FAKULTA INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Název:	Strojové učení v sociodemografické segmentaci zákazníků telekomunikační společnosti
Student:	Alex Eduard Marksfeld
Vedoucí:	Ing. Mgr. Jan Romportl, Ph.D.
Studijní program:	Informatika
Studijní obor:	Znalostní inženýrství
Katedra:	Katedra teoretické informatiky
Platnost zadání:	Do konce letního semestru 2018/19

Pokyny pro vypracování

Zásady vypracování:
1. Seznamte se základními datovými zdroji a nástroji využívanými k prediktivnímu modelování sociodemografických atributů ve společnosti O2 Czech Republic a.s.
2. Proveďte analýzu v O2 aktuálně existujících prediktivních modelů věku a pohlaví na základě využívání SIM karet. Zaměřte se na zhodnocení výkonu aktuálních modelů a na sémantickou kvalitu aktuálních trénovacích a testovacích dat.
3. Podejte shrnutí hlavních problémů v trénovacích a testovacích datech a navrhněte a implementujte algoritmy na odstranění těchto problémů.
4. Navrhněte, implementujte a otestuje alespoň dva typově různé prediktivní modely věku a pohlaví založené na algoritmech strojového učení.
5. Vyhodnoťte výkon všech vytvořených modelů a proveďte jejich srovnání s aktuálně nasazenými prediktivními modely.

Seznam odborné literatury

Peng R., Matsui E.: The Art of Data Science. Leanpub, 2015.
Hastie T., Tibshirani R., Friedman J.: The Elements of Statistical Learning, 2nd edition. Springer, 2009.
Caffo B.: Regression Models for Data Science in R. Leanpub, 2015.
O'Neil C., Schutt R.: Doing Data Science. O'Reilly, 2013.

doc. Ing. Jan Janoušek, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 4. prosince 2017

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

Machine Learning in Sociodemographic Segmentation of a Telco Company Customers

Alex Eduard Marksfeld

Department of Theoretical Computer Science
Supervisor: Mgr. Ing. Jan Romportl, PhD.

May 14, 2018

Acknowledgements

I would like to acknowledge my supervisor Mgr. Ing. Jan Romportl PhD. for recommending me interesting assignment from industry and for his constructive criticism and friendly advices. I would also like to thank Ing. Ondřej Pinta for his patience and advices.

Finally, I would like thank to my grandparents, mother Ivana and girlfriend Veronika for their invalueable support during the project work.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as school work under the provisions of Article 60(1) of the Act.

In Prague on May 14, 2018 ……

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Marksfeld, Alex Eduard. *Machine Learning in Sociodemographic Segmentation of a Telco Company Customers.* Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2018.

Abstrakt

Táto práca sa zaoberá použitím algoritmov strojového učenia na klasifikáciu veku a pohlavia u zákaznikov telekomunikačnej spoločnosti. Analyzuje už existujúci predikčný model a semantickú kvalitu dát, ktorej sa to týka. Boli ukázané rozdiely vo výkonnosti a rýchlosti dvoch algoritmov strojového učenia. Ďalej sa v práci experimentuje s využitím neuronových sietí na predikciu veku a pohlavia s úplne iným typom dat, aký bol použitý pri vytváraní dvoch predikčných modelov založených na algoritmoch strojového učenia.

Klíčová slova predikčný model, strojové učenie, Telco, Python, klasifikácia, XGBoost, Random forest, spracovanie dát

Abstract

This thesis is concerned with machine learning algorithms in order to classify the age and gender of Telco company customers. It provides the analysis of already existing predictive models and of the semantic quality of data, which were used in the training of this model. Differences in speed and performance were shown between two machine learning algorithms. Furthermore this thesis experiments with using neural network in order to predict age and gender with different types of data, than the ones used for creating the two machine learning models used for trainings.

Keywords prediction model, machine learning, Telco, Python, classification, XGBoost, Random forest, data preprocessing

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Machine learning, data mining and generally obtaining knowledge from data are one the most popular areas of computer science. Converting information into useful knowledge can be helpful for example in medicine, education, aerospace engineering. The field, which does this type of things, is called data science and it is one of the most growing fields in industry.

The definition of data science is not established yet, but generally speaking, it is a concept that unifies machine learning, statistics, data analysis and related methods in order to recognize patterns in a dataset and get actionable insights.

Establishing data science teams is becaming more and more popular in the recent years. These teams showed in the past, that they can convert data into something valuable for the company, such as money or social benefit. Predicting the nearest future in the business of some company is definitely one of the biggest assests the company can have from the strategic perspective.

The practical part of this bachelor thesis is dedicated to Telco company in order to use these models to score its own clients. Predicting sociodemographical information such as age and gender helps to improve products and services offered by this company, which will result in better satisfaction of its clients.

The actual predictive model for age and gender used in production was trained before three years ago and in recent months has failed to provide sufficient results. That is one of the major reasons why I have chosen to engage myself with the problematics in my thesis, besides building a new model writing a concise survey of the theory behind all of it, which would be a good source of information and knowledge about this issue for my collegues.

The main goal of my thesis is to create a prediction model, which will be capable of running in production. Likewise, the theoretical part of this thesis will also be helpful in improving and understanding similar models.

The text of the thesis will contain extensive description of data sources, issues in training/testing data sets and models created in the practical part

of my research.

The objectives of this thesis are to find out, what sort of data is the most important when predicting the age and gender of customers. The aim of the practical part is to design a stable, fast and accurate predictive model that will be used in the production of Telco company. One of the main objectives is also to find out, what kind of tools are the most powerful ones in terms of speed, reliability and maintainability.

Chapter 2

Machine learning

Artificial intelligence, abbreviated as AI, is a concept which is now overhyped and used by many companies just to attract attention. People are usually afraid of artificial intelligence, because they think it will damage us. But the majority of artificial intelligence researchers are working on projects, that are not meant to be destroying mankind. In this thesis, I will focus on machine learning, a part of artificial intelligence, sometimes also called narrow artificial intelligence. The term narrow in this case means, that it could solve just narrow set of tasks in contast to general artificial intelligence, which is basically a machine, that could apply its intelligence to solve any problem, rather than a specific one.

People are not able to remember, preprocess and store as much data as machines can and that is the major reasons, why we are using machine learning methods these days.

In this chapter, we will discuss popular supervised machine learning methods usef for probem classification and classification by itself. As I mentioned earlier, machine learning methods are not meant to teach a machine to think independently and solve any task given. These methods are using mathematics to extract not just information, but also knowledge from the given data.

Most computer algorithms are based on deterministic decisions, which means, that on the same input, an algorithm returns the same output everytime. One would say, that this behaviour is necesarry in any "good" algorithm. But this type of behaviour is not always desirable.

For instance, if we own a company that sells cars, we always want to find the best car for our customer. This is the moment, when "classical" algorithms stop working, because everyone has different preferences, amount of money, shopping history, etc. We would like to adjust to customer preferences in order to sell them the best car that fits them in the most efficient way. And this is the reason, why machine learning algorithms are used so extensively – they can manage tasks like this effectively.

The main principle of machine learning algorithms is the possibility of

adopting them to actions, that are performed so as their precision is the best possible. By precision, we mean in what way the actions performed are different from the actions, that should have been performed.

Machine learning merges a lot of concepts and ideas such as neuro science, mathematics, physics, etc., in order to achieve best results possible in the shortest time. For instance, a common and now widely used machine learning concept called artificial neural networks is heavily inspired by human brain. The artificial neural network is composed of neurons, which are mutually connected. The communication process between neurons is by signal, which are transformed by certain transfer function. We will discuss artificial neural networks in Section 6.

2.0.1 Types of machine learning algorithms

Machine learning algorithms are divided into four different categories according to how they learn and how they improve themselves.

- Supervised learning
- Unsupervised learning
- Reinforcement learning
- Semi-supervised learning

Each of the types is used for its own purposes. Specifically, tasks of supervised learning are divided into two categories.

- Classification
- Regression

We will discuss the classification in Section 2.2.

2.1 Supervised learning

Algorithms based on supervised learning are given a training set, where each pattern has its right answer. Algorithms then try to learn this training set in order to provide answer to an arbitrary input. In reality, there is no perfect classifier/regressor, but we want to find the best one. Let then f be the classifier, Θ the optimal parameters and y the vector of classes. The loss function compares each true class y with each estimated class $\hat{y}$. There are many loss functions.

2.2 Classification

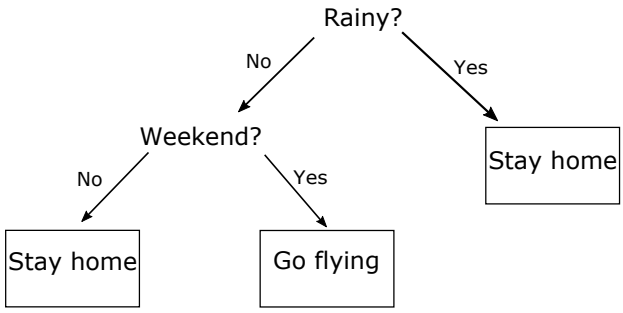
In general, classification is a distribution of output into separate disjoint sets called classes. There are two types of classification. When data are given as a set of observations without label or class and we want to find, if there are some clusters, it is called unsupervised learning. On the other hand, when we are given some data with predefined classes, our goal is to establish a pattern to classify a new observation into these predefined classes. This is also called supervised learning, which we will discuss later on. Because of the task of the practical part of my bachelor thesis, I will only focus on supervised learning and when I say classification, I mean the supervised approach.

The process, where we want to predict continues values instead of discrete, is called regression. For instance, Linear Regression [1] is a very popular regression method.

2.3 Decision trees

Imagine you want to fly your brand new Cessna at the weekend, what set of factors will affect your decision whether to go or not to go? The first one would be weather, the second one could be money, etc. The decision process would look like the graph in Fig. 2.3.

Figure 2.1: Test and training error as a function of model complexity



This method is called decision tree and it is one of the most popular machine learning methods. Decision trees are nonlinear method for both classification and regression problems. We can image a decision tree as a directed acyclic graph, but practically, they just split the feature space into a set of rectangles. The nodes in DT are divided into two categories. Internal node test values and leaf nodes specify output class or value. The decision process is made of a sequence of tests in these internal nodes and the decision by itself is in a leaf node.

[1] Linear Regression is a model and is exaplained here: `http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm`

Their main advantage in comparison with other machine learning models is the ability to interpret results which is a great feature for business cases. For example, if your manager asks you to clarify your decisions, you would rather give him a decision tree instead of weights exported from your artificial neural network.

On the other hand, it also has certain disadvantages, such as overfitting. Tendency to overfitting among decision trees is in inverse proportion with the size of the dataset. That means, that if the training dataset is small, the decision tree is very likely to overfit and vice versa.

There are many popular methods for building such trees, CART (classification and regression trees) and C4.5 are two of them.

2.3.1 Pruning in ecision trees

Pruning is a technique used mainly to increase the speed of result evaluation and to avoid overfitting of the decision tree. Like Eaton said: "Pruning of the decision tree is done by replacing a whole subtree by a leaf node" [11].

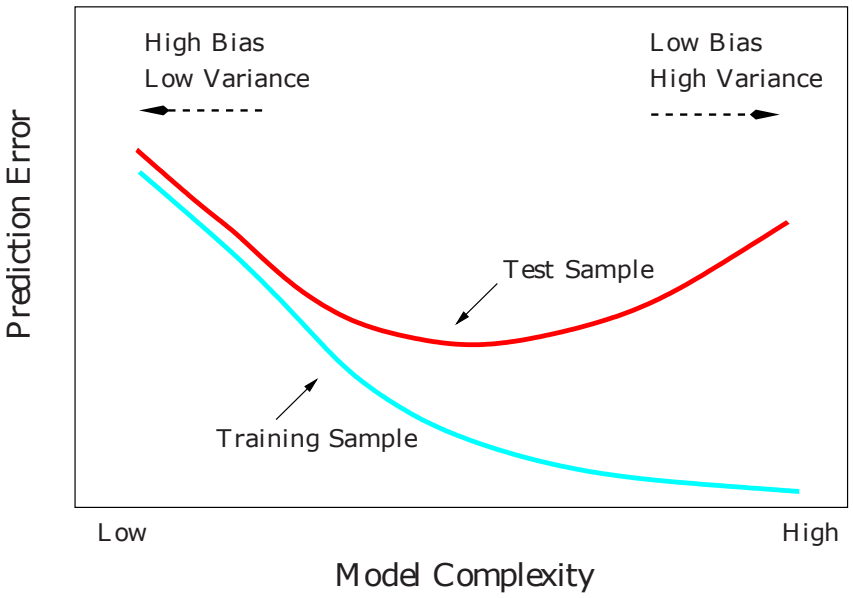
2.3.2 Information theory in Decision Trees

In order to have a proper definition of decision trees, we also want to know, how do we choose which attribute to split at each node. The answer is to find the feature that best splits the target class into the purest possible children nodes and the measure of (im)purity is defined in information theory. Information theory was originally used for compressing signals, but now it is a part of some machine learning methods, including decision trees.

2.4 Bias and variance of a model

Usually, decision trees have low bias, but high variance. In Section 2.5, we will talk about lowering variance while keeping the bias also low. Bias occurs when an algorithm has limited flexibility to learn real patterns in the training dataset. In other words, it is also known as underfitting. Bias could be also explained as how much the average accuracy changes with training dataset change. Variance, on the other hand, refers to the sensitivity of a model to specific training datasets. Having model with low bias and high variance, or having model with high bias and low variance is pretty usual. Let's say, we have two models – one with low variance and the other one with low bias. The first one, let's call it A, is based on regression. The second one, let's call it B, based on decision tree. How do we find the perfect trade-off between bias and variance? There is no right answer, but we can do some operations on these algorithms, that will make them probably more accurate. For example, the second model, B, can be pruned in order to reduce complexity.

Figure 2.2: Test and training error as a function of model complexity; source: [5]

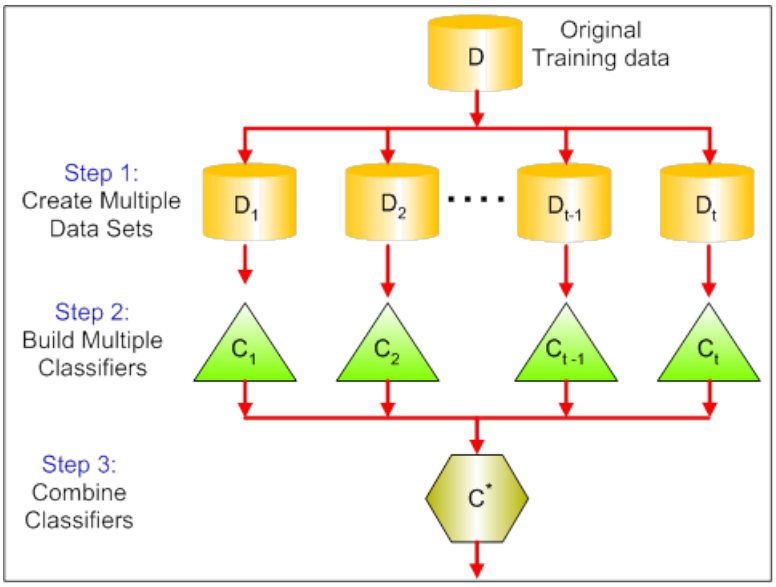


For example, if we had a completely unbiased model, it would fit every point exactly as it is. We do not want this kind of behaviour, because when we will test the model, it will give biased predictions based just on the training dataset.

2.5 Bagging and boosting

Bootstrap aggregating, abbrievated as bagging, is an ensemble technique used to average noisy and unbiased models in order to create a model with low variance. Let X be a training dataset with n samples. Bagging is a method that creates k bootstrap samples called $D_1, ..., D_k$ with a size of n', from the training data, where $n' < n$. It later trains distinct classifier on each D_i for i=1, ..., k. Finally, it classifies new instance by majority vote or by averaging, where $n' < n$. So basically, we learn models independently and merge the individual results into definite result.

Figure 2.3: Example of bagging model; source: https://www.analyticsvidhya.com/blog/2015/09/questions-ensemble-modeling/



Boosting is an ensemble technique where models are not made independently but sequentially. This means, that each model is dependent on the previous one, except the first one, of course. It applies the logic, that each model should be aware of mistakes that its predecessors had made. It converts many "weak" learners into a complex model.

2.6 Random forest

Random forest algorithm is based on bagging, which means that it grows many decision trees and then joins them into one ensemble model. It contains two sources of randomness – the first one is bagging and the second one is random input vectors. Random vector method means that at each node, the best split is chosen from a random sample of m attributes instead of all of them. Each generated decision tree is trained on a bootstrap sample of training data and each bootstrap sample contains a subset of input attributes. The use of the Random forest method with our hardware resources is very handful, because this method could be easily parallelised - each tree can be processed at the same time. For example – creating such model on a dataset containing approximately 1 milion rows and 100 columns lasts about 4 minutes on Edge[2], which is really fast, in contrast with other machine learning algorithms.

[2]Edge is a Hadoop node, mainly for computing node in the Telco company

2.7 Gradient boosting

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It is based on boosting technique, so the main idea is similar – build many weak learners sequentially and join them into a complex model. Usually, gradient boosting is used with decision trees and then they are called Gradient Boosted Trees. Gradient Boosted Trees use loss function and regularization to find out the best parameters in a given dataset. Regularization term tries to control the complexity of the model in such a way, that minimizes the risk of overfitting. Loss function computes the difference between the actual output and the predicted output. There are several loss functions, but I will only present here the Mean Squared Error. If we add training loss and regularization, we get what is called the objective function.

$$obj(\theta) = L(\theta) + \Omega(\theta) \tag{2.1}$$

where loss function can be defined as the Mean Squared Error

$$L(\Theta) = \sum_{n=0}^{N} (y_i - \hat{y_i})^2 \tag{2.2}$$

Random forest and boosted trees are basically the same model, they are just trained in a different fashion.

2.7.1 XGBoost

XGBoost stands for eXtreme Gradient Boosting and it is a distributed implementation of gradient boosting with emphasis on efficiency, flexibility and portability. It provides parallel tree boosting and in comparison with other gradient boosting implementation, it is a lot faster. I decided to use this model because of its great performance, execution speed and memory efficiency. Those two factors are extremely important not just in our case, but in general. This implementation has received a lot of attention during last few years thanks to its performance in Kaggle[3] competitions. In Section 5, we will discuss its performance in our case.

[3]https://www.kaggle.com/ - online platform for data scientists

2.8 Performance evaluation

When building a predictive model, we want to know, how accurate it is. Classification consists of two phases:

- Model creation
- Model evaluation

The goal of predictive modeling is to build a classifier with the best evaluating phase. In order to test a classifier, we need to split the training dataset into training and testing datasets in some suitable ratio. I will use a common 80:20 ratio throughout the work. If the training dataset would be smaller, the predictive model could not capture all the patterns in the whole dataset. On the other hand, if the training dataset would be way larger than the testing dataset, for example 95:5, the error would be smaller.

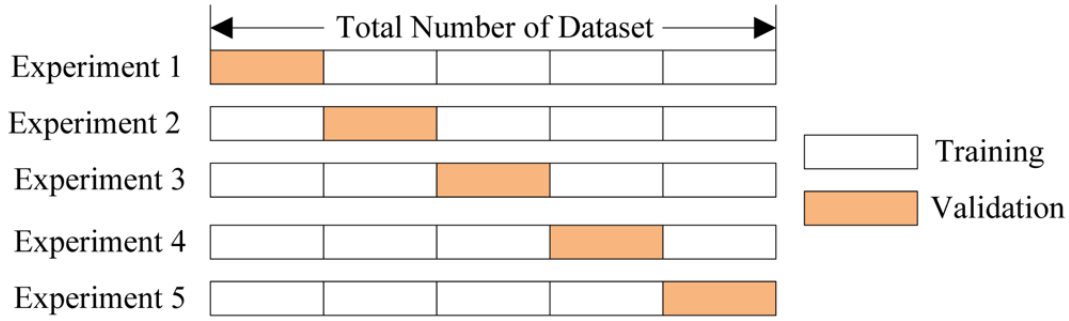
2.8.1 K-fold cross validation

Another way to evaluate a predictive model is by using K-fold cross validation. It is a process where dataset is split into K equal parts and then K iterations follows, where each iteration is made from two steps:

- The model is trained using $K - 1$ folds
- The trained model is validated on the remaining part of the data

Visualisation of K-fold cross validation is shown in 2.4. In this case, the K-fold cross validation was implemented using library *scikit-learn* and K was set to 10. The performance in 5.1 and 5.4 is a mean of cross validation accuracy.

Figure 2.4: 5-fold cross validation; source: `https://www.kaggle.com.dansbecker/cross-validation/code`



2.8.2 Confusion matrix

Confusion matrix is a method for performance evaluation that I have used. This method is mainly used for binary classification, but can be extended for multi-class problems.
"By definition a confusion matrix C is such that $C_{i,j}$ is equal to the number of observations known to be in group i but predicted to be in group j." [14] Basically, rows are representing predicted values and columns are representing actual values.

Table 2.1: General confusion matrix

		Actual values	
		Yes	No
Predicted values	Yes	TP	FP
	No	FN	TN

Values in 2.1 are:

- TP - true positive - predicted Yes and are actually Yes
- FP - false positive - predicted Yes but actually No
- FN - false negative - predicted No but actually Yes
- TN - true negative - predicted No and actually No

I will use relative values in confusion matrix. In age confusion matrices, dimensions will not be 2x2, but 5x5 because there are 5 age categories.

Chapter 3

Understanding data

Every supervised predictive model has to have a training dataset. It is essential to have reliable traning dataset, because everything derives from it. When the training dataset is not coherent and does not provide values representing reality, than we can not expect a predictive model to produce relevant results.

The Telco company provides a lot of data sources, but I will only focus on data sources used in the Data Science Team. Those sources contains just relevant data and are not pointlessly filled up with non-sense data.

3.1 Data flow

We will discuss data flow – how data come from our customers to our data warehouse.

From users' phones, data travel to so called BTS (base transceiver station) where they are stored and from there, they proceed to ETL servers which then send the data into our warehouse. From BTS to ETL servers and from ETL servers to our data warehouse, data travel by batches within defined unit of time.

Then, we also have our own preprocessing phases in the data warehouse as well. Those phases are necessary in order to store this data in reliable, maintainable, structured and consistent form. At the end of those preprocessing phases, we have a database which roughly corresponds to a third normal form[4]. A third normal form in data warehouse is not common, but Telco company has enough computing power and it provides some required security features.

[4]Third normal form is a normal form used in database normalisation. It is described here: http://courses.cs.vt.edu/ cs4604/Fall08/lectures/lecture15.pdf

Basically three kinds of data are used in Chapter 5:

- Network signalization
- Usage data
- Customer data

Network signalization is transfered directly from network elements to Big Data platform every 5 minutes and contains data from SS7[5] procols.

Usage data comes from billing mediation platform and contains information about sms, mms, calls and data usage. It is transfered in daily loads into our data warehouse.

Customer data contain for example information about customer's device. The Big Data platform is recieving this data from internal CRM systems and in daily loads.

3.2 Data sources and platforms

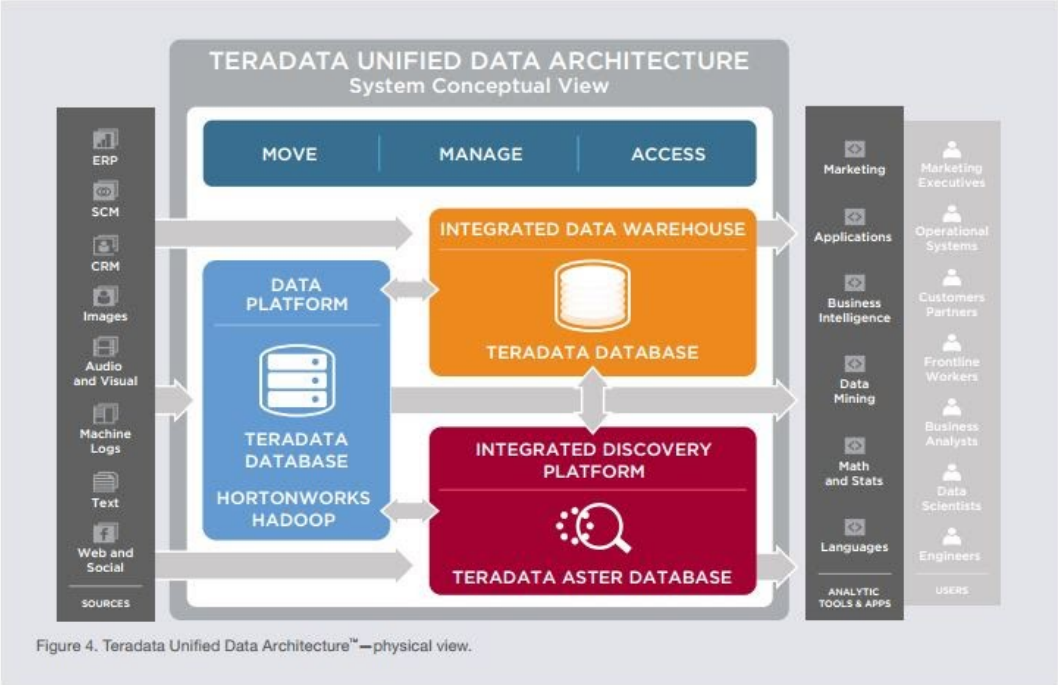
3.2.1 Teradata UDA

Teradata Unified Data Architecture is composed of three parts. From physical point of view its parts are: Hadoop, Teradata Aster and Teradata Integrated Data Warehouse.

The main purpose of Teradata Unified Data Arcitecture is to have maintainable and reliable architecture. Also, one of its values is to convert data into useful, actionable insights.

[5]SS7 means Signalling System No. 7 and is described here: `https://en.wikipedia.org/wiki/Signalling_System_No._7`

Figure 3.1: Teradata Unified Data Architecture; source: `https://www.informationweek.com/big-data/big-data-analytics/16-top-big-data-analyticsplatforms/d/d-id/1113609?piddl_msgorder=&image_number=17`



3.2.2 Hadoop

Apache Hadoop is an open-source software framework for distributed and scalable processing of very large data sets which implements the MapReduce programming model and is heavily inspired by Google File System (GFS). Apache Hadoop contains its own scalable and portable file system called Hadoop Distributed File System (HDFS) which provides high-throuhgput access to application data. HDFS uses TCP/IP sockets for communication between nodes and runs on native filesystem (Ext3, Ext4, XFS). Apache Hadoop was designed mainly for batch processing, not for streaming. We operate on running Hortonworks Data Platform 2. Hadoop in our computing environment is mainly intended to process our largest data – web traffic. We will not discuss web traffic usage, because primary data source for building this model is SIM card usage.

In the Telco company, there are multiple sources of data, but not every data source is suitable for modeling. We have a database structure intended to be a consolidated database source for building any model or preparing any ad-hoc analysis. We call this database OneTable.

It contains basically two parts – data warehouse and Aster part 3.2.3. Both parts are Teradata[6] databases.

3.2.3 Aster part

Aster is multi-genre advanced analytics platform. It allows its users to execute variety of things including R packages. Aster is capable of processing massive volumes of data and includes 3 analytic engines(SQL, MapReduce, Graph).

The main goal of Aster workflow is to establish new master entity, which roughly corresponds to a single customer. This entity can be further contacted with some offer based on its behavior. Aster part also has one table called OneTable which has every important attribute of a person.

3.2.3.1 Predictors

The number of predictors stored in OneTable exceeds 200. I will describe only those, that will be used in Section 5. Number of calls, duration of calls, number of incoming/outgoing text messages, amount of downloaded data in Mb, amount of uploaded data in Mb, connection lifetime is a subsample of usage data. Columns calculated from customer data are for example names of devices, spend of the customer, commencement date of the current contract, amount of the highest debt within last 12 months. Many predictors are calculated from the basic ones, for example average credit spending or average data usage during workweek.

3.3 Tools used for predictive modeling

3.3.1 Programming languages

The most widely used programming languages for predictive modeling in the Telco company are definitely R and Python. R is an interpreted programming language widely used among data scientist, statisticians and data miners around the world. It is a free software available under GNU General Public License for variety of operating systems.

The second programming language in the Telco company is Python, which I chose to do my practical part of the thesis about. We will discuss Python in separate subsection 3.3.2

Also, the actual predictive model for age and gender prediction, which we will discuss in Chapter 4.3, is written in R.

[6]Data and analytics solution, described here: https://www.teradata.com/

Other tools used for predictive modeling and continous integration are H2O[7], PySpark[8], the programming language Julia[9] and Gitlab[10] as a Git web-based repository.

3.3.2 Python

This subsection deals with programming language Python and libraries that have been used for implementation purposes.

Python is, like R, an interpreted programming language. Unlike R, Python is a more general purpose language. It is popular not only among data scientists, but also among web developers, ethical hackers and so on.

Python is open source and has broad community of active users. This helps with developing third-party libraries. Scientists are also taking care of these libraries and then use them for scientific purposes[1].

The first library used in the practical part of my work is a library called *NumPy*[2]. This library provides efficient manipulation with high-dimensional homogenous matrices, random number capabilities and tools for integrating C/C++ code. The kernel of this library is written in programming language C, which results in great computing speed.

The *Pandas* library offers data structures and functions intended for manipulating huge datasets. Basic structures are Series and DataFrame, where Series work with one-dimensional and DataFrame with two-dimensional data. These structures are based on NumPy library because of its performance. Pandas also offers effective reading and writing into the file, which I used in the practical part.

The library used for predictive modeling is called *scikit-learn*. The library provides simple and efficient tools for data mining and data analysis. It is based on *NumPy* and *Matplotlib*. What I really appreciate in *scikit-learn* is that it provides its own datasets for predictive modeling. This library was used for creating the random forest predictive model.

Matplotlib is a library used mainly for visualizing two-dimensional data. The library is written in Python and provides object-oriented API, which is easy-to-use. The visualisations in my thesis are also written in *Matplotlib*.

[7] H2O is an open source software mainly used for data analysis and is described here: https://www.h2o.ai/

[8] PySpark is Spark API for Python

[9] https://julialang.org/

[10] https://about.gitlab.com/

Chapter 4

Analysis of the existing predictive model

The actual predictive model used in production was written about three years ago. Within those three years, patterns in calls, texting and data usage have changed a lot. Usage of sms and calls has decreased whereas mobile data consumption has increased significantly. This information requires the Telco company to create a new model, which will recognize those new patterns in the behaviour of the users. Also, when GDPR[11] was released, some security issues occured.

From now, I will be more specific about the dataset. I will explain some necessary terms and what transformations did the author of the current model with his training and testing dataset in Subsection 4.2. I will also show the mistakes that have been made and how they should be avoided in Section 5.

4.1 Training and testing dataset

Back in the days, the Data Science Team in the Telco company was not using any versioning tools and the author of this model trained and tested the model on his own computer device. Unfortunately, right after creating this model, his computer crashed and all the training and testing datasets was gone. The source code written in R and SQL has been saved because they were stored also on our computing node, however the datasets were not. Versioning of files and workflow automation is critical even in the Data Science Team and nowadays it is perfectly set-up. But claims about training and testing dataset therefore cannot be verified.

[11]The General Data Protection Regulation is a regulation in European Union legislation and its details can be found here: `http://data.consilium.europa.eu/doc/document/ST-9565-2015-INIT/en/pdf`

I will be using the word party often from now on, so I would like to explain, what I mean by a party.

Definition 4.1.1. Party is a group of people sharing one billing information.

For instance, when a family consisting of a parent and two children are clients of the Telco company, they are considered a party. Also, when the parent is the payer of this party, we have information about his/her age and gender. We do not know the age and gender of the children.

The goal of the predictive model is to predict the age and gender of those Telco company customers, that have not fulfiled these two variables. Therefore the dataset should contain variables representing real age when predicting age, as well for the gender, respectively in order to train the predictive model. More specifically, variables called `party_age` and `party_gender` were chosen, where `party_age` corresponds to the age of each person and `party_gender` corresponds to the gender of each person.

The problem is, that column `party_age` is not reliable anymore and does not provide relevant information about a person, therefore including this column into training dataset does not make any sense at all. The relevant column about age is just column `age`.

4.2 Preprocessing

This phase is essential in understanding, where is the difference between my model and its ancestor.

At first, certain columns, which do not provide enough information, are deleted. Such columns may be those, which contain more than 90% null values. This can be controversial, because we have to know what does the null value mean – if it means that an error occured during measurement, or if it is just a regular value. It could be the best predictor within the whole dataset, but the author of this model told me, that he made a research on this and those values that he had deleted, could have been deleted for sure. This phase is done right in the database, using SQL.

The second phase of data preprocessing of this model contains some information from the Call Detail Records(CDR). Specifically, the author of this model basically provided components in a graph, where each node represents one person. Edges are between nodes that satisfy these conditions:

- have the same age
- have the same gender
- contact each other by call, sms or mms

So, when we want to know the age and gender of a user, we will consider how he/she interacts with each graph component. But this is not as good as it looks to be because it does not capture all patterns possible.

4.3 Modeling and performance evaluation

This model uses a XGBoost 2.7.1 implementation of the Gradient Boosting Technique. Hyperparameter experimenting showed that manipulation with maximum depth of a tree and the number of estimators does make sense. Everything else was not as important as those two hyperparameters. Setting maximum depth to 6 and number of estimators to 120 showed as the best hyperparameter setting with optimal results.

Age is classified into these five categories:

- 0-18
- 18-25
- 25-35
- 35-55
- 55-120

Each person from the training/testing dataset is then assigned to exactly one category.

The performance of this model is debatable because of the training and testing datasets.

As I mentioned in 4.2, the author of this model used variables, that were not reflecting real values. Training and evaluating the model with those kind of variables is not as efficient as it seems to be. It is due to the fact that when a model is trained and validated on wrong data, it seems that everything works fine, but in reality, its performance is not satisfactory enough.

Therefore real performance can vary from the reported performance shown in Table 4.1.

Table 4.1: Accuracy of Random forest predictive model in percentage

XGBoost	Accuracy in %
Age	70.5%
Gender	76.0%

	Category 1	Category 2	Category 3	Category 4	Category 5
Category 1	0.702	0.102	0.033	0.132	0.031
Category 2	0.057	0.683	0.141	0.097	0.022
Category 3	0.013	0.133	0.581	0.215	0.059
Category 4	0.032	0.041	0.110	0.629	0.187
Category 5	0.005	0.008	0.026	0.148	0.813

Table 4.2: Confusion matrix for age prediction

Chapter 5

Predictive modeling using tree based methods

While creating a new predictive model for age and gender, I tried to retrieve as much information from my collegue who did the current predictive model. Naturally, he has broad knowledge about this issue and we discussed possible improvements often.

5.1 Training and testing datasets and their preprocessing

In this section, I will describe how the datasets were preprocessed and what is the difference between variable `party_age` and variable `age`.

I mentioned in 4.1 that `party_age` does not provide reliable information about customers age. In terms of that, I used variable `age` as a label in order to train model on relevant dataset. `age` provides relevant information about the payer in a party, because it is calculated directly from birth number of a customer and it is updated each month, whereas `party_age` is no longer updated in our database.

Almost everyone in our database has filled `age`, but those people that are not payers, have just a prediction of their age here. The problem is, we do not store information about who the payer is in the party. Therefore, I had to delete all parties with more than one person. If I would not delete those, I would have a dataset that includes also predictions from current model and not real values.

The same applies for gender prediction, we have real gender only for the payers in the party and it is stored in `party_gender`. Non-payers also have values in this column, but it is just a prediction, not real value. So, we have to select parties with one person only.

Now I have datasets for both age and gender prediction with labels, but they are not preprocessed already. Next step is to delete columns with more than 90% of null values. This step deletes about one hundred columns, which speeds up the manipulation with our dataset significantly.

When I was dealing with dataset for age prediction, I realized that age distribution of our customers would not have to be even. I plotted graph 5.1, where we can clearly see in 5.1, that age categories are not evenly distributed. Those categories are perfectly fitted for marketing purposes, but predictive model created on this kind of data was very biased towards the last age category(55+). Hence we want to reduce the bias included in the data and we can do it in a simple way, to create evenly distributed dataset. If the first category contains about 40 thousands people, we would take 40 thousands people from each category. Of course, the size of the dataset will significantly decrease, but it is necessary in this case.

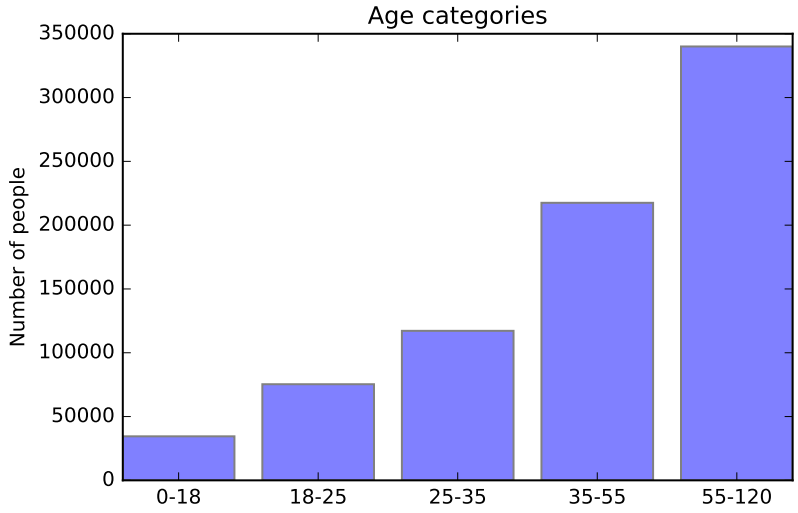


Figure 5.1: Age distribution

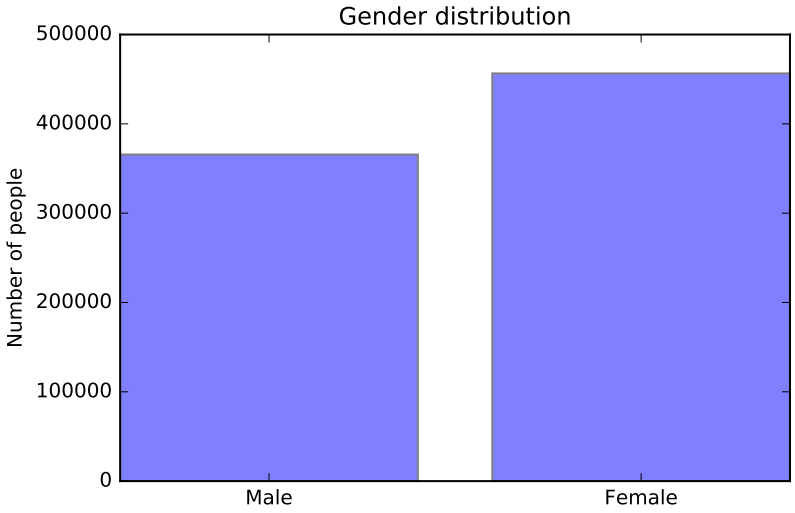


Figure 5.2: Gender distribution

5.2 Modeling

As stated in the assigment, I am supposed to create two different predictive models. The first model I have chosen is called Random forest. The second model is using XGBoost – the same model that my collegue used in 4.3. Results of both models for age and gender are in Table 5.1 and 5.4, respectively.

Both models are based on decision trees and like I mentioned in 2.3, the main advantage of methods based on decision trees is their interpretability and that is the reason why I chose these methods.

5.3 Performance evaluation

5.3.0.1 Age prediction

Predicting age from the business perspective is important. But when we want to target people, for example, that are between 25 and 30, we do not mind if we hit someone who is 24 or 31 years old.

Hence I implemented so called *relaxed accuracy*. It means, that if for example the person belongs to the 3rd age category and the prediction model categorizes the person to the 4th or 2nd age category, it is considered as correct prediction. *Relaxed accuracy* is 83.6% in Random forest, 88.9% in XGBoost.

I would like to show to the reader, what predictors are the most importatnt ones when predicting age. The three most important predictors in predicting age with XGBoost are:

1. Spending of a customer

2. Brand of cellphone

3. Number of outgoing and incoming sms

These features are available from XGBoost object attributes, specifically *feature_importances_*. Random forest, which is implemented using *scikit-learn* library, has also attributes called *feature_importances_* and the most imporant ones are:

1. LTE availability in device

2. Number of outgoing and incoming sms

3. Cellphone brand

Table 5.1: Accuracy of age prediction using Random forest and XGBoost in percentage

	Accuracy in %
Random forest	50.6%
XGBoost	57.1%

Table 5.2: Confusion matrix for age prediction using Random forest

	Category 1	Category 2	Category 3	Category 4	Category 5
Category 1	0.491	0.182	0.066	0.079	0.042
Category 2	0.226	0.460	0.224	0.097	0.028
Category 3	0.124	0.248	0.425	0.240	0.048
Category 4	0.109	0.093	0.247	0.435	0.205
Category 5	0.050	0.018	0.038	0.149	0.677

Table 5.3: Confusion matrix for age prediction using XGBoost

	Category 1	Category 2	Category 3	Category 4	Category 5
Category 1	0.680	0.171	0.062	0.057	0.036
Category 2	0.178	0.511	0.232	0.072	0.019
Category 3	0.065	0.230	0.442	0.239	0.036
Category 4	0.058	0.075	0.231	0.480	0.167
Category 5	0.019	0.014	0.032	0.151	0.742

5.3.0.2 Gender prediction

While predicting gender, we cannot use something like *relaxed accuracy*, as we did in 5.3.0.1. It is just a binary classification - we classify person as a male or as a female.

Accuracy of randomly guessing a person's gender would converge to 50%. The existing model 4.3 has about 76% accuracy and I would like to approach this accuracy.

The most relevant predictors using XGBoost 2.7.1 are:

1. Number of days spent using current device
2. Duration of calls
3. Connection lifetime

Using Random forest 2.6, the most relevant predictors are:

1. Amount of downloaded data
2. Spending of money
3. Amount of uploaded data

Table 5.4: Performance of gender prediction using Random forest and XGBoost in percentage

	Performance in %
Random forest	66.4%
XGBoost	70.9%

Table 5.5: Confusion matrix for gender prediction Random forest

	Female	Male
Female	0.674	0.352
Male	0.326	0.647

Table 5.6: Confusion matrix for gender prediction XGBoost

	Female	Male
Female	0.716	0.329
Male	0.284	0.671

Chapter 6

Predictive modeling using neural networks

Performing this experiment was inspired by paper [10], where its authors used only call detail records as a dataset. As this above what is stated in assignment, I decided to create this neural network just for gender prediction.

6.1 Artificial neural networks

Artificial neural networks (ANN) are one of the machine learning algorithms and are also used, but not exclusively, for classification like Random forest 2.6 or XGBoost 2.7.1. The inspiration for ANN is biological neuron, but I will not dive deeper into discussion about origin of ANNs. A neuron is the elementary building and functional block of neural network. Each neuron has its own inputs and outputs. Inputs are representing quantified information and each input has assigned weight. For weight w_i and value a_i of neuron i, where number of neurons in network is N, neuron performs the operation:

$$f(\sum_{i=1}^{N} w_i a_i + \theta_i) \tag{6.1}$$

where θ is the treshold of neuron activation. Although there is no formal definition, Kevin Gurney has the following explanation in his book: "A neural network is an interconnected assembly of simple processing elements, units or nodes, whose functionality is loosely based on the animal neuron. The processing ability of the network is stored in the inter-unit connection strengths, or weights, obtained by a process of adaptation to, or learning from, a set of training patterns." [3]

6.1.1 Deep learning

Deep learning is a part of machine learning algorithms, which uses multiple layers of nonlinear processing units for feature extraction and transformation. Like machine learning algorithms, deep learning can also be supervised or unsupervised 2.0.1. The term deep learning started to be popular after Geoffrey Hinton published an article titled "Learning Multiple Layers of Representation" [12].

Convolutial neural networks are example of deep feed forward neural network and they apply mathematical operation called convolution[12] instead of matrix multiplication at least in one of their layers. Convolutial neural networks take advantage of the following properties:

- local connections
- shared weights
- pooling
- use of many layers

Those type of neural networks are generally used in 1 dimensional signal or 2 dimensional image processing.

6.2 Preprocessing and datasets

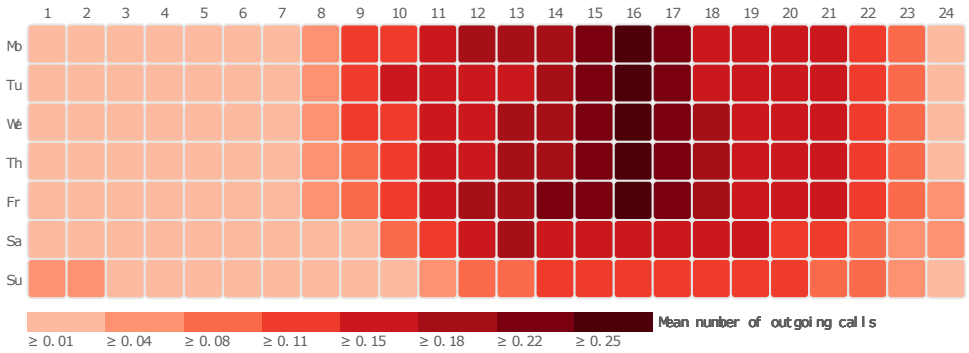
Data used for this experiment are a bit different from what I have used in 5. Call detail records are part of usage data 3.1 and are stored in Aster database 3.2.3. CDRs collects who calls or texts whom, for how long and from where. The location is not completely accurate as it uses the location of BTS. So, CDRs contains information about:

- interaction - text or call
- direction of interaction - in or out
- timestamp
- recipient ID
- call duration, if it is a call
- cell tower to which the phone was connected to

[12]Convolution is a special kind of linear operation and is described here: `http://deeplearningbook.org/contests/convnets.html`

I chose a week-matrix to represent the data. Week-matrix is a matrix, that has weekdays on the y-axis and hours of the day on the x-axis so each matrix has $7x24$ dimensions. I have four matrices, where each matrix represents a separate channel. Channels are average incoming number of calls, outgoing number of calls, incoming number of text messages, number of outgoing text messages. Sample matrix visualized as heat map is showed in 6.1.

Figure 6.1: Sample heat map; source: [10]



6.3 Modeling and performance evaluation

The architecture for convolutial neural network is the same as used by Montjoye et al. The architecture was designed to find inter-days and inter-hours patterns. As Montjoye et al. stated: "While an in-depth study is outside the scope of this paper, these results suggest that there are no strong inter-week patterns that are crucial for predicting demographic attributes." [10].

I used just four channels instead of eight used in [10]. This could cause a loss of information for sure, but I was not able to add another channels because of time pressure.

Accuracy of gender prediction is 63.3% with only four channels. State-of-the-art is 79.7% reported in [10] with eight channels.

Conclusion

The goals of this thesis were:

- Analyze current predictive model for age and gender in Telco company
- Create 4 predictive models based on machine learning algorithms - 2 for age and 2 for gender
- Test those models

In the thesis the following has been done:

- Mistakes made in current predictive model were found
- 4 prediction models were created and tested
- One additional predictive model based on artificial neural networks was created

Mistake in current model have been in dataset, specifically in label column for age prediction. I replaced the wrong column with the correct one in order to train model on dataset representing reality.

One of the goals was also to determine, whether at least one of the two predictive models is speed enough so it can run in production state. Both models were fast enough to run in production, so this goal was also accomplished. Training time of neural net created in 6 lasts longer than expected, but it was acceptable.

Prediction model using XGBoost 2.7.1 library outperformed prediction model using *scikit-learn* implementation of Random forest 2.6 in both age and gender prediction. Model based on XGBoost has also faster training and evaluating phase and is ready to be integrated in production state.

While I was creating predictive models for age, I realised that it would be maybe better to predict continous number instead of category. This would not affect marketing purposes, because to target, for example, college students,

we would pick those that have predicted age from 19 to 24. Subsequently, we would add 2 years to each side in order to tolerate some error in prediction.

Bibliography

[1] MCKINNEY, W.:*Python for Data Analysis*. Gravenstein Highway North, Sebastopol: O'Reilly Media, Octorober 2012, ISBN 978-1-449- 31979-3.

[2] VAN DER WALT, S.; COLBERT, S. C.; VAROQUAUX, G.: *The NumPy Array: A Structure for Efficient Numerical Computation*. 2011, [online]. Availble from: http://www.numpy.org/

[3] GURNEY, K.: *An Introduction to Neural Networks*. Taylor & Francis, 2003, ISBN 9780203451519

[4] PENG, R., MATSUI, E. *The Art of Data Science*. Leanpub, 2015, ISBN 9781365061462

[5] HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. *The Elements of Statistical Learning* 2nd edition, Springer, 2009, ISBN 978-0387848570

[6] CAFFO, B. *Regression Models for Data Science in R*, Leanpub, 2015

[7] O'NEIL, C., SCHUTT, R. *Doing Data Science*, O'Reilly, 2013

[8] BREIMAN, L., CUTLER, A. *Random forests*. [online] [cit. 2018-04-10]. Available from: `https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm`

[9] SONTAG, D. *Ensemble learning*. [online] [cit. 2018-04-09]. Available from: `http://people.csail.mit.edu/dsontag/courses/ml13/slides/lecture13.pdf`

[10] FELBO, B., SUNDSØY, P., PENTLAD, A., LEHNMANN, S., MONTJOYE, Y. *Modeling the Temporal Nature of Human Behavior for Demographics Prediction*. [online] [cit. 2018-03-05]. Available from: `https://arxiv.org/pdf/1511.06660.pdf`

[11] EATON, E. *Classification: Decision Trees & Overfitting*, [online] [cit. 2018-04-10]. Available from: https://www.cc.gatech.edu/~bboots3/CS4641-Fall2016/Lectures/Lecture3_1.pdf

[12] HINTON G.E.: Learning multiple layers of representation. *Trends in Cognitive Sciences*, volume 11, no. 10, 2007: pp. 428 - 434, ISSN 1364-6613, doi:http://dx.doi.org/10.1016/j.tics.2007.09.004. Available from: http://www.sciencedirect.com/science/article/pii/S1364661307002173

[13] GOODFELLOW, I., BENGIO, Y., COURVILLE, A.: *Deep learning*, 2016, MIT Press, [online] [cit. 2018-05-01]. Available from: http://www.deeplearningbook.org

[14] *scikit-learn* [online]. scikit-learn developers. [2018-04-20]. Available from: http://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html

Appendix A

Acronyms

CDR Call detail records

BTS Base transceiver station

Appendix B

Contents of enclosed CD

- `readme.txt` ..... the file with CD contents description
- `src` ..... the directory of source codes
  - `models` ..... implementation sources
  - `thesis` ..... the directory of LaTeX source codes of the thesis
- `text` ..... the thesis text directory
  - `thesis.pdf` ..... the thesis text in PDF format
  - `thesis.ps` ..... the thesis text in PS format