TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Henry Kukk 134660IAPB

PREDICTING SUCCESS TO PREVENT FAILURE IN JOOP

Bachelor's thesis

Supervisor: Martin Rebane
MSc

Tallinn 2017

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Henry Kukk 134660IAPB

JOOP AINE LÄBIVUSE ENNUSTAMINE LÄBIKUKKUMISE VÄHENDAMISEKS

Bakalaureusetöö

Juhendaja: Martin Rebane
MSc

Tallinn 2017

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Henry Kukk

15.05.2017

Abstract

The goal of this thesis is to find a machine learning algorithm among the simpler algorithms which is best suitable to predict the success of students in the course Java Object Oriented Programming in order to prevent failure.

In this thesis, multiple different machine learning algorithms are tested to figure out which one of them is best suitable to find a good match to the data. During the development process, Python language was used with Anaconda, a data science platform for it.

The conclusions and the application which was developed during this thesis can be used to pick out students who are in danger of failing the course in order to help them with extra work and suggest additional exercises.

This thesis is written in English and is 41 pages long, including 8 chapters, 7 figures and 1 table.

Annotatsioon

JOOP aine läbivuse ennustamine läbikukkumise vähendamiseks

Lõputöö eesmärgiks oli leida masinõppe algoritm lihtsamate algoritmide seast, millega oleks võimalik ennustada tudengite läbivust aines Objektorienteeritud programeerimine Javas (JOOP).

Lõputöös tesitakse erinevate masinõppe algoritmide suutlikust antud probleemi lahendamisel, analüüsitakse nende tulemusi ja leitakse parim sobivus antud algoritmide seast. Tarkvara arendamisel on kasutusel Python programeerimiskeel ja Anaconda laienduspakett sellele.

Töö käigus tehtud järeldusi ja rakendust saab kasutada tudengite nõustamisel ja katkestamise ohus tudengite välja noppimisel, et suurendada tulevikus eelnimetatud aine läbivuse protsenti.

Lõputöö on kirjutatud Inglise keeles keeles ning sisaldab teksti 41 leheküljel, 8 peatükki, 7 joonist, 1 tabel.

List of abbreviations and terms

Anaconda	Python science platform
JOOP	Object Oriented Programming in Java
LR	Linear regression
OOP	Object oriented programming
PR	Polynomial regression
Python	Programming language
Regression	Statistical process for estimating relationships
RF	Random forest
User Interface	Part of the application that the end user will use

Table of contents

List of figures

List of tables

1 Introduction

1.1 Background and problem

This thesis is going to solve the problem of predicting the result of the course according to the grades given to students. This thesis is based on the course “Object Oriented Programming in Java”. The subject deals with teaching students to use the OOP (Object Oriented Programming) principles to develop software and also introduces the students to the new features of Java 8. Since there has been a tendency for students to quit the course in the first part of the semester, after the first test, the thesis will work on trying to filter out students who are in danger of failing in order to prevent their failure. Last year, out of the total 139 enrolled students, 13 failed the course who got a result for test one. Additional 12 students did not even get a grade for the first test, they quit before the first test but had some homework submissions.

1.2 Goals and methods

The goal is to figure out who are the students that are in the danger of failing and to suggest extra work to them in order to prevent failure. We will try to find the best algorithm suitable to get the predictions and make some conclusion about it in the end. In order to do so, machine learning is being used to train the system to make predictions on whether the student would need help or not. We will be looking and the grades the student got for assignments as the data we are using to predict and some other grade, e.g. test result or failure/success as the predictable value. In order to solve this, the thesis will work with linear regression as well as with polynomial regression and random forest. In the end we will intend to be able to determine which algorithm is the best to use for the given problem and propose how to predict success.

2 Machine learning

2.1 What is machine learning

Machine learning is the subfield of computer science which deals with training systems to make predictions according to given data. There are multiple machine learning algorithms with which this thesis will also work with and try to find the best fit for the given issue. According to Arthur Samuel, a pioneer in the field, "[Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed." [1]. Alan Turning proposed the question "Can machines think?" in his paper "Computing Machinery and Intelligence" [2], in order to put it more into context, it would be suitable to replace the question with "Can machines think like we do?", that means can they make conclusions from tasks given. For us it is most clear to make assumptions according to some data given. Whether or clouds in the sky implicating that is going to rain or a cough implicating an oncoming illness. All of these conclusions in our context are made according to historical data and previous experience. The algorithm at initial run lacks any experience what so ever. That is why we are going to need to train it first. In order to train it we need data. And after that we can hope to start asking it questions on whether the grey clouds in the sky are actually a hint of upcoming rain or just atmospheric dust.

2.1.1 Machine learning uses

Machine learning can be used to solve a large scale of tasks. Whether it being the Tesla cars learning to drive better and safer in the traffic or the task in hand which deals with predicting students grades according to historical data [3]. Other uses range from Google trying to suggest sites to you [4] to Facebook suggesting what kind of pages you might like [5]. All of this magic is made possible by computers constantly dealing with data and trying to predict what would be of interest to you.

2.1.2 Machine learning algorithms

Machine learning algorithms are mainly spilt into bigger sets of algorithms. They are divided by what the outcome should be and mostly the choice boils down to a limitation of the data. Whether we have all of the predictions for each of the given rows or data, only some of them or none at all. According to that, the algorithms are split into unsupervised learning and supervised learning.

2.1.2.1 Unsupervised learning

In case of unsupervised learning, to keep it short we could describe it as working with a blindfold on. We will have a number of observations, a vector of measurements that is in accordance with all of them but we will not have an associated response [6]. That means, a linear regression model is out of the question, since there is nothing to predict. Unsupervised learning can be used to find relationships between variables or observations. For example, if some company would analyse the demographic characteristics of a group of people and try to fit them together as a group according to their spending habits. An example tool in the case of unsupervised learning is cluster analysis [6].

2.1.2.2 Supervised learning

In the domain of supervised learning the main difference is that we will have a value y for each of the observations [6]. That means, that we can construct a linear regression model to predict the outcome. This thesis will also deal with the concept of supervised learning. In our case, that means, that for each row of data (group of grades) we have an end result, what the user did get for this performance. For example, we have all the grades for the user that got 79 points on the exam and that is something we can tell the system. After that we can use a set of grades again to ask the computer, how well will this student do according to what you have learned from the data that we have given you earlier [6]. This is the basic concept of supervised learning. Probably the most popular and also the simplest supervised learning system is Linear regression, which we will also be using in this thesis.

2.1.3 Reinforcement learning

Although it is out of the scope of this thesis, it is nevertheless an important topic. This is the case which inspired by behaviourist psychology [7]. It is concerned with what the system has to do in order to get some reward. The most popular solution to this is that the software agent will have to maximise its result in the long term. For example, a really interesting paper was written on this topic by Dr. Tom Murphy VII Ph.D "The First Level of Super Mario Bros. is Easy with Lexicographic Orderings and Time Travel . . . after that it gets a little tricky" [8], where he basically uses reward system to teach the computer to play different NES console games on an emulator. In this case the rewards are mostly points that the user can collect, this is something the computer is able to understand and since it can also look to the future are bit, it can see what happens if some certain action is taken.

2.2 Important keywords

In order to understand the given thesis, we first need to set up some basics. Throughout this thesis we are going to work certain words which are defined here in order to avoid confusion in the later stages.

2.2.1 Keywords

The basis of the work is the algorithm; it describes the basic method which the system will use in order to make predictions later in the stages.

Dataset - Datasets is the amount of data that will be fed to the system in order to make decision. A dataset will have to consist of one or multiple independent variables, and one dependant variables, the dependent variable can also be missing in case we were to be dealing with unsupervised learning.

Independent variable - Variable which is used to in order to make decisions. These are in our case grades, which the student has got for homework assignments or tests, depending on what our scope of the job will be.

Dependent variable - this is the variable we are searching for. It can be anything which can be described to the computer. In our case it is the end grade of the student or test result whichever is the current scope.

Model - The result of the machine learning algorithm. This will be the outcome after the computer has done the calculations. If we are dealing with only one independent variable, we can get a graph or a scatter plot in order to see, what the system has generated.

According to that, we can actually read out what the result will be if we have some results already.

Training set - Is some amount of data that will be used to train the machine.

Test set - Is a set of data that will be used in order to test the machine after the training process has been done.

Test-train set spilt - It is a good practice to split the training and test sets somewhere around 20% and 80% in favour of the training set. If we keep the training set too little, we will end up with an inaccurate result and doing it the other way around would result in an accurate result but no way to prove it properly.

3 Technologies used

3.1 Python

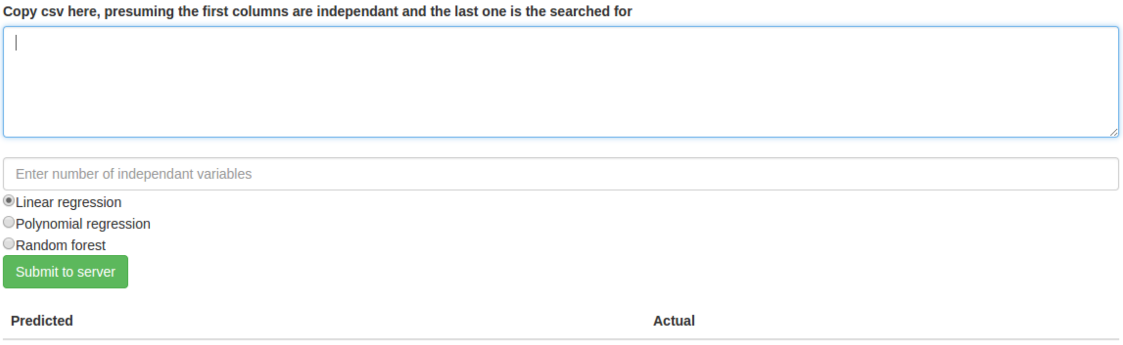
Throughout this thesis the main programming language which will be used is Python. We will use it with various packages and extensions. Python is an interpreted, object-oriented, high level programming language with dynamic semantics [9]. It is easy to get started with for people for are completely new to programming since it has a quite simple syntax and also provides the functionalities which are also used in other programming languages. Be it loops, variables or whatever one can imagine.

3.2 User interface

There is a simple web UI built using Python 3 with Flask framework, Bootstrap and CSS to allow the user of this tool to define a CSV text content with a certain format, tell the system which kind of algorithm we will want to use and the outcome will be an array of data which contains some actual value for the problem and the predicted value. It will be the test sets data which we will return. Since the scope of this thesis is not to make a product but rather to prove a point, there is not that much effort put into the generation of a beautiful UI with a rich UX.

3.2.1 Flask

Flask is a Python microframework for Python based on Werkzueg, Jinja 2 and good intensions [10]. It provides an easy way for Python to provide web application endpoints and rest services. Basically it is everything that one would need in order to build a web application within the range on what is needed in this given thesis.

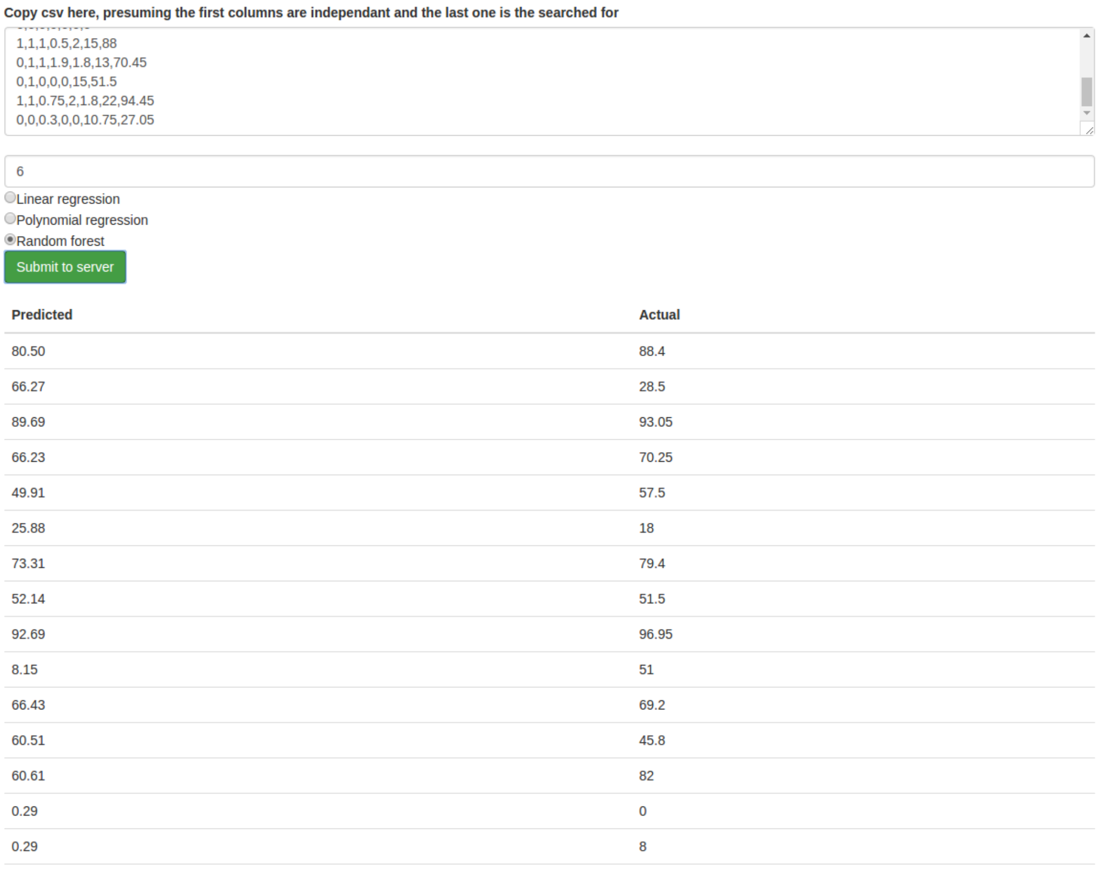


Copy csv here, presuming the first columns are independant and the last one is the searched for

Enter number of independant variables

● Linear regression
○ Polynomial regression
○ Random forest

Figure 1. UI landing page



Copy csv here, presuming the first columns are independant and the last one is the searched for

```
1,1,1,0.5,2,15,88
0,1,1,1.9,1.8,13,70.45
0,1,0,0,0,15,51.5
1,1,0.75,2,1.8,22,94.45
0,0,0.3,0,0,10.75,27.05
```

6

○ Linear regression
○ Polynomial regression
● Random forest

Predicted	Actual
80.50	88.4
66.27	28.5
89.69	93.05
66.23	70.25
49.91	57.5
25.88	18
73.31	79.4
52.14	51.5
92.69	96.95
8.15	51
66.43	69.2
60.51	45.8
60.61	82
0.29	0
0.29	8

Figure 2. UI after running algorithm with results

3.3 Machine learning

In order to use machine learning we are going to use Python and the Anaconda extension library. The main part will come from sklearn module and its subclasses. Sklearn is a set of “Simple and efficient tools for data mining and data analysis” [11]. In simpler words, in the scope of this thesis, it means that sklearn provides a set of tools to run machine learning algorithms on Python. If it is a possibility, we will also use plotlib to plot the graph and get a good overview of the data.

3.3.1 Anaconda

Anaconda is the leading open data science platform powered by Python. It contains over 100 of the most popular Python, R and Scala packages for data sciences. In addition to that there are over 720 packages which can be installed with conda, the “renowned package, dependency and environment manager, that is included in Anaconda” [12].

4 Simple linear regression

Simple linear regression is the most basic machine learning algorithm. It takes in an amount of X and Y values and makes a prediction according to them [6]. Simple linear regression is based on the assumption that there is approximately a linear relationship between X and Y [6]. Following the clouds example raised before, in simple linear regression we would only look at the percent of the sky covered with clouds and the millimetres it has been raining with the same conditions. Not considering the colour of the clouds or whether bugs are flying lower due to raised air pressure. It is effectively limited to only one X value and per one Y value. In order to deal with more than one value of X it is needed to use Multiple linear regression instead which will be handled in the later part of this thesis. As an equation it would look like $Y \approx \beta\ 0 + \beta\ 1\ X$ [6]. In our case Y would be the grade we are looking for, test one to build a certain case and X would be the points of the homework, in this case we will be using homework three and test one. We will be using the libraries embedded in Anaconda for python in order to train our machine learning algorithm with our training set and later plot graphs to analyse the results we got from the algorithm [6].

4.1 Homework 3 and test 1

First we are going to look at the homework three results and correspond them with the first test results. I am choosing homework three to make this model because the first two home works are too generic to get a picture in my opinion. The first homework was about creating classes and setters and getters. The second one was about the same but with some additional functionality.

4.1.1 Descriptions

4.1.1.1 Homework 3

The third homework was about creating a system for handling bank cards, some of them being credit cards with a limit and some of them being debit cards which had no credit money. Functions to print out transaction information and money left. It is also a good starting point because it has some similarities with the test one. Extending classes, overriding methods and distinguishing between different objects and their classes are all part of homework three.

4.1.1.2 Test 1

The basic idea behind test one was to create a system of something e.g. a system with different network adapters. Continuing with the network adapter example, first the student had to make a class for the network adapter with corresponding dummy methods, create connections, disconnect connections, check whether the adapter is connected or not, send a request and print the correct data to the console about the request and a method to print the information about the connection to the console. Next there had to be a wireless connection interface, which extended the previous class and had some additional functions in order to check the wireless connection strength. Before sending a request with the wireless adapter a check had to be made whether the connection is up and running. Next a static factory was required in order to create the adapters. There also had to be a Computer class which had one or many network adapters depending on its serial number. Setters and getters which never return null were also required. The last requirement was a shutdown method which will close all of the connections and print the information to the console.

4.1.2 Dataset

In the first case the dataset will consist of two columns and 139 rows. In the first column we will have the results for the homework three. Points ranging from 0 to 1, decimals allowed. In the second column we will have points the student got for test one. Ranging from 0 to 22 points, decimal points are allowed. We have all of the results from last year’s course and that means we are working with 139 rows of data. Currently we are taking all of the groups, mainly because simple linear regression only takes one independent variable and filtering the results out on the basis of group would make for datasets that are too few in rows to work with properly. There are 23 different values for the results of test one and 7 different values for the result of homework 3.

4.1.3 Regression on data

In order to make the first linear regression, we are setting up PyCharm to run with Anaconda in order to use the powerful statistics libraries which it provides. We are going to import the libraries and read the data from the CSV file which was exported from ained.ttu.ee. By now the dataset has been changed to match with the fact that any row, which is not present (was not submitted) is marked as 0. On the second column, the test one results have only the results which were for the first try of the test one. We do not look at the redo’s right now since because the student had more time to learn for the rework and also had already some experience with what the test might look like, the homework 3 result gets less relevant.

4.1.4 Results

After training the algorithm with the given data, that being one independent variable and one dependent variable, this is the graph that we can plot from the data.

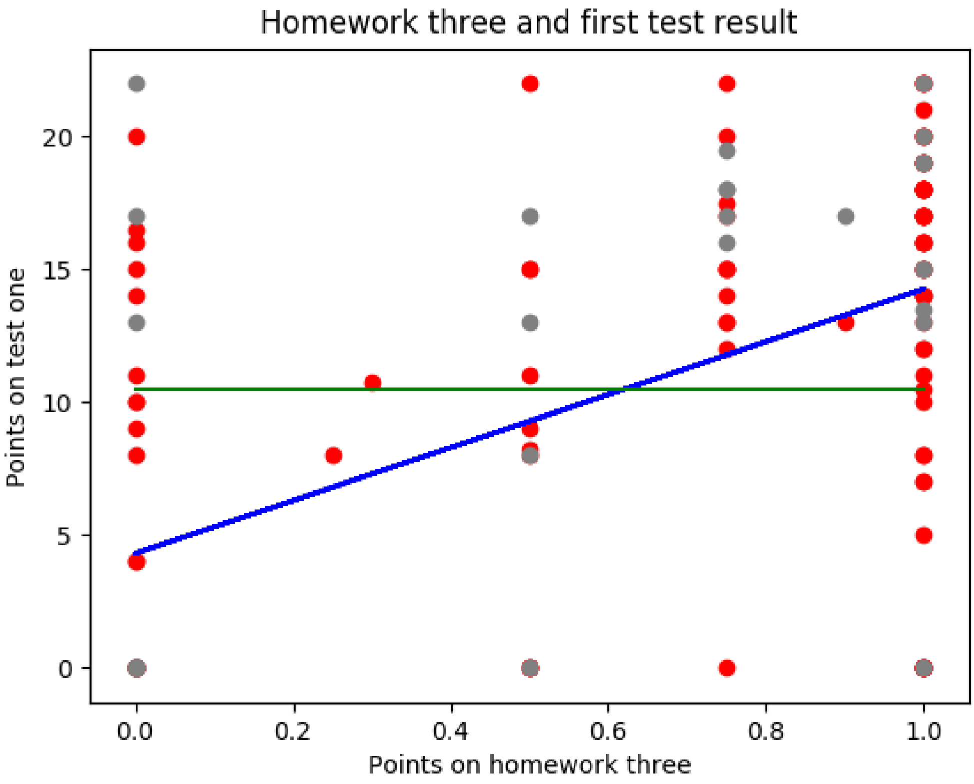


Figure 3. Linear regression. Homework three to predict test one.

The red dots represent the actual values of the training set data, the grey dots are for the test set data and the blue line is the prediction that the algorithm made according to the data that was presented to it. The green line is the amount needed to pass the first test. It becomes evident that first of student who had third homework results which were higher than 0.7 are likely to pass the first test. That is also supported by the red scatter dots representing actual data. Although there are other cases, it is necessary to understand that the homework cannot always be evaluated properly, although the idea was that all of the students who submitted a homework should have to defend it to the lecturer or the teaching assistants assigned to them, there were still cases where the students got their points from separate online grading, if there was not enough time to assess all of the students in the classroom and thus it cannot be proven that the students did the homework themselves or got help somewhere and how long did it take. Here it is important to note that it we are not currently dealing with whether the job was submitted in a rush or did the submitter still have time to spare until the deadline. The given data is analysed over all groups and times and the dataset size is 139 rows. In this case we can also note a weakness in the given dataset, that being that the grades for the first homework assignment are mostly 0, 0.75 and 1. Some of them between those values. This picture is representing the results when the algorithm is run with test data. This is now the part of the data which is not previously knew to the algorithm. As previously, the red dots represent the known data, the blue line is the prediction and the green line shows the pass level. Using a simple for loop that checks whether the user who passed the course according to the real data also passed according to the prediction reveals that the algorithm correctly gave points to 22 of the 28 results in the test set and was false on 6 occasions. That gives the algorithm as success rate of 78 %. We can say that given the fact that there is still a long way to go until the test at homework 3, it would be useful to run it through even the simple linear regression algorithm in order to seek out students who are in risk of failing the test. This current model takes into account only the first attempt grades, meaning that not all of the students who did not pass it on the first attempt never made it with the second one either, but it is a risk.

4.1.4.1 Improving

In order to get some better data, I am going to try and look at what are the weaknesses of the given dataset. First of all, as noted before the grades for the third homework only range from points coming from 0 to 1, although decimal points are allowed there still is going to be a lot of overlapping between the points of home works. That means that there will be some data which points the algorithm towards that when a student got 1 point for the homework he will easily pass and some data which points it towards that a student with 1 point did not pass the test. This will confuse the algorithm and make it harder to predict the outcome for a given number of points. In order to improve that part we are going to look at some other homework points which had a bigger range.

4.2 Homework 4 and test 1 linear regression

I will use the same test 1 as the dependent variable, the result which I am looking for and homework 5 as the independent variable, the result which I am predicting with. The main difference between homework three and homework five is that homework five was graded in the scope of 0 to 2 points, decimals allowed in contrast to homework three which was graded from 0 to 1 point.

4.2.1 Descriptions

4.2.1.1 Test 1

The description for test one can be above at the chapter about "Homework 3 and test one".

4.2.1.2 Homework 4

Homework 4 was held two weeks before the test. Although the time between homework 3 and homework 4 is only one week, since homework 4 points ranged from 0 to 2, decimals allowed, in contrast to homework 3 in which the points ranged from 0 to 1, decimals allowed, it should give a good overview on whether making the data more versatile, that being the points for homework 4 have 11 different values while points from homework 3 only have 7 different point values. With this we can check whether having a more versatile set of data will give us better predictions when talking about simple linear regression. The assignment of homework four was to learn the basics of Java 8 and its new features. That being to improve the code made in the practice lesson. The practice lesson assignment was about creating a banking system which uses static factory methods, inheritance and extending classes. On top of that, the homework was about using Java 8 Optional class to avoid null pointers and also streams to filter out given data according to some parameter given as a Predicate.

4.2.2 Dataset

The same logic was used to deal with missing data as before. If a student did not submit his work the work was counted as 0 in our case. There were 139 rows of data with two columns, first column represented the homework assignment grade given to the user and the second column showed the same test one result as used before. The points for the fourth homework range from 0 to 2, decimals allowed and the test results range from 0 to 22, decimal points allowed. There are 23 different values in the column for test 1 and 11 different values for the grades of homework 3.

4.2.3 Regression on data

In order to see the result, we are running a regression algorithm on the dataset given. The dataset is split up into two pieces, the train set and the test set with a ratio of 0.8 and 0.2 in favour of the train set. The process of running the regression in the chosen environment is more thoroughly described in the second Homework 3 and test 1.

4.2.4 Result

The graph below represents the results for the test. The red dots show the data in the training set, X axis shows the points the student got for the fourth homework assignment and the y axis show the points the corresponding student got for test one. The blue line shows the linear function the system came up with in order to predict the results and the grey dots show the data that was present in the test set.

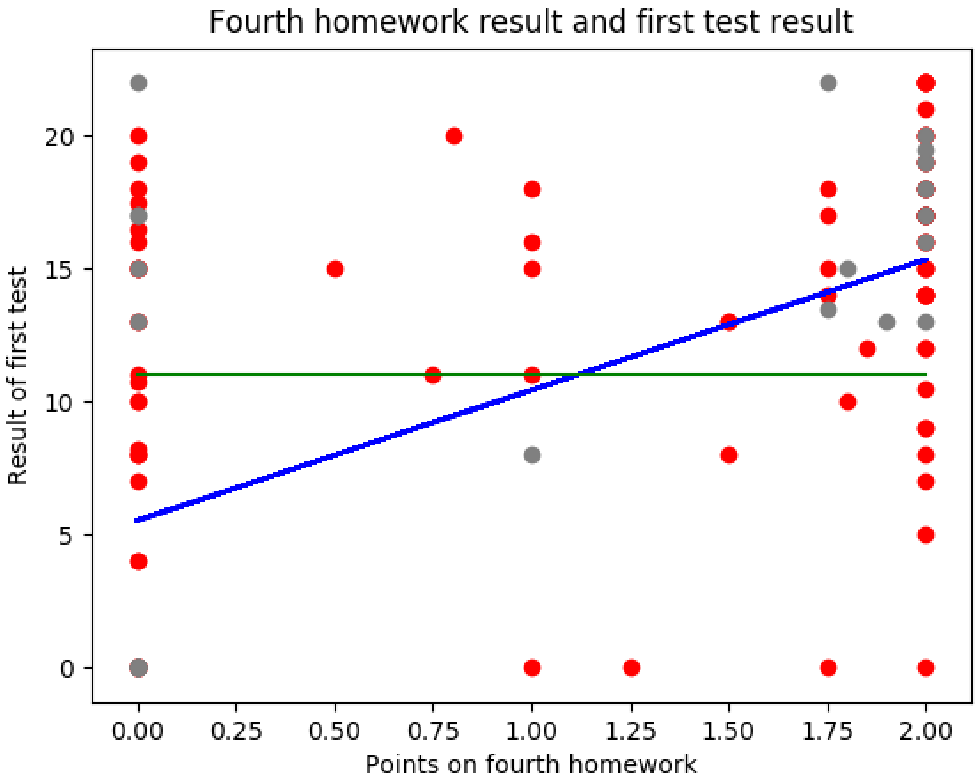


Figure 4. Linear regression. Homework four test one

Looking at the graph we can see that the linear function tells us that if the student got at least 1.1 points for the test, he is likely to pass. Here it is important to notice however that this data can be misleading to the algorithm. There are a lot of points that are actually on two ends of the spectrum. Writing a simple python for loop in order to determine how many students got 0 on their homework but actually passed the test and vice versa shows that 16 students who got 0 for their homework passed the test and 7 who got max for their homework did not. As the scatter graph shows, 0 and 2 were actually two of the most popular outcomes regarding points given to students. The statistical data, looking at the test set reveals that we now got 23 out of our 28 results in the test set correct, regarding whether the student will pass the test or not. That gives a success rate of 82% which is an improvement to the initial 78 % that we got on our first try with homework 3 and test 1. Regarding that our train set is rather small, it can be counted as a success and thus it is possible to say that if a dataset is more versatile, it leads to better results in linear regression. Given naturally that the problem one is solving actually is a linear problem. In order to further test this point, we are going to have another go at it.

4.3 Test 1 and test 2

This is the final try to predict the results of some assignments with simple linear regression, that being using only one independent variable to determine the dependant variable which we are looking for. In this case, I purpose, leaning on the last two results, that if a dataset is more versatile, that meaning it has more different values in the dependant and independent variable column and it can be described using a linear regression model, it will be better predictable. In order to test this hypothesis, I am going to run a last linear regression on data that consists of test one results and test two results.

4.3.1 Description

4.3.1.1 Test 1

Description for test one can be found at the chapter Homework 3 and test 1.

4.3.1.2 Test 2

Test two was a more complex system consisting of everything that has been learned during the course. In addition to dealing with classes and objects, there was also the need to use Java 8 functionalities in it and also to use multithreading and synchronization in order to avoid concurrent modifications. First of all, there was a base class which had some fields like name id and some other fields separating it from the other classes. Next there was a central system, which dealt with all of the objects. That means that the concept of extending classes was a necessity to use in order to pass this test. There were producers and consumers. That being there were some threads that produced the data for other threads to use. That was all being held in a central synchronized collection which was accessed by the producers and consumers. In addition, the consumer threads were different in the type of objects that they produce and each consumer also had its certain type which defined which type of data it needs ni order to fulfil its task. Of course it was necessary to end the systems work at some time so that it would not stay running for ever and that meant having a stopping condition and to kill all of the threads running at that point.

4.3.2 Dataset

The dataset consists of two columns. One representing the result for test one and the other column representing the result for test two. There are 96 (I have also sorted out students who did not get a result in the end for test one since that means they have basically failed the course and should not be taken into consideration in order to avoid confusion to the algorithm) rows of data in this particular way. If a student has no information on either of these rows, the result will be counted as 0 in order to make it readable for the computer. In this case we will also be using the results of both tests rework as well since the timeframe between the two tests is considerably large and whether the test was passed on first try is not that important in the given case in my opinion. Meaning that if a student failed the first try, the same student probably learned more for the second try, had a clearer understanding of what is going to come to the test and thus was able to collect knowledge and should be accounted with. It is also important to note that if a student took another attempt during examination session, they would only get half of the points possible. Meaning that a job graded with 17 points would give the student 8.5 points in the overall calculation. There are 16 different values for the grade of test one ranging from 10.5 to 22 points and 21 different values for the grade of test two, ranging from 0 to 21 points.

4.3.3 Regression

The running of the regression can be found in the chapter Homework 3 and test one. The test and train set spilt will be as always 0.2 to the test set and 0.8 to the train set. The only difference is that both of the variables, the dependant and the independent one have much more values in them, that means that if my hypothesis is correct, the result should be even more accurate than the last best result, being roughly 82 percent.

4.3.4 Result

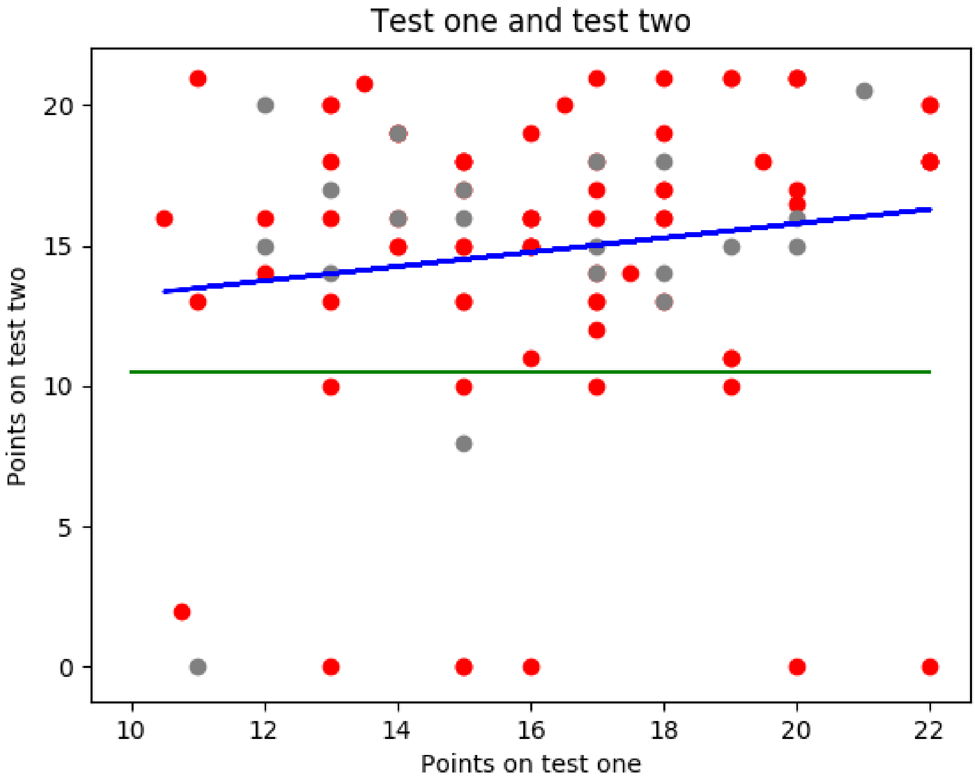


Figure 5. Linear regression. Test one to predict test two result.

In the picture above there is the result of the regression made by the algorithm. The result is quite interesting. It points out that if a student passed the first test, they are also likely to pass the second test. Looking at this picture, we also have some data that is actually 0 points for the test two and some points for test one. Analysing the results, it becomes evident that the algorithm does not deal with the problem that well. Although the analysis points out that it got the pass/fail correctly on 18 cases of 20 which it had in the test set, it would be a really good and useable result but, it is also evident looking at the graph that the linear equation never will predict failure. The blue line is always above the line of failure (the green line). So given that the student has passed the first test, the algorithm automatically presumes that the person will also pass the second one. Of course, if a student were to fail the first one, the results would probably differ but that is not a valid case in the real world, since if the first test is a failure, there is no third or fourth try.

4.4 Conclusion

To conclude the chapter on simple linear regression, it becomes evident that after the third test, it could be actually possible, if we could live with a few false positives to guide a student towards some extra work in order to prevent failure on the first test. In addition to that, the hypothesis that I purposed considering the versatility of the dataset, that the more versatile the dataset the better the prediction could hold water. Going up from 78% for the result if the independent variable has 7 distinguishable values to 82% if there are 11 different values for the independent variable, up to 90% if there are 20 some different values. Of course it is important to keep in mind that the conclusion can also come purely from the fact that if a grader has a bigger array of grades to give, the result is also more accurate. Also on the last dataset it is relevant that we are using the end result of both tests. So that in this case it is not important whether the test was passed on first or second try. Looking at only the blue line, then in the view of the model, it becomes evident that if a student passed the first test, they will also pass the second one. That is probably not a correct assumption. In order to only predict the outcome of the second test, it would be wiser to use an amount of homework assignments between test one and test two that better reflect the skillset a student has. But predicting the result of test two is not the main goal of this thesis.

5 Multiple linear regression

5.1 Multiple linear regression

Multiple linear regression, as the name suggests is a form of linear regression. In contrast to simple linear regression, multiple linear regression attempts to fit a set of independent variables to explain a certain value y on the dependant variable set [13]. If we continue with the example above, about clouds and the probability of rain, in case of multiple linear regression, we could add another variable, for example the level at which bugs fly to the equation in order to get a more accurate prediction on whether it is going to rain. Displaying multiple linear regression in a meaningful way is complex. Not because the outcome itself is complex but rather because there are multiple different variables as input and displaying them in a single graph would prove to be messy. However, that means that I will be writing more text as to explain on what is happening.

5.2 Regression to find course grade

The first run to use multiple linear regression in the form of this thesis will be to find whether the student is likely to pass the exam after giving the system all of the grades the student has got during the semester. We are going to use the results from last year's course, using all of the practical works to predict the value and the course total is what we are looking for. Here it is notable that some of the course total is made up as addition of the previous results and the exam together. So actually what this shows us is just whether the regression is able to work with the given data. It is actually not known to the system that the variable that we are searching, the dependant variable is actually a set of all of the independent variables. However, there still is the element of surprise in the aspect of theoretical works. That being tests that were taken in the lecture and those are not feed into the system to predict the outcome.

5.2.1 Preparation

5.2.1.1 Variables

In this case, since there are so many different variables, it does not make sense to elaborate much on what each of the columns is. Rather, below I have added an example row of the data that is going to be fed to the system.

1	1	0.5	1	1.8	17	0.25	2	2	2	1.6	4	13	12.5	**76.15**

Each of the columns represent a single result. The green variables represent the grades for homework assignments, the blue columns are the grades for the two practical work tests and the red value represents the exam grade. The bold column is the total amount the student got for the course in the end of the semester.

5.2.2 Dataset

The dataset for this given problem consists of 14 independent variables and 1 dependant variable. We have 139 lines of data with what we can work with and all of that is being used. Some of the given data is useless because of the dropouts, students who left the course midterm and did not get a final grade or test result. The dataset is now a mixture of columns which can range from 0 to 1 , 0 to 2, 0 to 22, 0 to 4, 0 to 21 ,0 to 28 and finally 0 to 100 for the whole grade of the subject. This should make for a quite versatile dataset and if the assumption made earlier is correct and the problem is actually a linear one, we should have quite a good result.

5.2.3 Regression

In order to run the regression algorithm, we are going to follow the same basic steps as in the previous examples. We will still split the data up into two parts, test set and train set with the sizes being 0.2 and 0.8 of the given dataset. That will give us a testset in the size of 28 and a train size of 111 rows. That should be enough to make assumptions after we have run the algorithm.

5.2.4 Result

In this case, since we are dealing with multiple linear regression, we do not have a good way of displaying the data which means, we will not have a graph. However, we still get the predictions the same way as we did earlier. The main result is that the prediction was correct in 26 cases of the total 28 student grades in the test set. Here, also being correct refers to whether the student passed the subject or not, getting grades that are higher than 50 points. We do not stress the algorithm currently to give us exact results. This gives us a success rate of 92% which seems good but it is not something that we could actually use. That is because in order to use it, we would have to know all of the grades already and that would not be much of a prediction, rather than stating the facts. In order to get some meaningful data out of the multiple regression model, we have to look at a better dataset.

5.2.4.1 Failures

The algorithm failed on two cases. The first case is where the data is following:

[1. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 18. 13] = 51

But given the points, the algorithm predicts a failure, 41.9 points to be precises. As we can see, the data in this row is considerbly misleading. Probably an edge case where the student was ill and could not participate in the course and only managed to do the first two homeworks, test two and the exam but still passed the course. The second case is following:

[1. 0. 0.5 0. 0. 10. 0. 0. 0. 0. 0. 0. 15. 7.5 = 50.5].

The student has done no homeworks but first and third, failed the first test but passed the second one. The model gives this student a points score of 45.6 points, which is actually not that far off from the actual result. But still, not correct.

5.2.4.2 Impact

It would not make sense to remove the rows from the data because both of them were in the test set. That means, they did not impact the generation of the model. That being said, since there were 2 of these rows in the test set which is 20% of the whole dataset we can assume, statistically, it is likely that there are at least the same amount of edge case rows in the training set, which impact the model generation.

5.3 Linear regression on first homeworks until test one and test one

In this regression we will use the first five homeworks and the test one result in order to predict the outcome of the course

5.3.1 Dataset

The dataset for the given regression will consist of 139 rows of data with each row containing 7 columns. The first 5 columns are the results for the homework assignments from homework one to homework 5 the 6th column is the result of test one and the 7th column represent the end result of the course. The CSV file is exported from ained.ttu.ee and the information columns, regarding the student information on who they are and other confidential data is removed since it is not necessary.

5.3.2 Regression on data

In order to run the regression I will import the train_test_split from sklearn and also LinearRegression from the same module. Running the regression will be the same as in the earlier stages when we were talking about Regression to find the grades of students according to all of the practical works.

5.3.3 Result

As earlier stated, there is no good way to display a multiple variable linear regression as a graph, since it is not a two dimensional function. However, we get the results and we can analyze them as a set. Firstly, it becomes evident that the linear regression is not too effective with this problem. The success failure ratio is now 23 correct, 5 incorrect. That leads us to a efficiency percentage of roughly 82%. That is a ratio which is informative but there ought to be a way to make the results more accurate. Failure cases. In the square parentheses are the points the student got for their assigments at hand, after the equals sign lays the actual result in the dataset and the predicted result is in the parentheses.

[0. 0. 0. 0. 0. 20.] = 28.5 (61.55)

[0. 1. 0. 0. 0. 15.] = 51.5 (40.58)

[1. 1. 0. 0. 0. 0.] = 51 (2.64)

[1. 1. 1. 0. 1. 13.] = 45.8 (56.96)

[0. 0. 0. 0. 0. 22.] = 22 (67.66)

As we can now see from the given data, two of the fail cases, marked in bold can be counted as edge cases. That means that they are not regular as to predict the skills of the student. For both of those caseses, the dataset contains enough examples to guide the algorithim to the false direction. Firstly the first row implicates a student who passed the test with max points, apart from the extra assignment, but probably dropped out or quit the course shortly afterwards. That conclsion is made by looking at the fact that the student collected only 8.5 points through the whole process of the course after test one.

The last row is an evident one - the student did not collect any points whatsoever after the first test and thus is a dropout case. However, for both of them, since the relevance of the first test is higher than the homeworks, the machine learning algorithm actually correctly presumes a high endpoint sum for the student but since they probably did not conitnue, that was not the case. Also, line 3, marked in red can be counted as another edge case. That being that the student did not pass the first test, but somehow managed to pass the whole course and get a grade after all. The model predicts an outcome of 2.64 points for the student but they got 51, just enough to pass the course. The other failure cases are quite near to the actual value and can not be counted as a total failure.

5.3.4 Conclusion

Concluding from the data presented and the analysis of the model generated it becomes evident that the problem can be solved, to some extent with a linear regression model. In the end, we got a success percentage of 82, which is a quite good percent to go with and make suggestions to make do some extra work in order to prevent failure. However, since there are other machine learning algorithms we can try out, it is possible that we can get an even better result.

6 Polynomial regression

6.1 Polynomial regression

In case of polynomial regression, we will end up with a set of values in the X matrix that is in corresponds with the degree which we have given the algorithm to generate the model on. It means that if b is a number, x is an independent variable, and the degree of the polynomial regression is two, we will end up with a model $y = b + b_1x_1 + b_2x_1^2 + .. + b_nx_1^n$ [6]. The graph of a polynomial regression will be a curve.

6.2 Polynomial regression on homework 4 and test 1

In order to show the difference between a graph of a linear regression and polynomial regression, I will start with a simple one independent variable graph which can be easily drawn and compared to the earlier Linear regression graph/model. We will run this with Homework 3 as the independent variable and test one as the dependent variable.

6.2.1 Dataset

The dataset which we will use will be the same as in the Linear regression model creation, the only difference will be that during our process, we will generate new columns to the data which will be bx^0 , bx^1 , bx^2 and bx^3. The first row is the constant row. The split will once again be 20% to 80% in favour of the train set.

6.2.2 Regression on the data

In order to run the polynomial regression, there are just a few more steps that need to be run beforehand. Firstly, we need to create a matrix of the features which also now will contain the polynomial features described in the "Dataset" partition of this chapter. Next, we will still need to create a linear regression object in order to create the model according to the data that we have newly generated. I have chosen the degree to be 3, that means that the last variable in the independent set will be x to the power of 3.

6.2.3 Result

After we have run the regression, we will be left with a model like this.

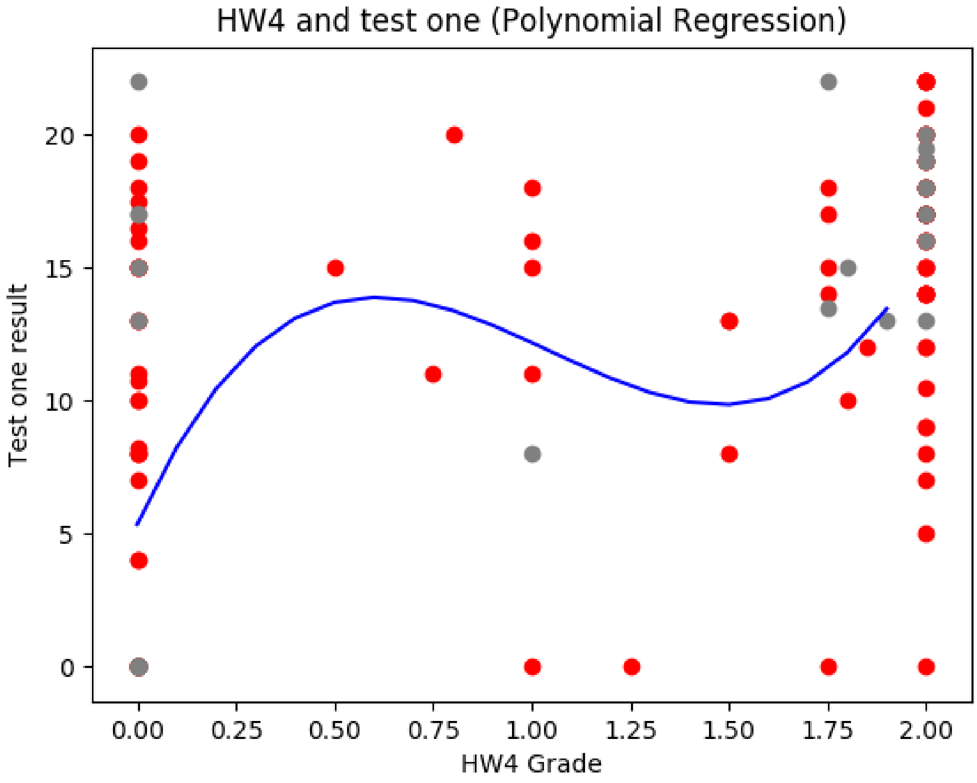


Figure 6. Polynomial regression. Homework four to predict test one.

Since the data in our actual test set is scattered around the spectrum, in order to get a good line of data, we will use the rearrange function in order to get a prediction for each of the values in the dataset. It will take the min and max in the dataset and predict a value for each of the given numbers. The step will be 0.1 points. Not doing that we would end up with a graph that has no significant value to the read from. The difference between the Linear regression model and the polynomial regression model becomes evident instantly. Instead of having a straight line as in Linear Regression model, we are now dealing with a curved line. For the given dataset, the line resembles one of a trigonometric function. This is because we do not have a single way the data is flowing. It is not a given fact, that if a student gets better grades on a Homework, they will automatically also get a higher point sum from the test. If we analyse the results with a simple loop, in order to determine what has been a false prediction and what not, we will come to the conclusion that the model is correct on 22 cases out of the total 28 in the test set. That means that we have a success percentage of 78%. That is actually lower than what we had with the Linear regression model on the same data. It would be tempting to presume that this means the

Linear regression is a better fit to dealing with this data, but it is better to try it again on multiple variables before making any conclusions.

6.3 Polynomial regression on grades until test one

This will be the final model that we are going to create in this thesis. It is to find out whether polynomial regression can be used in order to determine the end result given that we know the first 5 home works results and also the result of test one.

6.3.1 Dataset

In our dataset in this case we will be using all of the home works from one to five and also the end result of test one. Doing so we will try to predict the outcome of the course total, whether the student will pass or fail the course. The home works from 1 to 3 are graded with a result from 0 to 1, the home works 4 and 5 are graded with a result from 0 to 2 and the test one will have points ranging from 0 to 22. We will not scale the features here because it is valid to presume that the test with a higher score is more important, more relevant than the homework result.

6.3.2 Regression on data

In order to run the Polynomial regression on multiple variables, the steps are quite the same as they were on the single variable version. The only difference being that we are going to use more than one column in the X matrix representing the results for the tests. We will use a degree of 3 on the data, which means that in the end we will have x to the power of three in the train set.

6.3.3 Result

After running the regression, we will end up with a regression which we can use to predict the end result of a student's grade. We will not have a graph to display the outcome since the data we have is not displayable in a single two dimensional graph. To begin with, after running the regression on the test set and comparing the end result with the actual result we will have a miss in 5 cases of the total 28. That gives us a success rate of 82%. That is quite a good result being that the regression model we are using is quite simple and there are a lot of edge cases which will confuse the system.

6.3.3.1 Failure cases

The 5 failure cases are marked below, I will try to elaborate on them about why they have happened and how it could be possible to avoid them.

[0.   0.   0.   0.   0. 20.] = 28.5 (171.25)

[1. 1. 0. 0. 0. 0.] = 51.0 (-28.035)

[1.   1.   0.5   0. 0.5. 17.] = 69.02 (48.15)

[1.   1.   1.   0.   1. 13.] = 45.8 (52.60)

[0.   0.   0.   0.   0. 22.] = 22.0 (207.181)

There are two lines on which the outcome is understandable. Both of the lines are marked with a bold font. Both of them are really close to being correct. Being correct here meaning that the student did pass the course and the algorithm also predicts a success in the course. Analysing the other rows, it becomes evident that the other rows can be considered edge cases. Row one, the student did no home works what so ever but passed the first test with flying colours (almost a perfect result, perfect being 22), after that the student probably quit the course, at least did not get more than 8.5 points during the whole second part of the course. The second row is another edge case; the student actually did not pass the first test but somehow managed to pass the course nevertheless. This student was also talked about in the multiple variable course end result regression. The last row is another edge case where a student certainly did nothing but the first test and after that quit the course.

Looking at the predicted values in the parentheses it is also well visible that the algorithm takes a really high regard for students who did not do any home works but only did the first test. That is explainable by polynomial regression, being that the 22 points to the power of 3 will be a really large number and can affect the outcome of the model massively.

6.3.4 Conclusion

After running the regression and analysing the results it is evident that Polynomial regression, although a good tool is not powerful enough to deal with this data. We will have another go with Random forest regression in order to see whether that will give us a better result and final closure on what algorithm should be used to predict the end results of students.

7 Random forest

Random forest is another regression method which is based on building a number of trees which are bootstrapped on training samples. When we are building the decision trees, each time a split is considered, choosing the next move, a random sample of m predictors is chosen as spilt candidates from the full set of predictors p. That means that if building a random forest, at most steps the algorithm cannot choose from the majority of the available predictors [6]. It means, that to begin with, we will take a random K data points from the training set, then build a decision tree on these points. Basically, what we will do is end up with a sum of trees, which we will use to predict the value. In our case, the number of trees we will end up is defined in the code. When predicting, we will use all of the trees to make a prediction on what the Y is going to be and in order to get an answer, the algorithm takes the average over all of them. This means, that the Random Forest algorithm should be more capable when dealing with edge cases.

7.1 Homework 4 and test 1

In order to test the random forest regression, draw a graph of it to get an overview on what it is, we will start with a simple one independent variable regression. The same dataset will be used which we used in the Polynomial regression single variable example to show the graph and check which of the models is more accurate on predicting the results.

7.1.1 Running the regression

In order to run the regression, we have to call the RandomForestRegresson from sklearn module, give it a number of estimators and a random state. The train and test set size will be as always split in a ratio of 0.2 to 0.8 in favour of the training set, since we will want a model that is as accurate as possible.

7.1.2 Result

Below is the graph drawn out of results from the random forest regression. As by the usage of polynomial regression we have made a new array of values which range from 0 to 2 by a step of 0.1 points to get a good readable graph. Otherwise the outcome would be rather gibberish, since the graph would be jumping back and forth.

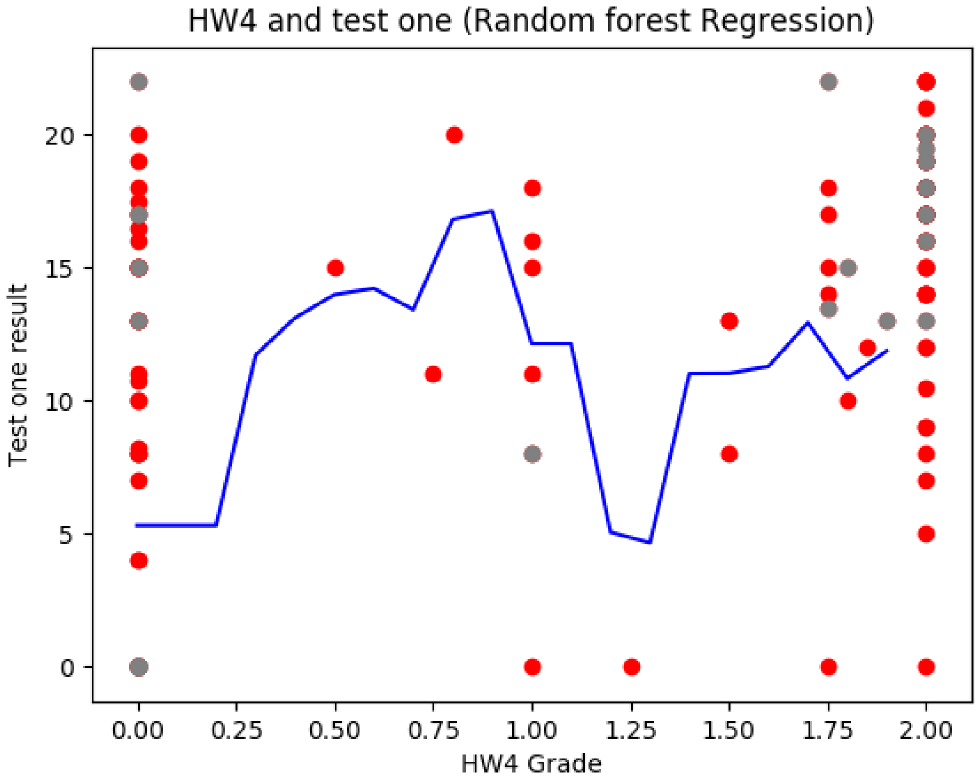


Figure 7. Random forest. Homework four to predict test one.

First of all, we will notice that the graph we get is totally different comparing it to the other graphs. It goes up and down at will and should actually be more considerate on what the outcome will be. However, when we start to look at the data, we will come to a conclusion that is completely different. Out of our 28 cases, we are correct on only 21 cases, that gives us a 75% of success. Considering we got better results with both, the linear model and also the polynomial model, I am currently not too optimistic about the fit of the random forest model to the given problem, that aside, we will still give it a go to predict the same end results as we did with the polynomial regression.

7.2 Random forest regression – First home works, test one

This model is made to figure out whether Random forest regression can be used in order to determine the end result given that we know the first 5 home works results and also the result of test one. Our aim is to get a better result than the 82% that we got with Polynomial regression.

7.2.1 Dataset

The dataset we will use will be the same as in the Polynomial multivariable regression, refer to that chapter to find information about the dataset.

7.2.2 Regression on data

In order to run the regression, we will use the same method as we did with one variable random forest regression, the only difference being that we will select more than one value in the X matrix.

7.2.3 Result

In the case of random forest regression on multiple values, we will not have a good graph to display the results. However, iterating over the results we will get a success rate of 23 of 28, the same success percent that we got using the polynomial regression.

7.2.3.1 Failure cases

```
[ 0.   0.   0.   0.   0.  20.] = 28.5 (66.27)
```

[1.   1.   0.   0.   2. 13.] = 57.5 (49.9)

```
[ 1.  1.  0.  0.  0.  0.] = 51.0 (8.15)

[ 1.   1.   1.   0.   1.  13.] = 45.8 (60.51)

[ 0.   0.   0.   0.   0.  22.] = 22 (66.69)
```

First of all, it becomes evident that most of the failure cases, apart from the row in bold are the same as in the polynomial regression part. However, looking at the estimates, it is more than evident that a random forest regression is a better suit for the given problem. The predicted values are really close to the actual values most of the cases. Apart from the two total edge cases which are the students who probably quit the course before the exam.

7.2.4 Conclusion

From this data we can summarize that random forest regression seems to be a powerful tool to solve this kind of problems. It is still not capable of dealing with edge cases, but those are probably the ones we can do nothing about. They do not correspond with the rules that are in hand.

8 Summary

The goal of this thesis was to find the best algorithm suitable to predict the success of a student in the course JOOP in order to prevent failure. In order to do so the writer of this thesis analysed different algorithms and their strengths and weakness on handling this given problem. In the beginning, the dataset was analysed in order to find the best starting point to display the graphs of the different algorithms and from there on the main focus shifted to actually predicting the results of the course according to some data given.

During the process an UI in order to enter a CSV string and define the number of independent variables was made. The UI will show a table which informs the user of the test set result. In one column, there is the actual value in the test set and in another column we keep the result the model predicted.

To sum it all up, it is evident that all of the machine learning algorithms dealt relatively well with the problem at hand. All of them could be used to suggest some extra work to a student before the end of the course in order to prevent total failure. None of the algorithms was actually good at dealing with edge cases, where the student was not in correspondence with the data at hand.

8.1 Comparison of methods

Since the algorithms percentage of success was the same, we will have to look at the results as a table to compare them with one another to see which of the algorithms was best at dealing with the given problem. Since the structure of the input data and also the output data was the same for each of them we will create a consolidated table in order to see the results. I have removed some rows which are too similar in structure in order to make the table more comprehensible.

HW1	**HW2**	**HW3**	**HW4**	**HW5**	**Test 1**	**RF**	**LR**	**PR**	**AC**
1,00	1,00	1,00	2,00	2,00	16,00	80,50	83,46	87,12	88,40
0,00	0,00	0,00	0,00	0,00	20,00	66,27	61,55	171,25	28,50
1,00	1,00	1,00	2,00	2,00	18,00	89,69	89,59	91,53	93,05
1,00	0,50	1,00	0,00	1,25	19,00	66,23	79,64	80,50	70,25
1,00	1,00	0,00	0,00	2,00	13,00	49,91	54,31	52,06	57,50
0,00	0,00	0,00	0,00	0,00	9,00	25,88	27,90	28,45	18,00
0,00	1,00	0,00	0,00	0,00	15,00	52,14	40,58	60,60	51,50
1,00	1,00	0,75	2,00	1,80	19,50	92,69	90,83	88,92	96,95
1,00	1,00	0,50	0,00	0,50	17,00	66,43	61,92	48,15	69,20
1,00	1,00	1,00	0,00	1,00	13,00	60,51	56,96	52,60	45,80
1,00	1,00	1,00	0,00	0,00	16,00	60,61	60,18	89,18	82,00
0,00	0,00	0,00	0,00	0,00	0,00	0,29	0,37	0,29	0,00
0,00	0,00	0,00	0,00	0,00	22,00	66,69	67,66	207,18	22,00
1,00	1,00	1,00	2,00	2,00	16,00	80,50	83,47	87,13	81,50
1,00	1,00	0,50	2,00	0,00	13,00	69,37	58,09	86,71	67,60
0,00	0,00	0,00	0,00	0,00	17,00	64,81	52,37	122,51	70,80
1,00	1,00	0,90	2,00	2,00	17,00	84,42	85,67	83,97	80,30
1,00	1,00	0,50	1,00	1,80	17,00	78,07	75,35	60,81	76,15
1,00	1,00	0,75	2,00	1,80	18,00	87,94	86,25	83,83	84,95

Table 1. Comparison between different algorithm's results

Looking at this table and seeking for a certain conclusion I would rank the algorithms as following: the winner, since it seems to have the least disturbance over all rows in the test set is the Random forest regression, after that comes the Linear regression and last place goes to Polynomial regression. This is certainly not to conclude that polynomial regression would be the worst algorithm, rather it is just the conclusion that for this given dataset and problem. The best dataset to use, picking from the ones that the author of thesis worked with seems to be all of the home works until test one and test one to predict the end grade. This will result a quite accurate prediction over all of the data at hand, accurate enough to suggest some extra attention to students who are in danger of dropping out.

8.2 Conclusion

As a result of this thesis, it can be concluded that the outcome of the course can be predicted using results of home works and first test as data. Random forest machine learning process is the best to use from the algorithms tested. Furthermore, it provides a good prediction on the students grade at the half-point of the semester and can be used to suggest extra work to a student. Although the algorithm is not able to deal with edge cases, it should not be considered a failure. From the view of the course at hand, it is accurate enough to suggest whether a student is strong or weak. If a student got 57 points and passed the course but the algorithm suggested 49, it should not be considered a failure, since 57 is a weak total and extra work should be considered regardless of passing the actual course. Keeping that in mind, it would not be unreasonable to expand the scope of passing to 60 points, making the success rate of the algorithm even better. But that is purely up to the person analysing the results and not the main focus of this thesis.

9 References

[1] N. Mccrea, “An Introduction to Machine Learning Theory and Its Applications: A Visual Tutorial with Examples,” United States.

[2] A. M. Turing, “COMPUTING MACHINERY AND INTELLIGENCE,” 1951.

[3] D. Smith, “Tech Insider,” Bussiness Insider, 2015.

[4] “Google Cloud Platform,” Google, [Online]. Available: https://cloud.google.com/products/machine-learning/. [Accessed 15 05 2017].

[5] Facebook, “Applied Machine learning,” Facebook, [Online]. Available: https://research.fb.com/category/applied-machine-learning/. [Accessed 15 05 2017].

[6] G. James, D. Witten, T. Hastie and R. Tibshirani, An Introduction to Statistical Learning with Applications in R, New York: Springer New York Heidelberg Dordrecht London, 2013.

[7] CTI Reviews, Psychology , Modules for Active Learning, 2014.

[8] D. T. M. V. Ph.D.*, *The First Level of Super Mario Bros. is Easy with Lexicographic Orderings and Time Travel ...after that it gets a little tricky,* 2013.

[9] “What is Python? Executive Summary,” Python Software Foundation, [Online]. Available: https://www.python.org/doc/essays/blurb/. [Accessed 15 05 2017].

[10] A. Ronacher, “Flask - Web development, one drop at a time,” [Online]. Available: http://flask.pocoo.org/. [Accessed 15 05 2017].

[11] “Scikit learn,” [Online]. Available: http://scikit-learn.org/stable/index.html. [Accessed 20 05 2017].

[12] Continuum Analytics, Inc., “Get superpowers with Anaconda,” Continuum Analytics, Inc., [Online]. Available: https://www.continuum.io/downloads. [Accessed 15 05 2017].

[13] M. Lacey, “Multiple Linear Regression,” [Online]. Available: http://www.stat.yale.edu/Courses/1997-98/101/linmult.htm. [Accessed 15 05 2017].