UNIVERSITY OF GOTHENBURG

An exploratory machine learning workflow for the analysis of adverse events from clinical trials

Master's thesis in Statistical Learning and AI

MARGARETA CARLERÖS

Mathematical Statistics
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2020

An exploratory machine learning workflow for the analysis of adverse events from clinical trials

MARGARETA CARLERÖS

UNIVERSITY OF
GOTHENBURG

Mathematical Statistics
University of Gothenburg
Gothenburg, Sweden 2020

An exploratory machine learning workflow for the analysis of adverse events from clinical trials
MARGARETA CARLERÖS

Supervisors: Jesper Havsol, AstraZeneca
Elisabeth Nyman, AstraZeneca
Bo Zhang, AstraZeneca

Examiner: Aila Särkkä, Mathematical Sciences, Chalmers University of Technology and University of Gothenburg

Typeset in LaTeX
Gothenburg, Sweden 2020

An exploratory machine learning workflow for the analysis of adverse events from clinical trials

MARGARETA CARLERÖS
Mathematical Statistics
University of Gothenburg

Abstract

A new pharmaceutical drug needs to be shown to be safe and effective before it can be used to treat patients. Adverse events (AEs) are potential side-effects that are recorded during clinical trials, in which a new drug is tested in humans, and may or may not be related to the drug under study. The large diversity of AEs and the often low incidence of each AE reported during clinical trials makes traditional statistical testing challenging due to problems with multiple testing and insufficient power. Therefore, analysis of AEs from clinical trials currently relies mainly on manual review of descriptive statistics. The aim of this thesis was to develop an exploratory machine learning approach for the objective analysis of AEs in two steps, where possibly drug-related AEs are identified in the first step and patient subgroups potentially having an increased risk of experiencing a particular drug side-effect are identified in the second step. Using clinical trial data from a drug with a well-characterized safety profile, the machine learning methodology demonstrated high sensitivity in identifying drug-related AEs and correctly classified several AEs as being linked to the underlying disease. Furthermore, in the second step of the analysis, the model suggested factors that could be associated with an increased risk of experiencing a particular side-effect, however a number of these factors appeared to be general risk factors for developing the AE independent of treatment. As the method only identifies associations, the results should be considered hypothesis-generating. The exploratory machine learning workflow developed in this thesis could serve as a complementary tool which could help guide subsequent manual analysis of AEs, but requires further validation before being put into practice.

Keywords: Machine learning; Adverse events; Clinical trials; Data mining.

Acknowledgements

This thesis would not have been possible without the help and support of several people. First and foremost, I would like to extend a huge thank you to my supervisors Jesper Havsol, Elisabeth Nyman and Bo Zhang for all their help throughout the past months. Their ideas and suggestions shaped this thesis in so many ways. I am also very grateful for the guidance and support that Klas Lindell, Ulrika Emerath and Lars Pettersson provided and for helping me understand how my thesis related to their work in patient safety. I also wish to thank Tom White, the originator of the adapted Virtual Twins method, for allowing me to use the method and for patiently explaining it to me; Isobel Andersson for her valuable suggestions; and Per Arkhammar for sharing his clinical experience. I feel very fortunate to have had the opportunity to work with all of you. Special thanks to Martin Karpefors for his role in initiating this thesis project and to everyone who I have encountered at Advanced Analytics and AI for your words of encouragement and for ideas shared over lunch, fika, and lately, virtual fika and virtual meetings. I would also like to thank my examiner Aila Särkkä for her input and for lending support. And lastly, I would like to thank my friends and family for standing by me during this journey.

Margareta Carlerös, Gothenburg, June 2020

Contents

List of Figures

List of Tables

1
Introduction

Before a new pharmaceutical drug can be used to treat patients it needs to be shown to be safe and effective. The analysis of a drug's safety aims to establish whether a drug has any side-effects of concern. Information about possible side-effects, known as *adverse events*, is collected during the testing of a new drug in humans and continues throughout the life cycle of a drug [1]. An AE is defined as

> *"any untoward medical occurrence in a patient or clinical investigation subject administered a pharmaceutical product and which does not necessarily have to have a causal relationship with this treatment..."*

and include

> *"...any unfavourable and unintended sign (including an abnormal laboratory finding, for example), symptom, or disease temporally associated with the use of a medicinal product, whether or not considered related to the medicinal product"* [2].

Thus, it must be determined whether an AE is possibly related to the drug or not, i.e. if it may be a side-effect. Unfortunately, traditional statistical testing is generally unsuitable for this purpose, mainly due to insufficient power and problems with multiple testing [3, 4, 5]. Therefore, analysis of AEs largely depends on descriptive statistics and requires substantial expertise to interpret.

The analysis of a drug's safety can be defined as a number of pattern finding tasks. For example, we may wish to understand which AEs are associated with a drug, as such AEs could be possible side-effects. Furthermore, being able to identify patient subgroups in which the risk of experiencing a particular drug side-effect is higher could be a step towards personalized treatment, whereby the treatment of patients experiencing specific side-effects could be adapted or steps could be taken to reduce the side-effect risk, if possible. Currently such analyses involve the manual review of a wide range of patient information.

Machine learning methods automatically learn to associate patterns in data with an outcome. Such methods could allow for a data-driven, comprehensive and objective analysis of AEs which could help guide subsequent manual analysis. The aim of this thesis is to develop an exploratory machine learning approach for the objective identification of drug-related AEs as well as the identification of patient subgroups

potentially having an increased risk of developing a particular drug side-effect. This is achieved using data from studies of a well-characterized drug where the true side-effects are considered to be known.

A background to the field is provided in Section 1.1 followed by a refined statement of the aim in Section 1.2. An outline of this report is found in Section 1.3.

1.1 Background

The goal of drug development is to develop an effective treatment with as few side-effects as possible. For this we need to study the *efficacy* (how well the drug is able to treat a specific condition) and safety of the drug. Such testing needs to occur before a drug can be placed on the market and be accessed by patients.

The drug development process proceeds in a number of stages, where the last stage of drug development, in which the drug is tested in human subjects, is known as *clinical development*. Clinical development is further divided into phase I, II, III and IV [6]. Studies performed as a part of these phases are known as *clinical trials*. Phase III clinical trials will be described in general in Section 1.1.1. Thereafter, different sources of safety data in phase III clinical trials, with an emphasis on AEs, are covered in Section 1.1.2. The reasons why analysis of safety data from such clinical trials is challenging is explained in Section 1.1.3. Lastly, Section 1.1.4 covers current practices in the analysis of AEs.

1.1.1 Phase III clinical trials

In a phase III clinical trial of a novel drug, the drug will commonly be compared to *placebo*, a compound that lacks biological activity but that has an appearance that resembles the drug. The different treatment regimens that a study subject can be assigned to in a clinical trial are known as *arms*. A study that contains a placebo arm is referred to as a *placebo-controlled* trial. The goal of a placebo-controlled phase III clinical trial is generally to show that the drug shows superior efficacy compared to placebo in treating a specific disease [6].

Typically subjects are randomly assigned to one arm, a process known as *randomization*, with an approximately equal number of subjects being assigned to each arm. Randomization helps ensure that study subjects in the treatment arms are roughly similar with respect to, for example, demographics, disease stage, medical history, concomitant medications and other *baseline variables* (variables that are known or measured before the first dose of treatment is administered) [6].

Furthermore, studies may be *blinded*, whereby subjects do not know which arm they were assigned to. In a *double-blinded* trial, neither the study subject nor the investigator or medical staff knows to which arm the patient belongs until after the trial [6]. Phase III trials are often performed as *multi-center studies* where patients from a number of hospitals across different countries are included.

1.1.2 Sources of safety data in clinical trials

There are several different types of safety data that are collected during a clinical trial. In fact, 70-80% of the information recorded during a clinical trial is estimated to relate to safety [7]. This information ranges from results of laboratory tests, to various clinical examinations, patient-reported outcomes and AEs [1, 3].

In a trial, AEs are generally recorded at regular intervals at a *study visit*, where the patient visits the study site. These include both AEs that the patient reports having experienced since that last study visit and AEs identified by a clinician during the study visit [3].

AEs are typically encoded using the *Medical Dictionary for Regulatory Activities* (MedDRA) [8]. This is a medical dictionary for classification of AEs that was developed in the 1990s and has since been widely adopted throughout the industry and by regulatory authorities [3]. In MedDRA, each AE is classified according to a 5-level hierarchy that from the highest level to the lowest level includes: *System Organ Class* (SOC), *High Level Group Term* (HLGT), *High Level Term* (HLT), *Preferred Term* (PT) and *Lowest Level Term* (LLT). Often only the PT and SOC of an AE are considered when summarizing or analyzing AEs. As of March 2020, MedDRA contains 24,289 different PTs belonging to 27 SOCs [9].

During a clinical trial, additional information is generally recorded in conjunction with each AE that is reported by a subject. These include, for example, the time of onset of the AE, the duration that the subject experienced the AE and the intensity of the AE (i.e. if it was a mild, moderate or severe case). Note that if a subject reported the same AE multiple times during the study, this may be recorded as separate events, although it will typically be presented as a single event in the final reporting. Furthermore, any events that have serious consequences for the patient will be characterized as *serious adverse events* (SAEs). AEs that are flagged as SAEs are given special attention when evaluating the safety of a treatment.

1.1.3 Why is analysis of safety in clinical trials challenging?

The analysis of safety signals from clinical trials is complicated due to a number of reasons which are outlined in this section. Firstly, for ethical and financial reasons, the size of trials should be kept small and their duration should be minimized. This makes it challenging to detect very rare AEs or AEs that take longer time to develop. However, a rare event or an event that occurs after some time may be severe and enough reason to withdraw the drug from the market [3]. A rule of thumb for the number of subjects who need to be enrolled to detect a single case of a drug-related AE with a certain incidence is the so called *rule of three*. According to this rule, if the incidence of a drug side-effect is 1 in n, $3\times n$ subjects must be enrolled in order to detect a single case [10].

Secondly, trials are commonly designed around an *efficacy endpoint* [1, 3], meaning that they are sized and powered in order to be able to test a hypothesis relating

to the efficacy of the drug in treating a certain aspect of the disease under study. One way in which possible side-effects of a drug may be identified is to compare the frequency of each AE between subjects receiving the drug and subjects receiving placebo, as a drug-related AE should be more common in subjects receiving the drug. However, since only a few patients usually experience each AE in a study, comparing the frequencies of AEs between treatment arms using statistical testing will not have sufficient power [1, 3].

The focus on efficacy also means that, in order to keep the trial size small and duration short, we want to observe a large effect in the group receiving the drug and to reduce the variance of this effect. For this reason, the subjects selected for a trial should preferably be a homogeneous group and we may choose to enroll only subjects who have a severe form of the disease. For example, we could exclude any patient who is taking a certain medication [3]. However, this could lead to the trial subjects not being representative of the patient population and could limit our ability to identify AEs that are possibly drug-related during clinical trials.

There is also the possibility that AEs are missed or misclassified during the study. For example, AEs that are experienced by patients in-between study visits, may end up not being recorded [3]. In addition, the way that the event should be encoded may be subject to interpretation and could therefore differ between study sites. For example, the same event could belong to multiple PTs.

Further complicating the analysis of AEs is that subjects may choose to drop out at any point during the study. Clearly, if a patient has spent a longer time in the study, there are more opportunities to report AEs [3]. Another possibility is that the subject dropped out due to an AE [3] which would normally be recorded along with the AE.

In a clinical trial there can be hundreds of different AEs that are reported and the total number may even exceed the number of subjects in the study [4]. Performing traditional statistical testing to determine whether the incidence of each AE is significantly over-represented in subjects receiving the drug compared to those receiving placebo leads to problems with multiple testing. Ignoring the problem of multiple testing will lead to false positive results, where AEs are identified as being treatment-associated where no such association exists [1, 4]. One alternative is to perform this hypothesis testing on a limited set of pre-specified AEs, but this instead suffers from a risk of false negatives as any treatment-related AEs that are not among the pre-specified AEs will be missed [1]. Identifying which hypotheses to test is a subjective task and can be challenging based on e.g. animal studies, as the behavior of the drug in humans may differ compared to animals [1, 4]. It is also unclear how to include information about the duration and intensity of the AEs, which may hold important clues about differences between treatment arms, in the analysis.

Lastly, the interpretation of AEs is context-dependent. Whether a particular AE is serious enough to stop the drug development will depend on the prognosis of the

patients without having received the treatment [3].

1.1.4 Analysis of adverse events

For the reasons mentioned above, safety evaluations in clinical trials mainly rely on descriptive statistics. These include a number of tables that summarize the frequencies and percentages of subjects per treatment arm who have experienced an AE [3, 11]. These are usually reported on the PT and SOC levels, as well as via separate reporting per treatment arm for SAEs and deaths. Interpreting these descriptive statistics requires considerable expertise and is subjective. To aid the analysis, various visualizations of the data may also be generated [3, 1, 11, 12]. Depending on the drug, AEs in different pre-defined subgroups, e.g. pediatric patients, may be studied. It can also be valuable to identify risk factors that are associated with a drug-related AE. Currently, such analyses are generally performed manually and *ad hoc*, although the use of data mining techniques has been suggested [3].

Machine learning is increasingly being used to analyze drug safety data, particularly in the *post-marketing* setting, when the drug has already been approved. Studies of the safety of a marketed drug is an important means to capture drug-related AEs that are uncommon, are the result of long-term drug use or those that only become apparent when the drug is released to the general patient population rather than the narrowly defined study population of a clinical trial [3]. Such data is published in large public pharmacovigilance databases and relies on spontaneous reporting [1, 13]. In the European Economic Area this database is called *EudraVigilance* [14] while the US counterpart is the *FDA Adverse Event Reporting System* (FAERS) [15]. The drawbacks of such databases include the under-reporting of AEs, the limited information available about the patient and the generally low quality of the data. As an alternative to these databases the use of machine learning to extract AEs from *electronic healthcare records* (EHRs) has been suggested, but this approach suffers from data privacy issues [16, 17].

Data collected during clinical trials is often available for selected research purposes that are unrelated to the original trial. Such datasets are typically in a tabular format, of high-quality and detailed, making them analysis-ready for machine learning methods.

1.2 Aim

The aim of this thesis is to develop an exploratory machine learning workflow that analyzes AEs at PT level from multiple placebo-controlled phase III clinical trials in two steps:

1. identification of AEs that are possibly drug-related;

2. identification of patient subgroups in which the risk of developing a particular drug side-effect is potentially increased.

1.3 Outline

This thesis is organized as follows. Section 2 introduces the data that will be used, originating from two phase III clinical trials of the drug *Symbicort* for the treatment of *chronic obstructive pulmonary disease*. Symbicort is considered to have a well-characterized safety profile. Section 3 provides a theoretical foundation of the tree-based supervised machine learning method as well as the model evaluation and interpretability methods that are used. In Section 4 it is explained how these methods can be combined into two different exploratory data mining methodologies, one for each aim. Results are presented in Section 5, followed by a discussion of the results and suggestions for further research in Section 6 and a conclusion in Section 7.

2

Data Description and Exploratory Data Analysis

This thesis is based on data from two randomized, double-blind, multi-center, placebo-controlled phase III clinical trials where the efficacy and safety of different therapies for *chronic obstructive pulmonary disease* (COPD) were evaluated.

COPD is a respiratory disease that is associated with long-term cigarette smoking, exposure to air pollution or recurrent lung infections [18], although these likely interact with other risk factors [19]. It is characterized by a range of symptoms including, but not limited to, progressive and irreversible airflow limitation causing shortness of breath and an increased inflammatory response in the lungs [20]. An estimated 174-384 million people have the disease worldwide and it accounts for over three million deaths annually [19].

Worsening of COPD symptoms is known as a *COPD exacerbation* and patients who experience frequent exacerbations have been found to have an accelerated disease progression [20]. *Symbicort* is an inhaled drug that can be used to reduce the risk of COPD exacerbations. Originally developed and approved for the treatment of asthma, the drug consists of two compounds, budesonide and formoterol. Budesonide is an inhaled corticosteroid which acts locally to reduce inflammation while formoterol is a so called long-acting β_2-agonist and is a bronchodilator. Both budesonide and formoterol reduce the risk of COPD exacerbations and this effect is enhanced when they are combined [20, 21, 22].

Symbicort has been on the market for the treatment of asthma since 2000 and for COPD since 2003. The safety profile of this drug is therefore considered to be well-characterized. Appendix A lists the currently recognized side-effects of Symbicort. As one of the aims of this thesis is to develop a method to identify possibly drug-related AEs, a comparison to the known drug side-effects can act as a validation of the results. For this reason, Symbicort will be the drug under study in this thesis.

The studies from which the data in this thesis is obtained are described in Section 2.1. The coding of the AEs present in this data is then explained in Section 2.2. Finally section 2.3 presents descriptive statistics based on the coded AEs, with section 2.3.1 discussing the consequences of excluding AEs experienced by only one subject on the descriptive statistics.

2.1 Origin of the data

The data in this thesis originates from the studies by Tashkin et al. (2008) [21] and Rennard et al. (2009) [22], henceforth referred to as *SHINE* and *SUN*. SHINE was a 6-month trial that was conducted across 194 sites in the US, Czech Republic, the Netherlands, Poland and South Africa between 2005 and 2006 [21], while SUN was a trial that followed patients during a 12-month period across 237 sites in Europe, the US, and Mexico between 2005 and 2007 [22].

Both studies consist of a number of treatment arms. Two of the arms that are common between both studies is a Symbicort arm (budesonide/formoterol pressurized metered-dose inhaler 160/4.5 $\mu g \times 2$ inhalations) and a placebo arm. Data belonging to these arms were pooled across SUN and SHINE. Subjects who received at least one dose of Symbicort or placebo were included in the analysis. Furthermore, only subjects where special permission had been granted to allow for reuse of data for other research purposes were included. This resulted in a dataset with a total of 723 subjects in the Symbicort arm (453 from SUN and 270 from SHINE) and 677 in the placebo arm (404 from SUN and 273 from SHINE) and an overall total of 1400 subjects.

2.2 Coding of adverse events

AEs were coded in a binary format at the PT level of the MedDRA hierarchy, where a 1 represented that the subject had experienced the event at any point during the study and 0 that the subject had not experienced the event. This binary encoding of AEs disregards any repeated occurrence of an event in the same subject, the duration of the event, as well as the intensity of the event (i.e. if it was considered to be a mild, moderate or severe case of the AE).

2.3 Exploratory data analysis

The number and percentage of subjects who reported at least one AE during SUN and SHINE is summarized by treatment arm in Table 2.1. Note that the numbers presented in this table may deviate from the numbers reported in the original studies as only a subset of data from SUN and SHINE is available for reuse in this thesis. Overall, 63% of subjects in the Symbicort arm had an AE while 58% in the placebo arm had an AE. The percentage of subjects with an AE is consistently somewhat higher in the Symbicort arm than in the placebo arm across both SUN and SHINE.

The longer study duration of SUN (12 months) compared to SHINE (6 months) is reflected in the higher percentage of subjects having any AE in SUN than in SHINE. Overall, 64% of subjects had an AE in SUN while only 55% had an AE in SHINE.

Among the subjects who experienced an AE, the median number of different AEs experienced by a subject was two in both the Symbicort and placebo arms. The

Table 2.1: Number and percentage of subjects who reported at least one AE by treatment arm and study. The studies SUN and SHINE refer to the subset of data from the original clinical trials that is available for reuse in this thesis. The numbers presented below can therefore deviate from the original studies.

	Symbicort	Placebo	Total
	454 (63%)	**392 (58%)**	**846 (60%)**
SUN	297 (66%)	248 (61%)	545 (64%)
SHINE	157 (58%)	144 (53%)	301 (55%)

frequency distributions of the number of different AEs per subject are shown in Fig. 2.1. These plots exclude the subjects who experienced no AE. The maximum number of different AEs observed in a subject was 17 in both the Symbicort and placebo arms. While the number of subjects who had 1 or 2 different AEs was similar across treatment arms, there were more subjects with over 2 different AEs in the Symbicort group.

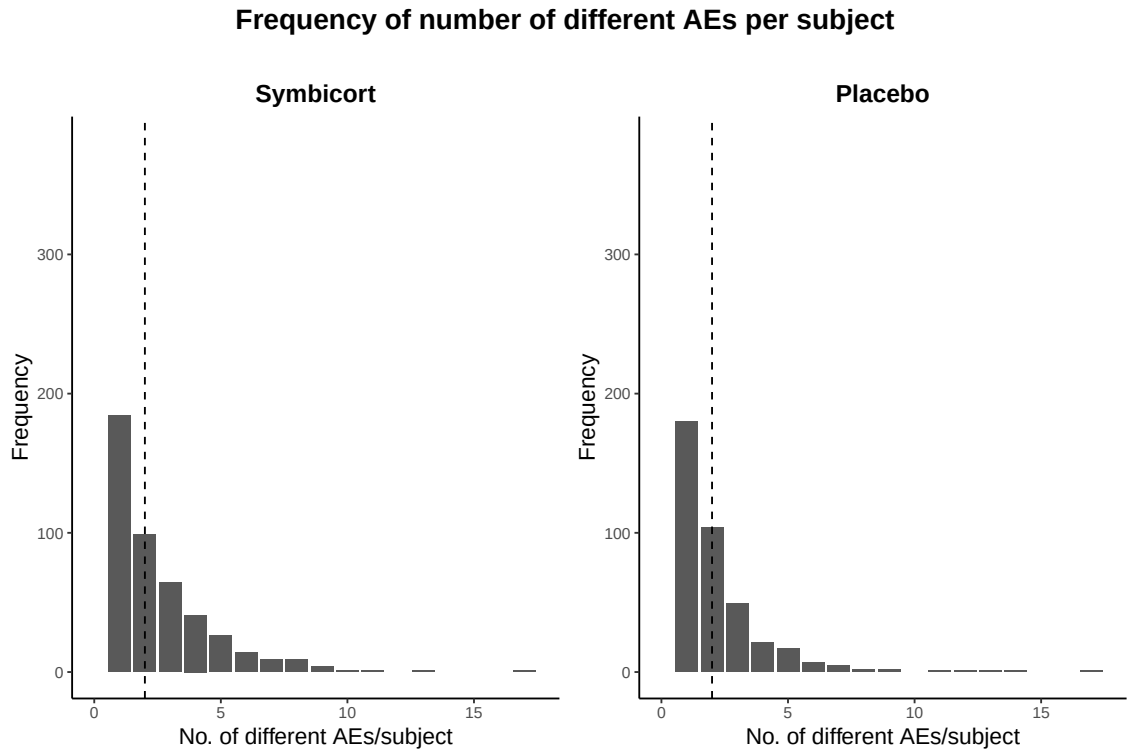


Figure 2.1: The frequency distribution of the number of different AEs per subject, after exclusion of subjects without any AE. The dashed vertical line indicates the median value.

The number of subjects who had a particular AE was generally low (Table 2.2). In both treatment arms, approximately 60% of AEs were only experienced by one

subject and 20% by two subjects. Thus only about 20% of AEs were experienced by more than two subjects in both treatment arms. Appendix B contains a list of the ten most common AEs.

Table 2.2: Percent of adverse events experienced by one, two or more than two subjects in the Symbicort and placebo arms.

Number of subjects experiencing the AE	Percent of different AEs Symbicort	Placebo
1	61%	63%
2	17%	18%
> 2	22%	19%

2.3.1 Exclusion of uncommon adverse events

AEs that occur only once across both treatment arms are likely of limited significance when determining whether an AE is drug-related or not. Table 2.3 presents the effect of removing such AEs. After removal of these 306 different AEs, the number of subjects in the Symbicort arm with at least one AE was 433 which corresponded to 60% (compared to 63% previously). In the placebo arm the corresponding value was 372 or 55% (previously 58%). The total number of AEs decreased from 1174 to 993 in the Symbicort arm and from 882 to 757 in the placebo arm. The total number of different AEs decreased by over half from 532 to 226, where 221 of these AEs were found in the Symbicort group and 196 in the Placebo group. Frequency distributions per treatment arm of the number of different AEs per subject after removal of the uncommon AEs are included in Appendix B.

Table 2.3: Number of subjects with at least one AE, total number of AEs experienced by subjects and the number of different AEs by treatment arm before and after removal of AEs occurring only once across both studies.

	Symbicort	Placebo	Total
Before removal of AEs occurring only once			
No. subjects with AE	454	392	846
Total no. AEs	1174	882	2056
No. different AEs	392	321	532
After removal of AEs occurring only once			
No. subjects with AE	433	372	805
Total no. AEs	993	757	1750
No. different AEs	211	196	226

3
Theory

A machine learning method can be thought of as a set of instructions for how a computer automatically learns from data [23]. The data could be, for example, a set of images or text segments. For simplicity we here assume that the data $\mathbf{x}$ is in a *tabular format* where each observation $i = 1, 2, \ldots N$ is represented by a row, each column $j = 1, 2, \ldots p$ represents some feature of the data and the values $x_{ij}, i = 1, 2, \ldots N$, for each feature j are either numeric or categorical. The process of learning from data is referred to as *training* or *fitting* a model [24]. Based on the instructions provided by the chosen machine learning method, a function $f(\mathbf{x})$ is fitted to the data. Specifically, the values of a number of parameters that are specified by the method are determined based on the data.

There are two principal ways in which this learning occurs: *supervised* or *unsupervised* [24]. In supervised learning, information about an *outcome* or *target* y_i is available along with the feature values $\mathbf{x}_i$ for each observation. The aim of a supervised learning task is to construct a model f that, given an observation $\mathbf{x}_i$, can be used to predict its target value y_i, i.e. $\hat{y}_i = f(\mathbf{x}_i)$. By comparing $\hat{y}_i$ and y_i we can measure how well the model fits to the data. This type of learning can further be subdivided into *classification* and *regression* problems, depending on whether the target values are categorical or numeric, respectively [24]. In contrast, no information about target values is available in unsupervised learning. This type of learning, which includes different data clustering methods, is largely concerned with discovering structure in data rather than making predictions. Unsupervised learning is therefore often used for exploratory purposes [25].

When training a supervised learning model we want it not only to generate predictions that closely resemble the true target values of the data used for training it, but more importantly we want it to be able to generalize to new data. It is therefore crucial that the model only learns about the signal in the training data rather than about any random noise that may be present, since the same random noise will likely not be present in new data. When a model starts learning about the noise in the training data it is said to *overfit* [24]. Often models that are more complex, i.e. contain more parameters that are determined from the data, have a higher tendency to overfit. This results in high variance in the predictions when such a model is applied to new data. On the other hand, the model must be sufficiently complex to be able to capture the signal in the data, otherwise the resulting model will have high bias. We must therefore balance both bias and variance when building the model in order to reduce the prediction error. This is known as the bias-variance trade-off [25].

One way in which we can control model complexity and other aspects of model fitting is through the choice of hyperparameters. A hyperparameter is a user-defined parameter in a machine learning method. The optimal value or setting of a hyperparameter depends on the data and is often determined using the training data in a process called *hyperparameter tuning*, described in Section 3.2.1. In addition, we can simultaneously monitor the goodness-of-fit and complexity of the model during training by defining a suitable objective function that is optimized throughout the training process (see e.g. Section 3.1.4).

The remainder of this chapter is organized into four parts. Section 3.1 describes different tree-based machine learning methods for classification and regression. Section 3.2 explains different choices for evaluating model performance. Model interpretability, being able to explain the predictions of a model or what the model has learned from the data, is covered in Section 3.3. Lastly, statistical testing, with a focus on categorical data analysis, is briefly described in Section 3.4.

3.1 Tree-based methods

This section focuses on tree-based methods for supervised learning. Variants of these methods such as *extreme gradient boosting* have been shown to produce models that in many cases outperform other types of machine learning methods, including neural networks, when applied to tabular data [26]. *Extreme gradient boosting* is the machine learning method that is used in this thesis. The theoretical foundation of this method is explained in several steps. Firstly, the basis for all tree-based methods, the decision tree, is described in Section 3.1.1. Thereafter, various ways of combining several decision tree models into one model are outlined in Section 3.1.2. Section 3.1.3 further explains one of these methods, *gradient tree boosting*. Finally, Section 3.1.4 details a modified version of gradient tree boosting, *extreme gradient boosting*.

3.1.1 Decision trees

A decision tree model consists of a hierarchically organized set of rules that splits the p-dimensional *feature space* of the data into regions $R_j, j = 1, 2, \ldots, J$ called *leaves*. The feature space consists of all possible values of the p features. Each of these regions is associated with a constant s_j which corresponds to the prediction $\hat{y}_i$ of the decision tree for all observations i that fall in the region R_j [24]. A schematic illustration of a decision tree is shown in Figure 3.1.

Each rule in the decision tree is commonly referred to as a *node* and at each node the data is split into two parts, a so called *binary split*. A node consists of a feature and a splitting point or criterion [24]. If the feature is numeric, then the splitting point can be viewed as a threshold. For a binary categorical feature the splitting criterion can instead be interpreted as the presence or absence of the feature.

A decision tree model should, by successively dividing the feature space into smaller regions, produce regions R_j that group together observations that have a similar target value. A decision tree that is used for a classification problem, a *classification tree*, will assign new observations to either the majority class of the training observations that ended up in the same leaf R_j or alternatively use the proportions of training observations belonging to the different classes to assign class probabilities to the new observations [24]. In contrast, a *regression tree*, may assign the mean of the training observations in a leaf R_j to any new observation that belongs to the same leaf in the regression problem [24].

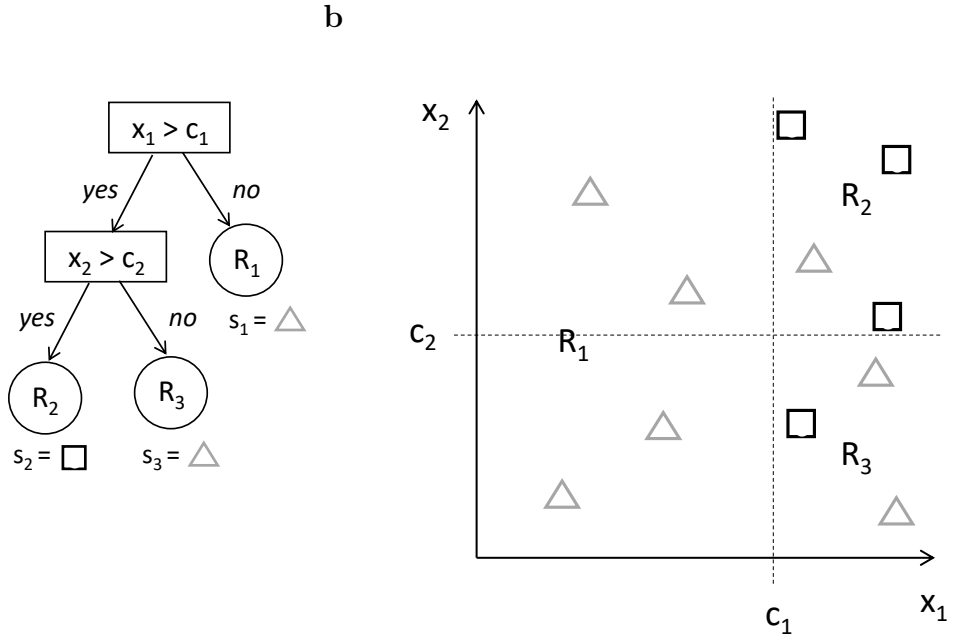


Figure 3.1: Schematic illustration of a decision tree for classification. The two classes are black squares and gray triangles. (a) Nodes are represented as boxes and leaves as circles. At the top node, feature x_1 is split at constant c_1. At the second node, observations with $x_1 > c_1$ are split on feature x_2 at constant c_2. Each leaf R_j, $j = 1, \ldots, 3$ is defined by a number of splitting criteria. The majority class of each leaf observed in the training data s_j, $j = 1, \ldots, 3$ is used to classify new observations. (b) The feature space upon which the decision tree in (a) is based, with the thresholds c_1 and c_2 shown as dashed lines. The data points originate from the training data.

When training a decision tree model we want to find the hierarchical set of rules that in the least number of splits possible divides the data with respect to the target values. Ideally we would test all possible trees that could be constructed. However, this is too computationally expensive. Instead a process called *recursive binary partitioning* is used to fit the tree to the training data in a *top-down* and *greedy* manner [24].

We start at the top node of the tree, the *root*, and identify the feature and splitting

point based on all observations in the training data that result in the best separation of the target values. This produces two child nodes and in each of these we again attempt to find the best feature and splitting point. The difference compared to the initial node is that only the observations that were assigned to the child node can be used to determine the optimal split, rather than all training observations. The node splitting is repeated until some stopping criterion is encountered. This tree growing procedure is greedy since it only considers the optimal split at a node rather than the split that will result in the globally optimal decision tree [24].

The best split at a node is determined by identifying the feature and splitting point that minimize some loss function. The choice of loss function is yet another way in which classification and regression trees differ. A common choice of loss function in a node R_n in regression trees is the residual sum of squares

$$\sum_{i:x_i \in R_n} (y_i - s_n)^2, \tag{3.1}$$

where we want to minimize

$$\sum_{i:x_i \in R_l} (y_i - s_l)^2 + \sum_{i:x_i \in R_r} (y_i - s_r)^2 \tag{3.2}$$

at each node R_0 that is split into R_l and R_r [24].

For classification the most common loss function is the Gini index, defined as

$$\sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) \tag{3.3}$$

with $\hat{p}_{mk}$ being the probability of class k in node R_m given by the training observations and K the number of classes. The Gini index corresponds to the sum of the variance over the K classes [24]. The total loss of the child nodes is calculated by weighting the Gini index of each child node by the proportion of training observations that ended up in the nodes after the split [25].

Several different loss functions can be used for classification and regression problems. Another choice that has to be made is the size of the decision tree. Since decision trees are prone to overfit to the training data, a risk that increases with tree size, different hyperparameters are available that limit the size of the tree that is grown. These include, but are not limited to, the maximum depth of the tree, the minimum number of observations that must be available in a node in order for a split to occur and the minimum number of observations in a leaf [27].

Decision trees have several advantages. Firstly, the models are generally quick to construct [25]. Another benefit is that the models are intuitive and easily interpretable by humans, at least when the number of nodes is small [24, 28]. The manner in which the splitting of the data is performed means that decision trees

can handle mixtures of numeric and categorical features, as well as features of different scale [25]. These models are also relatively robust with respect to outliers [25]. By selecting the most relevant feature in every split, decision tree models perform feature selection and are thereby not as influenced by irrelevant features as other types of models may be [25]. Finally, decision tree models can express non-linear relationships between features and target as well as interactions between features [25].

The main drawback of decision trees is their tendency to overfit, meaning that they will not generalize well to new data. Part of the instability of decision trees is due to their hierarchical structure since a suboptimal early split will affect all onward splits [25]. The greedy construction of trees means that the locally optimal split at a node is chosen rather than the split that yields the globally optimal tree. Another drawback is that these models are unstable when they are too complex given the underlying true structure of the data. For example, if the relationship between the features and target is linear, a decision tree model may result in a more complex model with greater risk of overfitting than if a linear model had been fitted to the data [24]. Regardless of the underlying relationship between features and target, a fully grown decision tree will likely fit not only to the signal in the data, but also to the noise [29].

Two common ways to minimize the overfitting of decision trees are *pruning* and *ensembles*. Pruning involves reducing the size of the decision tree after it has been constructed by defining a number of subtrees and choosing the subtree that produces the best balance between goodness-of-fit and complexity as the final model [24, 25]. Decision tree ensembles are described in the next Section.

3.1.2 Tree ensembles

While decision trees result in versatile models for performing classification and regression, they have a tendency to overfit to the data they were trained with. One solution to this problem is to combine several decision trees into a so called ensemble model. The idea behind ensemble models is that each of the component models has learnt about slightly different aspects of the data [30]. Together these models can likely make a better prediction than any one model alone could.

In order to construct an ensemble model using a number of decision trees, a tree ensemble, we must construct a set of different decision trees. To make each decision tree different, we must train each tree using a different set of data. How the training data for each tree in the ensemble is selected and how each tree is subsequently trained are the principal ways in which tree ensemble methods differ.

The two most common ensemble methods that are used for decision trees are *bagging* and *boosting* [24]. Bagging is short for bootstrap aggregating. In this method each model is constructed from a bootstrap sample. The final predictions are commonly produced by either averaging the predictions of the individual models or, in the

case of classification, the final class assignment can be determined by majority vote. Random forest is an example of a bagging method [29].

In bagging each of the trees is trained independently. In contrast, boosting models are trained sequentially, whereby the model that is constructed depends not only on the data but also on the models in the ensemble that have already been constructed. Boosting relies on training a sequence of so called *"weak"* learners [29]. Each tree is usually small which makes the training process slow as each tree only captures limited information about the training data. However, models that learn incrementally often outperform models that attempt to learn everything at once [24].

A popular boosting algorithm that was originally developed for classification problems is AdaBoost [31]. In AdaBoost, observations that were previously misclassified by the ensemble are given a higher weight when training the next classification tree to add to the ensemble. When combining the classification trees into the ensemble, trees that were more accurate are given a higher weight. The final prediction of the ensemble is determined using a weighted majority vote.

We can also consider the training of boosting models from the perspective of minimizing a loss function in each step. Here, each model that is added leads to a further reduction of the loss. This is one of the main ideas behind gradient tree boosting, a generalization of tree boosting that can be used for both classification and regression problems [32].

3.1.3 Gradient tree boosting

Gradient tree boosting [32] is a machine learning method in which several decision trees are trained sequentially [24]. Letting $b_m(\mathbf{x}_i)$ be an individual gradient boosted tree model constructed in step m and $f_{m-1}(\mathbf{x}_i)$ be an ensemble of $m-1$ gradient boosted trees, we can express model $f_m(\mathbf{x}_i)$ as

$$f_m(\mathbf{x}_i) = f_{m-1}(\mathbf{x}_i) + b_m(\mathbf{x}_i) \tag{3.4}$$

where

$$f_{m-1}(\mathbf{x}_i) = \sum_{\tau=1}^{m-1} b_\tau(\mathbf{x}_i). \tag{3.5}$$

Thus the final prediction is a sum of the predictions of all m models in the ensemble [25, 30].

During the training of a gradient tree boosting model, a loss function L is minimized in each step [25]. For binary classification problems (where f is the logit transform of the predicted probability) it is common to use the binomial deviance (or cross-entropy) as a loss function [25]

$$L(y_i, f(\mathbf{x}_i)) = \log\left(1 + e^{-2y_i f(\mathbf{x}_i)}\right) \tag{3.6}$$

while for regression problems the squared-error loss [25] is generally used

$$L(y_i, f(\mathbf{x}_i)) = (y_i - f(\mathbf{x}_i))^2 . \tag{3.7}$$

Algorithm 1 presents the pseudocode for gradient tree boosting. The initial model f_0 consists of a tree with just a single terminal node, i.e. the model predicts the same value for all observations and this value is optimal given the chosen loss function [25]. For example, if the loss function is the squared error loss, f_0 would predict the mean target value of the training data for all observations.

Each model that is added to the current ensemble f_{m-1} is trained using the pseudo residuals r_{im} rather than the original targets y_i in the training set. The pseudo residuals correspond to the negative gradient of the loss function with respect to the current prediction of f_{m-1} evaluated for each observation x_i in the training set, i.e. $r_{im} = -\partial_{f_{m-1}(x_i)} L(y_i, f_{m-1}(x_i))$. Note that the residuals will be continuous-valued for both regression and classification problems, which only differ by the loss function used [25]. This means that for both regression and classification problems we can use the pseudo residuals r_{im} to fit a regression tree. The only differences between regression and (binary) classification models are the loss function used and the need to convert the final prediction to a predictive probability via the logistic function [25].

Given the m^{th} regression tree, we calculate the score s_{jm} for each terminal node $j = 1, 2, \ldots, J_m$ defined by the terminal region R_{jm} that results in the smallest overall loss when added to the current predictions for observations x_i in the node. These optimal scores s_{jm} are then added to the current predictions for observations $x_i \in R_{jm}$.

The final model $\hat{f}(x)$ thus consists of an initial prediction by f_0 that has been incrementally adjusted such that the prediction for each observation increasingly reflects the true target value in each step. By using small trees, the learning of each tree will be limited. However, by fitting each new tree to pseudo residuals, which can be thought to represent what the current ensemble still has to learn about the data, each tree gets an opportunity to learn about different aspects of the data. Together these trees constitute a powerful ensemble model.

Algorithm 1: Gradient Tree Boosting

1) Initialize $f_0(x) = \arg\min_s \sum_{i=1}^{N} L(y_i, s)$;

2) **for** $m = 1, \ldots, M$ **do**

  a) **for** $i = 1, 2, \ldots, N$ **do**

    $r_{im} = -\partial_{f_{m-1}(x_i)} L(y_i, f_{m-1}(x_i))$

  end

  b) Fit a regression tree to the targets r_{im} giving terminal regions R_{jm}, $j = 1, 2, \ldots, J_m$.

  c) **for** $j = 1, 2, \ldots, J_m$ **do**

    $s_{jm} = \arg\min_s \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + s)$

  end

  d) Update $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} s_{jm} I(x \in R_{jm})$

end

3) Output $\hat{f}(x) = f_M(x)$.

Gradient boosted trees can be tuned in several ways. Firstly, there is the choice of loss function, which is dictated by the type of task (classification or regression) and by data-specific factors. Secondly the size and number of trees to construct must be adapted to the data [24]. For example, the size of a decision tree determines the order of interactions between features that can be expressed by the model. A decision tree with only a single splitting node, a so called *decision stump*, can only capture main effects, while a tree with two splits in the hierarchy before the terminal node is reached can also capture interactions between two features [25]. Another factor to consider when choosing the size of the decision tree is that smaller trees have a smaller risk of overfitting and are therefore preferable over larger trees [25]. Usually all trees are set to be the same size and the choice of tree size is determined by cross-validation (see Section 3.2.1) [25]. Similarly, fitting too many trees to the data can cause the gradient boosted tree model to overfit. Again, the optimal number of trees should be determined by cross-validation.

To prevent overfitting of the model regularization can be performed. A common technique is called *shrinkage* [24] whereby the contribution of each tree in the ensemble is reduced by scaling it by some constant $c \in (0, 1)$, often referred to as the *learning rate*. A smaller learning rate will require a larger number of trees to be trained in order to achieve comparable performance on the training set [25].

3.1.4 Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is an implementation of the gradient tree boosting method with several enhancements [33]. Instead of minimizing a loss function, XGBoost minimizes an objective function $\mathcal{L}_m$ that in addition to the loss function also includes a regularization term $\Omega(b_m)$ which limits the complexity of the model

$$\mathcal{L}_m = \sum_{i=1}^{N} L\left(y_i, f_{m-1}(\mathbf{x}_i) + b_m(\mathbf{x}_i)\right) + \Omega(b_m) \tag{3.8}$$

where

$$\Omega(b_m) = \gamma T + \frac{1}{2}\lambda \left\|\mathbf{s}\right\|^2, \tag{3.9}$$

T is the number of leaves and $\mathbf{s}$ is a vector of leaf values of the newly added gradient boosted tree b_m and where γ and λ are constants.

Apart from the inclusion of a regularization term, XGBoost differs from regular gradient tree boosting through a number of improvements in computational efficiency [33], some of which are described below. These computational improvements allow for the training time of XGBoost to be reduced and enable handling of larger data sets.

Take for example the problem of finding the best split in a tree. This could be solved using an exact greedy algorithm that evaluates all possible splits. If the feature is continuous, this involves first sorting the data by feature value, then checking all split points and subsequently repeating this for all features. An alternative to this computationally expensive approach is to use an approximate method, where only a number of candidate split points are evaluated. One of the contributions of XGBoost is how to identify these candidate split points. XGBoost also has the ability to handle sparse data well, which arises e.g. through one-hot-encoding of categorical features. XGBoost has the additional benefit of performing the sorting of the data, which is necessary both in the exact greedy and approximate split finding algorithms, in a parallelized manner. This further helps to speed up the computations.

3.2 Model performance evaluation

When constructing a model we may use a loss function to monitor how well the model fits to the training data in order to prevent underfitting and regularization to control the complexity of the model and thereby avoid overfitting. However, no information is provided regarding how well the model has captured the true signal in the data and how well it will generalize to new data. For this purpose it is essential to have the model to generate predictions based on a new, previously unseen, dataset. By comparing the model prediction with the true target for each observation in the new dataset, a measure of the performance of the model can be computed [24].

There are two main purposes for evaluating the performance. The first is *model selection*, i.e. choosing the best model for the data. The models may be based on different machine learning methods or different choices of hyperparameters for the same type of machine learning method. In this setting the new dataset is often

referred to as the *validation set* [25]. The second purpose is *model evaluation*, to understand how well the trained model will perform when deployed. Here, the new dataset is called the *test set* [25].

The subdivision of the available data into training, validation and test sets means that less data will be available for fitting the model. If the dataset is small, making the random division of data into these three parts representative poses an additional challenge [24]. We may counteract these effects in low-data situations by using *k-fold cross-validation* (Section 3.2.1) for model selection or evaluation. Different performance metrics that can be used to compare or evaluate models are described in Section 3.2.2.

3.2.1 K-fold cross-validation

In k-fold cross-validation the available data is randomly divided into k roughly equal-sized parts or *folds*. One of the folds is used for validation or testing, depending on the purpose of the performance evaluation. The observations in the remaining folds are then used for training the model. Once the model has been fitted it is evaluated on the fold that was held out for validation or testing using a suitable metric. By repeating this model training and evaluation so that each fold acts as a validation or test set exactly once, we obtain k trained models and k measures of model performance. Usually the average of these k performance measures are used as an estimate of the model's performance [24].

When the aim of performing k-fold cross-validation is model selection, the model that achieves the best model performance rather than the model performance per se, is of primary interest [24]. When the compared models differ only with respect to the choice of hyperparameters, model selection is referred to as *hyperparameter tuning*. The hyperparameters that result in the best model performance are chosen as the final hyperparameters in the model.

The most common choice for k is either $k = 5$ or $k = 10$ [24]. Choosing a smaller k may lead to a biased estimate of the true model performance as less data will be available for fitting the model than if k had been large. A larger k, on the other hand, risks increasing the variance of the model performance measure as all models are trained on approximately the same dataset and the performance measures become correlated. This causes the mean performance of the models to have high variance.

3.2.2 Performance metrics

In this section, performance metrics for binary classification models are first described, followed by a metric used for regression models.

In binary classification the data is divided into two classes, which are typically referred to as the *positive* and *negative* class. The positive class is considered to be the class of particular interest. In order to define the classification metrics it is helpful

to consider the confusion matrix in table 3.1.

Table 3.1: Confusion matrix showing possible outcomes of a binary classification model. Rows are true classification, columns are predicted classification. TP = true positive, FN = false negative, FP = false positive, TN = true negative.

		Predicted	
		+	-
Actual	+	TP	FN
	-	FP	TN

This matrix presents the four different outcomes that can result when we perform binary classification. TP is the number of true positives, i.e. the number of observations with a positive target label that the model classified as positive. Similarly, TN, represents the number of actually negative observations that the model classified as negative (true negative). The sum of TP and TN are the number of accurately classified observations. FN is the number of false negatives, the positive observations that were missed by the model, while FP is the number of false positives, the negative observations that were incorrectly called positive by the model. FN and FP represent misclassified observations.

A common classification metric is *accuracy* [30], defined by

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP}. \tag{3.10}$$

This metric represents the proportion of correctly classified observations. If we are instead interested in the proportion of actual positives that were classified as positives, then *sensitivity*, or equivalently *recall* or *true positive rate* (TPR) [30] is the metric of choice

$$sensitivity = \frac{TP}{TP + FN}. \tag{3.11}$$

Specificity is the analogous metric for the negative class [30] and is defined as

$$specificity = \frac{TN}{TN + FP}. \tag{3.12}$$

We may also be interested in the proportion of actually negative observations that are incorrectly classified as positive, the *false positive rate* (FPR). This corresponds to 1-specificity [24].

Many binary classification methods result in models that produce a probability that an observation belongs to the positive class. This probability can be converted into a prediction label, i.e. *"positive"* or *"negative"* using a threshold. Typically, this threshold is set at 0.5, such that observations receiving a probability higher than

0.5 are classified as positive, and otherwise as negative. However, the choice of a 0.5 threshold may not be optimal for the data [24].

The *receiver operating characteristics* (ROC) curve presents a way to visualize the performance of a model for all choices of thresholds [24]. It plots the TPR versus the FPR and can be thought of as the proportion of true negatives identified as false positives that we must tolerate in order to detect a proportion of the true positives. A ROC curve that passes through the point (0,1) is a perfect classifier with TPR = 1 and FPR = 0. A diagonal ROC curve suggests that the classifier is no better than a random prediction.

For easier comparisons between classifiers, the *area under the ROC curve* (AUC), is often calculated [30]. A perfect classifier has AUC=1 while a random classifier will receive a value close to 0.5. The AUC is a more expressive performance metric than accuracy as it takes all thresholds into account and thereby gives a more complete picture of how well the model captures the signal of the positive class in the data.

The *coefficient of determination* or R^2 is a commonly used metric for evaluating regression models. It is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - f(\mathbf{x}_i))^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2}, \tag{3.13}$$

where $\bar{y}$ is the mean outcome value of the validation or test data [34]. A regression model which perfectly fits the data has a R^2 value equal to 1.

3.3 Model interpretability

While model performance is an important consideration when deploying a model, the interpretability of models has recently received increasing attention [35, 36, 37]. Some types of models are generally more interpretable than others [28]. For example, in decision tree models a prediction can be explained by a set of rules. Unless the tree is large, such a model will be interpretable. In contrast, a model like XGBoost is a *black box* model as its predictions may be generated by contributions of several hundreds of decision trees. Black box models often outperform simpler models, so they are therefore frequently used in practice. In many cases there is a trade-off between interpretability and model performance [35].

There are several reasons why model interpretability matters. Firstly, understanding the prediction made by the model offers insight into what the model has learned about the data [35, 38]. Secondly, understanding which features a model relies on when making a prediction can help us to discover possible flaws and biases in the data and thereby lead to the construction of better and more fair models [35, 28, 39]. As the adage goes: *"garbage in, garbage out"*. Lastly, model interpretability is an important means for gaining trust in the models, as models and algorithms are increasingly integrated into different aspects of our daily lives [28, 39].

Recent advances in model interpretability allow us to begin to understand what high-performing models have learned. One of the most significant recent contributions to the model interpretability field has been the application of Shapley values [40], that originate from cooperative game theory, to explain model predictions [41]. The Shapley value is described in Section 3.3.1, an efficient way to exactly compute it for tree-based models is presented schematically in Section 3.3.2 and how Shapley values can be combined to provide a global understanding of model behavior is explained in Section 3.3.3.

3.3.1 Shapley value

Additive feature attribution methods calculate the contribution of each feature i to the model prediction. These are so called *local methods* of model interpretability, that explain the prediction for a single observation $\mathbf{x}$ [41]. The presence of a feature in a model can either lead to an increase in the value predicted by the model (have a positive contribution) or lead to a decrease in the predicted value (have a negative contribution). By summing over the contributions of all p features we obtain a value that expresses by how much the information contained in the features of observation $\mathbf{x}$ causes the prediction to shift relative to a model prediction ϕ_0, which is the average model prediction $\mathbb{E}[f(\mathbf{X})]$ based on the entire dataset [42]. Ideally, we want the sum of ϕ_0 and the individual feature contributions $\phi_i(f, \mathbf{x})$, $i = 1, 2, \ldots, p$ to equal the model prediction, i.e.

$$f(\mathbf{x}) = \phi_0(f, \mathbf{x}) + \sum_{i=1}^{p} \phi_i(f, \mathbf{x}). \tag{3.14}$$

This property is called *local accuracy*. Additional desirable properties for additive feature attribution methods are *consistency* and *missingness* [41]. Consistency means that if a change in the model causes the true importance of a feature in the model to increase, then the calculated feature contribution should either stay the same or increase, regardless of which other features are present. Missingness states that a feature which has no impact on the prediction should be assigned a contribution of zero. For example, if an additive feature attribution method were to be applied to a decision tree where feature i was not present in any of the nodes, we want $\phi_i(f, \mathbf{x}) = 0$ to hold.

The *Shapley value* is the only additive feature attribution method that satisfies all three properties [41]. This value is obtained by calculating how the model prediction changes on average when feature i is added to all possible groups of ordered features (including the empty set). Letting $\mathcal{R}$ be the set of all feature orderings, R a feature ordering and P_i^R the set of all features that appeared before feature i in ordering R, the Shapley value for feature i given the model f and the observation $\mathbf{x}$ is

$$\phi_i(f, \mathbf{x}) = \frac{1}{p!} \sum_{R \in \mathcal{R}} \left[f_{\mathbf{x}}(P_i^R \cup \{i\}) - f_{\mathbf{x}}(P_i^R) \right]. \tag{3.15}$$

Since the prediction is not affected by the order in which features are added, e.g. $f_{\mathbf{x}}(\{a, b\}) = f_{\mathbf{x}}(\{b, a\})$ for two features a and b, we can instead redefine Eq. 3.15 using sets of features [43]. Letting S be a subset of all features F excluding feature i, the Shapley value for feature i is

$$\phi_i(f, \mathbf{x}) = \sum_{S \subset F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left[f_{\mathbf{x}}(S \cup \{i\}) - f_{\mathbf{x}}(S) \right]. \tag{3.16}$$

Here $|S|!$ is the number of permutations of the features in set S, $(|F| - |S| - 1)!$ the number of permutations of the features appearing after set S and feature i and $|F|!$ is the number of permutations of all features. The factor

$$\frac{|S|!(|F| - |S| - 1)!}{|F|!} \tag{3.17}$$

weights the sum in Eq. 3.16 to convert the calculations over sets of features into calculations over all feature orderings. This alternative formulation simplifies the computation of the Shapley value and is therefore used in practice.

When computing $f_{\mathbf{x}}(T)$ for some subset T of features for observation $\mathbf{x}$ we must take into account the subset of $\bar{T}$ of features that are dropped from observation $\mathbf{x}$ [42]. This is expressed by the *marginal expectation*

$$f_{\mathbf{x}}(T) = \mathbb{E}\left[f(\mathbf{x}_T, X_{\bar{T}}) \right] \tag{3.18}$$

where $X_{\bar{T}}$ is the distribution of the dropped features, $T \cap \bar{T} = \emptyset$ and $T \cup \bar{T} = F$. It was previously argued [41] that the *conditional expectation*, $\mathbb{E}\left[f(\mathbf{x}_T, X_{\bar{T}}) | X_T = \mathbf{x}_T \right]$ should be used to calculate $f_{\mathbf{x}}(T)$, however [42] showed that using the conditional expectation can lead to irrelevant features being assigned non-zero Shapley values, i.e. a violation of the missingness property.

Fortunately, initial implementations of Shapley values for model interpretability, such as ***SH****apley* ***A****dditive Ex****P****lanations* (SHAP), used the marginal expectation rather than the conditional expectation for easier computations [41]. The use of the marginal expectation was initially motivated by the fact that it is the same as the conditional expectation if the features are independent [41]. Since SHAP already implemented the marginal expectation, the modified theoretical rationale for the computation of $f_{\mathbf{x}}(T)$ did not necessitate any change to the algorithm used in SHAP.

The Shapley value represents the difference between the average prediction $\mathbb{E}[f(\mathbf{X})]$ and the prediction $f(\mathbf{x})$ for observation $\mathbf{x}$ that can be explained by feature i [42]. To explain a prediction, the Shapley value must be calculated for all p features, which is computationally expensive. Different approaches have been developed to calculate approximate Shapley values, such as the *Shapley sampling values* method [43]. These methods can be used to explain the predictions of any machine learning model.

Methods for exact low-complexity computation of Shapley values have also been developed for explaining predictions of certain types of machine learning models, such as *TreeExplainer* for tree-based models.

3.3.2 TreeExplainer

TreeExplainer is a computationally efficient method for the exact computation of Shapley values for tree-based models [26]. As was explained in Section 3.1.1, the predictions made by a decision tree are determined by a set of rules that together define a leaf. Instead of performing computations for each subset of features, TreeExplainer performs calculations per leaf [26]. TreeExplainer can also do this for sums of trees and can thereby be used to calculate Shapley values for black-box tree-based models like XGBoost in an efficient manner.

3.3.3 Global feature attribution

Shapley values are local and only express feature attributions of a single observation $\mathbf{x}$. While this helps us to understand why the model makes a specific prediction, it does not provide a global understanding of how the model behaves. To gain a global understanding we could repeat the calculation of Shapley values for all observations and features in the dataset [26]. With tools such as TreeExplainer, this approach has become computationally feasible in practice [26].

3.4 Statistical testing

This section covers statistical testing of categorical data, where data can be represented as a contingency table of counts (Table 3.2). For example, we may wish to know whether subjects from a certain population are more likely to belong to a category A than subjects from another particular population are. To test this hypothesis we may use Fisher's exact test, which is especially suitable for small sample sizes [44].

Table 3.2: A contingency table of two populations showing the number of belonging to category A and B as well as category and population totals.

	Population 1	Population 2	Total
Category A	n_{A1}	n_{A2}	$n_{A.}$
Category B	n_{B1}	n_{B2}	$n_{B.}$
Total	$n_{.1}$	$n_{.2}$	$n_{..}$

3.4.1 Fisher's exact test

The null hypothesis of Fisher's exact test is that the probability of belonging to a category A is independent of the population,

$$H_0 : \pi_{A|1} = \pi_{A|2}, \tag{3.19}$$

where the categories and populations are defined as in Table 3.2, $\pi_{A|1}$ is the probability of subjects from population 1 belonging to category A and $\pi_{A|2}$ is the corresponding probability for subjects from population 2.

Given that the population and category totals $n_{.1}$, $n_{.2}$, $n_{A.}$, and $n_{B.}$ are fixed, n_{A1} has a hypergeometric distribution [44] and the probability that n_{A1} assumes a particular value k is

$$P(n_{A1} = k) = \frac{\binom{n_{A.}}{k}\binom{n_{B.}}{n_{.1}-k}}{\binom{n_{..}}{n_{.1}}}. \tag{3.20}$$

For the one-tailed alternative hypothesis $H_A : \pi_{A|1} > \pi_{A|2}$, the p-value is the sum of the probability of the observed value of n_{A1} and the probabilities of observing all possible values for which k is larger than this value.

4
Methods

Machine learning models learn to associate patterns of input features with an outcome. Both aims of this thesis can be expressed as such learning tasks: the first aim, to identify possibly drug-related AEs, involves finding AEs that are associated with a treatment arm; while the second aim, to identify patient subgroups in which the risk of a particular drug side-effect is potentially increased, concerns elucidating which patient characteristics that are associated with an increased risk of developing a specific drug side-effect.

The objective is thus to use modelling as an exploratory data mining tool, instead of producing a model for predictive purposes. Put differently, we focus primarily on *what* the model has learned rather than *how well* the model has learned. Recent developments in model interpretability (Section 3.3), such as the Shapley value and its implementation, SHAP, has given us tools to understand what the model has learned.

A challenge when performing exploratory analysis is the validation of the results. For this reason we have chosen to analyze the AEs from clinical trials of a well-characterized drug, Symbicort. Any suspected drug-related AEs identified can thereby be compared to the currently recognized side-effects of this drug. However, the possible risk factors for developing a particular drug side-effects have been less explored and the findings of the second aim of this thesis should therefore be considered hypothesis-generating.

The method for the identification of possible drug-related AEs is outlined in Section 4.1 and is followed by the method used to identify variables potentially associated with an increased risk of having a particular drug side-effect (Section 4.2).

4.1 Identification of possibly drug-related adverse events

The method that is described here is an adaptation of the *inside-out data mining* method described by Southworth & O'Connell (2009) [5]. The principle of this method is to first construct a model to predict the treatment a subject received based on the AEs they experienced during the study and then inspect what the model has learned. In this manner the model can help us to understand which AEs or patterns of AEs that are possibly associated with a drug.

While we recognize that in reality it is not the case that AEs determine treatment, we must remind ourselves that a model does not represent a causal relationship but rather a correlation between the features and the outcome. As correlation, unlike causation, is bidirectional, we are free to use the AEs to predict which treatment the subject received.

The subjects in the data used for this thesis have either received Symbicort (encoded as 1) or placebo (encoded as 0), thus the problem is a binary classification task. As described previously (Section 2.2), the AEs are encoded in binary format. AEs that occur only once across both treatment arms are excluded as it is not possible to evaluate the relevance of such AEs and they likely have limited impact in the model. The consequences of removing these AEs is discussed in Section 2.3. Table 4.1 contains a schematic view of the data, where each row represents a different subject.

Table 4.1: Schematic view of the data used to identify drug-related adverse events. Each subject is represented on a separate row, while the columns indicate whether the subject had different adverse events (1=yes, 0=no) and which treatment the subject received (1=Symbicort, 0=placebo).

Subject	AE 1	AE 2	AE 3	Treatment
Subj 1	1	1	0	1
Subj 2	1	0	1	1
Subj 3	0	0	0	1
Subj 4	0	1	0	0
Subj 5	0	0	0	0

With this data an XGBoost classification model with a cross-entropy loss function is tuned using 5-fold cross-validation with AUC as metric. Table C.1 in Appendix C lists the hyperparameters and the respective values that are investigated using grid search, whereby models are constructed using all combinations of hyperparameter values. The model that achieves the highest average AUC is selected and re-trained using all the data. The trained model and all data are then used to calculate a SHAP value for each subject-AE pair, which differs from the original inside-out data mining method in which permutation importances are calculated for each AE. One advantage of using the SHAP value over permutation importance is that the SHAP value allows for the feature importance to be calculated on a subject level, in addition to on a global level.

The SHAP value represents the contribution of the AE to the treatment prediction for a specific subject, compared to the average prediction, which in this case is the probability that a subject belongs to the Symbicort arm of the study. If the SHAP values for an AE are further away from zero (either in the positive or negative direction), it means that that AE is more informative to the model and impacts the

prediction of the model more than an AE where the SHAP values of all subjects are close to zero. Thus, if we take the absolute value of the SHAP values for all subject-AE pairs and then average these by AE, we expect AEs with a higher mean absolute SHAP value to be more important for the prediction than AEs where the mean is close to zero.

Sorting the AEs from highest to lowest mean absolute SHAP value we obtain a ranking of the AEs from highest to lowest importance in predicting any treatment. Boxplots of the raw SHAP values are then used to understand which treatment the highest ranking AEs are associated with. Any high-ranking AEs where the presence of the AE is generally associated with a positive SHAP value, will be considered as possible drug-related AEs, i.e. possibly related to Symbicort. The findings are validated by comparing them to the currently recognized side-effects of Symbicort (Appendix A). Since there are 723 subjects who received Symbicort in the dataset, being able to detect side-effects with an incidence $< 0.1\%$ is considered unlikely. AEs which are associated with a negative SHAP value are related to the placebo treatment, according to the model.

Tree-based models such as XGBoost can capture interactions between features. In order to understand what patterns of AEs the model has learned to associate with a treatment, the SHAP values of each subject are transformed into a two-dimensional space using *t-distributed Stochastic Neighbor Embedding* (t-SNE) with the default hyperparameters and then plotted as a scatter plot, with each dot representing a subject. t-SNE is a non-deterministic non-linear dimensionality reduction method that can be used for producing lower dimensional representations of data for visualization purposes [45]. By highlighting the plot by the highest-ranking AEs, we can investigate which treatment-related AEs co-occur.

4.2 Identification of subgroups potentially having an increased side-effect risk

Typically, drug side-effects only occur in a subset of patients. It is therefore of interest to understand which factors are associated with an increased risk of developing a particular side-effect. A similar analysis is commonly performed using efficacy endpoints, where the aim is to identify subgroups of patients with differential treatment effect, and is known as *subgroup analysis*. While the use of subgroup analysis for safety data has been suggested [3], these methods have generally not been developed for this purpose.

In this section a method is described that is based on the *Virtual Twins* (VT) subgroup analysis method proposed by Foster et al. (2011) [46]. This method was initially developed with the aim of identifying subgroups of subjects in placebo-controlled clinical trials who experience an enhanced effect when treated with a drug. Here an adaptation of the VT method is used to identify risk groups of patients at an increased risk of developing the Symbicort side-effects *oral candidiasis*

(a fungal infection in the mouth) and *dysphonia* (hoarse voice).

Section 4.2.1 introduces the application of the adapted VT method to the characterization of drug side-effects. The variables that are considered in the subgroup analysis are briefly described in Section 4.2.2 and the classification and regression models that form the basis of the adapted VT method are presented in Section 4.2.3.

4.2.1 Adapted Virtual Twins method

When applying the adapted VT method to the analysis of suspected drug side-effects we are interested in identifying patient characteristics that are associated with an increased risk of having the particular AE when treated with the drug. However, AEs that are drug-related may also be present in subjects not receiving the drug. Thus, in order to understand which variables are linked to the drug, we must correct for variables that are associated with the development of the AE in general, regardless of treatment. This is what the adapted VT method aims to achieve.

The adapted VT method (Fig. 4.1) involves the construction of three models, where the two models that are initially developed are used as a basis for the construction of the third model. As a first step, data from subjects who received the drug treatment is used to train a model to predict the probability of a subject experiencing a particular drug side-effect. A second model is subsequently trained for the same prediction task, but instead using data from subjects who received the placebo treatment. We will hereafter refer to these models as the drug model and placebo model, respectively.

These two models are then used to generate a prediction for each subject, i.e. each subject receives two probabilities of getting the AE: one based on the model for the treatment they received and one based on the treatment they did not receive. The latter prediction is referred to as the virtual twin. The method described hereafter differs from the original VT method.

The next step in the adapted VT method involves subtracting the predictions of the placebo model from the predictions of the drug model, following a logit transformation of the probabilities into scores. The differences in scores are then used as outcomes in a regression model that is trained using all data (i.e. input variables from all subjects). By calculating the SHAP values of the trained regression model and computing the mean absolute SHAP value by variable, we can learn which variables are most informative to the model when explaining any differences in scores between the drug and placebo models, as these will have the highest mean absolute SHAP value.

Variables that can explain differences in the risk of experiencing the side-effect under different treatments in the same subject are the result of an association between the variable and the side-effect risk that has been picked up by either the drug or placebo models or by both models. Therefore, the identified most informative vari-

ables are used as a starting point to explore which variables may be risk factors for the development of the drug-related AE, rather than general risk factors for developing the AE or noise stemming from the placebo model.

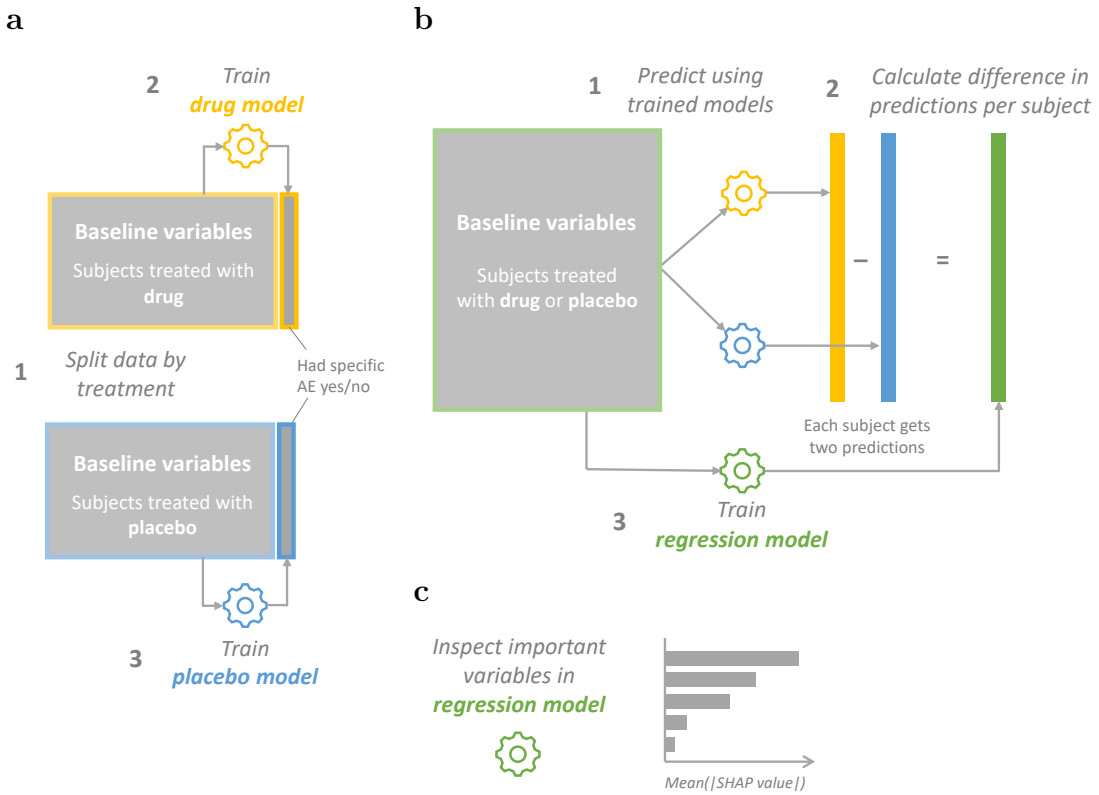


Figure 4.1: (a) Data consisting of a number of variables measured at baseline (before first dose of treatment is administered) as well as information regarding whether a subject had a specific adverse event is divided according to which treatment each subject received. For subjects who received drug treatment, a model (the drug model) is trained to predict the probability of a subject having a particular adverse event based on the baseline variables. A second model (the placebo model) is trained for the same prediction task, but instead using data from subjects who received placebo treatment. (b) The trained drug and placebo models are then applied to all subjects, such that each subject gets two predictions: one based on the treatment they received and one based on the treatment they didn't receive (the virtual twin). Predictions from the placebo model are subtracted from the drug model. A regression model is then trained using the baseline variables as features and the difference in predicted probabilities as outcomes. (c) The mean absolute SHAP values of the baseline variables in the regression models are calculated, allowing for variables that are most informative to the regression model when explaining any differences in model predictions to be identified. Such variables are associated with an increased risk of having the AE according to the drug model, placebo model or both models.

In order to understand the nature of the identified signal, we again use SHAP values but now from the drug and placebo models. SHAP values for the drug model are

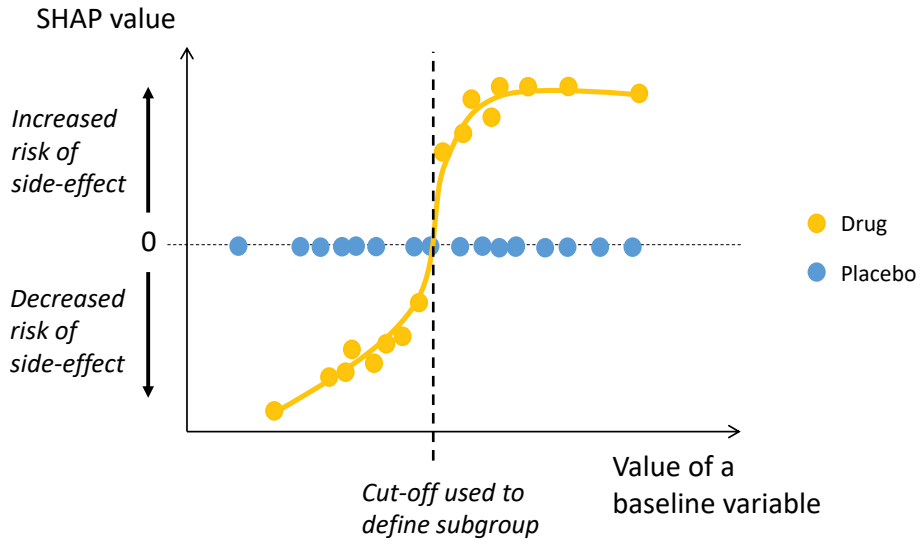


Figure 4.2: Illustration of a plot of the SHAP values versus observed baseline variable values for the drug (yellow) and placebo (blue) models. Each point represents a different subject. A positive SHAP value indicates an increased risk of the drug side-effect, while a negative SHAP value indicates a decreased risk, according to the model. In this particular example the placebo model does not identify any association between the variable and the risk of experiencing the side-effect. In the drug model the risk is increased for higher values of the variable. By fitting a curve through the points and calculating the value at which the SHAP value is zero (horizontal dashed line), a cut-off (vertical dashed line) can be calculated which can be used to define the subgroup of patients potentially having an increased risk of the side-effect.

computed based on subjects who received the drug and SHAP values for the placebo model are similarly computed based on subjects who received placebo. We can then inspect the SHAP values of these two models for each of the identified informative variables by plotting the SHAP values against the observed baseline variable values (Fig. 4.2). Here a positive SHAP value represents that the model has learned that a variable confers an increased risk of having the side-effect in a particular subject, while a negative SHAP value represents a decreased risk. A SHAP value of zero means that the variable does not influence the model prediction for a subject and is neither associated with an increased nor decreased risk of having the side-effect.

Thus, if the SHAP values of a model are zero for all subjects we can conclude that this model does not identify any association between the variable and the risk of the side-effect. For a drug-specific risk factor of the AE we expect the drug model to have non-zero SHAP values while the SHAP values of the placebo model are all zero, as is illustrated in Figure 4.2. A general risk factor, on the other hand, will result in non-zero SHAP values in both the drug and placebo models.

The plots of the SHAP values against variable values will also inform us for which

variable values the risk of the AE is higher and for which values the model assigns a lower risk. This information is used to define cut-off values where the risk of the side-effect is higher and thereby define the subgroup of patients who are at an increased risk of the side-effect (Fig. 4.2). Specifically, for continuous variables a LOESS (locally estimated scatterplot smoothing) curve is fitted per treatment arm and the value at which this curve has a SHAP value of zero is used as the cut-off. In case the cut-off cannot be established for one of the treatments, the same cut-off is used for both treatments. Validation of the identified subgroups is performed using Fisher's exact test.

4.2.2 Variables included in the subgroup analysis

A range of variables that were either known or measured at the beginning of the study, before the first dose of the treatment was administered, are considered in the subgroup analysis. These include variables related to demographics, laboratory data, results from various medical examinations, medical history, patient-reported outcomes and concomitant medication. In addition, the study the subject belongs to is included as a variable.

Categorical variables are encoded as dummy variables with one level removed to reduce correlation between variables. Levels of categorical variables that only contain one subject are removed and any missing values in the remaining levels are replaced by the respective modes. Ordinal variables are recoded as integer values. Any missing numeric values are imputed using *k-nearest neighbors* with $k = 5$, whereby the mean variable value of the k closest subjects (by Euclidean distance following normalization) is used. Finally, one variable is removed in each pair of highly correlated variables.

4.2.3 Classification and regression models

For each studied side-effect, two XGBoost classification models with cross-entropy loss functions are constructed that use the variables mentioned in Section 4.2.2 to predict the side-effect; one using data from subjects who received Symbicort and the other using data from subjects who received placebo. These models are tuned using grid search and 5-fold cross-validation with AUC as metric. Per treatment arm, the model having the highest cross-validation AUC is selected and re-trained with all the data from subjects receiving the particular treatment. All hyperparameter values which are investigated are listed in Table C.2 in Appendix C.

The regression model that builds on the pair of drug and placebo models is an XGBoost regression model with a squared-error loss function that is tuned and trained in an analogous manner, using R^2 as metric. The model with the highest cross-validation R^2 is selected and re-trained using all data. Table C.1 in Appendix C contains the hyperparameter values that are searched during the tuning of this model.

5
Results

Results of the analyses aiming to find possible Symbicort-related AEs and variables associated with an increased risk of developing specific side-effects when treated with Symbicort are presented in Section 5.1 and 5.2, respectively.

5.1 Identification of possibly drug-related adverse events

A description of the selected model is found in Appendix D.1. This model achieved a mean AUC of 0.56, as can be seen from the ROC curve in Figure 5.1, and performs only slightly better than a random classifier (red dashed line).

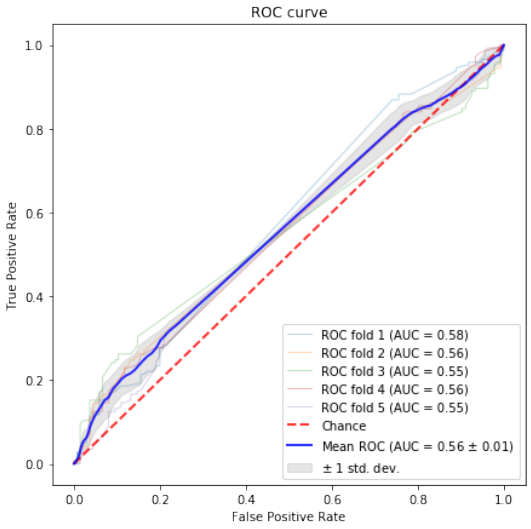


Figure 5.1: Receiver operating characteristics (ROC) curves following 5-fold cross-validation of the selected classification model for assigning subjects to a treatment arm based on their experienced adverse events. The blue line indicates the mean ROC curve, the dashed red line the expected ROC curve for random classification and the shaded gray area represents ± 1 standard deviation of the mean ROC curve. The mean area under the ROC curve (AUC) is 0.56.

The ten AEs with the highest mean absolute SHAP values in the model are shown in Figure 5.2. These AEs, will hereafter be referred to as the highest-ranking AEs and represent the AEs that are most informative in the model when predicting treatment arm. The most important AEs according to the model are oral candidiasis (a

fungal infection in the mouth), COPD and dysphonia (hoarse voice). While COPD is a disease that all subjects in the study suffer from it can also occur as an AE. The criteria for this include that the COPD symptoms are serious, that the subject manifests new symptoms of the disease or that the COPD symptoms cause the subject to discontinue the study [21].

The known side-effects of Symbicort that are identified by the method are highlighted in yellow in Figure 5.2 and include oral candidiasis, dysphonia, dizziness, muscle spasms and anxiety. Oral candidiasis corresponds to *"candida infections in oropharynx"* and dysphonia is included in the side-effect *"mild irritation in the throat, coughing, hoarseness"*. Both of these side-effects occur in approximately 1% to 10% of patients, while dizziness, muscle spasms (*"muscle cramps"*) and anxiety (*"agitation, restlessness, nervousness, sleep disturbances"*) occur in 0.1% to 1% of patients, according to Appendix A. COPD, nasopharyngitis, dyspnoea, pneumonia and sinusitis are related to some treatment (either Symbicort or placebo) according to the model, but are not known drug side-effects and are shown in gray in Figure 5.2.

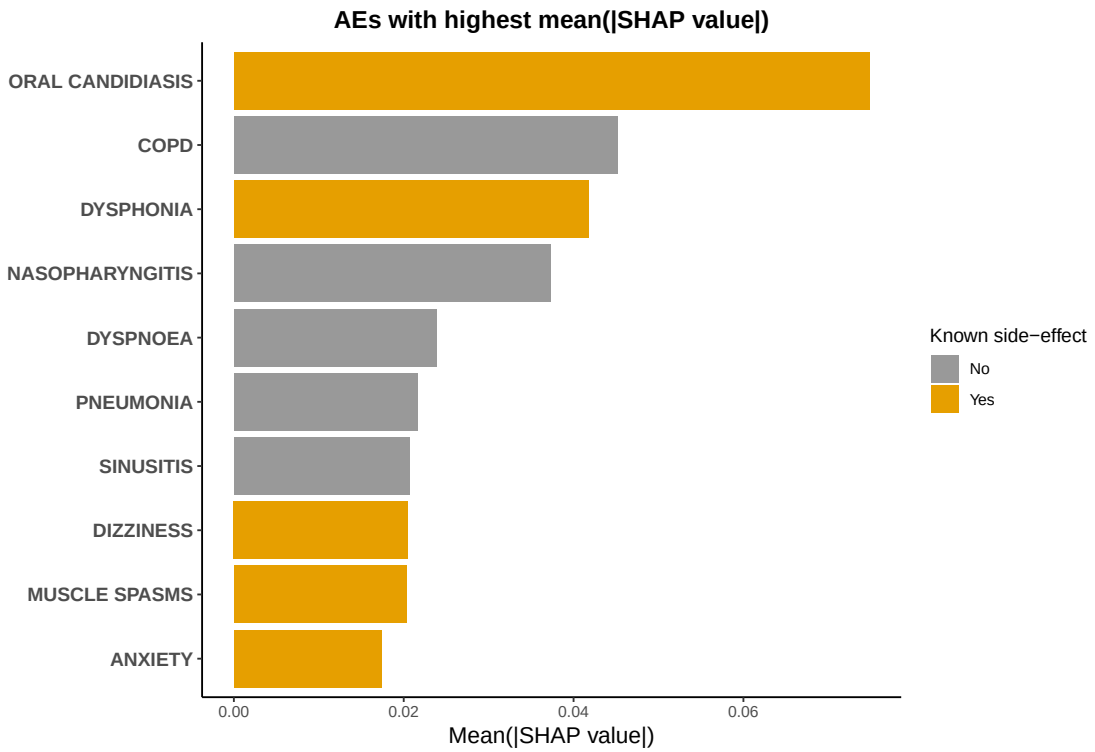


Figure 5.2: The ten adverse events with the highest mean absolute SHAP values, representing AEs that are most informative in the model when predicting treatment arm. Known Symbicort side-effects are highlighted in yellow, while other AEs are shown in gray.

In order to determine which treatment the model associates with each AE, we inspect the SHAP values of the highest-ranking AEs. In Figure 5.3, the SHAP values of all subjects have been grouped by AE and whether they had the AE or not; yel-

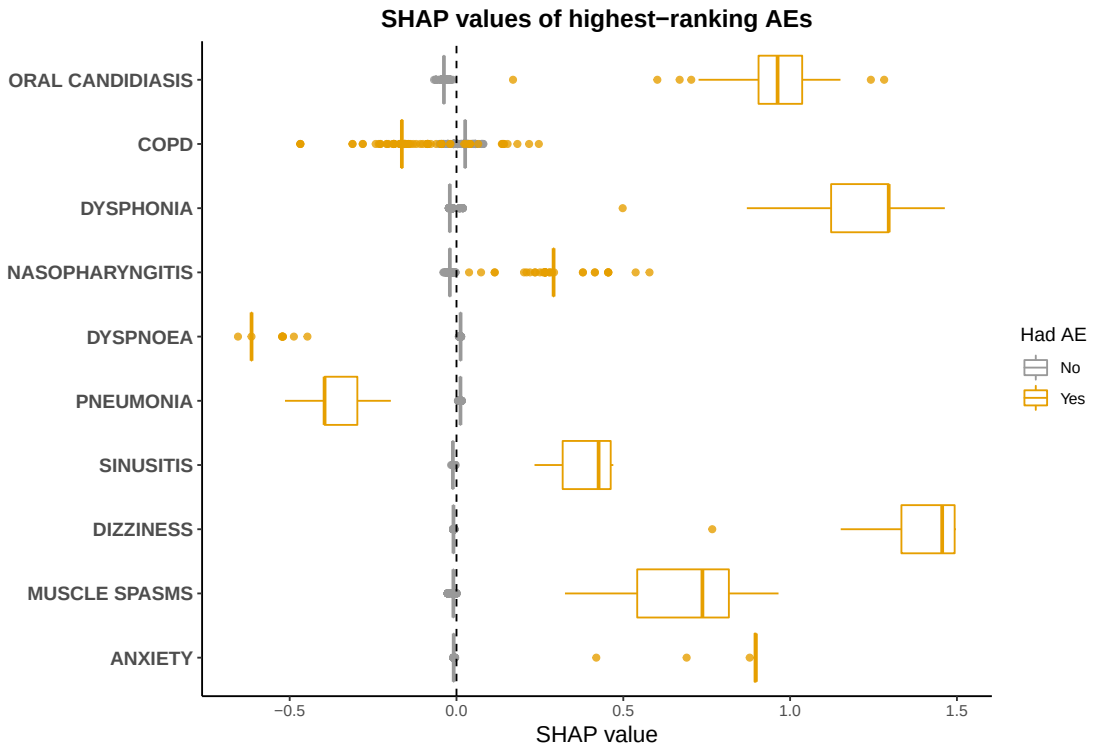


Figure 5.3: Boxplots of SHAP values for the highest-ranking AEs, grouped by if subject had AE (yellow) or not (gray). Adverse events for which the yellow boxplots appear to the right of the dashed line are to be considered as suspected drug side-effects.

low boxplots represent subjects who had the AE and gray boxplots subjects who did not have the AE. In this context a positive SHAP value corresponds to an increased probability that the subject received Symbicort according to the model, while a negative SHAP value that the model assigns a decreased probability that the subject received Symbicort.

As expected, the absence of an AE (gray boxplot in Figure 5.3) is generally associated with a SHAP value close to zero. Since most AEs are uncommon, the absence of an AE is not considered informative to the model. On the contrary, the presence of an AE (yellow boxplot) is more informative to the model when predicting treatment arm and may result in a non-zero SHAP value.

According to the model, oral candidiasis, dysphonia, nasopharyngitis, sinusitis, dizziness, muscle spasms and anxiety are possible side-effects of Symbicort while COPD, dyspnoea (shortness of breath) and pneumonia are associated with placebo treatment. In addition, these findings are even supported by the data, as the incidence of these AEs is consistently higher in the respective treatment arm that the model associates the AEs with (Appendix E).

Out of the seven possible drug-related AEs identified by the method, five are known

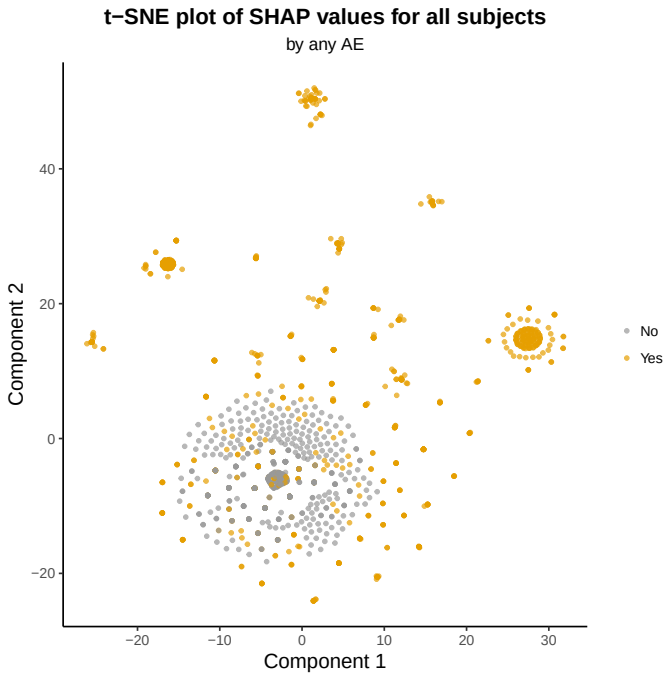


Figure 5.4: t-SNE plot of the SHAP values for all subjects. Yellow dots indicate subjects who had at least one adverse event, while gray dots represent subjects who did not experience an adverse event.

side-effects of Symbicort: oral candidiasis, dysphonia, dizziness, muscle spasms and anxiety. All of these AEs except anxiety are significantly over-represented in the Symbicort arm by Fisher's exact test at a significance level of 0.05 (Appendix E). The lack of a statistically significant result for the AE anxiety would have led to this AE not being identified as a possible side-effect if traditional statistical testing alone would have been employed.

The three AEs that the model associates with placebo treatment are not considered to be side-effects of Symbicort (Appendix 2.3). In addition, both COPD and dyspnoea (shortness of breath) are expected complications in COPD patients not receiving active treatment.

The model thus correctly classifies eight out of the ten highest-ranking AEs. Nasopharyngitis and sinusitis were both incorrectly classified as possible Symbicort side-effects. However, these results are not statistically significant (Appendix E) and may be due to the limited amount of data available for analysis.

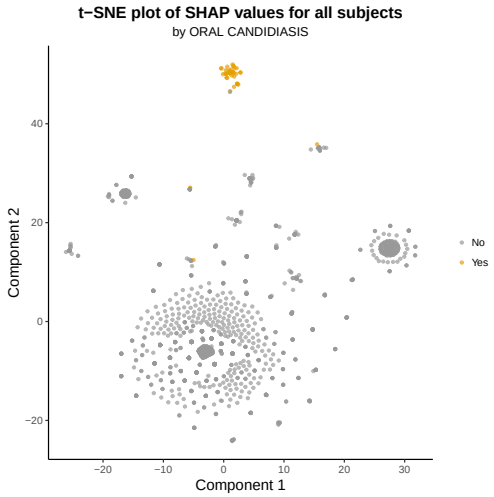
The remainder of this section explores the patterns of AEs that are learned by the model by using a two-dimensional representation of the SHAP values of each subject (Figure 5.4), where each dot represents a subject. This can be thought of as a simplified view of what the model has learned and does not necessarily correspond to reality. In Figure 5.4 subjects without any AE are displayed as gray dots and are clustered together, while subjects with at least one AE are displayed as yellow dots.

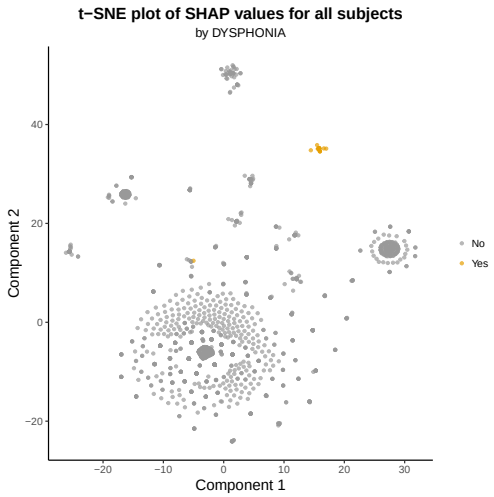
Some of the subjects with some AE are interspersed with the gray dots and are thus indistinguishable from subjects with no AE according to the model. Other subjects with some AE are represented as clusters at various distances from the cluster of subjects with no AE.

Highlighting the t-SNE plot by some of the highest-ranking AEs (Fig. 5.5) allows us the see which patterns of AEs the model has learned to associate with treatment. This allows general co-occurrences of AEs to be identified. In addition, this visualization can also reveal whether the model has learned that an AE belongs to one pattern (represented as a distinct cluster in the t-SNE plot) or several patterns (where the AE is present in multiple clusters).

Figure 5.5 shows that oral candidiasis and dysphonia are largely distinct clusters far away from the cluster of subjects with no AE. According to the model, the patients who are affected by oral candidiasis are different from the patients who are affected by dysphonia. Nasopharyngitis, on the other hand, is spread out across several clusters. This AE, which is also known as the common cold, affects a wide range of subjects and this is reflected in its more complex pattern of occurrence in Figure 5.5.

For the AEs most related to placebo, the AEs dyspnoea and pneumonia are mainly represented as distinct clusters (Fig. 5.6). COPD forms a large cluster, but also occurs in several other clusters. This is unsurprising since all subjects have a diagnosis as COPD and are at risk of COPD complications. Noteworthy in both Figure 5.5 and 5.6 is that the highest-ranking AEs appear as clusters that are farthest away from the cluster of subjects having no AE.





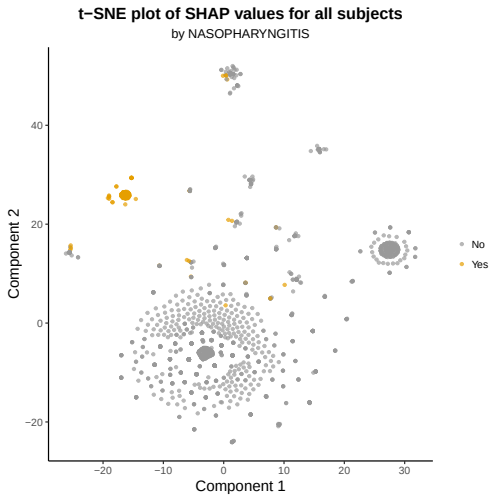
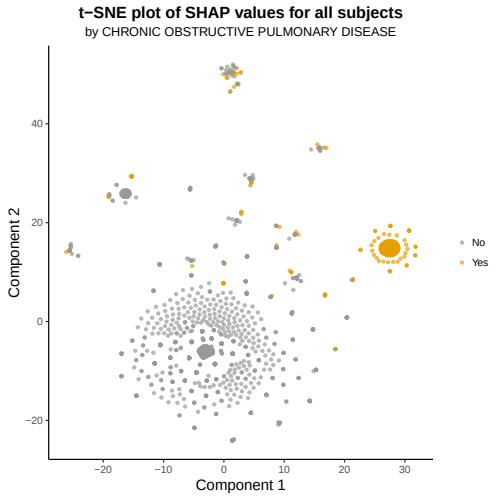
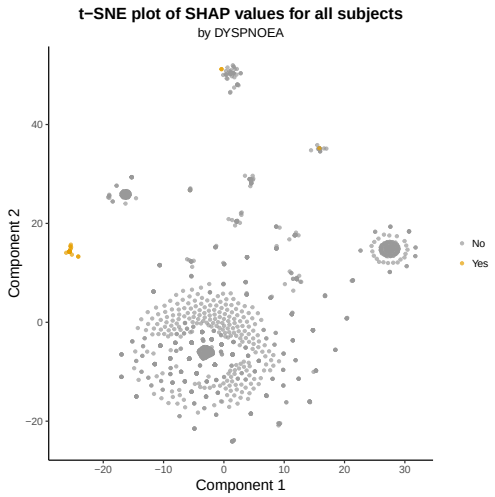


Figure 5.5: t-SNE plots of the SHAP values for all subjects, with subjects who experienced the three most important identified possibly drug-related adverse events highlighted in yellow: oral candidiasis (top), dysphonia (middle) and nasopharyngitis (bottom).





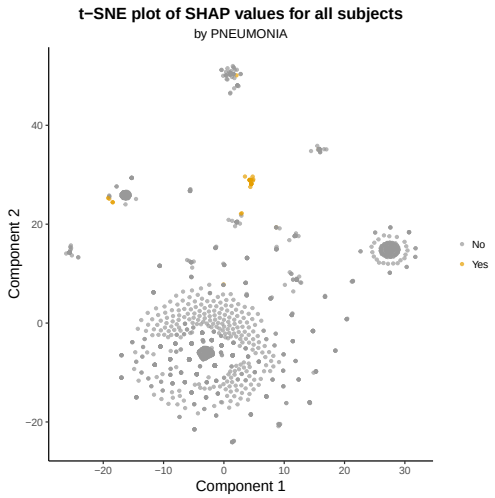


Figure 5.6: t-SNE plots of the SHAP values for all subjects, with subjects who experienced the three most important identified placebo-related adverse events highlighted in yellow: chronic obstructive pulmonary disease (top), dyspnoea (middle) and pneumonia (bottom).

5.2 Identification of subgroups potentially having an increased side-effect risk

The analyses aiming to identify variables associated with an increased risk of having the side-effects oral candidiasis and dysphonia when treated with Symbicort are presented separately in Section 5.2.1 and 5.2.2, respectively. Each analysis includes 607 variables that were measured or known prior to each subject receiving the first dose of the study treatment.

5.2.1 Oral candidiasis

The selected Symbicort, placebo and regression models that form the basis of this analysis are described in Appendix D.2.1. In the Symbicort arm 44 out of 723 subjects reported experiencing oral candidiasis, while 13 out of 677 subjects reported the event in the placebo arm. Figure 5.7 displays the corresponding ROC curves of the Symbicort and placebo models. The ROC curve of the Symbicort model shows that this model is clearly better than a random classification (mean AUC 0.78). In comparison, the ROC curve of the placebo model indicates a somewhat poorer performance of this model (mean AUC 0.72), which may be a consequence of the small number of subjects who reported having oral candidiasis in this treatment arm. The regression model achieved a mean 5-fold cross-validation R^2 of 0.97, meaning that the model fits very well to the data. However, as this model is based on the predictions of the Symbicort and placebo models, any weaknesses in the underlying models may be transferred to the regression model.

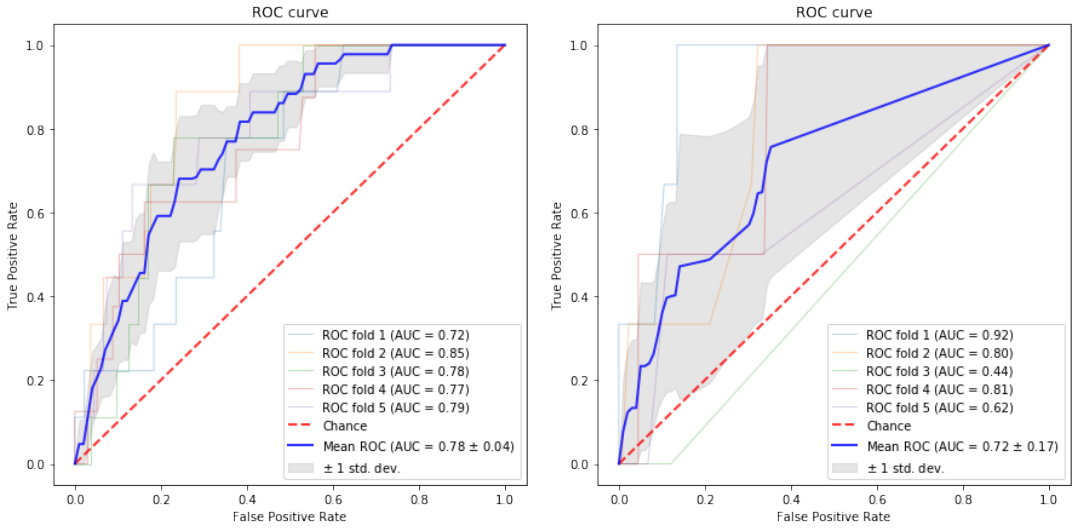


Figure 5.7: Receiver operating characteristics (ROC) curves following 5-fold cross-validation of the selected Symbicort model (left) and placebo model (right) for oral candidiasis. The blue line indicates the mean ROC curve, the dashed red line the expected ROC curve for random classification and the shaded gray area represents ± 1 standard deviation of the mean ROC curve. The mean area under the ROC curve (AUC) for the Symbicort and placebo models is 0.78 and 0.72, respectively.

The five variables with the highest mean absolute SHAP values in the oral candidiasis regression model are presented in Figure 5.8 and include: country US, antibiotics use, the concentration of neutrophils (a type of cell which is a part of the immune system) in the blood, smoking status and whether the patient suffered from anxiety at the initial study visit.

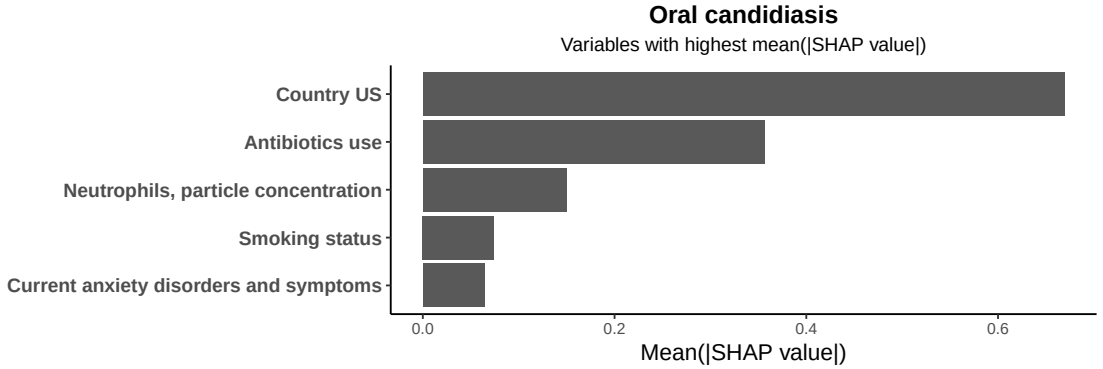


Figure 5.8: The five variables with the highest mean absolute SHAP values in the oral candidiasis regression model.

According to Figure 5.9, the Symbicort model associates subjects based in the US, antibiotics use, low neutrophil concentration and habitual smoking with in an increased risk of oral candidiasis. The placebo model identifies no such associations, but has instead learned that anxiety is related to an increased risk of oral candidiasis.

Performing statistical analysis based on these findings shows that US subjects and subjects taking antibiotics have a significantly increased risk of having oral candidiasis in both treatment arms by Fisher's exact test (Table E.2 in Appendix E). However, the over-representation of oral candidiasis in these patient subgroups is more pronounced in subjects taking Symbicort than subjects taking placebo.

A low neutrophil concentration exhibits a significantly higher risk of developing oral candidiasis in subjects receiving Symbicort (9.5% at lower neutrophil concentration versus 3.9% at higher neutrophil concentration, $p<0.01$), but not in subjects receiving placebo (Table E.2). Similarly, regular smoking appears to increase the risk of having oral candidiasis in subjects receiving Symbicort but not in subjects receiving placebo, although the raw p-value is close to 0.05 in the Symbicort arm and would likely not be significant after multiplicity adjustment. Anxiety was significantly related to an increased risk of oral candidiasis in the placebo arm only.

In sum, based on the models and available data, a low neutrophil concentration and smoking status are variables that are potentially linked to an increased risk of experiencing oral candidiasis in patients treated with Symbicort. Country US and antibiotics use are general risk factors of oral candidiasis, as they confer a higher risk independent of treatment.

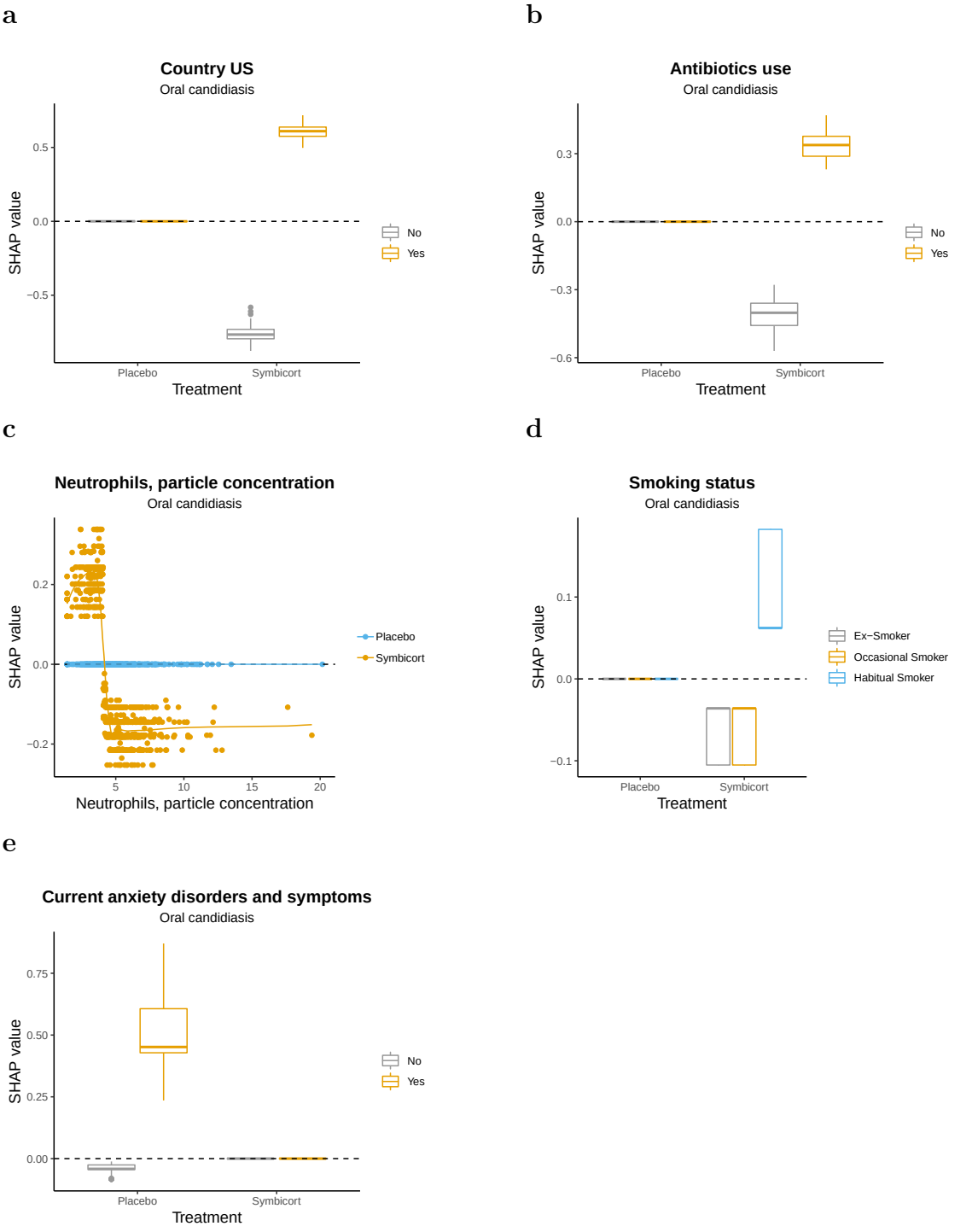


Figure 5.9: SHAP values of the Symbicort and placebo models plotted against variable values for the variables (a) country US, (b) antibiotics use, (c) neutrophil concentration, (d) smoking status and (e) current anxiety disorders and symptoms. The dashed horizontal line represents a SHAP value of zero, i.e. no impact on the model prediction. In (c) a LOESS curve has been fitted to the SHAP values from each model. According to plots (a)-(d) the Symbicort model associates these variables to the risk of oral candidiasis, while plot (e) shows that anxiety is linked to oral candidiasis by only the placebo model.

5.2.2 Dysphonia

The Symbicort, placebo and regression models used for this analysis are described in Appendix D.2.2. There were 21 out of 723 subjects who experienced dysphonia in the Symbicort arm, while only 5 subjects out of 677 had dysphonia in the placebo arm. The mean AUC of the Symbicort and placebo models is 0.72 and 0.81, suggesting that the placebo model performs better than the Symbicort model (Figure 5.10). However, the ROC curves of both models exhibit high variance, especially the placebo model. The R^2 of the corresponding regression model was 0.84.

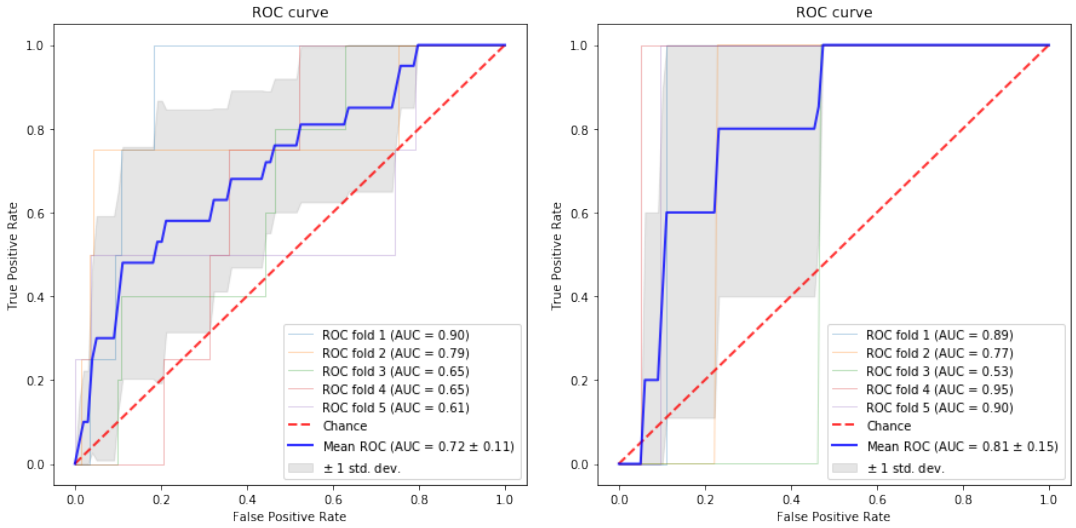


Figure 5.10: Receiver operating characteristics (ROC) curves following 5-fold cross-validation of the selected Symbicort model (left) and placebo model (right) for dysphonia. The blue line indicates the mean ROC curve, the dashed red line the expected ROC curve for random classification and the shaded gray area represents ± 1 standard deviation of the mean ROC curve. The mean area under the ROC curve (AUC) for the Symbicort and placebo models is 0.72 and 0.81, respectively.

The five variables with the highest mean absolute SHAP values in the regression model are displayed in Figure 5.11 and these are: pre-bronchodilator forced vital capacity (FVC; the volume of air that the patient can exhale after taking a deep breath), FEV1 reversibility (the percent increase in air volume that the patient can exhale in one second following treatment with a bronchodilator), platelet concentration (components of the blood essential for blood clotting), sitting diastolic blood pressure and months since first COPD symptoms.

High FEV1 reversibility and low platelet concentration are related to an increased risk of dysphonia according to the Symbicort model, but not according to the placebo model (Figure 5.12). A lower sitting diastolic blood pressure is linked to an increased risk in both models, suggesting that this is an independent risk factor for development of dysphonia. Interestingly, the associations learned by the Symbicort and placebo models exhibit opposite trends for the variables pre-bronchodilator FVC and months since first COPD symptoms.

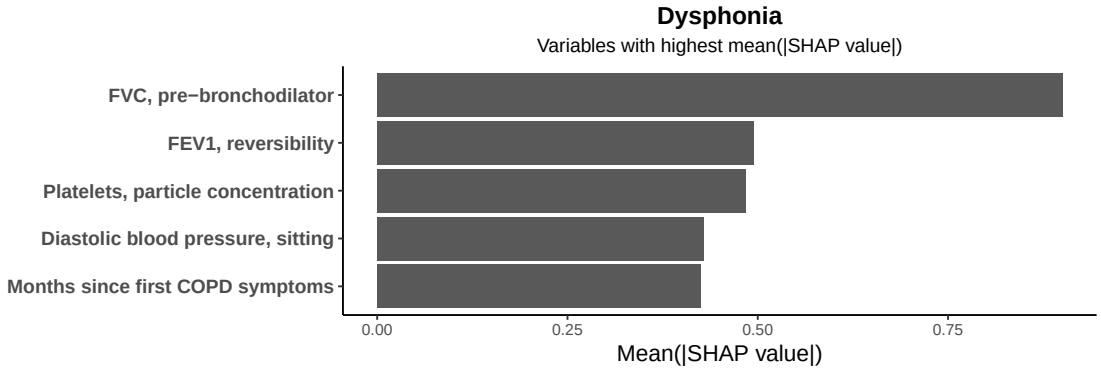


Figure 5.11: The five variables with the highest mean absolute SHAP values in the dysphonia regression model.

Statistical testing of these results (Table E.3 in Appendix E) shows that FEV1 reversibility is not a significant risk factor of dysphonia in either treatment arm, but recent onset of COPD symptoms is a significant risk factor in the Symbicort arm. The raw p-value is, however, close to 0.05. In addition, both high FVC and low platelet concentration are significantly associated with dysphonia in the Symbicort arm whereas low diastolic blood pressure is significantly associated with dysphonia independent of treatment.

From these results we can conclude that low platelet concentration, high FVC and possibly also recent onset of COPD symptoms are potential factors associated with an increased risk of dysphonia in patients treated with Symbicort and that low diastolic blood pressure is likely an independent risk factor for the development of dysphonia.

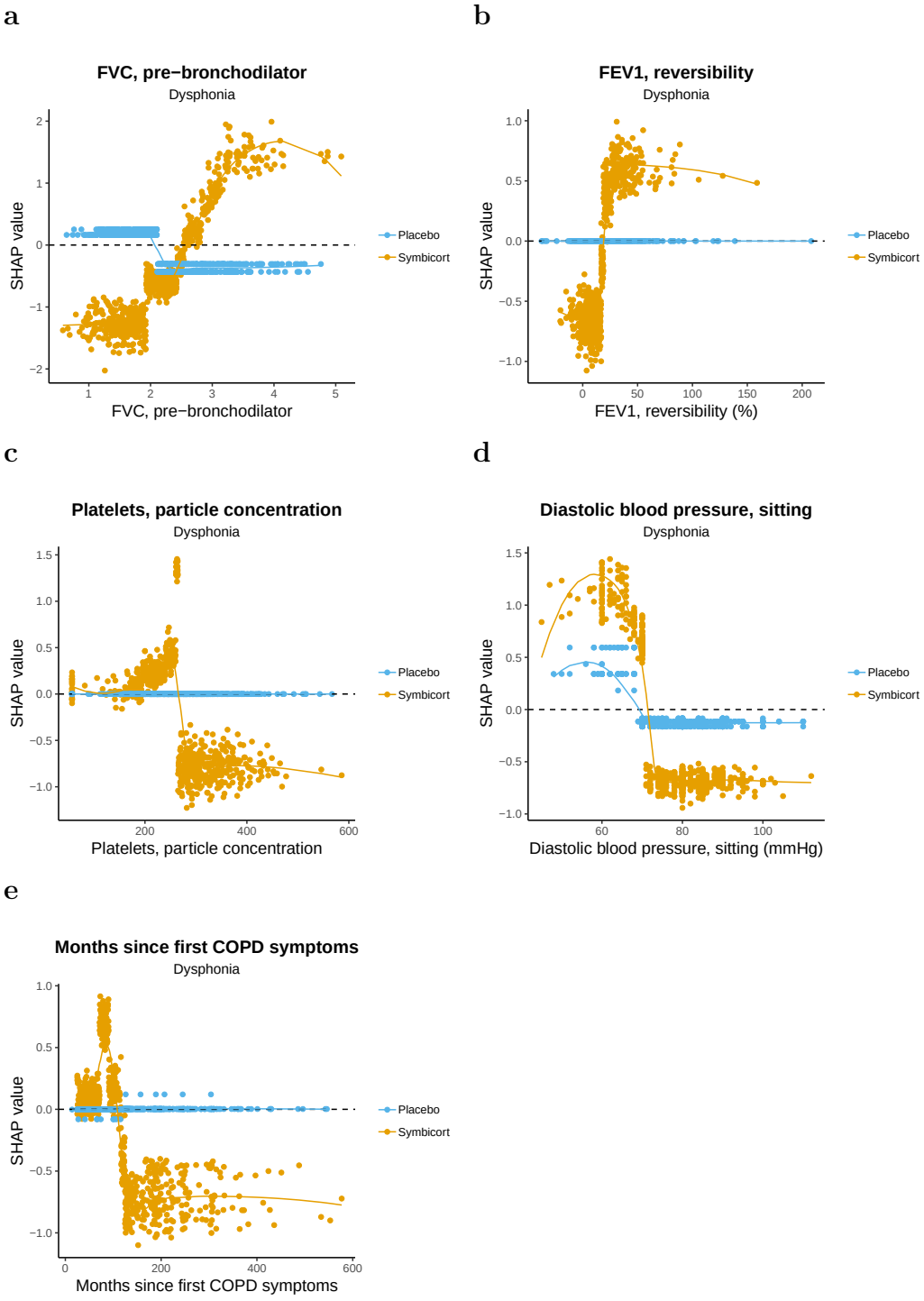


Figure 5.12: SHAP values of the Symbicort and placebo models plotted against variable values for the variables (a) pre-bronchodilator FVC, (b) FEV1 reversibility, (c) platelet concentration, (d) sitting diastolic blood pressure and (e) months since first COPD symptoms. The dashed horizontal line represents a SHAP value of zero, i.e. no impact on the model prediction. A LOESS curve has been fitted to the SHAP values from each model. According to plots (b) and (c) the Symbicort model associates these to the risk of dysphonia, plot (d) shows that both the Symbicort and placebo models identify a similar pattern, while plots (a) and (e) that the models identify opposite patterns with respect to dysphonia risk.

6
Discussion

In this thesis machine learning was used as an exploratory data mining tool to analyze adverse events from clinical trials. This enabled known drug-related adverse events to be identified in a data-driven manner as well as hundreds of different variables to be searched with the aim of identifying potential patient risk factors associated with an increased risk of developing a specific drug side-effect.

Using data from two phase III clinical trials of Symbicort for the treatment of COPD, the adapted inside-out data mining model in combination with statistical testing identified four true side-effects of Symbicort: oral candidiasis, dysphonia, dizziness and muscle spasms. The results for oral candidiasis and dysphonia were highly statistically significant and these side-effects have a higher expected frequency (1%-10%) than dizziness and muscle spasms (0.1%-1%) according to Appendix A. Anxiety, a known side-effect of Symbicort was identified as a potential drug-related AE by the model, but the result was not statistically significant. This is likely a consequence of the limited number of subjects included in the analysis.

Nasopharyngitis (the common cold) and sinusitis were also flagged as potential drug-side effects by the model and appeared more frequently in the Symbicort arm than in the placebo arm, but these differences were not statistically significant. Both nasopharyngitis and sinusitis are not considered side-effects of Symbicort and should be regarded as false positive findings by the method. The fact that these AEs were not statistically significant demonstrates the value of performing statistical testing in this setting to reduce the risk of false positive findings. However, false positive findings may still arise due to multiple testing. Given the limited number of observations, false negative findings may also result from such testing, as was seen with the AE anxiety. Thus, the initial results generated by the machine learning model provide a highly sensitive early indication of the possibly drug-related AEs.

Certain known common Symbicort side-effects, such as palpitations, were not identified by the method. However, these events were not frequently reported in the data (Fig. B.1). The low frequency of these AEs in the data may have several explanations. Firstly, these AEs may take a longer time to develop, and would therefore not be detectable within the limited duration of the clinical trials. Secondly, the homogeneous patient population included in the clinical trials may not be representative of the true patient population.

The adapted inside-out data mining method aims to identify possible drug side-

effects by finding AEs that are over-represented in subjects receiving the drug compared to subjects receiving placebo. However, the over-representation of AEs in subjects receiving the drug is not sufficient in order for an AE to be classified as a side-effect. For example, it needs to be determined whether it is reasonable to assume that the drug caused the AE. The *Bradford Hill Criteria* is one method for how to perform such an investigation [47]. In addition, certain serious AEs are known to often be drug side-effects, and may be considered side-effects even when just reported by a single subject. Lists of such AEs, known as *Designated Medical Events*, are published by regulatory authorities, e.g. the European Medicines Agency [48]. Therefore, the adapted inside-out data mining method should be considered as complementary to other established methods for AE analysis.

The AEs in this thesis were coded in binary format. This ignores the intensity, time of onset, duration as well as repeated occurrences of an event and thereby fails to capture differences in burden of the AE between treatment arms. Depending on the patient population and drug under study, differences in the burden of AEs may be an important way in which AE profiles differ between drug and placebo arms. Such differences would not be captured by the current method. Instead, the current method relies on a different spectrum of AEs arising in the drug arm compared to the placebo arm.

Machine learning methods generally perform better when more data is available. Pooling of data from different sources is a common way of increasing the amount of data. However, if the patient populations from which the data originates are different, this may result in findings that are artefacts of the pooling rather than the signal in the individual datasets. This is known as Simpson's paradox. In this project data was pooled from two different studies. While the inclusion and exclusion criteria of these studies are similar, the patient populations may have been different. For example, the set of countries from which patients were recruited differed.

Another important factor that may have influenced the AEs that were observed is the differing exposure of subjects to treatment, which had two principal causes. Firstly, the duration of the included studies differed, with SHINE following subjects during 6-months while SUN was a 12-month study. This may have caused AEs that take a longer time to develop to be observed only in SUN but not in SHINE. However, for events with a short time to onset the expected impact of the longer study duration in SUN would likely have been minimal. The binary coding of events also meant that repeated occurrences of events, which would have been more likely in the 12-month study, were ignored. Secondly, subjects who discontinued the study were included in the analysis. One reason why subjects choose to discontinue is if they experience an AE. Since the subjects in this study were COPD patients, one can hypothesize that subjects who received placebo treatment experienced complications of their COPD disease which could have been serious enough to cause discontinuation. This may explain why a slightly lower number of events were observed in the placebo arm.

The original paper describing the inside-out data mining method reported a classi-

fication error of 0.25 and 0.39 for data from two clinical trials to which the method was applied [5]. In contrast the model constructed here had a mean AUC of 0.56. The poor performance of the model may be a consequence of the large number of subjects reporting no AE during the studies. In each arm approximately 40% of subjects did not report any AE. Considering also events where the incidence was similar in both arms, the model would be expected to not perform better than randomly guessing for these subjects. How well the model must perform in order to yield trustworthy results is a valid question.

The mean absolute SHAP value was used as a crude measure of ranking the AEs according to being most likely to be possibly related to any treatment. Interestingly, in the t-SNE plot of the SHAP values, the mean absolute SHAP value appeared to be positively correlated with the distance to the cluster of subjects with no AE. However, one must exercise caution when interpreting t-SNE plots as only neighborhoods but not distances and densities are preserved [49]. The algorithm is also stochastic and will therefore yield somewhat different results each time it is run [45]. In spite of the stochastic nature of t-SNE, the positive correlation between the highest-ranking AEs and the distance to the main cluster of subjects having no AE still held upon re-running the t-SNE algorithm.

The t-SNE plot indicated that the model had learned that the oral candidiasis and dysphonia AEs affect different sub-populations of patients. This is consistent with the finding by the adapted VT method that very different risk factors were associated with developing these AEs when treated with Symbicort.

The adapted VT method identified both independent and possibly Symbicort-specific risk factors. Due to the limited availability of data and in particular the low event rate, these results should be interpreted with caution. It should also be noted that the identified variables only show an association with the AE and any causal link cannot be established based on this data, thus the findings should be considered as hypothesis-generating. In addition, the true factors driving the development of the AE may not have been measured.

Subjects in the US and subjects taking antibiotics were found to be at a higher risk of developing oral candidiasis independent of treatment. In SHINE it was reported that subjects in the US had a higher incidence of AEs overall compared to subjects in non-US regions (63.6% versus 46.8%) and that this held also after adjustment for differences in exposure [21]. The increased risk of oral candidiasis in US subjects could be an effect of differences in reporting practices or different practices when administering the medication. For example, it is known that rinsing the mouth after using an inhaler with an inhaled corticosteroid (such as budesonide, one of the components of Symbicort) can help reduce the risk of developing oral candidiasis [50]. The increased risk of oral candidiasis in patients taking antibiotics is considered to be an effect of changes to the microbiome in the mouth caused by the antibiotic [51].

A low neutrophil concentration was identified as a possible risk factor of oral can-

didiasis in patients taking Symbicort, but did not show any significant association to the condition in the placebo arm. Neutrophils are part of the innate immune system and form an integral part in the defense against opportunistic infections [52], such as *Candida albicans*, the yeast that overgrows in oral candidiasis. The inhaled corticosteroid component of Symbicort has an anti-inflammatory action and one can speculate that it may interact with neutrophils. This finding is however only an association and is based on relatively few events.

Associations between smoking and an increased risk of development of oral candidiasis have been found in other studies [53]. The lack of a significant association between smoking and oral candidiasis risk in the placebo arm in this thesis may be the result of the limited amount of data that the analysis is based on, as only 13 subjects in the placebo arm experienced oral candidiasis.

Dysphonia was associated with a low diastolic blood pressure irregardless of treatment and with a low platelet concentration, a high FVC and possibly also recent onset of COPD symptoms in patients taking Symbicort. Studies in patients undergoing hemodialysis have found that a low systolic blood pressure was linked to an increased incidence of dysphonia, but this was not the case for diastolic blood pressure [54]. As a low platelet concentration, a high FVC and recent onset of COPD symptoms only exhibit an association with an increased risk of dysphonia in subjects treated with Symbicort, these variables may be a proxy for other underlying factors which are driving the development of dysphonia. These findings are also based on a limited set of subjects.

A challenge when applying the adapted VT method to search for risk factors of side-effect development is the need for placebo data and a sufficient incidence of the AE in the placebo arm. The availability of placebo data may be limited due to ethical concerns, especially when depriving study subjects of active treatment can have serious consequences for the patient. Even when placebo data is available, the number of events in the placebo arm must be sufficient in order to build a trustworthy model. As the AEs that are selected for this analysis (oral candidiasis and dysphonia) are drug side-effects, they will be expected to have a lower incidence in the placebo arm.

Simulation studies could aid our understanding of the characteristics of the adapted inside-out data mining and adapted VT methods. Of particular interest will be how well the models need to perform in order to provide reliable guidance.

7
Conclusion

The exploratory machine learning workflow developed in this thesis offers an objective means of analyzing AEs from clinical trials that is complementary to current practice. In addition, the comprehensive analysis of subgroups could be a step towards personalized treatment. Future studies should aim to validate the workflow in order to establish the necessary performance characteristics of the models.

Bibliography

[1] Xia HA, Jiang Q. Statistical Evaluation of Drug Safety Data. Therapeutic Innovation Regulatory Science. 2014;48(1):109–120.

[2] ICH. E2A Clinical Safety Data Management: Definitions and Standards for Expedited Reporting; 1994. https://database.ich.org/sites/default/files/E2A_Guideline.pdf.

[3] Zink RC, Marchenko O, Sanchez-Kam M, Ma H, Jiang Q. Sources of Safety Data and Statistical Strategies for Design and Analysis: Clinical Trials. Therapeutic Innovation Regulatory Science. 2018 Mar;52(2):141–158.

[4] Phillips R, Hazell L, Sauzet O, Cornelius V. Analysis and reporting of adverse events in randomised controlled trials: a review. BMJ Open. 2019 Mar;9(2).

[5] Southworth H, MO O. Data mining and statistically guided clinical review of adverse event data in clinical trials. Journal of Biopharmaceutical Statistics. 2009;19(5):803–817. Available from: https://doi.org/10.1080/10543400903105232.

[6] Turner JR. New Drug Development. New York, NY: Springer New York; 2010. Available from: https://doi.org/10.1007/978-1-4419-6418-2_1.

[7] Gong Q, Tong B, Strasak A, Fang L. Analysis of safety data in clinical trials using a recurrent event approach. Pharmaceutical Statistics. 2014 Nov;13(2):136–144.

[8] ICH. MedDRA;. https://www.meddra.org/.

[9] ICH. What's New - MedDRA Version 23.0;. https://www.meddra.org/sites/default/files/guidance/file/whatsnew_23_0_english.pdf.

[10] Onakpoya IJ. Rare adverse events in clinical trials: understanding the rule of three. BMJ Evidence-Based Medicine. 2018;23(1):6–6. Available from: https://ebm.bmj.com/content/23/1/6.

[11] Ma H, Ke C, Jiang Q, Snapinn S. Statistical Considerations on the Evaluation of Imbalances of Adverse Events in Randomized Clinical Trials. Therapeutic Innovation Regulatory Science. 2015 Feb;49(6):957–965.

[12] Wang W, Whalen E, Munsaka M, Li J, Fries M, Kracht K, et al. On Quantitative Methods for Clinical Safety Monitoring in Drug Development. Statistics in Biopharmaceutical Research. 2018;10(2):85–97.

[13] Diao G, Liu GF, Zeng D, Wang W, Tan X, Heyse JF, et al. Efficient methods for signal detection from correlated adverse events in clinical trials. Biometrics. 2019;75(3):1000–1008.

[14] Agency EM. EudraVigilance;. https://www.ema.europa.eu/en/human-regulatory/research-development/pharmacovigilance/eudravigilance.

[15] Food, Administration D. Questions and Answers on FDA's Adverse Event Reporting System (FAERS);. https://www.fda.gov/drugs/surveillance/questions-and-answers-fdas-adverse-event-reporting-system-faers.

[16] Izem R, Sanchez-Kam M, Ma H, Zink R, Zhao Y. Sources of Safety Data and Statistical Strategies for Design and Analysis. Therapeutic Innovation Regulatory Science. 2018 Aug;52(2):159–169.

[17] Liu F, Jagannatha A, Yu H. Towards Drug Safety Surveillance and Pharmacovigilance: Current Progress in Detecting Medication and Adverse Drug Events from Electronic Health Records. Drug Safety. 2019;42(1):95–97.

[18] Porta M, Last JM. A Dictionary of Public Health. Oxford University Press; 2018.

[19] Rabe KF, Watz H. Chronic obstructive pulmonary disease. The Lancet. 2017;389(10082):1931–1940.

[20] Barnes PJ, Burney PGJ, Silverman EK, Celli BR, Vestbo J, Wedzicha JA, et al. Chronic obstructive pulmonary disease. Nature Reviews Disease Primers. 2015;1(1).

[21] Tashkin DP, Rennard SI, Martin P, Ramachandran S, Martin UJ, Silkoff PE, et al. Efficacy and Safety of Budesonide and Formoterol in One Pressurized Metered-Dose Inhaler in Patients With Moderate to Very Severe Chronic Obstructive Pulmonary Disease: Results of a 6-month Randomized Clinical Trial. Drugs. 2008;68(14):1975–2000. Available from: https://doi.org/10.2165/00003495-200868140-00004.

[22] Rennard SI, Tashkin DP, McElhattan J, Goldman M, Ramachandran S, Martin UJ, et al. Efficacy and tolerability of budesonide/formoterol in one hydrofluoroalkane pressurized metered-dose inhaler in patients with chronic obstructive pulmonary disease: results from a 1-year randomized controlled clinical trial. Drugs. 2009;69(5):549–565. Available from: https://doi.org/10.2165/00003495-200969050-00004.

[23] Samuel AL. Some Studies in Machine Learning Using the Game of Checkers. IBM J Res Dev. 1959;3:210–229.

[24] James G, Witten D, Hastie T, Tibshirani R. An introduction to statistical learning: with applications in R. Springer; 2013.

[25] Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning Data Mining, Inference, and Prediction, Second Edition. Springer New York; 2009.

[26] Lundberg SM, Erion G, Chen H, DeGrave A, Prutkin JM, Nair B, et al. From local explanations to global understanding with explainable AI for trees. Nature Machine Intelligence. 2020;2(1):2522–5839.

[27] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12(85):2825–2830. Available from: http://jmlr.org/papers/v12/pedregosa11a.html.

[28] Ribeiro MT, Singh S, Guestrin C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. CoRR. 2016;abs/1602.04938. Available from: http://arxiv.org/abs/1602.04938.

[29] Bonaccorso G. Machine Learning Algorithms. 1st ed.; 2017.

[30] Marsland S. Machine learning : an algorithmic perspective. Second edition. ed.; 2015.

[31] Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences. 1997;55(1):119–139.

[32] Friedman JH. Greedy Function Approximation: A Gradient Boosting Machine. Annals of Statistics. 2000;29:1189–1232.

[33] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16. New York, NY, USA: Association for Computing Machinery; 2016. p. 785–794. Available from: https://doi.org/10.1145/2939672.2939785.

[34] Rice JA. Mathematical Statistics and Data Analysis. 3rd ed. Belmont, CA: Brooks/Cole; 2007.

[35] Lipton ZC. The Mythos of Model Interpretability. Queue. 2018 Jun;16(3):30:31–30:57. Available from: http://doi.acm.org/10.1145/3236386.3241340.

[36] Doshi-Velez F, Kim B. Towards A Rigorous Science of Interpretable Machine Learning; 2017.

[37] Gilpin LH, Bau D, Yuan BZ, Bajwa A, Specter M, Kagal L. Explaining Explanations: An Overview of Interpretability of Machine Learning. 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA). 2018 Oct;Available from: http://dx.doi.org/10.1109/DSAA.2018.00018.

[38] Caruana R, Lou Y, Gehrke J, Koch P, Sturm M, Elhadad N. Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '15. ACM; 2015. p. 1721–1730. Available from: http://doi.acm.org/10.1145/2783258.2788613.

[39] Preece AD, Harborne D, Braines D, Tomsett R, Chakraborty S. Stakeholders in Explainable AI. CoRR. 2018;abs/1810.00184. Available from: http://arxiv.org/abs/1810.00184.

[40] Shapley LS. A value for n-person games. Contributions to the Theory of Games. 1953;2(28):307–317.

[41] Lundberg SM, Lee SI. A Unified Approach to Interpreting Model Predictions. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, et al., editors. Advances in Neural Information Processing Systems 30. Curran Associates, Inc.; 2017. p. 4765–4774. Available from: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[42] Janzing D, Minorics L, Blöbaum P. Feature relevance quantification in explainable AI: A causal problem; 2019.

[43] Štrumbelj E, Kononenko I. Explaining prediction models and individual predictions with feature contributions. Knowledge and Information Systems. 2013;41(3):647–665.

[44] Agresti A. An Introduction to Categorical Data Analysis. 2nd ed. Hoboken, New Jersey: John Wiley Sons, Inc; 2007.

[45] van der Maaten LJP, Hinton GE. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research. 2008;9(nov):2579–2605.
[46] Foster JC, Taylor JMg, Ruberg SJ. Subgroup identification from randomized clinical trial data. Statistics in Medicine. 2011 Apr;30(24):2867–2880.
[47] Hill AB. THE ENVIRONMENT AND DISEASE: ASSOCIATION OR CAUSATION? Proceedings of the Royal Society of Medicine. 1965;58(5):295–300.
[48] European Medicines Agency. Signal management; 2020. Available from: https://www.ema.europa.eu/en/human-regulatory/post-authorisation/pharmacovigilance/signal-management.
[49] Schubert E, Gertz M. Intrinsic t-Stochastic Neighbor Embedding for Visualization and Outlier Detection. In: Beecks C, Borutta F, Kröger P, Seidl T, editors. Similarity Search and Applications. Cham: Springer International Publishing; 2017. p. 188–203.
[50] Capes Jost B, Abdel-Hamid KM, Friedman E, Jani AL. The Washington Manual Subspecialty Consult Series: Allergy, Asthma, and Immunology Subspecialty Consult. Lippincott Williams Wilkins; 2003.
[51] Agrawal A, Singh A, Verma R, Murari A. Oral candidiasis: An overview. Journal of Oral and Maxillofacial Pathology. 2014;18(4):81.
[52] Abbas AK. Basic immunology : functions and disorders of the immune system. Sixth edition ed.; 2020.
[53] Soysa N, Ellepola A. The impact of cigarette/tobacco smoking on oral candidosis: an overview. Oral Diseases. 2005;11(5):268–273. Available from: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1601-0825.2005.01115.x.
[54] Zumrutdal A. An overlooked complication of hemodialysis: Hoarseness. Hemodialysis International. 2013;.

A

Symbicort side-effects

Table A.1 lists descriptions of the side-effects of Symbicort and is adapted from the *Investigator's Brochure* (version 10, 28 June 2019) of this drug. This document represents the current state of knowledge of the safety and efficacy of Symbicort. It is revised regularly and is based on data collected during drug development as well as post-marketing clinical trials and spontaneous reporting.

Table A.1: Frequencies and descriptions of the known side-effects of Symbicort.

Frequency	Description
Common (1% to 10%)	Palpitations Candida infections in oropharynx Headache, tremor Mild irritation in the throat, coughing, hoarseness
Uncommon (0.1% to 1%)	Tachycardia Nausea Muscle cramps Dizziness Agitation, restlessness, nervousness, sleep disturbances
Rare (0.01% to 0.1%)	Cardiac arrhythmias, e.g., atrial fibrillation, supraventricular tachycardia, extrasystoles Immediate and delayed hypersensitivity reactions, e.g., dermatitis, exanthema, urticaria, pruritus, angioedema and anaphylactic reaction Bronchospasm Skin bruising
Very rare (<0.01%)	Angina pectoris Signs or symptoms of systemic glucocorticosteroid effects, e.g., hypofunction of the adrenal gland Hyperglycemia Depression, behavioral disturbances

B

Exploratory data analysis supplementary material

Figure B.1 displays the ten most common AEs in the data. The most common AE was COPD, followed by nasopharyngitis and oral candidiasis.

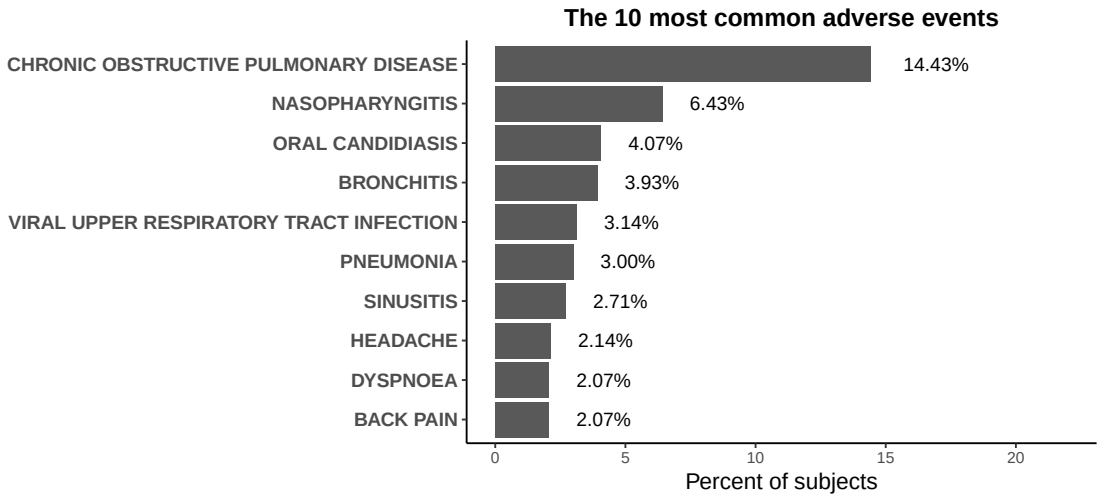


Figure B.1: The ten most common adverse events across the Symbicort and placebo arms.

The frequency distribution of the number of different AEs per subject, after removal of AEs that were experienced by only one subject, by treatment is shown in Figure B.2.

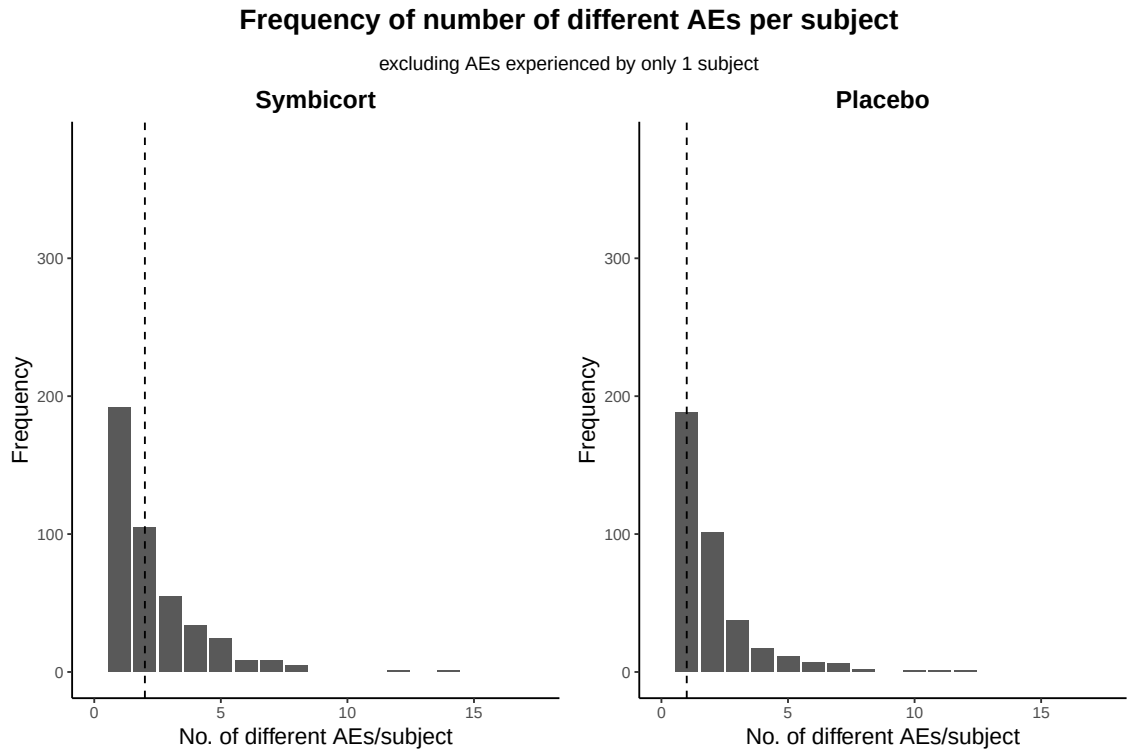


Figure B.2: Frequency distribution of the number of different adverse events per subject by treatment, after exclusion of adverse events that were experienced by only one subject. The dashed line represents the median number of adverse events per subject, which was two in the Symbicort arm and one in the placebo arm.

C

XGBoost hyperparameters

The hyperparameters investigated during the tuning of the XGBoost models are presented in Table C.1 and C.2.

Table C.1: Hyperparameter values investigated in the XGBoost classification model for predicting treatment arm from adverse events as well as in the XGBoost regression model for predicting the difference in scores between the drug and placebo models.

Hyperparameter	Description	Values investigated
n_estimators	The number of trees to construct	100, 200, 300, 400
max_depth	Maximum depth of the tree	2, 3, 4, 5
learning_rate	Learning rate	0.01, 0.05, 0.1, 0.2, 0.3
lambda	L2 regularization	0.5, 1, 2
min_split_loss	Minimum loss reduction to split at node	0, 1, 5, 10
min_child_weight	Minimum sum of instance weight needed in a child	1, 2, 3

Table C.2: Hyperparameter values investigated in XGBoost classification models for predicting the probability of a subject experiencing a particular adverse event.

Hyperparameter	Description	Values investigated
n_estimators	The number of trees to construct	100, 200, 300, 400
max_depth	Maximum depth of the tree	2, 3
learning_rate	Learning rate	0.01, 0.05, 0.1, 0.2, 0.3
lambda	L2 regularization	0.5, 1, 2
min_split_loss	Minimum loss reduction to split at node	0, 1, 5
min_child_weight	Minimum sum of instance weight needed in a child	1, 2
max_delta_step	Maximum update of any leaf	1, 3, 5, 7, 9

D

The selected models

Descriptions of the selected models are presented here.

D.1 Model for identification of possibly drug-related adverse events

The hyperparameters of the XGBoost model that achieved the highest mean cross-validation AUC is shown in Table D.1

Table D.1: Hyperparameter values of the selected XGBoost model for predicting treatment arm from adverse events.

Hyperparameter values
n_estimators = 200
max_depth = 4
learning_rate = 0.05
lambda = 0.5
min_split_loss = 0
min_child_weight = 1

D.2 Models for characterization of drug side-effects

The Symbicort, placebo and regression models that are used in characterizing the AEs oral candidiasis and dysphonia are presented here.

D.2.1 Oral candidiasis

The hyperparameters of the selected Symbicort and placebo models for oral candidiasis are shown in Table D.2. while Table D.3 shows the hyperparameters of the corresponding regression model.

Table D.2: Hyperparameter values of the selected XGBoost models for classifying subjects receiving Symbicort (left) and placebo (right) according to their probability of experiencing the adverse event oral candidiasis.

Symbicort model hyperparameter values	Placebo model hyperparameter values
n_estimators = 100	n_estimators = 200
max_depth = 2	max_depth = 2
learning_rate = 0.1	learning_rate = 0.01
lambda = 0.5	lambda = 0.5
min_split_loss = 5	min_split_loss = 0
min_child_weight = 2	min_child_weight = 2
max_delta_step = 3	max_delta_step = 1

Table D.3: Hyperparameter values of the selected regression model for oral candidiasis.

Hyperparameter values
n_estimators = 400
max_depth = 3
learning_rate = 0.1
lambda = 0.5
min_split_loss = 0
min_child_weight = 3

D.2.2 Dysphonia

The selected Symbicort and placebo models are presented in Table D.4 and the corresponding regression model in Table D.5.

Table D.4: Hyperparameter values of the selected XGBoost models for classifying subjects receiving Symbicort (left) and placebo (right) according to their probability of experiencing the adverse event dysphonia.

Symbicort model hyperparameter values	Placebo model hyperparameter values
n_estimators = 200	n_estimators = 100
max_depth = 3	max_depth = 2
learning_rate = 0.3	learning_rate = 0.3
lambda = 0.5	lambda = 0.5
min_split_loss = 0	min_split_loss = 1
min_child_weight = 2	min_child_weight = 1
max_delta_step = 3	max_delta_step = 3

Table D.5: Hyperparameter values of the selected regression model for dysphonia.

Hyperparameter values
n_estimators = 400
max_depth = 2
learning_rate = 0.2
lambda = 0.5
min_split_loss = 0
min_child_weight = 3

E

Statistical analyses

The results from statistical analyses using Fisher's exact test are presented here. Note that the p-values that are reported have not been adjusted for multiplicity. The percent and frequency of the ten highest-ranking AEs identified in Section 5.1, including results from Fisher's exact test, are shown by treatment in Table E.1. Similarly, variables identified as potential risk factors for developing oral candidiasis (Section 5.2.1) and dysphonia (Section 5.2.2) are presented in Table E.2 and E.3, respectively.

Table E.1: Percent and frequency of the ten highest-ranking adverse events by treatment.

AE	Symbicort (n=723)		Placebo (n=677)		p-value (Fisher's exact test)
	% with AE	No. with AE	% with AE	No. with AE	
Oral candidiasis	6.1	44	1.9	13	<0.01
Chronic obstructive pulmonary disease	13.4	97	15.5	105	0.29
Dysphonia	2.9	21	0.7	5	<0.01
Nasopharyngitis	7.6	55	5.2	35	0.06
Dyspnoea	1.4	10	2.8	19	0.09
Pneumonia	2.4	17	3.7	25	0.16
Sinusitis	3.5	25	1.9	13	0.10
Dizziness	1.4	10	0.3	2	0.04
Muscle spasms	2.4	17	0.7	5	0.02
Anxiety	1.5	11	0.6	4	0.12

Table E.2: Percent and frequency of subjects with oral candidiasis by treatment arm for the five most important variables identified. P-values are computed by Fisher's exact test.

Variable (Reference range)		Symbicort % with AE	Symbicort No. with AE /total	Symbicort p-value	Placebo % with AE	Placebo No. with AE /total	Placebo p-value
Country US	No	1.2	5/416	< 0.01	0.8	3/397	0.01
	Yes	12.7	39/307		3.6	10/280	
Antibiotics use	No	2.0	8/399	<0.01	0.8	3/379	0.02
	Yes	11.1	36/324		3.4	10/298	
Neutrophils, particle concentration (1.8-8 GI/L)	< 4.18	9.5	27/284	<0.01	1.5	4/266	0.58
	≥ 4.18	3.9	17/439		2.2	9/411	
Smoking status	Ex-Smoker	4.2	18/428	0.03	1.0	4/385	0.09
	Occasional Smoker	6.7	2/30		3.7	1/27	
	Habitual Smoker	9.1	24/265		3.0	8/265	
Current anxiety disorders and symptoms	No	5.6	37/656	0.17	1.0	6/626	< 0.01
	Yes	10.4	7/67		13.7	7/51	

Table E.3: Percent and frequency of subjects with dysphonia by treatment arm for the five most important variables identified. P-values are computed by Fisher's exact test.

Variable (Reference range)		Symbicort			Placebo		
		% with AE	No. with AE /total	p-value	% with AE	No. with AE /total	p-value
FVC, pre-bronchodilator	< 2.60	1.7	9/523	<0.01			
	≥ 2.60	6.0	12/200				
	< 2.07				1.4	4/280	0.17
	≥ 2.07				0.3	1/397	
FEV1, reversibility	< 19.09	2.0	9/452	0.07	0.9	4/423	0.66
	≥ 19.09	4.4	12/271		0.4	1/254	
Platelets, particle concentration (130-400 GI/L)	< 265.58	5.0	20/398	<0.01	1.1	4/375	0.39
	≥ 265.58	0.3	1/325		0.3	1/302	
Diastolic blood pressure, sitting	< 71.420	5.9	12/203	<0.01			
	≥ 71.420	1.7	9/520				
	< 69.30				3.4	3/88	0.02
	≥ 69.30				0.3	2/589	
Months since first COPD symptoms	< 108.39	4.3	16/376	0.03			
	≥ 108.39	1.4	5/347				
	< 122.24				0.8	3/390	1.00
	≥ 122.24				0.7	2/287	