1 slide

Hello everyone and welcome to the Machine Learning course. My name is Alexei Platonov and I will be your teacher for the next eight lessons and practices. In this course, we will cover the theoretical aspects of machine learning and the practical ones. From a practical point of view, we will analyze the basic methods of working with data and machine learning algorithms using the Python language and the machine learning library ScickitLearn. In practical classes, you will get acquainted with this library and learn how to solve simple machine learning tasks that will become your basis in solving serious tasks at work. In the lecture course, we will learn the mathematical core of machine learning, we will examine how machine learning algorithms work from the mathematical point of view, which underlies all machine learning algorithms. The theoretical part of the course may seem a bit complicated, because it requires a good mathematical preparation and the ability to perceive a mathematical text from the student. However, in spite of this, the theory for understanding machine learning is important because it allows you to outline the boundaries of what machine learning allows you to do and what not. Also, a good understanding of the mathematics underlying machine learning is important if you need to improve an existing algorithm in a practice.

2 slide

Before we begin, let's look at the main topics that we will touch on during the course. As I said, the course is divided into two parts - theory and practice, and the goal of the course is to study the mathematical core of machine learning and gain experience in solving machine learning problems. The course consists of eight lectures and practical works. In this course, we will consider supervised machine learning and the basic paradigms underlying this class of machine learning tasks. More specifically, we will consider metric classification methods, rule-based classification, linear regression and classification methods, and Bayesian or probabilistic approach to machine learning. We will also cover topics such as evaluating machine learning models, developing features for solving machine learning problems, and also consider basic algorithms that solve the problem of automatic processing of texts. In a practical course, we will try to consider all of the above methods for solving machine learning tasks. Also, in the middle of the course there will be an intermediate test to determine the understanding of lecture and practical material and two practical works devoted to machine learning competitions where you will have to split into small teams and solve machine learning tasks, getting a larger result than other teams. At the end of the course, you will get final test in the theoretical and practical parts of the course.

3 slide

It is necessary to identifying the rules for interacting with the teacher on the course. Each week one lecture and one practical task will be upload for students. The results of the practical assignment should be sent to the teacher’s mail at the end of the week on Friday no later than 22 hours Beijing time. An intermediate test will be send as one of the parts of the practical task at 15 of Beijing on the day, and answers to it must also be provided before 22 hours of Beijing time. Variants of practical tasks and tests will be fixed to your email and the main ways of communication with the teacher is email and DingTalk. Every Monday, Wednesday and Friday from 20 to 23 hours of Beijing time, the teacher will be able to answer your questions in DingTalk chat. Also we will have consultations every Thursday at 10 hours of Beijing.

4 slide

Here is a plan of today’s class. We will observe a supervised machine learning task statement and examples of real-world tasks. Also we will learn an effect of overfitting.

5 slide

So, let's get down to the task of supervised machine learning. In general, machine learning is a branch of applied mathematics that is devoted to the development of algorithms and methods for searching for a structure in numerical data. For example, it can be an algorithm for determining the face in an image, which is represented by a matrix of numbers, each of which encodes a pixel in the picture. Supervised machine learning is a subsection of machine learning that involves a learning sample with answers to each object of interest. Machine learning considers the set of objects - this is a mathematical representation of real-world objects, expressed in numerical form. For example, these may be vectors and matrices describing a patient in a hospital. In supervised machine learning each such object of the real world is associated with a certain answer, a label that characterizes the answer to the problem being solved. For example, in the case of a patient in a hospital, this may be a label of diagnosis or a label defining a method of treating a person. In supervised machine learning, an unknown function Y is always assumed, which determines how objects are assigned labels. It is the search for this unknown function that supervised machine learning is involved in. Thus, the machine learning task with the teacher can be considered in the following form - a certain finite subset TS of objects, each of which corresponds to the answer Y, acts as input for the task. It is necessary to find a decision function that would be as close as possible to the unknown function Y(x). Thus, several questions appear immediately. First, how to describe real-world objects for machine learning in the form of vectors and matrices? Secondly, how to find the decision function? And finally, how to evaluate machine learning algorithm?

6 slide

Before moving on to the first question, it is necessary to discuss the types of machine learning tasks. First of all, there is supervised machine learning, which we examined on the previous slide. An example of a supervised machine learning task can be the task of handwriting recognition. In this a task, photos of the text are objects, and digitized texts are answers. Our goal is to find an algorithm that could translate a photo into text. Machine unsupervised learning implies that we do not have the right answers - there are only objects for processing. An example of such task can be the search for a hierarchical structure in textual data, for example, the construction of a tree of thematic sections of literary texts. Semi-supervised learning assumes that we have some kind of small sample with answers, but for all other data we do not know answers. This situation often arises on very large datasets when it is impossible or expensive to get answers for all objects. Finally, reinforcement learning assumes that the algorithm exists in some environment and interacting with it receives responses in the form of signals from the environment and reactions to its actions. A striking example of such tasks are autopilots for cars and aircraft.

7 slide

Let's look at how to mathematically describe the object of machine learning. In the task of machine learning, an object can be anything - texts, pictures, people, complex technical systems, etc. However, machine learning involves processing numerical information. Thus, an object in machine learning is always represented in the form of a vector, which is a collection of numbers - values ​​of attributes. A feature is a certain function of the learning object, which numerically describes some characteristic of the object. For example, for a text document, the feature may be the number of words in it. Characteristics can be divided into classes depending on the range of values and allowed values ​​for it. In general, four groups of features can be found — binary features, nominal features, ordinal features, and numerical features. Binary features can give only two values ​​- 0/1, -1/1, etc. Nominal features take values ​​from a finite number of integers, for example, 1,2,3,4. At the same time, the values ​​of the nominal features does not correlate with others in any way - they are just numerical marks. Ordinal features are nominal features, but there is an order relation between the values ​​of the feature. Finally, the most natural representation of a feature for us is a numerical feature, which can take any numerical value. A vector composed of the values ​​of the attributes of a certain object is called a feature description.

8 slide

It should be noted that for different tasks the same objects can be composed of completely different features. For example, we can solve the problem of medical diagnostics where the object is a person. The vector of feature in this task can be gender, age, place of residence and human temperature. In the movie recommendations task, a vector can consist of a flag for registering user in the system, age, label of a favorite genre, and user rating in the system. The types of feature are the same, their number also coincides, but their interpretation is completely different.

9 slide

The second part of supervised learning is the type of answer. Moreover, the type of answer most strongly affects the class of supervised machine learning tasks. Let's look at these types a little more. The binary classification task assumes that there can be only two answers for a task’s object, for example, 0 or 1. A well-known example of a binary classification problem is the spam filtering task in which label 1 means “spam” and label 0 means good writing. In a multiclass classification problem, the answer can take a finite number of integer values, for example, 1,2,3, etc. We have already examined an example of a handwriting recognition problem. In such task, the classification object will be one letter of the Latin alphabet from the picture, and the answer will be a number from 1 to 26, where each digit encodes one letter. There are also tasks in which an object can take several values ​​from a finite number of classes at once. Then the response is encoded by the flag vector, where each flag is responsible for its class label, and if it is 1, then this means that the object belongs to this class. An example of such classification problem for many intersecting classes is the problem of determining the subject of a text document, when the same document can contain several main topics. The task in which we need to restore some numerical function is called as the regression learning task. This is the most natural class of tasks for us. An example of such task is predicting the temperature the next day, predicting the required number of products for delivery to the store. Finally, there is a class of tasks in which it is necessary to sort the objects in a certain unknown order, which reflects the degree of importance of the object. A striking example of such a problem is the task of ranking of search results in information retrieval.

Slide 10

To better understand the types of machine learning tasks, let's look at examples of some tasks from the real world. We have already mentioned the task of medical diagnosis. In such task, the object of classification is a person who can have a very large number of symptoms so-called syndrome. The answer to the patient admitted to the hospital is the method of treatment or diagnosis that he needs to make. Examples of features may include gender, the presence of a particular symptom, such as a headache, and age, pulse, blood pressure, and severity of condition. Each machine learning task has some specifics. In medical diagnosis, a striking feature is that in this task there are often missing values ​​- not all analyzes can be collected, but for some part we need to give an answer for the patient. Another feature is that the algorithm that we get must be interpreted, that is, the doctor must understand it. Otherwise, he will not have the right to make a diagnosis without understanding the response of the algorithm.

Slide 11

In the task of credit scoring, the one who solves it is faced with the question - is it possible to give a person a loan in a bank on his profile. The questionnaire includes a large number of questions, for example, a person’s age, gender, education, how many times a person has changed jobs, whether he is working now, whether a person already has a loan and whether he has received it before, whether there is any kind of savings for a person. In this task, all the many features that we saw earlier also arise. In this problem, the question of probability estimation also arises. We want to understand whether a person can not repay the loan. It is important for the bank to assess risks in order to understand how much money it will lose. In addition to the fact that there may be omissions in the data, questionnaires that a person fills out may contain inaccurate data. A person can simply lie to have a greater chance of obtaining a loan. The machine learning algorithm must somehow be able to understand this.

Slide 12

Another interesting task is the task of texts categorization. Imagine that you are developing a system that automatically collects data from the Internet and you have a task to group all the texts of the pages in a hierarchy by topic. For example, these can be scientific texts and fiction, which, in turn, can be divided into texts in physics, mathematics, or in science fiction and historical drama. The metadata, such as the author of the text and the date of its publication, may be features in the task. But what is important in such task, the features will be the presence of certain words in the texts. For example, a feature of the presence of the word "spaceship" may suggest that the text is devoted to the subject of "astronautics" or "science fiction." A specificity of the task is that it has a very large number of such features - one for each known word useful in the task. Often such task has very small samples - we have few texts in which the subject will be immediately put down. Moreover, the document can correspond to several topics and we need to take this into account somehow. Most likely, this will require the development of separate binary classifiers for each of the available topics.

Slide 13

An example of the task of regression learning is the task of assessing the value of real estate. Imagine that you want to sell your apartment, you have several examples of flats with a known price and you trust these estimates. Now you need to name the price of your apartment. How to do it? You can look at the features with which other apartments can be described and use them to build a formula that, adding them together with some weights, will return the value of the apartment. This formula is called linear regression and it returns the numerical value of the apartment value. Similar tasks just relate to the learning of regression. A feature of this task is that after receiving some model it is important to revise it, since the cost of an apartment may change over time. Many training examples may also contain incomplete data. Moreover, not all features can be stacked in a simple scheme of weighing and adding and they must be somehow transformed in order to obtain an acceptable result.

Slide 14

An example of a task that everyone knows is the task of ranking search results. Imagine that you have a database of text documents and you want to find some information in these documents. You can review them manually, but if there are a lot of them, then it will take all your time. Based on your request, you need to somehow find a lot of documents that can at least partially satisfy your interest, and then sort these documents in the order that they fully answer your question. This is precisely the task of ranking. Features, as in the case of the task of categorizing texts, are also words here, only now these are user’s query words and we look at how often the word from query appears in the document. We can also try to evaluate the importance of a page based on how often other documents link to it. This is called page rank. The problem is that the learning set here is really huge. If we are developing a search engine on the Internet, then we will have to process huge amounts of data. So, our algorithms should be as efficient as possible. Also, a problem often arises in this problem, consisting in the fact that it is difficult to come up with new features that would improve the quality of the algorithm.

Slide 15

Let's move on to what the machine learning algorithm is all about. The machine learning algorithm is a function that takes as a parameter a feature description of an object and returns a answer in one of the formats that we discussed above. The algorithm can be either a simple mathematical function or a complex algorithm with its own logic. Examples of the simplest algorithms are a group of linear classifiers and regressions. On the slide you can see that these are algorithms that take the sum of the values ​​of all feature values of object with some weights. If we are talking about regression, then the answer obtained by the formula is the response of regression, if we are talking about classification, then in such model it is customary to take the sign of the resulting amount. For such a classifier, a negative value will correspond to a class label of -1, and a positive value corresponds to label 1.

Slide 16

As I said earlier, a function, a machine learning algorithm, can be very complex. For example, a neural network is a complex function, which is represented in the form of a computational graph that connects many functions and the sequence of their calculations and the relationship between them. But there is one object that contained in any machine learning algorithm. Any machine learning algorithm contains a set of parameters or a vector of parameters that determines the behavior of the model. Formally, the machine learning algorithm is a certain decision model G and its set of parameters that affects the model answer. The process of training a model is the search for a set of model parameters that allows you to obtain optimal values or answers ​​during testing of the algorithm.

Slide 17

Moreover, we must understand that there are countless algorithms and decision functions. On the picture, you can see the set of points that was obtained by calculating the function y(x) with the addition of a little random noise. In this problem, we know which function generated the values ​​and we can immediately learn the regression model for it in the correct form, that is, with the calculation of the sine and square of X. But, imagine that we do not know the original function, and this is exactly what happens when we want to train the model on a real task. In this case, with some error, we can assume that the first algorithm, which simply sums the powers of the number x, also works well on the shown segment. Thus, already at this stage, we can conclude that the problem of machine learning in the general case does not have a strict solution - it is always the art of choosing a model and a set of features that can quite well restore an unknown law. In a sense, machine learning is like searching for the laws of nature.

Slide 18

When we chose the machine learning algorithm and decided on its set of parameters, we must somehow find out the values ​​of these parameters. For example, in the problem on the previous slide, these were weights for various functions of X. But how do you know them? For this, there is a stage called “model training”. Formally, a learning algorithm is a function that takes a training sample and returns a machine learning algorithm. If you dig deeper, then for supervised machine learning, a training sample is a set of objects in the form of a features description and answers for them, and an algorithm is a model with known weights. That is, the process of learning an algorithm is just the search for its weights using information about the objects of the problem being solved. After we trained the model, it must be tested. Testing the model is that we select objects and answers on which the model was not trained and watch how often it is mistaken. If the error rate suits us, then we can start using this model to solve practical problems.

All this may seem very abstract, but we actually already know the learning algorithm. Remember the least squares method for linear regression from mathematical statistics - this is machine learning!

Slide 19

As I said earlier, we must evaluate the quality of the machine learning algorithm. To do this, there are loss functions. The loss function is a function that takes some object of the problem being solved, the answer for it, the machine learning algorithm, and returns the numerical value of the algorithm error. For example, if we solve the classification problem, then such a function can return 1 if the algorithm incorrectly classified the object and 0 if it is correct. For the regression problem, it turns out to be convenient to consider the square of the error, where the error is the difference between the answer of the model and the correct answer. Finally, in order to understand how much the machine learning algorithm is mistaken as a whole, we use the quality function Q, which in most cases considers the average error value on a set of objects and correct answers.

Slide 20

Once we mathematically determine the quality function on a variety of objects, we can determine the algorithm for training the model, that is, search for its weights! In fact, any machine learning task is an optimization problem for the quality function Q. Indeed, if the quality function returns the numerical value of the algorithm error on a set of objects, then we must choose the algorithm’s weights or choose such an algorithm to minimize the value of the error function. This is exactly what is written in the formula on the slide. However, as soon as we got some kind of machine learning algorithm, the question before us arises: did we manage to find the very law of nature or did we just find some optimal values ​​of the model parameters on which quality suits us? Is the trained algorithm extrapolated to new, previously unknown objects? In other words, will the value of the function Q on the new data be also small?

Slide 21

It turns out that everything is far from simple and the machine learning algorithm, like a diligent student, can simply learn answers. This effect is called overfitting. It is due to the fact that in the process of solving the optimization problem from the previous slide, we were able to find such values ​​of the model parameters that we show a small number of errors in the training sample. But if we try to apply the algorithm to new data, we will see a large number of errors. Let's look at an example of overfitting. Suppose we have a function y(x). We know her again, but we want to try to train the model of polynomial regression. In such an algorithm, the value of the answer will be the sum of the powers of X taken with some weights. Let's take the blue points from the graph as points for training, and we will check the trained algorithm on red points. The function y(x) is specially chosen so that it cannot be restored by polynomial regression.

Slide 22

Let's see what the function will look like, calculated by blue dots for different values of the model parameter n, where n determines the degree to which the polynomial is necessary for us. As we can see the polynomial of the third degree is completely different from the original function.

Slide 23

Let's try to increase the power of the polynomial regression to 10. It seems to be better….

Slide 24

Let's take the power value equals to 20. Better yet - we really draw lines well through the points of the training set.

Slide 25

With a power of 30, we almost ideally drew lines through the blue dots. But what if we try not only to connect the points along the blue points of the training set, but also see how the function will look if we compute it also in the red points...

Slide 26

Boom! As we can see, if we consider the function also in red dots, then it does not at all look like the original function. This is overfitting. We made our model more difficult, we selected good weights and she could not find the pattern that was originally laid down in the task. But how can we see this when learning an algorithm is more complicated than just a function of X?

Slide 27

To do this, it is useful to build the learning curves. This is a plot that shows how the quality function behaves on the training set and on the test set. The blue line shows the value of the quality function in the training set. As you can see with each step, it is less and less - we successfully solve the optimization problem. But, the red line, which shows the value of the error function on the test set, shows a completely different behavior - it begins to grow sharply when the power of the polynomial regression is greater than 10. This is how we can notice the overfitting of the algorithm.

Slide 28

To summarize, we can say that overfitting occurs when the machine learning algorithm is so complex and has such a large number of parameters that it can simply learn the answers of the training sample. But this does not allow us to find the law to which the data obeys. We can also define a term named as underfitting, which proves the opposite - our model is too simple and incapable of learning, since it does not have enough degrees of freedom to solve the optimization problem. Remember, it is always useful to monitor the complexity of the model and draw learning curves.

Slide 29

How can we get rid of overfitting? The first option is to somehow limit the number and values ​​of the algorithm parameters. In machine learning, this is called regularization. In the case of the previous problem, it was enough for us to limit the degree of the polynomial. Further on the course, we will talk about more complex methods of regularizing the machine learning algorithm. Also, we can define the theoretical formula of error on new unknown data and optimize it. This is an interesting and effective method, but, unfortunately, it cannot be applied to all algorithms and it is quite complicated from a mathematical point of view. In this course, we will not learn it. Finally, we can very carefully use specific training methods and work with training sets, such as cross-validation. In such methods, the model receives the training sample in parts and tries to extract the maximum information from them, and use the rest for self-testing during training. We will talk about such methods later in the course.

Slide 30

So, today we are introduced to the definition of machine learning. We examined how you can describe real-world objects with features and how you can classify machine learning algorithms according to the types of answers that the algorithm should produce. We looked at examples of machine learning tasks. Finally, we moved on to the mathematical model of the machine learning algorithm, saw that learning involves solving a mathematical optimization problem. Finally, we examined the effect of overfitting. In the next lecture, we will talk about the simplest class of machine learning algorithms for solving the classification problem, namely, methods based on the metric approach.

Slide 31

Thanks for watching. Have a nice day!