype recognition is a great addition to these systems because it is very important to understand what the driver's next step can be to prevent him from making mistakes on the road. We analyzed 2 machine learning algorithms, namely cluster analysis - k-means.

Method and two-stage cluster analysis. It was concluded that the k-means method was more suitable for solving the problem. Acknowledgement: RFBR, project number 19-29-06036, funded the reported study.

REFERENCES

[1] Ilyukhin A.V., Chantieva M.E., Gematudinov R.A., Shukhin V.V. Cluster structures and the theory of percolation in computational materials science of construction composite materials .Bulletin of the Moscow automobile and road state technical university (Moscow Administrative Road Inspectorate). 4(27)(2011)97-101.

[2] M. E. Chantieva, K. A. Dzhabrailov, A. V. Iluhin and R. A. Gematudinov, Software Optimization Methods for Composite Materials, 2019 Systems of Signals Generating and Processing in the field of on Board Communications, Moscow, Russia, (2019).1-4. https://doi.org/10.1109/SOSG.2019.8706771

[3] Dzhabrailov K., Gorodnichev M., Gematudinov R., Chantieva M. Development of a Control System for the Transportation of Asphalt Mix with the Maintenance of the Required Temperature. In: Popovic Z., Manakov A., Breskich V. (eds) VIII International Scientific Siberian Transport Forum. TransSiberia 2019. Advances in Intelligent Systems and Computing, 1116(2020). Springer, Cham. https://doi.org/10.1007/978-3-030-37919-3_35