# **Python Machine Learning**

## ITEC 230 – Emerging Technologies

### Artjom Leonov

2019 - 12 - 04

## Introduction

Nowadays machine learning is finding its use in almost any industry, thus is in a high demand. It covers wide range of usage starting from something simple like a spam filter and ending with robots who are able to think and make decisions. I wanted to get a grasp of machine learning field for quite a while but it always seemed too complicated to start with. I knew that it is required to go through a lot of theory in order to succeed in this field and be able to do meaningful work, that’s why I wanted to get a sufficient learning source for this. The best source I could find was a python machine learning course by data science engineer from Amazon, which I found very informative and practical as well. The course introduces machine learning principles through theoretical explanation with the following demonstration with Python programming language. I find this approach good as it allows to apply theoretical knowledge right away and to understand the concepts better.

## Goals

The goal of this project was to get to know the base theory behind machine learning and to try to apply and experiment with as much of it as possible. Later through out the course I set up more specific learning goals that satisfy my previously mentioned initiative such as:

* Expand my Python knowledge
  + Learn to plot figures and graphs using Matplotlib
  + Learn to use scientific libraries like pandas and numpy
* Build spam filtering system
* Build movie recommender system

## Obstacles encountered

### **Excessive theory**

The course that I am following was created by a very experienced data science engineer, thus he has a good idea which topics are important to cover to get a solid basic understanding of machine learning. Correspondingly, he goes through a lot of theory introducing various scientific concepts giving life-applicable examples and going into detail on some of them. Therefore, it is pretty hard to keep in mind every principle covered in the course and it can get impossible to continue without going back and taking more time on each of the topics. Also, some of the introduced information was never applied later in the following progressions, so it is under a risk to be forgotten too. The theoretical information acquired throughout this course covers topics like deviation, probability, data distribution, correlation, regression, entropy, decision trees, etc. All of this is extremelly useful information and I will strongly enjoy applying it in my future works, just it is a lot of theory to digest.

### **Time limitation**

As mentioned before, the course I am following covers a lot of theoretical knowledge as well as many scientific Python libraries such as pandas, numpy, scikit-learn and matplotlib. The theory introduced and the capabilities of those libraries sync together extremely well and it is just a must to take a good amount of time experimenting with code snippets in order to understand the way the theory is applied. Given that, it is pretty challenging to accomplish something meaningful in the scope of this course as it last only 4 weeks. My initial goal was to get a sufficient base knowledge so I could progress based on that in the future, so I tried taking my time going through materials of the course.

## Process

### **Python**

Python is a well-spead programming language that can be applied in almost any field of computer science, this is why I was very glad that the course I am following uses this programming language to build systems that apply machine learning. I was already familiar with the language, but I have never got my hands on any scientific libraries, so this was a good experience and starting point for me. One of the libraries that I worked with is NumPy, a powerful library that adds support of multi-dimensional arrays and matrices with an excessive collection of high-level mathematical functions to manage those data structures. It allows programmers to manipulate data in very flexible ways, structure and analyze it. Pandas is another scientific library which provides programmers with some two and one-dimensional data structures like DataFrames and Series that are very useful for working with and analyzing data. Scikit-learn is a machine learning library and it features various regression, classification and clustering algorithms and is designed to work with NumPy and Pandas, making them an extremely powerful toolset for data science. Lastly, the 2d plotting library Matplotlib is an extension to Numpy which can produce figures and graphs in a variety of formats. I have practiced working with all of those libraries throughout this course and I can see so many use cases for applying this knowledge already at this point.

### **Statistics and probability**

Machine learning relies a lot on statistics principles, so it was essential to go through some basic things like mean, median and mode and learn some new, more complicated concepts. The most interesting of them were measuring covariance and correlation, which allows us to tell if attributes of 2 different elements are related to each other. It can be accomplished by applying conditional probability. There are some rules in statistics that apply only to dependant or independent components, this way we can judge if any sort of relationship between elements exists. Later I will apply this principle in supervised learning for the spam filtering system. Apart from theoretical knowledge, the course introduced an advanced visualization library Seaborn which is based on Matplotlib and can produce more attractive and informative statistical graphics like box and whiskers or heatmaps.

### **Supervised and unsupervised learning**

Supervised and unsupervised learning concepts are essential parts of machine learning today. Both of the principles are used to discover different patterns in big data that can result in finding some meaningful insights. Unsupervised machine learning is less commonly used concept between the two and it can produce surprising results, that humans normally would not come up with. Unsupervised learning allows to discover some not obvious patterns between elements as the system makes sense of the data just on itself. On another hand, supervised machine learning is much more commonly used and is more easy to get a grasp on. In supervised learning, the algorithm “learns” from examples with the “correct” answers. Later on the model is used to predict the answer for new and unknown scenarios. An example of that is a spam classifier that I implemented using Naive Bayes classifier on a big set of classified emails. The library Scikit-learn made it really simple to “teach” my model and produce good results for simple mails. However, when I split the training data into train/test subsets to see how efficient my model is on the real mails the chance of failing in identifying a spam letter was pretty high (40%), so it was a good start, but there is definitely a lot of room for improvement. Some more concepts that were introduced in this course and could be a good example of a potential improvement for the spam filter are decision trees, ensemble learning and stacking, which is a technique of running multiple different models at once and combining the results. I will definately put more time and practice into those techniques later on and will try to implement a better spam filter.

### **Recommender systems**

The main specialty of the teacher of the course I am following is recommender systems, so he gives some great insights and examples. Recommender systems try to predict what product a user would be interested in. The main technique used for those systems is collaborative filtering which can be user or item-based. User-based one compares users and makes decisions following logic like if people are similar then their preferences are similar too. This technique has some shortcomings which are met in item-based collaborative filtering technique. This technique focuses on items instead of users which makes the system work faster, harder to falsify and generally produce better results. I used this technique for implementation of movie recommender system. Pandas library makes it easy to create a correlation matrix for a big set of movies (provided by GroupLens) based on rankings from users. This matrix can be used to identify movies similar to the ones that user liked or did not like to give him good recommendations. It is impressive to see it working and being able to adjust the variables and the logic to see how it affects the predictions.

## Future work

There is a lot of things to learn in this field and I will definitely continue expanding my knowledge of it as I found it very interesting and applicable. For now, I would set my closest goals as follows:

* Recap all the learned material
* Research more on entropy, decision trees and ensemble learning
* Apply mentioned above knowledge on the spam classifier system
* Improve the recommender system by applying test/train technique to “learn” system to judge how well it finds correlations
* More research on different machine learning techniques
* Data mining research
* Research on Deep Learning and Neural Networks

## Conclusion

The main goal of the project was successfully completed, I acquired some solid base knowledge on machine learning and am ready to further develop my skills in this field. I find materials introduced in the course that I am following very useful and applicable, so I don’t regret going through all this theory and, in fact, will definitely go through the learned material again. Machine learning seems like a field with extreme potential which is rapidly developing day by day. I am glad that I had a chance to take my time learning this material from a valuable and very informative source. I am looking forward to finishing the course I am following and applying the learned knowledge in my future developments.