Exploring the use of machine learning in health insurance risk adjustment

Final Report, 29 May 2020

i2i | Gupta Strategists |

Ministerie van Volksgezondheid, Welzijn en Sport

Table of Contents

About this report

This report describes the results of a broad exploration of the possibilities that machine learning algorithms can offer in the area of health insurance risk adjustment schemes. The exploration was performed by Gupta Strategists and i2i (intelligence to integrity) and has been commissioned by the Dutch Ministry of Health, Welfare and Sport.

The scope of this research is the Dutch somatic risk adjustment model of 2020. This model aims to predict somatic healthcare costs at the level of individual policyholders as accurately as possible, based on a large number of individual characteristics. In this project we explored the impact of replacing the normally used Ordinary Least Squares (OLS) regression model with a machine learning algorithm.

Source code

The source code of the analyses performed in the scope of this study, including explanatory notes, can be downloaded here:

https://github.com/intelligence2integrity/ml-risicoverevening

Project Team Members

The team for this project was:

- **Gupta Strategists**: Roxanne van Donselaar-Busschers, Daan Livestro, Sjors Oudshoorn
- **i2i**: Birgitta de Gruijter, Jules van Ligtenberg, Diederik Perdok, Michel Taal

The project team was assisted by dr. Tom Heskes, professor in Data Science at Radboud Universiteit, Nijmegen, The Netherlands.

For more information about this study you can contact:

- Daan Livestro (Daan.Livestro@gupta-strategists.nl), or
- Birgitta de Gruijter (Birgitta.de.Gruijter@i2i.eu)

Summary

The Dutch risk adjustment system is working well in terms of its main objectives: creating a level playing field for health insurers and removing risk selection incentives wherever possible. This is why the system is leading, globally, and innovations within our national scheme are closely followed abroad. Nevertheless, the current model appears to have reached its peak of possibilities, as becomes apparent from the fact that parties have agreed that the system is ready to make a switch to a maintenance system in the short term. That is why now is a good time to no longer look at just incremental improvements to the current method, but to evaluate the method per se. Against this background, this study explores the potential added benefit of machine learning techniques for health insurance risk adjustment.

Machine learning is used all around us, e.g. by supermarket chains in the segmentation of customers, by Facebook to be able to place targeted online adverts, and by utility companies to determine the need for energy. Vis-a-vis risk adjustment, some careful experience has been gained in (international) studies and earlier WOR[1] studies. Several studies suggest that machine learning models may be better able to predict the costs of healthcare than the current risk adjustment model.

This study is a broad exploration of the possibilities that machine learning algorithms can offer for health insurance risk adjustment. In doing so, the study focuses on improving the adjusting effect by replacing the current OLS model with a machine learning algorithm. Suggestions for improvements to the current system are a by-catch.

Design of the Study

By means of a literature review and expert opinion, possible algorithms have been identified and assessed based on five criteria: practical applicability, evidential value, expected adjusting effect, expected robustness and interpretability of results. This has led us to choose these five algorithms:

- **M1 – Decision Tree**. The simplest class of all tree-based models is applied to the OT dataset. Like OLS, it produces easily interpretable results because the algorithm achieves modeled healthcare costs per policyholder through a clear decision tree.
- **M2 – Piecewise Regression.** We apply this type of regression to the OT dataset enriched with age in number of years. It can be regarded as a logical extension of OLS. In this model, age is used to segment the data for separate linear regressions.
- **M3 – Random Forest.** We apply this more complex tree-based algorithm to the enriched OT dataset. The algorithm can handle non-linear interactions in the data very well, which gives it great predictive power, but produces difficult-to-interpret results because it is an average of several individual decision trees.
- **M4 – Gradient Boosting Machine.** We apply this algorithm to the source dataset, i.e. the OT data set supplemented with age in years and underlying data for morbidity criteria. Gradient Boosting Machines are the most powerful of all tree-based algorithms and tend to score very well in international machine learning competitions. The adjusting effect is expected to be favorable, although, as a result, this algorithm too compromises on interpretability.

[1] Werkgroep Ontwikkeling Risicoverevening, a Dutch body of experts and consultancies advising the government on annual improvements to the health insurance risk adjustment scheme

- **M5 – Artificial Neural Network.** We apply the Artificial Neural Network on the source dataset. This algorithm has been successfully applied to risk adjustment in several studies but is one of the most difficult to interpret models due to the large extent to which it allows interactions.

In this study the models have been optimized for R^2. This is the most commonly used measure within risk adjustment, and it ensures that the prediction of a model is an approximation of the average actual costs.

The selected algorithms vary to the extent that they deviate from the current methodology and thus allow for a broad exploration of the possibilities of machine learning for risk adjustment.

To avoid overfitting in the development and validation of the models, 30% of unique policyholders were randomly assigned to a test set for this study. The remaining 70% of policyholders form the training set. All algorithms were exclusively trained based on this training set. In developing each model within the training set we used 10-fold cross validation. After the final definition of the model parameters, the metrics of each model are calculated for the test set.

Finally, interviews with stakeholders provided important insights into the consequences of applying machine learning to the current working method within the risk adjustment cycle.

Results

In this study, a first improvement in adjusting effect (R^2) from 35.1% to 38.5% was achieved in a limited construction period of 3 months using machine learning techniques. This demonstrates that machine learning techniques potentially have added value for risk adjustment in the Netherlands.

In addition to an improvement in R^2, based on which optimization took place, detailed metrics have been calculated, from which a number of additional conclusions can be distilled:

- **Model M1 - Decision Tree** scores slightly lower than the current model on nearly all metrics. The metrics are negatively impacted on an individual, subgroup and insurer level.
- **Model M2 – Piecewise Regression** achieves an improvement on R^2 relative to the current model. Remarkably, a comparison with OLS on all individual segments except for the segment of 0 to 8-year-olds shows an improvement of R^2. On the other hand, however, there are substantially more policyholders with a negative standard amount. This is mainly due to the fact that the model uses an age segment of 0 to 8-year-olds, and therefore does not include the deviating cost pattern of 0-year-olds as a separate segment. The modeled healthcare costs for part of the 8-year-olds are therefore negative. At a subgroup and insurer level, a small deterioration can be observed on the majority of metrics. Incidentally, the results on the metrics are better if this model is applied in a slightly modified variant, by performing a regular OLS on the age segments found.
- **Model M3 – Random Forest** scores better than the current model on all individual metrics. With the same data, the model shows an improved R^2 from 35.1% to 36.3% compared with OLS. At the subgroup and insurer levels, different metrics show both an improvement and a deterioration. The addition of continuous age has a limited positive impact on the metrics for this model.
- **Model M4 – Gradient Boosting Machine** achieved good results on nearly all metrics. With the same data, the model shows an improved R^2 from 35.1% to 36.3% compared with OLS. The added value compared with OLS only really becomes apparent when extra source data are added: the R^2 then improves to 38.5%, while an OLS model achieves an R^2 of 36.1%

with the same data. What stands out in particular, are the improvements on individual measures; these are the best of all models. Only the bandwidth of the results of medium-sized and major insurers deteriorates slightly.

- **Model M5 – Artificial Neural Network** achieves good results on all metrics. With the same data, the model shows an improved R^2 from 35.1% to 36.3% compared with OLS. As with the Gradient Boosting Machine, the added value compared to OLS only really becomes apparent when we add extra source data: the R^2 then improves to 38.2%, while an OLS model achieves an R^2 of 36.2% on the same data. The complete trained model, including additional data, shows improvements at an individual, subgroup and insurer level compared to an OLS model using the same additional data set. The results at the insurer level are particularly favorable and the best of all models. All bandwidths become narrower. This also becomes apparent from the highest mean absolute result shift (MARS).

The main risk features, i.e. the features that contribute most to the adjusting effect, remain more or less the same across the different models. Multiyear high costs (MHC) and multiyear high costs of nursing and home care (MHCN) are among the most important features in each model – as with the OLS, they remain equally important to arrive at a good prediction of healthcare costs. The relative relevance of MHCN does decrease in models M1 through M5 relative to baseline model M0, while the relevance of MHC goes up. This effect is the greatest in the *tree-based* models M1, M3 and M4. A further remarkable effect is that features such as Medical Aid Cost Group (MACG), Physiotherapy Diagnosis Group (PDG), Nature of Income (NOI), and Socio-Economic Status (SES), in models M4 and M5 are not among the top-10 most important features.

Apart from conclusions regarding the potential value of machine learning models for risk adjustment, this study also led to some in-depth insights that may be important for the regular risk adjustment process:

- The Decision Tree analysis characterizes five very large groups of policyholders based on only a limited number of features. In all, these five groups comprise 44% of policyholders. The features used for these five groups show that relatively healthy policyholders in particular can be captured well with fewer features than used by the current OLS. It is likely that OLS overfits on some of these groups - the use of more adjustment criteria for these policyholders may deteriorate the prediction. The OLS model may benefit from this insight by taking into account the interaction by setting parameters that the decision tree does not use (PCG, e.g., for one of the groups) to 0 for the group in question in the OLS.
- A further interesting observation follows from the Piecewise Regression analysis. The relative added value of a high-risk category, for example MHC8, compared to the reference class, MHC0, decreases among older policyholders. Receiving the MHC8 standard amount in accordance with the regular OLS would have led to an overestimation of the costs of healthcare. Whether this is really the case cannot be concluded with certainty on the basis of this study. We do see, however, that for the majority of policyholder groups the results are better predicted based on MHC and age segment model M2 than the baseline model. Further research could home in on this observation to determine whether segmentation of features like MHC according to age could improve the adjustment result.

Consequences of applying machine learning in risk adjustment

In addition to the positive results of the use of machine learning models (outcome) in terms of R^2, the consequences of applying machine learning (feasibility) are also an important criterion in the appeal of machine learning for risk adjustment.

If one of the machine learning techniques was to replace the current OLS model, this will have consequences for all aspects of the risk adjustment cycle. In this study we looked at:

- **Organizing parameters**: Legislation, stakeholders and infrastructure
- **Improvement cycle**: Performance and interpretation of improvement studies
- **Maintenance cycle**: Consequences for the different process steps – Data Phase, OT and Standard Amounts Phase
- **Implementation**: Consequences for the settlement process and for ex-post mechanisms
- **Interpretation and incentive effect**: Validity, stability and homogeneity, transparency and simplicity, and incentive effect

As for the decision tree (M1) the main concerns are the stability of the outcomes. In many ways, piecewise linear regression (M2) is similar to the current OLS model, meaning that the implementation of M2 should be relatively straightforward.

Models M3, M4 and M5, specifically, have considerable implementation thresholds. Application to replace the current OLS may require changes to the regulatory framework, because these models do not lead to "weights per criteria" as required by the Dutch Healthcare Insurance Act. However, conducting studies into the improvement and maintenance cycles also becomes more complex under these models. In addition to creating and testing (risk) *features*, some extra time is needed to optimize hyperparameters, and when detailed underlying data is used (as in M4 and M5) it is conceivable that a very detailed estimate of policy holder counts by each individual feature is no longer possible.

Conclusion: how to continue with machine learning in risk adjustment?

A number of models tested in this study are easily applicable and could offer added value in the short term:

- Model M2 (Piecewise Linear Regression) is a logical extension of the current OLS. With limited further development, it could potentially be suitable to replace the OLS soon, in particular to better deal with interactions between age and regular adjustment features such as PCG or DCG (Diagnosis Cost Group). For the same purpose, the model is also very useful in the improvement cycle.
- Model M1 (Decision Tree) is easy to implement and has led to the identification of large, cost-homogeneous groups in this study. Periodically conducting a study like this can help provide insight into interactions between features in a simple manner, and determine whether, for example, interaction terms can provide added value.

Models M4 (Gradient Boosting Machine) and M5 (Artificial Neural Network) both are very promising models. The results are convincing when these models are given more data to use, but even when using exactly the same data, these models show (slightly) better results than the OLS. It is important to note that the machine learning models on the limited data sets have not been extensively optimized, and that further improvement may still be possible. Even if these models will not *replace* the OLS as yet, they can quickly prove to be of added value as an additional research technique, as a benchmark for the 'best achievable' adjustment result (for example, as part of the OT), and as part of partial studies into the improvement cycle.

Especially when more information will be added, these models clearly show added value. The ability of these models to extract predictive power from interactions seems even more apparent. Note also that this was only an exploratory study, and it may be that even better results can be achieved

in the future with further optimization. On the other hand, these are also the models whose implementation will have greater consequences for the current way of working, which means that the barrier for implementation is higher.

Recommendations for further research

In addition to the specific recommendations for implementation of the different models, this study also leads to a number of more general recommendations for further research into machine learning used in risk adjustment:

- Our main recommendation is to run these machine learning models *in parallel* with the regular OLS for several years in order to gradually gain experience with the robustness and implementation aspects of these models, and also to address the main uncertainties. In addition, the best performing models could be tested retrospectively over several years.
- It would also be interesting to study how machine learning models perform when less desirable aspects of the current model are omitted. In this study, models were trained on *at least the same features* as available in the current model, but not on a *more limited* set of features. How valuable are these models if, for example, MHC or MHCN is not included as a feature?
- In this study only the somatic model has been explored. In future research it would also be useful to explore the use of machine learning for other risk adjustment models, in particular the mental healthcare model. It would also be useful to explore the extent to which machine learning can help achieve a reduction in number of required models in general.
- The main goal of this study was to find models that lead to the highest possible R^2. In further studies, it would be useful to look at other target functions, such as the result on subgroups.

Recommendations for the regular risk adjustment cycle

Finally, this study has led to insights that may be relevant for further research within the regular risk adjustment cycle.

- When evaluating risk adjustment as is, it is important to also test other steps and not just the model development itself. In this study, for example, the question was raised to what extent the scaling of the OT data set to the policyholder estimate has an impact on the quality of the adjustment. How bad is it if we are not able do this very precisely, for example because the features are too detailed? In addition, it can be useful to test separately the combining of different models (somatic, MHC and deductible) for a total result per policyholder, for example. The main recommendation in this context is to test the *entirety* of the steps from the creation of the dataset to the performance of the final determination in a protocolized and integrated manner.
- The Decision Tree analysis identified a few major groups of healthy policyholders for whom feeding more information into the model seems unfavorable. It is worth studying whether adjusting the OLS so that it accounts for the groups identified in the decision tree analysis can lead to a better adjustment result in the current adjustment model.
- Piecewise Regression shows that MHC in particular shows possible interaction with age. Whether this is in fact the case cannot be determined with certainty on the basis of this study. What we do see, though, is that for the majority of policyholder groups model M2 better predicts the results than the baseline model, based on MHC and age segment. Further research could look into this observation more closely and determine whether segmentation of features such as MHC by age could improve the adjustment result.

- Breakdown of PCG based on the number of defined daily doses (ddds) seems to add value to the predictive power, also in the case of an OLS model. It is useful, for example in the case of major maintenance, to explore whether a more fine-grained model, or at least even more specific threshold values per PCG, can lead to a better adjustment model. An improvement in the adjustment effect must, however, be expressly weighed up against potentially undesirable incentives in terms of effectiveness.
- Age in years, number of patient days, number of healthcare products, number of diagnoses and number of procedures are all possible *features* that may increase the predictive power of the OLS model and are therefore worth looking into. It is especially important to include the incentive effect in the evaluation.
- Splitting DCG into dx groups seems to improve the predictive power. This possibility has already been explored in the recent overhaul of the DCG (WOR 988) and in the pre-OT 2020 (WOR 990), and a more detailed diagnosis categorization has been included. This is why we will not make any further recommendations for this in the present report.

1 Introduction

1.1 Immediate cause

The Dutch risk adjustment system is working well in terms of its main objectives: creating a level playing field for health insurers and removing risk selection incentives wherever possible. This is why the system is leading, globally, with hardly any other countries that have a similar system in place. Nevertheless, the current model appears to have reached its peak of possibilities, as becomes apparent from the fact that parties have agreed that the system is ready to make a switch to a maintenance system within the short term. This is why now is a good time to no longer look at just incremental improvements to the current method, but to evaluate the method *per se*. Against this background, this study explores the potential added benefit of machine learning techniques for risk adjustment.

Machine learning is used all around us, e.g. by supermarket chains in the segmentation of customers, by Facebook to be able to place targeted online adverts, and by utility companies to determine the need for energy. Vis-a-vis risk adjustment, some careful experience has been gained in (international) studies and earlier WOR studies. The study by Ismail (2018) [1] which applied Random Forest and a Gradient Boosting Machine to the OT data, suggests that machine learning models may be better able to predict costs of healthcare than the current risk adjustment model[2]. Internationally, research performed by Rose (2016), to name one example, has demonstrated that machine learning techniques can help improve the predictive power *and* simplify the formula[2]. Also, in some WOR studies, experience has been gained with the application of machine learning in the construction of risk categories [3], [4].

In this study, we explore the possibilities that machine learning algorithms offer for risk adjustment. In doing so, the study focuses on improving the adjusting effect by replacing the current OLS model with a machine learning algorithm. Suggestions for improvements to the current system are a by-catch. The potential of machine learning algorithms to improve the adjusting effect is threefold: 1) discovering relationships in the data that remained hidden so far; 2) broadening restrictive model assumptions; 3) the ability to add more and more detailed data to the model. The latter also offers as a potential advantage that the risk adjustment model may become less dependent on the existing data structure with carefully constructed risk categories.

Given the limitations of the current OLS regression model and the potential gains that can be achieved with machine learning techniques, we are exploring the possibilities that machine learning may or may not offer for Dutch risk adjustment. In doing so, we have charted the changes this would entail for the various stages in the risk adjustment cycle.

1.2 Objective

This study serves to provide a broad exploration of the value of machine learning applications for risk adjustment, including a description of applicable methods, results of application, consequences for the performance of risk adjustment and possible suggestions for improvement within the current system. As such, this report answers the central question below and the associated subquestions.

[2] https://equalis.nl/verslaat-machine-learning-het-huidige-risicovereveningsmodel; study not publicly available

Central question: *'What is the added value of machine learning in the context of risk adjustment?'*

Subquestions:

1. Which techniques are useful to further study in terms of their applicability for risk adjustment?
2. Which results do these techniques give in terms of their adjusting effect?
3. What other results do these techniques provide that can be used in the current adjustment model?
4. What other consequences (advantages and disadvantages) of applying these techniques for risk adjustment are there?
5. Advice to be formulated based on the elaboration of the above questions concerning the application of machine learning in risk adjustment.

1.3 Design of the Study

For this study we have used the Overall Test (OT) data for 2020, the standard dataset developed each year to build the Dutch risk adjustment model. In addition, the National Health Care Institute (Zorginstituut Nederland) has supplied additional source files with regard to the morbidity criteria in the OT data. Both the OT data and the additional source files have been validated against WOR 973 and reference files [5].

Based on a literature review and expert opinion, possible algorithms have been identified and assessed based on five criteria: practical applicability, evidential value, expected adjusting effect, expected robustness and interpretability of results. This has led us to choose these five algorithms: Decision Tree, Piecewise Regression, Random Forest, Gradient Boosting Machine and Artificial Neural Network. The selected algorithms vary in terms of the extent to which they deviate from the current methodology and thus allow for a broad exploration of the possibilities of machine learning for risk adjustment.

To avoid overfitting in the development and validation of the models, 30% of unique policyholders were randomly assigned to a test set for this study. The remaining 70% of policyholders form the training set. All algorithms were exclusively trained based on this training set. After the final definition of the model parameters, the metrics of each model are calculated for the test set.

Finally, interviews with stakeholders provided important insights into the consequences of applying machine learning to the current working method within the risk adjustment cycle.

1.4 Structure of the report

The next chapter describes the literature review of machine learning applications in risk adjustment and the selection of algorithms for this study. With that, this chapter will answer subquestion 1. Chapter 3 describes the data sources used and the application of this data. Chapter 4 describes the working of all models and presents the qualitative results of the different models (subquestion 2). Chapter 5 explains the results of the different models and describes the lessons that we can learn from this study in terms of the present adjustment model (subquestion 3). Chapter 6 lists the implications of the application of machine learning algorithms for the adjustment system (subquestion 4). And Chapter 7 gives advice on the applicability of machine learning to risk adjustment as well as recommendations for further research (subquestion 5).

2 Frame of Reference for Machine Learning Application in risk adjustment

This chapter describes the selection of machine learning algorithms that we tested in this study. To this end, section 2.1 first gives a relevant demarcation of machine learning applications within risk adjustment. Section 2.2 sets out the selection criteria we use for this study. Section 2.3 describes the different categories of machine learning that can be applied. And finally, section 2.4 sets out the arguments for choosing five algorithms based on the defined criteria.

2.1 Demarcation machine learning application with risk adjustment

2.1.1 The somatic adjustment model has 12 adjustment criteria and over 200 binary risk categories

In this study, we limit ourselves to the somatic risk adjustment model - this model aims to predict somatic healthcare costs at the level of individual policyholders as accurately as possible. The current somatic risk adjustment model uses Ordinary Least Squares (OLS) regression on 12 adjustment criteria:

- Age and gender
- Nature of Income (NOI)
- Socio-Economic Status (SES)
- People Per Address (PPA)
- Region
- Primary diagnosis cost group (pDCG)
- Secondary diagnosis cost group (sDCG)
- Pharmacy Cost Group (PCG)
- Medical Aid Cost Group (MACG)
- Physiotherapy Diagnosis Group (PDG)
- Multiannual High costs of Nursing and Homecare (MHCN), and
- Multiannual High Costs (MHC)

The adjustment criteria have been subdivided into approx. 200 binary risk categories for which standard amounts are calculated (see source [5] for a specification of these risk categories).

For the application of machine learning to risk adjustment, we have generalized the terminology of criteria and risk categories to ‘variables’, because in many cases there is no actual risk category anymore. In addition, we speak of 'modeled healthcare costs' (also called: 'modeled value' or 'model value') when discussing the 'prediction' that a machine learning algorithm (abbreviated to ML algorithm) makes for the costs of healthcare of an individual based on the variables.

2.1.2 Machine learning is relevant at various data application levels

In this study, we will explore the application of machine learning to the Overall Test (OT) data set, but also to data sets with a richer level of information. We do this because machine learning is ideally suited to establish relationships in data sets with multiple continuous and discrete variables. For this study, we distinguish four relevant application levels (see Figure 1):

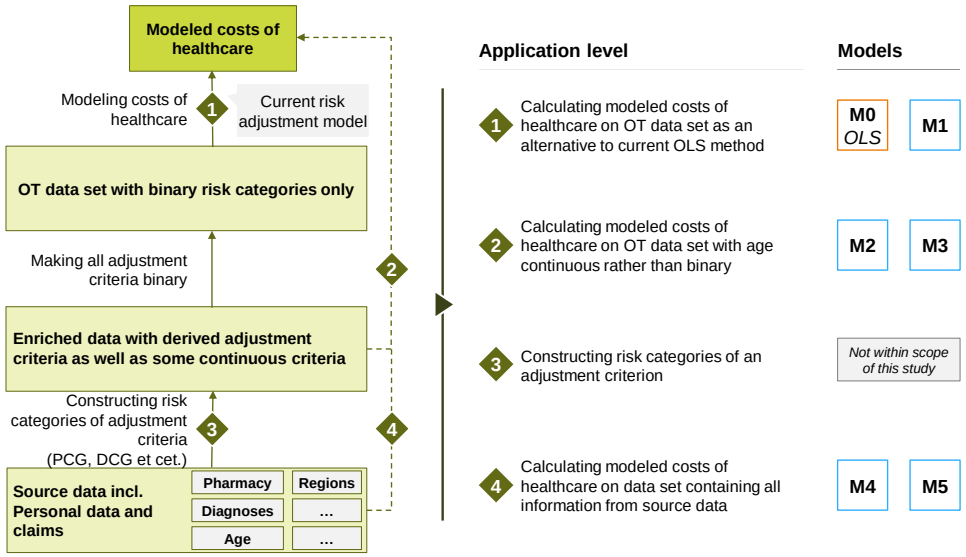
a) **Modeling costs of healthcare on the OT dataset.** The OT dataset comprises the 12 criteria divided over approx. 200 binary risk categories as described in section 2.1.1.
b) **Modeling costs of healthcare on an enriched OT dataset.** This enriched OT dataset is very similar to the current OT dataset, but it contains age as a continuous variable instead of in the form of five-year risk categories.

c) **The construction of risk categories.** In a separate step, risk categories are constructed from combinations of variables in a source file for each adjustment criterion. This application level is not an adjustment method in its own right - for example, the ultimate risk categories follow OLS regression to arrive at modeled health care costs.

d) **Modeling costs of healthcare on the source dataset.** No manipulation of variables has been performed on this dataset yet - no risk categories have been defined. For this study, for practical as well as theoretical reasons, we will only use the source data[3] for the morbidity criteria and existing risk categories for the other criteria.

For this study, we will explore all application levels based on which direct costs of healthcare are modeled, i.e. application levels a, b and d. For each of these levels we will calculate the effects of two models (including OLS as M0 model). As regards application level c, it better fits within regular major maintenance of a specific adjustment criterion than within this study. Besides, it is potentially very research-intensive. This is why we will not calculate the effects of application level c.

Figure 1: six machine learning models are explored on four different data levels.



2.1.3 Supervised regression algorithms are applicable for risk adjustment

There are many machine learning variants, each with its own application. Logically, not every algorithm is suited for risk adjustment. In risk adjustment costs of healthcare are modeled based on structured data and it calls for *supervised regression* algorithms (see Figure 2):

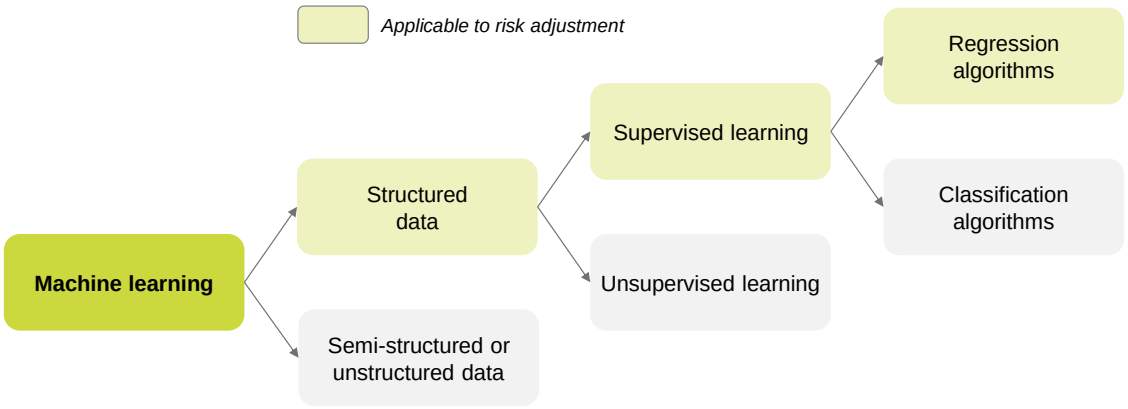
- **Structured data** – The risk adjustment is tabulated and thus has a clear structure. Each row in the table corresponds to *one* record (policyholder). Each column represents a variable of the policyholder, e.g. age or NOI. The other data format is (semi-)unstructured data, such as text files and images. We only consider algorithms that are applicable to structured data.
- **Supervised** – If the desired outcome is known, we speak of *supervised* machine learning. On application levels a, b and d, the outcome – i.e. the modeled costs of healthcare – are known. When applying *unsupervised* machine learning the desired outcome is not known. The third

[3] By using the source data for the morbidity criteria only (age and gender, PCG, DCG, MACG, PDG), we ensure that the data not contains unnecessary personal data. Moreover, the morbidity criteria are the most predictive in the current model and best fit the underlying idea of risk adjustment, making them the best candidates for further research.

level of application, the constructing of risk categories, is an example of this. Because we will not further explore the third application level, unsupervised algorithms have not been studied for this report.

- **Regression** – Finally, risk adjustment is a form of regression: predicting a continuous numerical value. There are algorithms that are stronger in categorization: assigning data points to previously known categories. In this study we will only explore algorithms that can be used for regression.

Figure 2: Risk adjustment calls for algorithms that belong to the category of supervised regression on structured data.



2.2 Selection criteria algorithms

As already described in section 2.1, it is a precondition for the selection of machine learning algorithms for this study that the algorithm is able to model costs of healthcare on the basis of tabular data (i.e. *supervised regression* algorithms). In addition, all selected algorithms must be an existing algorithm with which previous experience has been gained in (semi-)scientific studies.

Furthermore, we have used five selection criteria for the final selection of algorithms. They are:

1. **Practical applicability:** To what extent can the method be used in the present risk adjustment cycle? The current model has been applied for years and provides stakeholders with clarity based on annually published standard amounts. Many ML algorithms, however, do not yield standard amounts, but a complex combination of variables, or they use a different method, such as setting off policyholders against a large dataset. These algorithms score *lower* on this criterion. With other ML algorithms, the calculation time scales non-linearly with the number of records, so that these could not be researched during the course of this study. Such algorithms *do not* meet this criterion and therefore cannot be used.

2. **Evidential value:** To what extent has the technique been used before, either in literature or in a practical setting, for risk adjustment or similar problems? In many countries, OLS regression is the standard method for risk adjustment [6] which is why it is included in many comparative studies. As for other algorithms, the number of scientific and practical applications is smaller. Such algorithms get *lower* scores on this criterion.

3. **Adjusting effect:** To what extent do the modeled costs of healthcare match the actual costs? In recent years, the current model has consistently been able to model more than 30% of variation in actual costs of healthcare[5], [7]. ML algorithms that are expected to better model the costs of healthcare of policyholders score *higher* on this criterion.

4. **Robustness:** To what extent does the method produce stable results and is it robust for other hyperparameter settings? Every year, the adjustment model of the previous year is applied to the new data set – the shifts of standard amounts per risk category are usually very limited [5] and can largely be explained by increased costs of healthcare. However, with more complex algorithms it is quite possible that these are more sensitive to shifts in the underlying data. Such algorithms get *lower* scores on this criterion.

5. **Interpretability:** To what extent is the method easy to understand and can the results be explained easily? The results of the current adjustment model are easy to interpret - the standard amounts that are linked to the different risk categories can be applied independently to the entire policyholder population. Moreover, the relative level of the standard amounts are often easily explained on the basis of known relationships between health status and subsequent costs. However, there are also ML algorithms that are not as easy to interpret, for example because they categorize policyholders on the basis of combinations of many variables. Such algorithms get *lower* scores on this criterion.

Finally, in this exploratory study we aim to research the widest possible spectrum of ML algorithms. This means that we are looking for algorithms that are based on different principles and starting points, and that vary in terms of the extent to which they interfere with the current methodology. In the event of an equal score on the other criteria, we prefer the algorithm that is the *least similar* to the other selected algorithms.

2.3 Results literature review

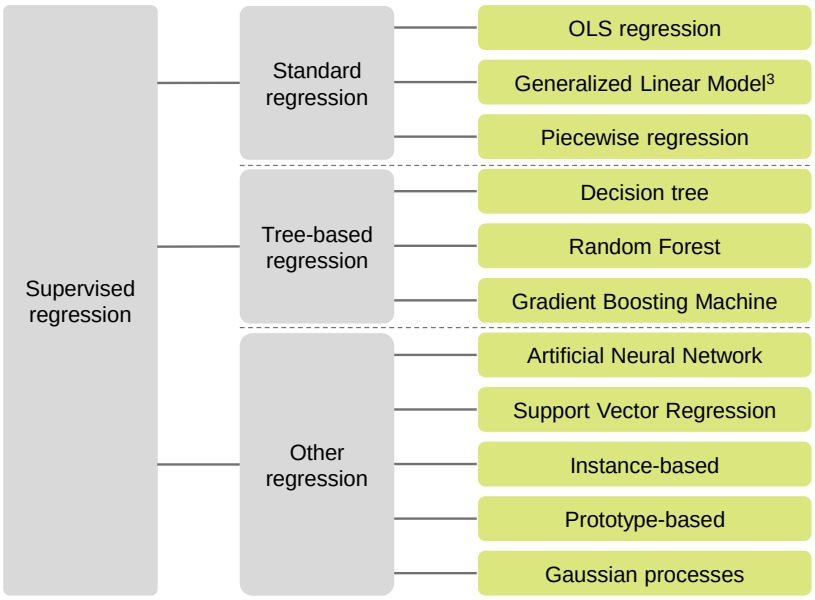
Machine learning is a very broad notion. There is a great diversity of algorithms, with different properties and varying applicability for risk adjustment. As described in section 2.1 and section 2.2 in this study we have only explored ML algorithms that can perform supervised regression on tabulated data and with which experience has been gained in (semi-)scientific research on similar issues. All the different types of machine learning that we describe in this section meet these preconditions.

The literature review shows that for supervised regression there are 11 common types[4] (see Figure 3). In turn, each of these types has many specific designs. For example, a Generalized Linear Model can use linear regression, logistic regression or Poisson regression. At a later stage of this study, we will describe the precise application of the selected algorithms.

Within the supervised regression algorithms, we distinguish between standard regression algorithms, tree-based algorithms and other regression algorithms. A description of these three categories and the underlying algorithms is given below.

[4] Naturally, more types of supervised regression are conceivable; these either fall into one of the categories described or have not been successfully applied to similar issues.

Figure 3: Eleven algorithms have been considered in this study, subdivided into 3 categories: standard regression, tree-based regression and other regression.



2.3.1 Standard regression algorithms

The standard regression analyses are among the most commonly used machine learning algorithms that rely heavily on statistics. This type of algorithm seeks to establish the relationship according to a certain comparison between the explanatory variables and the dependent variable (or modeled value, for risk adjustment this is the costs of healthcare), whereby the square of the distance of all data points to the modeled values is as small as possible.

Standard regression methods are very practicable: they are easy to calculate using conventional computing power and result in standard amounts that everyone can apply to individual policyholders. These algorithms are widely used in literature, although the evidential value differs per application form. The expected adjustment varies per algorithm but is generally lower than that of other machine learning classes. The algorithms, however, are relatively robust for changes in the underlying data[6] and results are easy to interpret because they yield standard amounts. Interpretation becomes more difficult as the model becomes more complex.

We distinguish three forms, which increasingly add complexity:

1. **Ordinary Least Squares regression** is among the most commonly used regression types. The algorithm searches for the combination of weighting factors of all variables leading to the line with on average the least squared distance between the modeled values and the actual values. This is the method used for the current risk adjustment model. The results of linear regression are easy to interpret and very stable over several years, but as for their adjusting effect they lag behind more advanced algorithms[8].
2. **The Generalized Linear Model** is a generalization of OLS regression, where the link function, which describes the relationship between the variables and the model values, also allows for nonlinear interactions. The chosen link function form strongly determines the predictive power

[5] Generalized Linear Model not including OLS regression, because for risk adjustment it is relevant to regard this as a separate category.
[6]This is true as long as one does not feed too many variable into the model, which would result in overfitting of the data.

of this model. The use of non-linear link functions may lead to better adjustment than OLS regression if appropriate interactions between variables are modeled; interpretability does suffer, to some extent.

3. **Piecewise Regression** (also referred to as segmented regression) applies regression to subsets (segments) of the data. Segments are made on one or more variables. The algorithm then performs a regression on each of the segments. Although per segment any type of regression is possible, this study only considers the repeated application of linear regression per segment, to facilitate a proper comparison with OLS regression. In previous studies, piecewise regression offered more predictive power than OLS regression, when applied to risk adjustment with extreme outliers[8], [9]. As is the case with the generalized linear model, this extra adjusting effect does mean interpretability suffers, because it yields separate sets of standard amounts per segment.

2.3.2 Tree-based algorithms

Tree-based algorithms use a selection tree, where the data set is split into subsets at each step (usually two). The algorithm calculates per step which variable and which split value best divide the data set. Each subset is split once more, often (but not necessarily) based on a different variable. A decision tree can be used on its own in a Decision Tree algorithm, or it can be combined with the Random Forest and Gradient Boosting Machine algorithms.

Tree-based algorithms have proven to be applicable to risk adjustment[8]–[11]. These algorithms often have a good adjusting effect and deliver robust results. As the algorithms become more complex, the adjusting effect increases as well, although this often happens at the expense of interpretability.

4. A **Decision Tree** algorithm uses a single decision tree, which can contain hundreds of splits for large datasets with many variables. The algorithm clusters the complete data into subsets, each with their own specific features and associated modeled value. In most cases, the Decision Tree leads to results that are easy to interpret, with a good adjusting effect if the data contains clear clusters [8], [10].
5. The **Random Forest** algorithm does not use a single decision tree but adjusts on the basis of the average result of several small and independently modeled decision trees. While Decision Trees tend to overfit, the RF algorithm averages the outcome of different trees to arrive at a more generalizable model of healthcare costs. This does require, however, that the individual trees correlate with each other as little as possible. For this reason, each tree is created based on an arbitrary subset of data and variables. The algorithm repeats this process several times, so that eventually it can contain many hundreds of simple decision trees. Also, the algorithm is robust to changes in the dataset by using more than one prediction. On the other hand, the model is not as easy to interpret compared to using a single decision tree [1], [2], [8], [9], [12], [13].
6. **Gradient Boosting Machines** also use multiple simple decision trees. The algorithm learns these decision trees sequentially; each new iteration of the tree uses the residual error of the previous decision tree to further train the model. As a result, the modeled outcome approaches the actual value slightly closer with each step. During adjustment, the algorithm adds the results of every iteration to arrive at a final modeled value of a policyholder's healthcare costs. Both in literature and in international machine learning competitions[7] this proves to be one of the more powerful forms of regression with a strong predictive capability and robust outcomes.

[7] E.g. the identification of fraud: kaggle.com/c/ieee-fraud-detection

As with the Random Forest algorithm this goes at the expense of the interpretability of the outcomes [1], [9], [13], [14].

2.3.3 Other regression algorithms

Five further regression algorithms cannot be classified into any of the above categories: Artificial Neural Networks (ANN), Support Vector Regression (SVR), Instance-based Regression, Prototype-based Regression and Gaussian Processes. The types of machine learning are diffuse in their operating principles and expected outcome on the selection criteria. What is clear, however, is that they have not been applied to risk adjustment as often as the aforementioned algorithms, and that in general they are less applicable to large data sets with many variables.

7. **Artificial Neural Networks** (also called *deep learning*) model complex non-linear relationships between variables and model values in a network consisting of 'neurons' organized in different layers. The input data constitutes the first layer and the modeled values are the final layer. There can be various hidden layers in between. In each layer, all neurons independently process the 'input signals' and transmit an 'output signal' to the next layer of neurons. Each neuron receives a signal from the neurons in the higher layer and sends a signal to the neurons in the lower layer. Thus, there are potentially a great many highly layered interactions between the input variables. Each signal has a numerical weight. The algorithm searches for the optimal combination of weights of all signals in such a way that the model values approximate the actual values as closely as possible. Artificial Neural networks have been successfully tested on the modeling of healthcare use by policyholders [9]. The literature review shows that these algorithms can be more accurate than linear regression, but because of the complex interaction possibilities between variables they lead to results that are difficult to explain.
8. **Support Vector Regression** is a specific form of regression used to capture linear and non-linear relationships. The algorithm combines the available variables so that as many points in the dataset as possible are within a defined distance (the margin of error) from this line. Sometimes only using linear combinations of the variables is not sufficient: in this case a specific kernel can be chosen to better separate the data points. The kernel transforms the existing variables, for example through a multiplication or a logarithmic function. Thanks to this transformation, the relationship in the data can often be captured better. Unfortunately, the number of calculations of a support vector regression algorithm does not linearly increase with the number of data points, making it difficult to apply this to large data sets [11].
9. **Instance-based** algorithms file all previously known data points as a training set. They compare each new record with all records in the training set and determine the model value based on comparability. Instance-based algorithms are susceptible to redundant variables that affect comparability with other data points: *dimensionality reduction* algorithms (see text box) help prevent variables with low predictive power from influencing regression. This form of machine learning also does not scale linearly with the number of records in the dataset, making it difficult to apply to health care modeling of a large population [8], [9], [14], [15].
10. **Prototype-based algorithms** are similar to instance-based algorithms but use prototypes as a summary of a (large) number of records. The model finds these prototypes by, for example, first clustering the data or by first reducing the number of records used in the instance-based algorithms. The closer a prototype resembles a predictable record, the heavier this prototype will weigh in the estimate of the model value. As with the instance-based method, these algorithms are not suited for large data sets due to the large amount of computing power required.
11. **Gaussian Processes** model functions with uncertainty – uncertainty is low near known data points but increases between data points. For this, the algorithm assumes that the observed

results jointly follow a Gaussian distribution. The model value of records between existing data points depends on the chosen algorithm settings. This form of machine learning is also difficult to apply to large data sets because it scales non-linearly with the number of records in the set.

The above algorithms are all capable of full regression on the dataset. There are further techniques that are used regularly: *regularization* and *dimensionality reduction.*

Regularization and dimensionality reduction are used in machine learning to prevent overfitting on the data. This may be relevant, for example, in piecewise regression models, where the data set is first segmented before performing a regression, so that suddenly there are many more variables in relation to the size of the data sets to which regression is applied. Through a *penalty* on the number of variables to be included or a *principal component analysis*, high-noise variables are weighted less heavily in the model than high-predictive variables – or even are deleted. It follows from literature that this can increase the predictive power of algorithms on new records [15]. This technique has not been used in the models included this study, so this may still be a starting point for future research.

2.4 Selection of algorithms for this study

The 11 forms of machine learning have been scored on the five criteria from section 2.2 based on our literature review and expert interviews (see Table 1). The explanation of the scores follows from the description per algorithm in section 2.3.

For two reasons, five algorithms have not been selected for further research:

1. Support Vector Regression, Instance-Based Regression, Prototype-Based regression and Gaussian Processes all scale non-linearly with the number of records[8]. Thus these algorithms are not practically applicable for this study. Moreover, there is relatively little evidential value for these algorithms; other algorithms are used more often in international machine learning competitions as well as in scientific literature.
2. Generalized linear model regression was not chosen because we only want to test *a single* standard regression technique besides OLS for diversity. Piecewise regression and generalized linear model algorithms score equally on all criteria. Ultimately, piecewise regression is preferred, because the current OLS method is already a specific variant of a generalized linear model.

[8] There are methods to reduce the number of calculations of these algorithms, but these generally decrease the predictive power of the algorithm.

Table 1: the assessment of 11 forms of machine learning on five criteria: practicability, evidential value, adjustment effect, robustness and interpretability. A low score is unfavorable on that criterion, a high score is favorable. When an algorithm cannot be applied for risk adjustment due to a criterion, this is indicated in orange.

	Practicability	Evidential value	Adjustment effect	Robustness	Interpretability	Selection
Standard regression						
OLS-Regression	High	High	Low	High	High	**M0**
Generalized Linear Model	High	Medium	Medium	High	Medium	-
Piecewise regression	High	Medium	Medium	High	High	**M2**
Tree-based regression						
Decision Tree	Medium	Medium	Medium	Medium	High	**M1**
Random Forest	Medium	High	High	High	Low	**M3**
Gradient Boosting Machine	Medium	High	High	High	Low	**M4**
Other regression						
Artificial Neural Network	Medium	Medium	Medium	Medium	Low	**M5**
Support Vector Regression	n/a	Medium	Medium	Medium	Low	-
Instance-based	n/a	Low	Medium	High	High	-
Prototype-based	n/a	Low	Medium	High	High	-
Gaussian processes	n/a	Low	Medium	High	Low	-

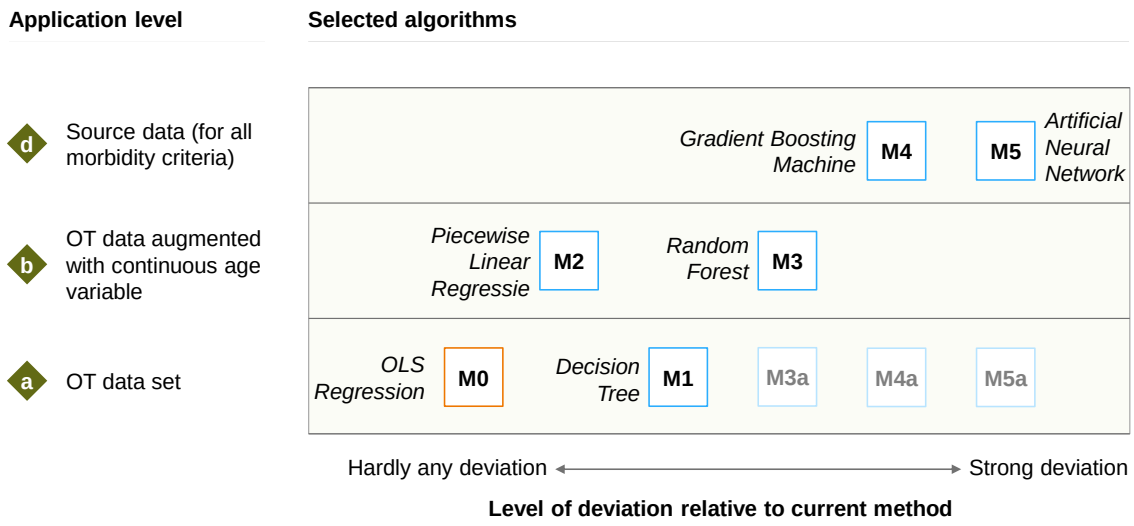
Based on the scoring of the algorithms, we have made the following selection of models (see also Figure 4)[9]:

- **M0 – OLS.** This is the model that is currently applied to the OT data set. It serves as a baseline for the comparison of the other five models.
- **M1 – Decision Tree**. The simplest class of all tree-based models is also applied to the OT dataset. Like OLS, it produces easily interpretable results because the algorithm achieves modeled healthcare costs per policyholder through a clear decision tree.
- **M2 – Piecewise Regression.** This type of regression we apply to the enriched OT dataset. It can be regarded as a logical extension of OLS. In this model, age will be used to segment the data for separate linear regressions.
- **M3 – Random Forest.** We apply this more complex tree-based algorithm to the enriched OT dataset. The algorithm can handle non-linear interactions in the data very well, which gives it great predictive power, but produces difficult-to-interpret results because it is an average of several individual decision trees.
- **M4 – Gradient Boosting Machine.** We apply this algorithm to the source dataset. Gradient Boosting Machines are the most powerful of all tree-based algorithms and tend to score very well in international machine learning competitions. Their adjusting effect is expected to be favorable, although as a result this algorithm too compromises on interpretability.

[9] Often, several underlying algorithms are suitable per model. The final selection of the algorithm will be made at a later stage of this project.

- **M5 – Artificial Neural Network.** We applied the Artificial Neural Network on the source dataset. This algorithm has been successfully applied to risk adjustment in several studies but is one of the most difficult to interpret models due to the large extent to which it allows for interactions.

Figure 4: Five machine learning models were selected for further exploration in this study, on three application levels[10].



During the study, the supervisory committee was very interested in also calculating the new models without adjusting the application level. This way, the effect of applying a different prediction model can be separated from the effect of enriching the data that models have at their disposal to make the prediction. In Figure 4 these additional model variants (M3a, M4a and M5a) are rendered in grey. Please note that it is not possible for model M2 (Piecewise Regression) to only use the OT data set, because the continuous age is required to define the segments.

In this study, we will report the results for these additional model variants only to a limited extent, namely wherever it is relevant to distinguish the effect of the model from the effect of the application level.

[10] As described in section 2.1 this study does not explore algorithms for application level 3 – the determining of the adjustment criteria (e.g. PCGs and DCGs) from the source data.

3 Data Sources and Application in this Study

This chapter describes which data was available and how this data was applied in this study. Section 3.1 describes the data sources used and the way in which these were validated. Section 3.2 then describes which input parameters based on these data sources were used for the different models. Section 3.3 discusses how overfitting is prevented in this study by dividing the dataset into a training set and a test set and by using *10-fold cross validation* within the training set. Finally, section 3.4 shows that the selection of the test set has been carried out in such a way that it creates a sufficiently representative subset of the complete data set.

3.1 Available data sources

For this study, we enriched the 2020 OT file with underlying source files that relate to age and morbidity criteria. The National Health Care Institute provided four further files for this:

1. **Personal data 2016 and 2017**. These files contain the age on 30 June 2016 and 2017 in number of years for all policyholders. In all, these files contained the ages of 16,912,322 and 17,014,864 unique policyholders, respectively. For 241,559 unique policyholders (89,925 policyholder years) from the OT file, no age information is available in either file. This is why the National Health Care Institute provided two additional files with the year of birth of all people who were insured at any time in 2016 and 2017. With this additional data, the age can be determined for all policyholders for whom age data was lacking[11].
2. **Claims file pharmaceutical care 2016.** This file contains all claims of non-hospital medication use within somatic care (i.e. exclusive use of hospital medication and add-on medication). Regarding these claims the following were given: ATC (Anatomic Therapeutic Chemical) code, product code, ZI (Z-index) article code and Defined Daily Dosage (DDD) factor[12]. In total, this file contains 242,383,160 claims lines of 11,556,060 unique policyholders. This data has been validated by establishing that for each policyholder in the OT file with a PCG classification, at least one associated product (including the information from the add-on file) has been claimed.[13].
3. **Claims file add-on medication 2016.** This file contains all add-on medication claims. This file contains 1,417,953 claims lines of 197,445 unique policyholders. An implementation table has been supplied separately with a translation of claims codes to ATC codes and DDD value. This dataset has been validated in combination with the pharmaceutical care claims file, as described above.
4. **Claims file DTC[14] somatic care 2016**. This file contains claims across all DTC trajectories: DTC product codes, diagnoses, specialty codes and claim codes. In total, this file contains 22,262,516 claims lines of 6,980,142 unique policyholders. This data has been validated by establishing that for each policyholder in the OT file with a PCG classification at least one associated diagnosis or claim code has been claimed.

Appendix A summarizes all data from the source files and describes what data is available for the different models.

[11] For policyholders where there is a discrepancy between the datasets, we take the average age across the files.
[12] The National Health Care Institute linked this data based on the G-Standard.
[13] For just 17 policyholders this allocation cannot be reproduced.
[14] DTC = Diagnosis-Treatment-Combination, the product structure by which hospital care claims are processed

3.2 Input parameters of the models

The data described in section 3.1 are made available to models M1 through M5 to varying degrees. Model M1 (Decision Tree) only uses the OT file. M2 and M3 use the OT file plus age in years. For this, we linked data from the sets personal data supplied to the OT file.

M4 and M5 make use of additional data on the use of pharmaceuticals and Medical Specialist Care (MSC) diagnoses. Such claims files can contain many highly detailed data items per policyholder. The pharmaceuticals database alone contains on average more than 20 claim lines per policyholder using pharmaceuticals. And for each of these rules, information such as dose, ATC code, etc. is available. This information can be edited to which results in highly relevant input for the eventual model. For example, if we want a model to take into account factors such as 'total annual dose per ATC code, 'total number of claims', 'number of ddd per pcg', etc., pre-processing is necessary - machine learning models do not do this automatically.

For this reason, it is necessary to determine so-called *features* that a machine learning algorithm can actually use to determine a model. The construction of features is not new: for example, PCGs and DCGs are (complex) features that are constructed on the basis of source data to provide targeted input to the model per policyholder. But numerous other features per policyholder are conceivable, e.g.: number of pharmacy prescriptions, number of packages per ATC code, or a binary variable per diagnosis or per diagnosis cluster.

It is impossible to conduct an exhaustive search for the best features in the limited scope of this study. This is why for this study, upon consultation with the supervisory committee, only a number of promising features were tested in models M4 and M5, namely:

- **Dx groups**. A binary variable per Dx group that indicates whether a diagnosis has been registered for a policyholder within the relevant Dx group in the previous year. These are 198 additional variables.
- **PCG_DDD.** A continuous variable per PCG that gives the number of DDDs (summed over all medicines within the relevant PCG) of a policyholder within the relevant PCG. This is 36 additional variables[15].
- **Patient_days_number.** A continuous variable with the number of patient days of a policyholder in the previous year. This is determined on the basis of averages in the Dutch Healthcare Authority (NZa) healthcare product profiles of the supplied healthcare products,[16] because healthcare activities are not available.
- **Procedures_number.** A continuous variable with the number of surgical procedures of a policyholder in the previous year. This was determined on the basis of averages in NZa standard profiles of the supplied healthcare products, because healthcare activities were not available to us.
- **Diagnoses_number.** The number of unique diagnoses per policyholder in the previous year.
- **Specialties_number.** The number of unique specialties visited by a policyholder in the previous year.
- **Healthcare_products_number.** The number of healthcare products of a policyholder in the previous year.

[15] We treat hypertension separately from diabetes, so instead of a PCG type I diabetes with hypertension and PCG type II diabetes with hypertension, we use type I diabetes, type II diabetes and hypertension as three separate categories. As a result, the effect of hypertension can also be included separately, since hypertension poses a greater risk of heart failure, renal dysfunction and CVA.
[16] Based on the version published on 17 November 2016

Eventually, model M4 and M5 were designed to include the features as rendered in Table 2. Due to the long computation time, no variant effect has been calculated for M4 including diagnoses_number, specialties_number and healthcare_products_number. Age in years was available for model M5 but did not lead to a better model during training.

Table 2: List of features in models M4 and M5.

	M4 – Gradient Boosting Machine	M5 – Artificial Neural Network
OT-data	X	X
Age in years	X	
Dx groups	X	X
PCG_DDD	X	X
Patient_days_number	X	X
Procedures_number	X	X
Diagnoses_number		X
Specialties_number		X
Healthcare_products_number		X

3.3 Preventing overfitting

3.3.1 Division of the data into training set and test set

Within machine learning it is customary to evaluate an algorithm *out-of-sample*, i.e. on a different dataset than the one on which the algorithm was trained. This method helps to prevent drawing the wrong conclusions as a result of overfitting. In the event of overfitting, the algorithm actually models random variation rather than variation driven by the input variables, which means that the model has a high predictive power on data that are already known, but that it does not generalize to unknown data very well. To avoid overfitting in this study, 30% of the data (5,158,453 policyholders)[17] were randomly assigned to a test set, the remaining 70% being part of a training set.[18,19]

All algorithms are trained exclusively with using the training set: the data contained therein is used to design the model, for example by calculating the standard amounts or determining relevant splitting criteria in a decision tree. The working of the models was only tested once on the test set at the end of the study, after the final determination of all model parameters.

3.3.2 Cross-validation

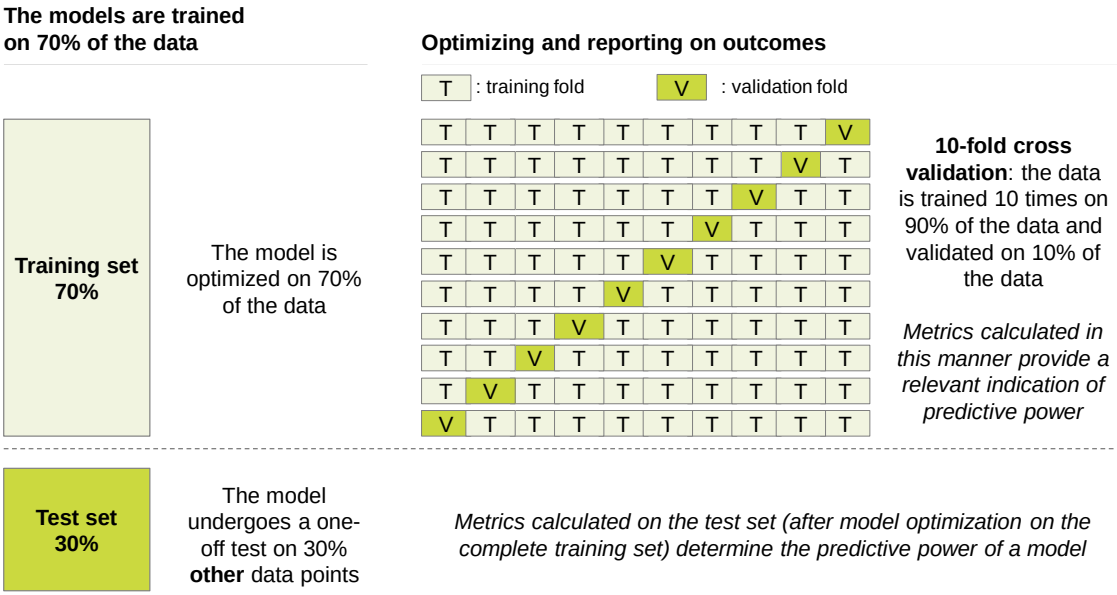
To prevent overfitting within the training set, we use 10-fold cross-validation. In this method, the training set is split into 10 random subsets or *folds*[20]. The model is always trained on 9 folds and tested on 1 fold. This means that 9 folds are input for designing the model, for example determining standard amounts per risk category or drawing up the decision tree. For the policyholders in the remaining fold, the model is then applied to model healthcare costs. Because the model performs

[17] This factors in policyholders who by switching in the current years, cover several rules.
[18] Ideally, we should separate the training set from the test set over time, and thus use data set of the subsequent year as the test set to actually evaluate the predictive capability of the model.
[19] For a subsequent study, we recommend to separate 40% of the data as a test set, as it turned out that the predictive power of all models on the test set is slightly better than on the training set.
[20] The same folds are used for every model.

this 10 times, each time using a different fold for validation, the entire training set is predicted *out-of-sample*[21].

The split between training set and test set and the effect of the 10-fold cross validation within the training set is visually represented in Figure 5.

Figure 5: Splitting data into a training set and a test set and applying 10-fold cross validation to determine out-of-sample metrics.



3.4 Characteristics of the training set and test set

The test set consists of 30% randomly selected policyholders. The training set contains the remaining 70% of the policyholders. Table 3 presents the descriptive statistics of the complete OT data set, training set and test set[22].

The average healthcare costs are 5 euros (0.2%) higher in the test set compared to the full set. At 51 euros (0.6%), the standard deviation of the healthcare costs is slightly higher in the test set than in the full set as well. Age, gender and the number of policyholders with a PCG, pDCG, sDCG, PDG, MACG, MHC or MHCN are very similarly distributed over the training set and test set, with a maximum deviation of 0.04 percentage point (percentage male).

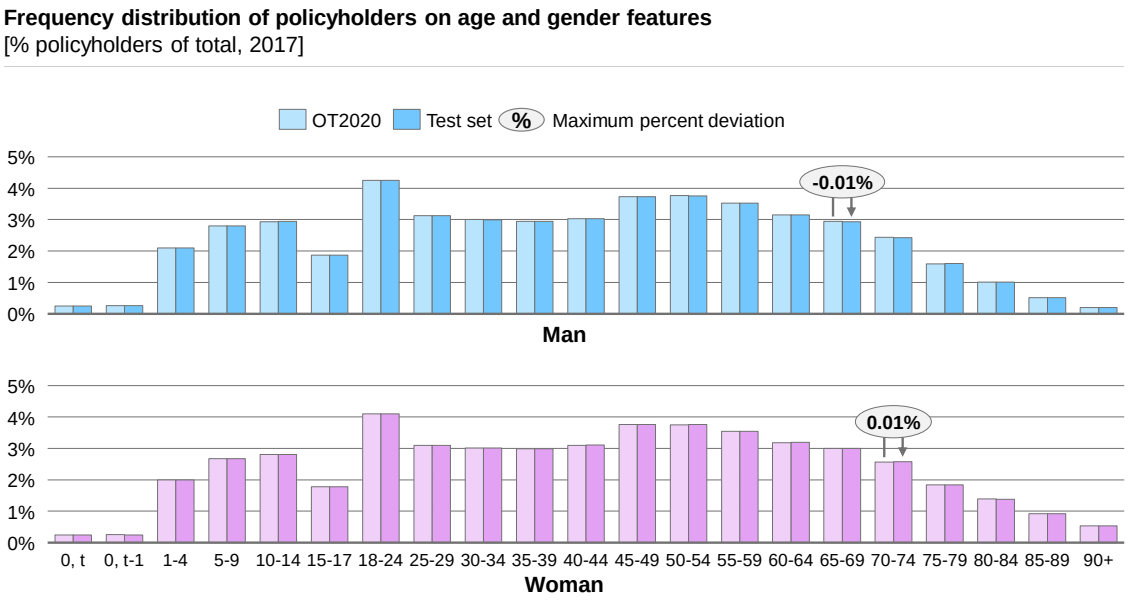
[21] Due to limited computing capacity, model parameters are first tested on a small set through hold-out validation. The 10-fold cross-validation is only applied if this results in a promising combination of model parameters.
[22] Naturally, these statistics are only determined after final determination of the models.

Table 3: Descriptive statistics of the complete set (OT2020), the training set and the test set.

	OT2020	training set	test set
Number of policyholders	17,194,536	12,036.083 (70%)	5,158,453 (30%)
Number of policyholder years	16,839,948	11,787,901	5,052,047
Average costs of healthcare	€ 2,355	€ 2,352	€ 2,360
Standard deviation costs of healthcare	€ 8,525	€ 8,503	€ 8,576
% man	49.45%	49.47%	49.43%
Median of age segment	40-44	40-44	40-44
% with PCG > 0	16.71%	16.70%	16.71%
% with more than 1 PCG	3.73%	3.73%	3.72%
% met pDCG > 0	8.88%	8.88%	8.88%
% met sDCG > 0	3.90%	3.90%	3.89%
% with PCG > 0	1.93%	1.93%	1.93%
% with MACG > 0	3.75%	3.75%	3.74%
% with MHC > 0	46.06%	46.06%	46.06%
% with MHCN > 0	2.38%	2.38%	2.37%
% with PCG, pDCG, sDCG, PDG, MACG, MHC or MHCN > 0	97.69%	97.69%	97.68%

The distribution of the different age-gender features in the complete set and the test set is shown in Figure 6. The largest absolute deviation for men is 0.01 percentage point, within the 65-69 age segment. For women, the 70-74 yrs age segment is the least equally distributed one, with a difference of 0.01 percent point as well.

Figure 6: Frequency distribution of policyholder years over age and gender features.



For each of the adjustment criteria, it was examined whether it is distributed significantly differently within the full set and the test set. The Chi-Square significance test has been used for all adjustment criteria, with the exception of PCGs. This test examines whether two datasets have a significantly different distribution of risk categories without making assumptions about the underlying distribution. The analysis of the distribution of PCGs calls for a different approach. Unlike the other adjustment criteria, the underlying risk categories are independent of each other:

a policyholder can have several risk categories[23]. A binomial test has been performed for every risk category. This test compares the probability of a specific PCG in the full set with the frequency actually found in the test set.

We have not found any statistically significant deviations between the two data sets for any of the adjustment criteria studied, except for PCG 37 (extremely high cost cluster 3), which occurs more often in the test set than expected. PCG 37 has a prevalence of <0.001% and deviates significantly with a p-value of 0.005.

To summarize, we conclude that the test set and training set match very well.

[23] With 12 exceptions, all PCGs are independent. These exceptions are described in the 'PCG ATC Reference file PCG's 2020', published by the National Health Care Institute. These exceptions are not included in the statistic comparison.

4 Results

This chapter describes the predictive value and adjusting effect of the five models studied. Section 4.1 gives a more detailed explanation of the working of the algorithms and the selection hyperparameters. Section 4.2 describes the outcome of the models on the usual standards for risk adjustment.

4.1 Description of the models

In this section we will describe the working of the tested machine learning algorithms. Each of these machine learning algorithms uses different *hyperparameters* that influence the functioning of the algorithm. A hyperparameter represents an algorithm setting, such as the number of *hidden layers* of an artificial neural network, the depth of trees in a Random Forest or the minimum number of data points per cluster in a decision tree.

The combination of hyperparameters with the highest found 10-fold cross-validated R^2 determines the eventual model settings. With this selected combination of hyperparameters, the algorithm is then trained once more on the entire training set. Finally, the trained model is used to model costs of healthcare for the policyholder in the test set.

In this study the models are optimized for R^2. This is the most common metric within risk adjustment (see, for example, WOR 973 [5] and Kan (2019) [15]). Minimizing a quadratic error, as happens with R^2 means that the prediction of a model is an approximation of the average real costs rather than the median of the costs (as is the case with optimization to MAPE). This is appropriate for a risk adjustment model in which on average insurers have to be properly compensated for the risk profile of their policyholder pool and in which the result remains relatively stable during switching movements.

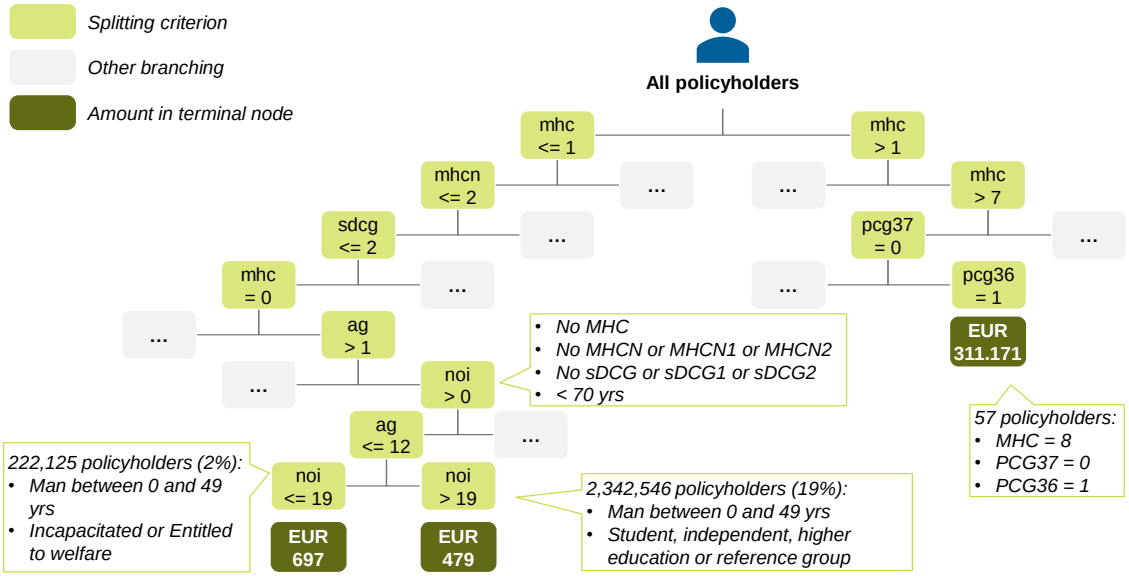
4.1.1 M1 – Decision Tree

Figure 7 shows a simplified representation of the decision tree on the training set. Using the risk features available in the OT dataset, the CART decision tree algorithm (Breiman et al., 1984 [16]) has divided the entire dataset into small subsets.

The algorithm uses the training set to shape the model. With each split, the algorithm calculates which criterion and which split value decreases the variation within the relevant subset the most. Thus, the risk features used first have the highest predictive value. When the variation decreases insufficiently, the subset of data is too small or the maximum number of sequential splits has been reached, a set is not split any further: on this *terminal node*, the algorithm calculates the average of healthcare costs of all associated policyholders. With the resulting tree, the algorithm places every policyholder to be adjusted in one of the terminal nodes. The corresponding value gives the modeled healthcare costs of this policyholder.

The tree shown in Figure 7 contains 872 unique combinations of risk features; the longest branch of this tree uses 23 risk features (9 of which as unique risk features, i.e. all except NOI, region and PCG), the shortest of only 4; the largest final group contains 19% of the policyholders, the smallest groups only 9 policyholders.

Figure 7: Simplified representation of a Decision Tree; 3 of the 872 terminal nodes are shown[24]



A compressed data set was used in training the model: all policyholders with the same unique combination of risk categories have been combined into a single policyholder set with weighted average costs of healthcare. This strongly reduces the computation time of the model. Appendix B describes all the settings of the algorithm used to create this tree; the main ones being:

- Minsplit = 60. The algorithm only examines whether a split is desirable if the subset to be split contains at least 60 unique policyholder sets.
- Minbucket = 9. The algorithm places a minimum of 9 policyholder sets in each *terminal node* to calculate the adjustment amount.
- CP (complexity parameter) = 0.000015. The algorithm splits a subset when the incremental improvement in R^2 is greater than the complexity parameter. Potential splits that do not provide sufficient improvement are not performed.
- MaxDepth = 30. The algorithm examines up to 30 consecutive splits.

Finally, the model ignores all combinations of risk categories that were insured for a total of just one day in the forecast year. These policyholders often lead to considerable outliers, because the healthcare costs per policyholder are scaled to a full year. Naturally, this operation is *not* performed on the validation and test set. The advantage of this procedure is that none of the *terminal nodes* get an exceptionally high adjustment amount from a single outlier in the training data. A decision tree is relatively sensitive to outliers, so that predictive power improves with the exclusion procedure. Incidentally, a similar procedure did not appear to lead to greater predictive power for the Random Forest, although this has not been exhaustively tested and it cannot be ruled out that it could still lead to some improvement.[25].

[24] Important: the numbers shown are only the ones within the training set, i.e. on 70% of the data

[25] A different procedure was also tested for both the Decision Tree and Random Forest: placing a cap on the healthcare costs to be included from the training set. This cap is 90 standard deviations above the average healthcare costs for each age-gender category. For policyholders with higher healthcare costs, the costs are equated to the cap. The total capped amount is then distributed uniformly across all policyholders within the

4.1.2 M2 – Piecewise Regression

The Piecewise Regression algorithm breaks up the OT data set into segments based on the age of the policyholders. For each of these segments, the algorithm separately calculates the standard amounts associated with the different risk categories. This results in 7 age segments, each with its own standard amounts.

The algorithm investigates which splits at the variable 'age' show the most interaction with healthcare costs and the other risk features. For this it uses a multivariate analysis that is very similar to the current OLS model[26]. Every possible age split is examined; the split that shows the most favorable effect based on the OLS model is actually applied. We prevented overfitting by enforcing a minimum number of policyholders per split. However, this does affect, for example, 0-year-olds, whom the model does not select as a separate group. This model can be improved in various ways in terms of the adjusting effect and explicability. The adjusting effect could be improved by performing a lasso regression per segment. This yields a linear model with standard amounts but uses a penalty at the level of the standard amounts to prevent overfitting. The explicability of the resulting standard amounts can be improved by running a regular OLS on the age segments founds. We will actually perform the latter in Chapter 5 to interpret the outcomes in order for these to be applied in the current OLS.

This algorithm uses two hyperparameters that deviate from default values. See Appendix C for the other hyperparameters:

- % fraction = 1%. The model calculates the most appropriate age split based on 1% of the training set. This corresponds to approximately 100,000 policyholders during the 10-fold cross-validation procedure. Larger fractions increase the required computing power considerably, while analyses performed by us with fractions of 2.5% and 5% show that the quality of the final model does not improve.
- Minsize = 10.000. Each subset must comprise at least 20,000 (2 x Minsize) policyholder years. Smaller sets are not split any further.

4.1.3 M3 – Random Forest

The Random Forest algorithm uses the training data to model a large number of independent decision trees. Each decision tree makes a random selection with return of the policyholders from the training set - such that the number of selected policyholders corresponds to the size of the training set. The algorithm can select a policyholder multiple times for the same tree, so that the trees are trained on uneven and independent data sets. The algorithm also makes a random selection of 20 risk features per split. Given these selections, the algorithm develops the best possible decision tree (see section 4.1.1 for the working of a Decision Tree). Figure 8 visualizes the working of this algorithm.

The algorithm determines the average modeled healthcare costs for all policyholders based on each of the 200 decision trees. The average of these 200 predictions then forms the definitive forecast for the relevant policyholder.

---

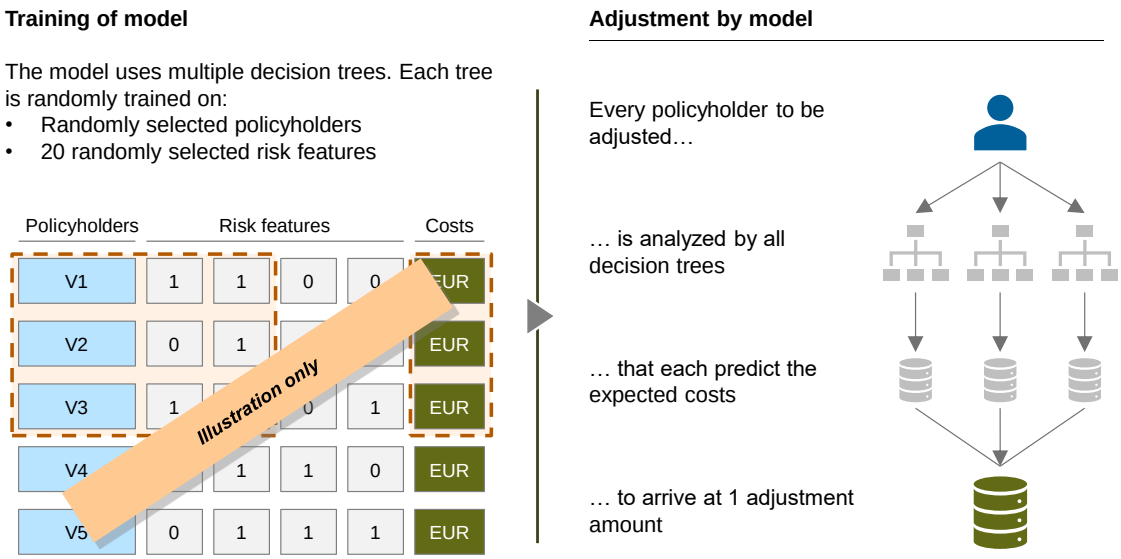
same age group in the training set. For the Decision Tree, this method did not work as well and for the Random Forest, the method did not lead to better predictive power.

[26] With three differences: 1) age is a continuous variable with *per age segment* a linear relationship between age and costs of healthcare; 2) gender is a separate binary variable; 3) age is no longer part of NOI, SES and PPA: they now have 7, 4 and 4 risk categories respectively.

All hyperparameters of this algorithm are described in Appendix D. The main settings of this are:

- Sample.fraction = 1. The total number of random policyholders per tree always equals the number of policyholders in the training set.
- Replace = True. Each decision tree uses the data of randomly selected policyholders from the training set *with* return – this allows a policyholder to be selected multiple times for the same tree.
- Mtry = 20. Every split uses 20 randomly selected risk features for the relevant policyholders to shape the algorithm.
- Numtrees = 200. The algorithm trains 200 independent decision trees, each of which models the costs of healthcare of all policyholders.
- Min.node.size = 30. Each split contains at least 30 policyholders.
- Max.depth = NULL. There is no limit to the depth each tree can reach.

Figure 8: The Random Forest algorithm creates 200 independent decision trees - each of these trees considers 20 risk features per split and a random combination of policyholders from the training set



4.1.4 M4 – Gradient Boosting Machine

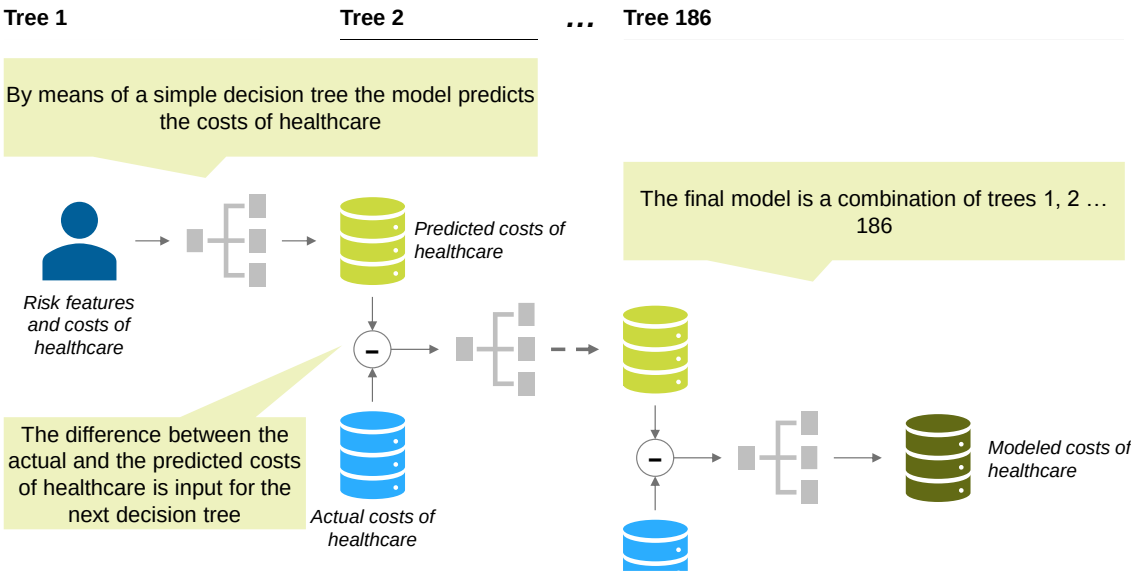
The Gradient Boosting Machine also uses simple decision trees to model the healthcare costs of policyholders. Instead of applying these trees in parallel, like the Random Forest algorithm, the trees are placed in series. This is shown by Figure 9.

The first tree makes a modeling based on the actual healthcare costs in the training set. The second tree compares the modeled costs with the actual costs - the difference between them is used by this tree as input to determine the splits. This is repeated a total of 9 times; the combination of these consecutive trees forms a predictive model of healthcare costs.

The model repeats this procedure 22 times in total. These 22 combinations of 9 trees jointly form the eventual model. Each of these combinations of trees is independent of the others - this reduces the risk of overfitting.

To further reduce the risk of overfitting, all unique feature combinations that appear in the file for exactly one day are removed. This same procedure was applied to the individual Decision Tree (see section4.1.1).

Figure 9: Concept representation of the implemented Gradient Boosting Machine.



A Gradient Boosting algorithm has many hyperparameters to be optimized. The main ones are presented below; in Appendix E includes a detailed list.

- Nrounds = 9. The algorithm uses 9 sequential decision trees, which jointly lead to a modeling of the healthcare costs.
- Number_of_tracks = 22. The model repeats the above procedure 22 times - each procedure is completely independent of the other procedures. The average prediction of all procedures forms the final modeling of the costs of healthcare.
- Subsample = 0.632. Every decision tree uses 63.2% of the available data points to base the splits on.
- Min_child_weight = 32. A subset is only split further if it contains data from at least 32 policyholders.[27]

4.1.5 M5 – Artificial Neural Network

Neural networks may appear very complex, but, like OLS, for example, they can in fact be traced back to one (very long and non-linear) formula. To illustrate this, 0 contains a simple elaboration of the working of an Artificial Neural Network.

The neural network to model healthcare costs implemented for this study is shown schematically in Figure 10.

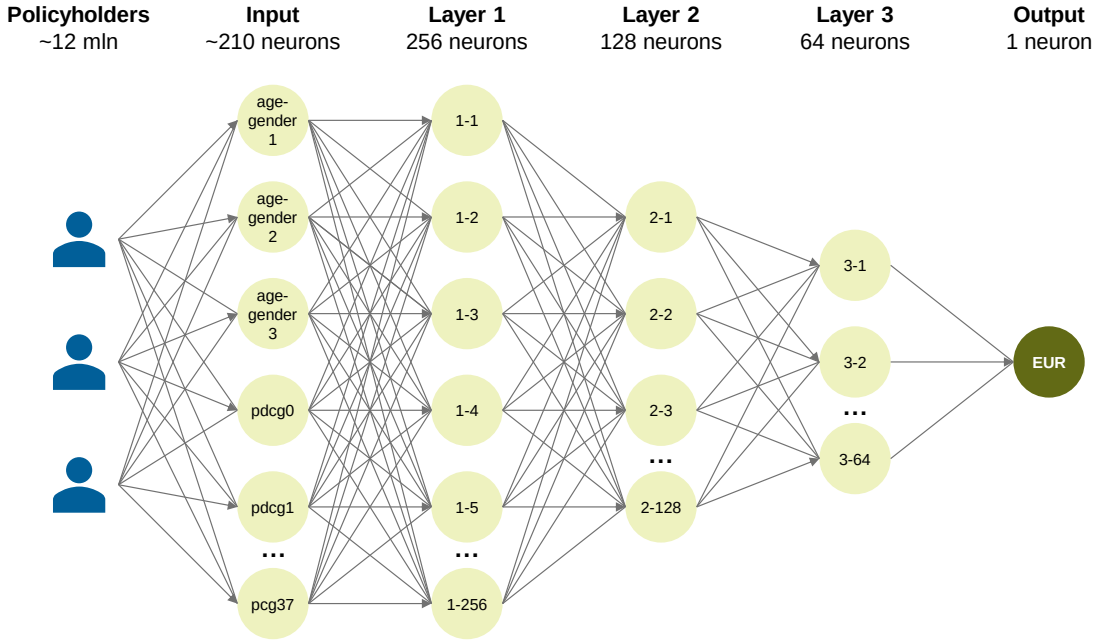
The *input layer* contains the data for all policyholders per risk feature from the training set. The data is sent to each of the 512 neurons from the first layer. Each neuron makes a weighted sum of the activations of the neurons in the previous layer. Next, a non-linear activation function (the

[27] The hyperparameter min_child_weight depends on the number of the procedure (1 t/m 22). The actual value used equals 32 * procedure number / 22

ReLU) is applied to calculate the activation of this single neuron. Exceptions are the input neurons and the output neuron. The activation of the input neurons is simply equal to the value of the relevant feature, and the output neuron combines all signals from this second hidden layer into a single prediction of healthcare costs and therefore has no activation function.

Ultimately, the algorithm searches for the best combinations of factors with which the neurons weigh the relevant input signals. Translated to the figure below, this means that every 'arrow' gets a weighting factor. Each neuron, in turn, has a 'bias' of its own, a number that influences the average activation of the neuron over the entire training set. Therefore, the 'bias' is comparable to the *intercept* in a linear model.

Figure 10: The implemented Artificial Neural Network uses two hidden layers with 512 and 256 neurons.



The *weighting factors* and *biases* are optimized using *batches* – combinations of data points – of 2,048 policyholders.[28, 29] Ultimately, each data point from the training set is fed into the model as input 45 times (this is the number of *epochs*). With every repetition of batches and epochs the model further optimizes the weighting factors. In this optimization procedure the algorithm uses the *drop-out* function: with every iteration 35% of the neurons from the hidden layers are randomly deactivated. This forces the algorithm to independently train the weighting factor as much as possible, and it limits the risk of overfitting.

The complete list of hyperparameters of the Artificial Neural Network has been included in Appendix G. The main parameters on this list are:

- Number of *hidden layers* = 2; number of neurons per layer = 512 and 256, respectively. The number of layers and neurons is related to the complexity of the relationship between the risk features and the costs of healthcare to be modelled.

[28] The version of this model on OT-data (M5a) uses 1,024 batches
[29] For a complete overview of the differences between models M5 and M5a, see Appendix G

- *Dropout rate* = 0.35. Every iteration of the training deactivates 35% of the neurons from the hidden layers. Note that during the forecast all neurons are active (however, the activations of neurons from hidden layers are multiplied by 0.65).
- *Batchsize* = 2.048, number of *epochs* = 45 and *shuffle* = true. The algorithm adjusts the weighting factors after evaluating the prediction on 2,048 policyholders. The algorithm uses every data point 45 times for training. After each epoch, the data is randomly distributed over new batches.

4.2 Predictive power and adjusting effect

This section describes the predictive power of the models and the adjusting effect of the models.

4.2.1 Predictive power of models

Table 4 shows the predictive power of the different models: the R^2 on the test set. For the sake of comparability the R^2 is also reported for M0 on the test set of exactly the same data sets as used for models M1-M5. Models M4 and M5 were mainly optimized for data level d[30]. It cannot be ruled out that the reported outcomes on the OT data set (level a) may still improve through further optimization of the hyperparameters.

Table 4: Predictive power of the six models (R^2)

	M0 OLS	M1 Decision Tree	M2 Piecewise linear regression	M3 Random Forest	M4 Gradient Boosting Machine	M5 Artificial Neural Network
OT dataset (data level a)	35.1%	35.0%		36.3%	36.3%	36.3%
OT with age continuous (data level b)	35.1%		35.3%	36.3%		
OT with source data (data level d1)	36.1%				38.5%	
OT with source data (data level d2)	36.2%					38.2%

Green *values are better than M0;* Orange *values are not as good as M0. Important: data level d1 = data included in model M4, data level d2 = data included in model M5)*

The Gradient Boosting Machine (M4) achieved the greatest predictive power with an R^2 of 38.5%. The OLS scored 2.4 percent points lower on the same data set. M3, M4 and M5 were better able to predict healthcare costs of policyholders based on risk features in the OT dataset than OLS. They all achieved an R^2 of 36.3% on the OT data set, outperforming OLS by 1.2 percentage points. Piecewise linear regression (M2) also has greater predictive power than OLS on the same data set by 0.2 percent point. However, this relatively limited improvement is accompanied by an almost sixfold increase in the number of calculated weights, because weights are determined per age segment. The Decision Tree (M1) has 0.1 percent point less predictive power than OLS on the OT dataset, but uses substantially fewer unique combinations of insurance features for this.

Appendix I shows the R^2 of each of the models determined through 10-fold cross validation in the training set and per fold in the training set. Although it is actually not desirable to draw conclusions about individual folds, it does give an idea of the robustness of the results of the models. The Gradient Boosting Machine achieves the best prediction on 8 out of 10 folds. The Artificial Neural

[30] For a complete overview of the data used per model see section 3.2

Network algorithm gives the prediction on the other two folds, but on these folds too the Gradient Boosting Machine only scores 0.2 percent point lower.

4.2.2 Normalization of the prediction

The sum of the modelled costs of healthcare od all policyholders must equal the sum of actual costs[31], so that in a given year payments do not exceed the available budget. The 2020 baseline model, trained on the complete OT dataset and using OLS regression, always meets this requirement. This is not true for models M0 through M5. This has two causes:

1. Models M1, M3, M4 and M5 are not designed to model for the average healthcare costs. Therefore, the sum of the modeled costs may deviate from the sum of the actual costs.
2. All models M0 through M5 predict healthcare costs using the training set - an uneven distribution of risk features and actual healthcare costs between the training set and the test set may lead to a deviation of modeled expenditure from the total available budget.

This is a known problem, experienced by Ismail (2018), among others [1]. To be in line with the macro budget, the modeled healthcare costs per model have been normalized to the actual healthcare costs by means of a correction factor. For models M0, M1, M2, M3, M4a, M5a, M4d and M5d these correction factors are 1.0016, 1.0013, 1.0034, 0.9965, 1.0118 and 09980, respectively. The greatest correction factor is needed for M4, the Gradient Boosting Machine, where without normalization the sum of the modeled healthcare costs is more than 1% lower than the sum of the actual healthcare costs.

The predictive power shown in table 4 and discussed in section 4.2.1 is based on unadjusted healthcare costs, as this best reflects the predictive power of the model. The reported metrics in the next section have been calculated on the basis of the normalized modeled healthcare costs per individual.

4.2.3 Adjusting effect of models

The adjusting effect of all models is assessed using metrics at four levels: standard amounts, individuals, subgroups and insurers. Table 5 shows the metrics that were calculated on the test set. The 2020 baseline set was trained and tested on the complete dataset, which is why it is not representative of the comparison with models M1-M5. The criteria in column M0 - OLS are in fact determined in the same way in the test set: the standard amounts are determined on the training set to then model healthcare costs in the test set.

A number of conclusions can be drawn from the calculated criteria of the five developed models and the current model:

- **Model M1 - Decision Tree** scores slightly lower than the current model on nearly all metrics. The metrics are negatively impacted on an individual, subgroup and insurer level.
- **Model M2 – Piecewise Regression** achieves an improvement on R^2 relative to the current model. On the other hand, however, there are substantially more policyholders with a negative standard amount. This high number is caused by the model assuming a linear relationship between age in years and healthcare costs. However, for the age group 0-8 yrs, policyholders born in the adjustment year are significantly more expensive, so this relationship does not exist. The modeled healthcare costs for part of the 8-year-olds are therefore negative. At subgroup and insurer level, a small deterioration can be observed on the majority of metrics. Incidentally, the results on the metrics are better if this model

[31] That is, actual costs in the OT dataset, therefore actual costs 2017.

is applied in a slightly modified variant, by performing a regular OLS on the age segments found. For more details see section 5.3 and Appendix L.

- **Model M3 – Random Forest** scores better than the current model on all individual metrics. With the same data the model shows an improved in R^2 from 35.1% to 36.3% compared with OLS. At the subgroup and insurer levels, different metrics show both an improvement and a deterioration. The addition of continuous age has a limited positive impact on the metrics for this model.
- **Model M4 – Gradient Boosting Machine** achieved good results on nearly all metrics. With the same data the model shows an improved in R^2 from 35.1% to 36.3% compared with OLS. The added value compared with OLS only really becomes apparent when extra source data are added: the R^2 then improves to 38.5%, while an OLS model achieves an R^2 of 36.1% with the same data. What stands out in particular, are the improvements on individual measure; these are the best of all models. It is just the bandwidth of the result of medium and major insurers that deteriorates slightly.
- **Model M5 – Artificial Neural Network** achieves good results on all metrics. With the same data the model shows an improved in R^2 from 35.1% to 36.3% compared with OLS. As with the Gradient Boosting Machine, the added value compared to OLS only really becomes apparent when we add extra source data: the R^2 then improves to 38.2%, while an OLS model achieves an R^2 of 36.2% on the same data. The complete trained model, including additional data, shows improvements at an individual, subgroup and insurer level compared to an OLS model using the same additional data set. The results at the insurer level are particularly favorable and the best of all models. All bandwidths become narrower. This also translates into the highest mean absolute result shift (MARS).

Appendix H compares all models with an OLS on the same dataset. For models M3-M5, which use additional source data, this provides additional information. It appears that M3-M5 outperforms an OLS model on the same data on all individual benchmarks. E.g. the R^2 of OLS on the same data as M3 is 35.1% and for the same data as M4 and M5 the R^2 is 36.1%. In summary, we see that M3-M5 are all an improvement over OLS on the same data.

Figure 11 shows the financial result per age (in years) of the models within the test set. All models follow largely the same line - volatile under- or overcompensation among policyholders under 5 or over 70, and a relatively stable and accurate forecast between these ages. On average, undercompensation and overcompensation among policyholders under 40 is lowest for model M4, where models M0 and M2 predict most accurately among policyholders over 70 and 80, respectively.

Figure 12 shows the financial result per age (in years) of the models within the test set. For M5, the financial result is closer to zero for 16 of the 24 risk bearers. This is consistent with the reported developments on the bandwidth of the financial result. M4 performs similarly for many risk bearers but reduces the bandwidth less because the financial results of the outliers improve less strongly. M1, M2 and M3 actually increase the bandwidth by a stronger negative and positive financial result on the two extreme risk bearers.

Table 5: Metrics of models on test set

Level	Metric		2020 Baseline model	M0 OLS	M1 Decision Tree	M2 Piecewise linear regression	M3 Random Forest	M4 Gradient Boosting Machine	M5 Artificial Neural Network	M3a[b] Random Forest	M4a[b] Gradient Boosting Machine	M5a[b] Artificial Neural Network	M0d1[c] OLS on data of model M4
Individual	R^2 x 100%		34.4%	35.1%	35.0%	35.3%	36.3%	38.5%	38.2%	36.3%	36.3%	36.3%	36.1%
	CPM x 100%		33.5%	33.6%	33.6%	32.9%	34.2%	35.7%	35.7%	34.1%	34.0%	34.1%	34.1%
	MAPE		1,980	1,984	1,983	2,004	1,965	1,920	1,920	1,969	1,971	1,968	1,969
	Standard deviation outcomes		6,906	6,909	6,915	6,898	6,843	6,726	6,741	6,845	6,843	6,843	6,854
	# with negative standard costs		15,054	4,412	-	62,137	-	58	544	-	19	-	4,390
Subgroups	MAPE on all subgroups		1,011	1,132	1,124	1,176	1,091	1,047	1,058	1,080	1,141	1,099	1,114
	Res. 15% lowest costs in t-3		107	112	171	140	144	107	111	159	159	150	108
	Res. 15% highest costs in t-3		-124	-129	-147	-105	-131	-118	-90	-191	-124	-158	-149
Insurer	R^2 x 100%		99.1%	98.9%	98.3%	98.6%	98.8%	99.0%	99.0%	98.8%	98.3%	98.9%	98.9%
	MAPE		26	29	38	31	30	26	21	31	32	30	28
	Bandwidth of results[f]	All	302	310	347	394	347	302	235	315	434	294	323
		Excl. 2	115	180	191	176	161	138	96	167	165	185	148
		Small	284	279	319	335	292	258	230	265	409	274	259
		Medium	154	157	192	179	167	160	138	172	168	160	178
		Large	64	63	99	64	71	77	57	79	60	84	67
		Not-concern	167	186	209	176	161	138	146	167	183	185	148
		Concern	302	310	347	394	347	302	235	315	434	294	323
	MARS[g]		8	-	14	3	4	12	19	6	6	6	4

Green *values are better than M0;* Orange *values are not as good as M0.*

[a] As reported in WOR 973 and reproduced for this study; the small difference on result 15% lowest costs t-3 possible by rounding up or very limited differences in data

[b] Model effect calculated on just OT dataset

[c] By way of illustration, OLS performed on dataset of model M4. For a comparison with other OLS models on further datasets, see Appendix H

[d] Large difference between baseline model and models M0-M5 is the result from the baseline model being determined on the complete data (~17 million policyholder years) while the other models only used 30% of the set (test set)

[e] Per model the subgroups were defined on the basis of the subgroups from the baseline models (1.85 million subgroups)

[f] This rules shows the bandwidth of the results are policyholder level where the two risk bearers that always determine the actual bandwidth have not been considered

[g] The MARS was compared with for all models, including the 2020 baseline model M0 – OLS

Figure 11: The financial result in Euros per policyholder year according to age in yrs

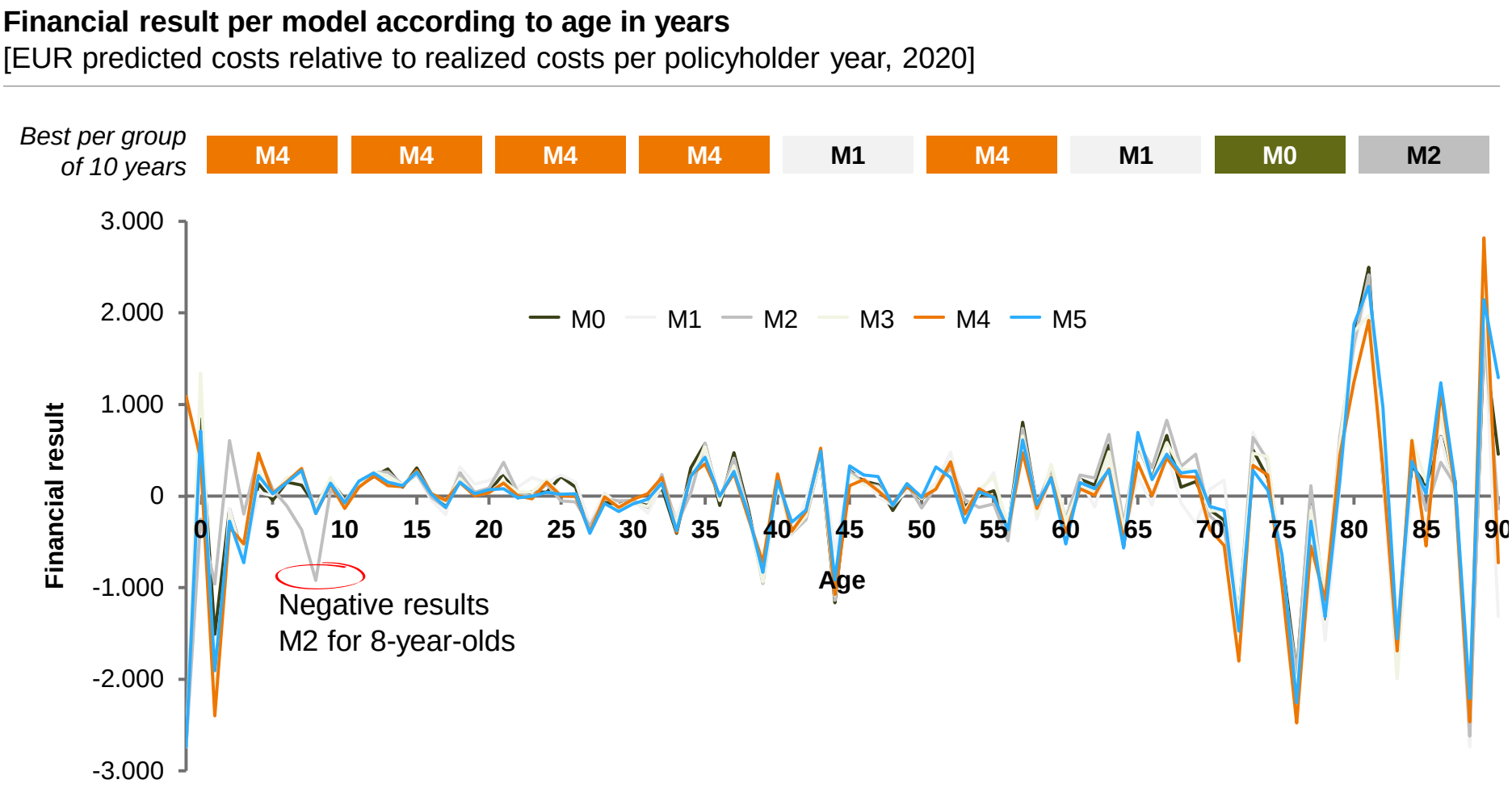
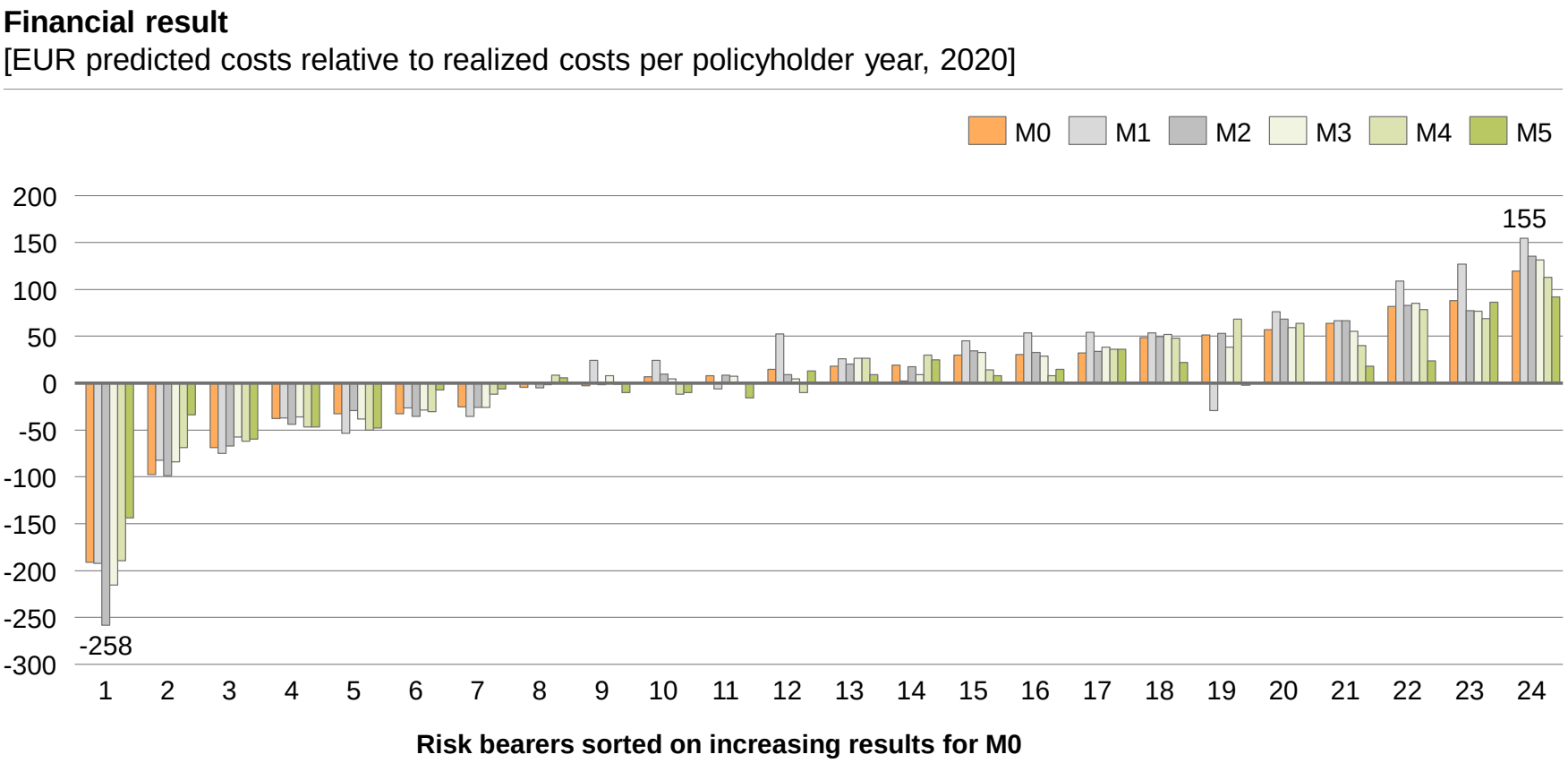


Figure 12: The financial results in Euros per policyholder year for 24 risk bearers.



5 Relevance of outcomes for the present risk adjustment model

In this chapter we will look at the results of the machine learning algorithms through an OLS perspective. That is, we interpret the results of the algorithms to determine what lessons can be learned for adding or adjusting adjustment criteria and risk categories. With this the algorithms tested in this study can be used as a testing ground for OLS in the research phase.

Section 5.1 shows the relative importance is of the risk features for all the individual models. On the one hand, this gives an indication of the added value of replacing OLS with a machine learning algorithm, for example because a model attaches less importance to non-morbidity criteria such as MHC and MHCN. On the other hand, it gives an indication of the value for the current OLS of new risk features that have been added to the models. Sections 5.2 and 5.3 go deeper into the risk categories with great importance from Decision Tree and Piecewise Regression respectively. These two algorithms are the most similar to the current OLS and provide handles to derive relevant new risk categories for OLS. Finally, section 5.4 summarizes which risk features and risk categories deserve further research within OLS.

5.1 Relative importance of risk features

Table 6 shows the 10 main risk categories of each of the models. Appendix J contains these results for the supplementary models on OT data (M3a, M4a, M5a).

Table 6: Ten main risk categories per model, numbered 1 (main) to 10

Risk category	M0 OLS	M1 Decision Tree	M2 Piecewise linear regression	M3 Random Forest	M4 Gradient Boosting Machine	M5 Artificial Neural Network
PCG	1	5	1	4	6	3
MHCN	2	3	4	2	2	2
MHC	3	1	3	1	1	1
pDCG	4	4	5	3	3	5
Age and gender	5	6	9	9		4
sDCG	6	7	6	6	10	9
PPA	7	2	8	7	7	6
MACG	8	9	7	8		
PDG	9		10			
NOI	10	8		10		
SES		10				
Age in years	N/A	N/A	2	5	4	N/A
Number of patient days	N/A	N/A	N/A	N/A	5	10
DDDs in PCG 37	N/A	N/A	N/A	N/A	8	7
DDDs in PCG 33	N/A	N/A	N/A	N/A	9	
Number of hospital products	N/A	N/A	N/A	N/A	N/A	8

Important: in the above table an empty cell means this risk feature does not occur in the top 10. N/A = not applicable because not available in the present dataset.

For this purpose, all risk categories have been ranked per model from most to least significant category. To be able to make this ranking, *permutation feature importance* was applied to the test set. This is a commonly used method for which the packages used include standard routines. With this method one feature at a time is mixed within the dataset between policyholders to then

determine another R^2. This has the simulated effect of a risk feature no longer being included, and will thus lead to a lower R^2. We consider the extent to which R^2 decreases as a measure of importance of the feature in question for the model on hand. The 10 most significantly contributing newly added risk categories are shown in Table 7.

Table 7: Ten main added risk categories per model, indicated with *x*

Risk category	M4 Gradient Boosting Machine	M5 Artificial Neural Network	M0d1 OLS on M4 data	M0d2 OLS on M5 data
Age in years	x	N/A	x	N/A
Number of patient days	x	x	x	x
ddds in PCG 37	x	x	x	x
ddds in PCG 33	x			
Number of procedures	x			
Dx 175 (dialysis)	x	x		x
ddds in PCG 36	x			
ddds in PCG 13	x			
Dx 1731 (chemo or immunotherapy)	x	x	x	x
ddds in PCG 24	x			
Number of healthcare products	N/A	x	N/A	x
Dx 1732 (chemo *and* immunotherapy)		x		
Number of diagnoses	N/A	x	N/A	x
ddds in PCG 26		x	x	x
Dx 400002 (pediatric oncology)		x	x	
Dx 21099 (leukemia)		x	x	x
Dx 176 (ventilation)			x	x
Dx 41013[a]			x	
Dx 21015[b]			x	
ddds in PCG 8				x

Important: in the above table an empty cell means this risk feature does not occur in the top 10 of most important new features. N/A = not applicable, because not available in the present dataset. a = Malignant neoplasm of the respiratory system and other neoplasms, b = Malignant neoplasm of the lymphatic and blood-forming tissue

Table 6 shows that there is only a limited shift of the main risk features. MHC and MHCN are among the top 4 most important features in each model - they remain important to arrive at a good prediction of healthcare costs. This effect is the greatest in the *tree-based* models M1, M3 and M4. The attribute greater value to MHC because these models have fewer risk features to choose from. Within these models, MHC is just one risk category, while in M0, M2 and M5 it is nine binary risk categories.

Age in years is an important contributing new feature in 3 models to which it has been added. The same is true for the number of patient days and ddds in PCG 37.

It follows from Table 7 that multiple added risk categories have an impact on the eventual predictive power of the model in question. For example, adding Dx risk categories on top of existing pDCG and sDCG categories to the models leads to a greater predictive power - in M4 and M5 but also in the OLS model.[32] Besides, four completely new risk features are used by the models for the prediction: the number of patient days, the number of surgical procedures, the number of healthcare products and the number of unique diagnoses.

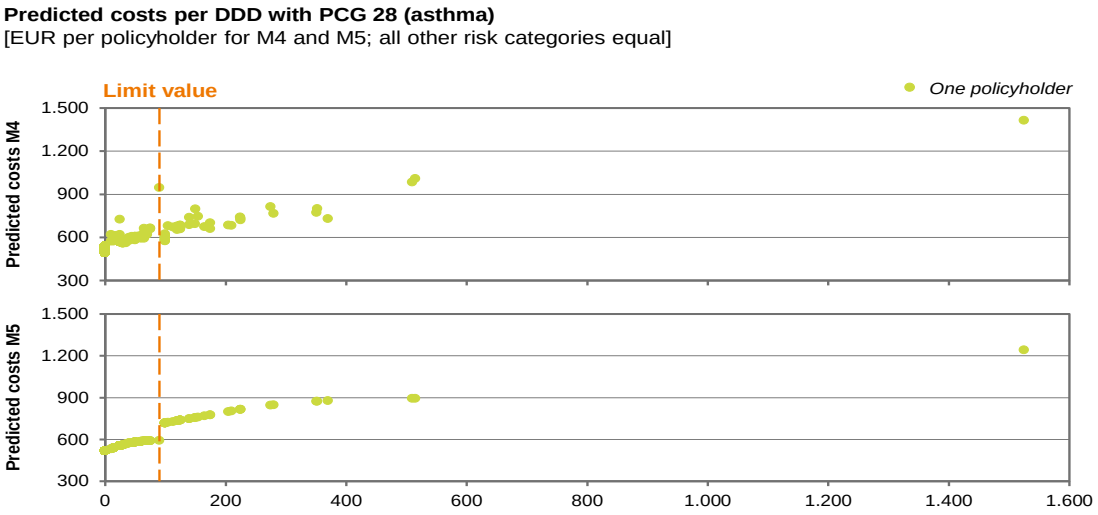
Incidentally, the above tables only indicate what happens if certain features are omitted in the *already trained* models. It would be interesting to see how, for example, a neural network would perform - and which features would contribute the most - if it were *specifically trained* on a dataset that did not include MHC as a feature.

Models M4 and M5 use ddds per PCG in addition to the existing binary PCGs (which for each policyholder with a number of ddds in the relevant PCG above a limit value is equal to 1). By analyzing the relationship that these models establish between the number of ddds and healthcare costs, we can develop hypotheses and further test the appropriateness of the chosen limit value and the impact of more or fewer ddds on healthcare costs. Figure 13 shows an example of this for PCG 28 (asthma).[33] For this analysis, we looked at the largest possible group of policyholders who are identical on all risk features except for the feature PCG 28. Then we looked at the number of ddds on PCG 28 within this subset and the predicted cost of healthcare. The artificial neural network in particular shows a clear, approximately linear relationship. For this specific PCG, there are clear additional costs for policyholders who have used fewer than the cap of 90 ddds compared to no ddds. But even over the cap the use of ddds results in higher modeled costs of healthcare. The patterns shown by model M5 is also interesting. At 90 ddd the predicted costs of healthcare get a boost because over this value the binary variable for PCG 28 is 'on' as well. both above and below the cap, though, we see a clear (linear) relationship between the number of ddds and the predicted costs of healthcare.

[32] we would like to stress that the dx groups were added as *supplementary* features, i.e. not to substitute pDCG and sDCG. It was decided to show the additional value as much as possible relative to existing model and feature choices in the current risk adjustment model. Of course it is also possible to have dx groups replace the current DCG structure, see for example the recent major maintenance DCGs (WOR 988) in which some options have been explored.

[33] This example was chosen because it concerns a relatively large group of policyholders, meaning that it was still possible to identify a sufficiently large group that was the same on *all* features except for this PCG. The analysis merely serves as an illustration.

Figure 13: Relationship between the number of ddds (on the x-axis) and modeled costs of healthcare in Euros (on the y-axis) for a group of policyholders of which all risk categories are equal except for PCG 28 (asthma)



5.2 Interpretation outcomes Decision Tree for OLS

As Figure 7 in the previous chapter show, the Decision Tree shows a number of large groups of policyholders with a limited number of features. See Table 8 for an overview of the five largest groups of policyholders clustered by the Decision Tree[34].

The features used for these five groups show that relatively healthy policyholders in particular can be captured well with fewer features than used by the current OLS.

Table 8: Five largest policyholder groups defined by Decision Tree; the features of these groups are defined here.

	Group 1 19% of policyholders	**Group 2** 9% of policyholders	**Group 3** 7% of policyholders	**Group 4** 6% of policyholders	**Group 5** 4% of policyholders
Gender	Man	Woman	Woman	Man	Man
Age	0_{t-1} – 49	0_{t-1} – 24	35 – 59	0-100	50 – 59
MHC	0	0	0	1	0
MHCN	0, 1 or 2	0, 1 or 2	0, 1 or 2	0, 1 or 2	0, 1 or 2
sDCG	0, 1 or 2	0, 1 or 2	0, 1 or 2	0, 1 or 2	0, 1 or 2
NOI	Student, highly educated, independent or reference	-	Highly educated, independent or reference	Student, highly educated, independent or reference	-
PCG	-	-	PCG0	PCG0	-
PPA	-	-	Single household or other	-	-
pDCG	-	-	-	pDCG = 0	-
PDG	-	-	-	PDG = 0	-

[34] Based on the 70% of policyholders in the training set.

This begs the question whether the current model overfits for these groups of policyholders. Therefore, the prediction of the OLS was compared with the prediction of the decision tree on these five groups. The metrics of OLS for these five largest groups of policyholders have been summarized in Table 9; the relevant metrics of the Decision Tree are zero by definition – for the modeled costs of healthcare on the subset equal the average costs of the subset.

Table 9: The prediction of OLS on five largest policyholder groups of the decision tree.

Measure	Group 1	Group 2	Group 3	Group 4	Group 5
R^2 x 100%	-0.4%	-0.7%	-0.1%	0.0%	0.4%
CPM x 100%	7.9%	3.9%	3.1%	-8.7%	-1.6%

Green *values indicate that OLS yields a better outcome on this measure for this group;* -Orange *values indicate that the decision tree model yields a better outcome on this measure for this group*

The table above does not give unambiguous results on the relative squared deviation (R^2) or the relative absolute deviation (CPM). Often, OLS performs better on one measure, but not as good on the other measure. The above results do show that is likely that OLS overfits on some of these groups – the use of *more* adjustment criteria for these policyholders may weaken the prediction.

The OLS model may benefit from this insight by taking into account the interaction by setting parameters that the decision tree does not use (e.g. PCG for group 1) to 0 for the group in question in the OLS.

Research by Buchner (2017) applies this two-step approach of calculating a decision tree to find relevant interaction terms for an OLS to a data set of approximately 2.9 million German policyholders [17]. The addition of the 95 interaction terms founds improves the R^2 from 25.43% to 25.81%. Van Veen (2018) applied the same method to the Dutch adjustment system and finds similar results: the addition of 145 found interaction terms improves the R^2 from 25.56% to 27.34% [18]. By only adding 3rd order interaction terms, 7 in this study, the R^2 only improves 0.08 percent point. It is therefore likely that the addition of the interaction terms found in this study will only lead to an improvement in the adjusting effect.

5.3 Interpretation outcomes Piecewise Regression for OLS

Further to the general results of Chapter 4 Appendix K presents an overview of the results of the Piecewise Regression model per age segment relative to the OLS model. Piecewise Regression shows a better performance an all segments except the first (0-8 yrs). The difference is particularly striking for the 17-26 and the 34-41 segments. Apparently having a separate model per age segment really makes a difference, and it is mainly the problem in the group of 0 to 8-year-olds that decreases the overall result of Piecewise Regression somewhat.

The standard amounts of model M2 are difficult to interpret, because there is a weight attached to the continuous variable for age. In addition, the model does not use the restrictions commonly used in the OLS model, so there is no reference class. This is why we used a regular OLS as an additional analysis within the segments determined by the Piecewise Regression algorithm. This has a limited, but positive impact on the metrics (for details see Appendix L).

With these adjusted standard amounts, the interpretation of results is often intuitive. For example, almost all standard amounts are positive (with the exception of the decline categories), they often vary around the standard amounts of the baseline model and the standard amounts generally

increase with heavier DCGs, MHCs and MHCNs. Appendix M shows the standard amounts per age segment for all morbidity criteria.

These standard amounts per age segment lead to an interesting observation: the relative added value of a high-risk category, for example MHC8, compared to the reference class, MHC0, decreases among older policyholders. This is shown by Figure 14. Receiving the MHC8 standard amount in accordance with the regular OLS would have led to an overestimate of healthcare costs.

Whether this is in fact the case cannot be determined with certainty on the basis of this study. What we do see, however, is that for the majority of policyholder groups model M2 better predicts the results than the baseline model, based on MHC and age segment. Particularly striking are the better results among policyholders aged 54 (M2 prediction is EUR 50 closer to actual costs than M0) and policyholders with MHC8 (M2 prediction is EUR 6.713 closer to actual costs than M0). Further research could home in on this observation to determine whether segmentation of features like MHC according to age could improve the adjustment result.

Figure 14: The relative additional costs of the different MHC risk categories per age segment relative to the reference category MHC0. Older policyholders receive a relatively lower contribution for a high MHC feature than younger policyholders.

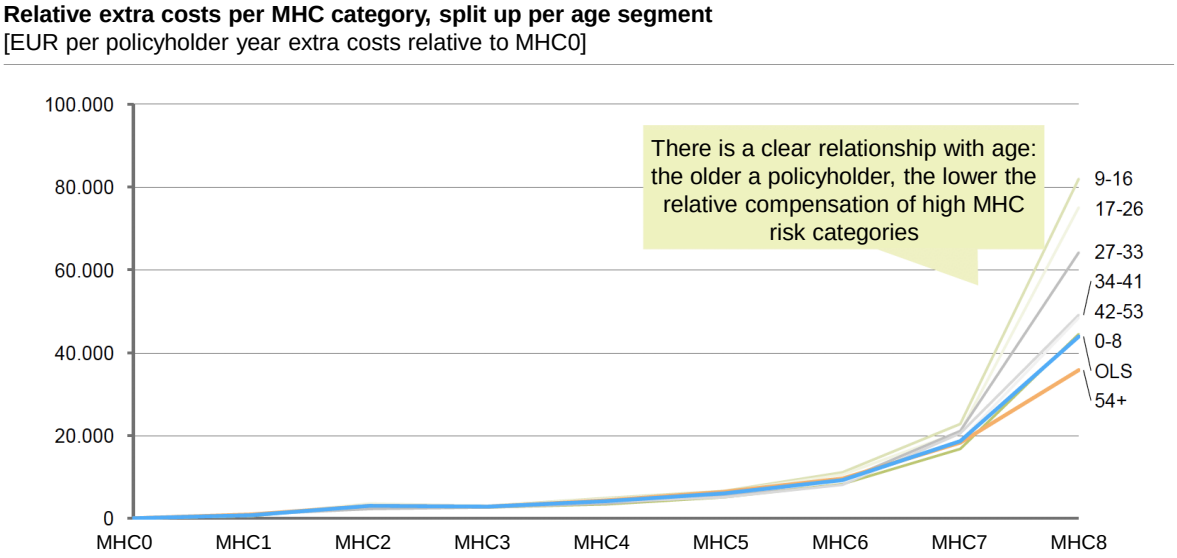


Table 10: Financial result M2 relative to M0. Negative amounts are an improvement of the financial result by using M2.

	Age segments:							
	0-8	9-16	17-26	27-33	34-41	42-53	54+	**Total***
No MHC	18	-26	-10	1	4	1	-52	**-15**
1	10	-137	-53	-104	16	-14	-46	**-38**
2	-1,464	-2,008	-59	-587	-190	35	-26	**-193**
3	57	-67	320	-149	130	-70	-26	**-42**
4	423	-268	-88	-555	-246	61	78	**60**
5	731	206	-346	-544	430	-35	-31	**20**
6	833	842	349	-54	-549	-447	22	**64**
7	-1,631	-3,007	-640	-1,005	1,930	581	-158	**-467**
8	1,576	-11,217	-38,827	11,506	-3,041	-5,039	-7,161	**-6,713**
Total*	**9**	**-61**	**-29**	**-45**	**7**	**-9**	**-50**	**-30**

** The total is not a sum of underlying components because a weighted average based on the number of policyholders was used.*

5.4 New risk features for OLS

In this chapter potential improvements of the OLS have been identified in various ways, based on the outcomes of the machine learning algorithms. This is not an exhaustive list, because for this study it was only a 'by-catch'.

We will judge the possible modifications of the current OLS based on the assessment criteria from the assessment framework (WOR 871): adjusting effect, efficiency, manageable complexity and validity and measurability [19]. In addition, we use the criterium of applicability: the extent to which the modification is applicable within the current adjusting system and for the current risk categories. Table 11 gives a summary of the results.

Table 11: Overview results assessment criteria for possible modifications of OLS.

	Applicability	Adjusting effect	Efficiency	Manageable complexity	Validity and measurability
Age in years	+ / -	+/-	+	+	+
Dx groups	++	+	-	-	+
Number of ddds per PCG	-	+	- -	-	+
Number of HC products	+ / -	++	-	+	+ / -
Number of patient days	+ / -	++	-	+	+
Number of diagnoses	+ / -	++	-	+	+
Features e.g. MHC, split up into age segments	++	+	+	+/-	+
Interaction terms for specific groups[35]	++	+	+	+/-	+/-

- The **applicability** has been assessed as very positive for the adjustments that concern a new binary risk category, because it fits in well with the current OLS. The assessment is neutral for age in years because it is added as a continuous variable or about 100 binary variables. The same is true for the number of HC products, number of patient days and number of diagnoses. These risk features could also be added to OLS as a risk features with a number of binary classes. The assessment is negative if a continuous variable is added to the model.
- The **adjusting effect** of age in years got a neutral assessment because Table 7 shows that this does not contribute a lot for OLS. This is in contrast to the number of healthcare products, the number of patient days and the number of diagnoses that add 0.6, 0.6 and 0.4 percent point R^2 to OLS, respectively. The assessment is positive for other potential adjustments, because there are clear indications that these adjustments can improve the adjusting effect. This has not been tested for all adjustments in this study.
- The **efficiency** of adjustments for which no new information has been added to the model has been assessed as positive, because there has been no change in the efficiency incentives of insurers and providers. Adding more detailed information about age does not

[35] E.g. as the groups defined in Table 8.

affect efficiency either. As for adding one or more Dx groups[36] to the model, efficiency is negatively impacted because incentives arise in the registration of diagnoses that lead to the categorization of a policyholder in a Dx group. The same is true for the number of healthcare products, the number of patient days and the number of diagnoses, which might lead to unwanted incentive effects. Adding a continuous variable with the number of ddds per PCG has a very negative effect on efficiency, because without a lower limit a policyholder immediately receives a higher adjustment contribution when prescribing a medicine from the reference table.

- The **manageable complexity** of the modifications that do not add interaction terms has been assessed as positive because the model hardly loses on simplicity and transparency. For modifications that do add interaction terms, the manageable complexity is neutral, because the number of categories increases only to a very limited extent. However, the interactions do decrease the transparency of the classification of the adjustment criteria. For dx groups and PCG-ddd, we score a '-' on complexity, because there are many extra variables involved that may be transparent but that make implementation more complex (think; regular maintenance, decisions yes/no trend, assessment plausibility, questions from insurers about specific categories).
- The validity of most adjustments has been assessed as high, because these risk categories have a logical predictive value for future costs of healthcare. This is true to a lesser extent for the number of healthcare products, because this number also depends on the maximum lead time of a healthcare product and the first month in which a healthcare product is registered. As for the interaction terms for specific groups too, the relationship between the newly constructed risk categories and future costs of healthcare is less obvious. The **measurability** of all adjustments is high, as the required data does not increase relative to the current OLS.

This possibility has already been explored in the recent overhaul of the DCG (WOR 988) and in the pre-OT 2020 (WOR 990), and a more detailed diagnosis categorization has been included. Naturally, these outcomes are more important than the first exploration we are conducting here.

6 Consequences of applying machine learning in regular risk adjustment cycles

In addition to the positive results of the use of machine learning models (outcome) in terms of R^2, the consequences of applying machine learning (feasibility) are also an important criterion in the appeal of machine learning for risk adjustment. In this chapter we will explore the consequences of applying machine learning techniques in the regular risk adjustment cycle, if one of these techniques were to replace the current OLS. We will start the chapter with a summary of our findings, which we will explain in more detail in the following sections. Section 6.1 explores the parameters to be organized for each of the models. Next, sections 6.2 and 6.3 describe the implications for the improvement and maintenance cycles. In section 6.4 the consequences for implication are explained, while section 6.5 concludes with the impact the models have on interpretation and incentive effect.

If one of the machine learning techniques was to replace the current OLS model, this will have consequences for all aspects of the risk adjustment cycle. Table 12 shows that the implementation of models M3, M4 and M5 in particular would have a great impact on the cycle as set up at present. Application in lieu of the current OLS may require modification of current legislation, and studies into the improvement and maintenance cycles will become more complex. In many ways, piecewise linear regression (M2) is similar to the current OLS model. In that sense, the implementation of M2 would be less complicated.

Table 12: Consequences of applying machine learning in risk adjustment.

		M1	M2	M3	M4	M5
		Decision Tree	Piecewise linear regression	Random Forest	Gradient Boosting Machine	Artificial Neural Network
Organizing parameters	Legislation	3	3	1	1	1
	Stakeholders	3	3	3	3	3
	Infrastructure	3	3	2	2	3
Improvement cycle	Implementation improvement cycle	2	2	2	1	2
	Interpretation improvement studies	3	3	1	1	1
Maintenance cycle	Data phase	3	3	3	3	3
	OT	2	2	1	1	1
	Standard amounts phase	2	3	2	1	1
Implementation	Adjustment	2	3	2	1	1
	Ex-post mechanisms	2	3	1	1	1
Interpretation and incentive effect	Validity	2	3	1	1	1
	Stability and homogeneity	2	3	3	3	3
	Transparancy and simplicity	3	3	2	2	2
	Incentive effect	2	3	2	1	1

Legend: 1 = major consequences, 2 = some consequences, 3 = no or limited consequences

The following caveats are important when interpreting Table 12:

- The table show the consequences of any of these techniques replacing the current OLS. Comparison with the current working method is not the main purpose of this analysis, but it does help clarify the consequences of the use of these models.
- The fact that consequences are significant in some cases is not a value judgment or a sign of impracticability. It does mean that the barrier for implementation will probably be higher.

In the following sections we will home in on the aspects as summarized in Table 12.

6.1 Organizing preconditions

Legislation

In Sections 32 and 34 of the Healthcare Insurance Act make provisions for risk adjustment. Article 32 reads: 'Each year, by governmental order, rules will be set concerning the calculation of the adjustment contributions'. These rules at least stipulate that the amount of the adjustment contribution is calculated on the basis of criteria that are equal for all health insurers, including at least the number of policyholders with a health insurer and a number of policyholder features. Also, a contribution is linked to each criterion, including the statistical substantiation.

Section 34 then specifies that ultimately on 1 April of the fourth year following the calendar year for which the contributions, as meant in Sections 32 and 3 have been granted, the National Health Care Institute will determine final contributions. The determination of an adjustment contribution in any case implies a recalculation of the adjustment contribution on the basis of the actual number of policyholders of the health insurer in the relevant year and the actual distribution of the policyholder features over these policyholders, insofar as the necessary information has been submitted to the National Health Care Institute in a timely manner.

Thus it has been established in the Healthcare Insurance Act that adjustment has to take place based on policyholder features, and that a contribution must be linked to every feature. In the Healthcare Insurance Decree (Section 3.6, par. 2) this is further specified: 'Our Minister assigns weights to all categories of the said criteria.'

Standard amounts resulting from the OLS comply with this wording. As do models M1 and M2: for these model both include a weighting per criterion (in model M2) or per group of criteria (in model M1) which is equal for all policyholders. Models M3-M5, on the other hand, no longer include a 'statistically substantiated contribution to each criterion'. After all, under these models, the contribution per criterion varies between individuals because it depends on the total of features of each individual.

This analysis is a first exploration, not a comprehensive legal review. The exploration shows that models M3-M5 requires at least a thorough legal review and possible a legislative change. For this reason we score these models on this aspect as 'major consequences'.

Stakeholders

If models M1-M5 were to be used instead of M0, no additional parties would be required for data collection or implementation. This is why in Table 12 we score this aspect as having 'no consequences' for all models.

Infrastructure

Making an OLS model for 17 million policyholders and about 200 risk features does not require special computing power, but some machine learning models do require significant computing power. Machine learning models are often developed on cloud infrastructure, which can easily be scaled up or down. Given the sensitivity of the data required for risk adjustment this is not a desired method, though. For this study, we used our own infrastructure, which is designed to process large amounts of healthcare data in complex analyses.

In conducting this research, we experienced noticeable limitations of this infrastructure, particularly in the development of models M3 and M4 (Table 13). Training these models took several days, and we also experienced steeply increased run times when we made more data available to these models (see, for example, the distinction between model M3a on OT data, vs.

M3 on OT data plus continuous age). It is important to realize that these run times are heavily influenced by the number of models that are trained simultaneously. In this study, in which many models were developed simultaneously in a short period of time and model calculations often ran in parallel within the same server capacity, this was an important contributing factor.

In essence this problem can be resolved: adding on extra infrastructure will solve the issue. For this study, too, we were able to overcome the problem by temporarily adding extra server capacity. This is it mainly a factor to take into account in the research design and budgeting.

Naturally, this problem is only relevant when training the model. The *application* of a trained machine learning model is simple and fast, same as a trained OLS model.

Table 13: Run times for training models per fold and overall (in hours)[37]

	M0	M1	M2	M3	M4	M5	M3a	M4a	M5a
HOURS *(per fold and total)*	OLS	Decision Tree	Piecewise linear regression	Random Forest	Gradient Boosting Machine	Artificial Neural Network	Random Forest	Gradient Boosting Machine	Artificial Neural Network
Min	0.24	0.03	0.13	17.94	4.41	1.05	7.27	0.27	0.79
Max	0.38	0.05	0.15	18.34	9.44	1.09	11.60	0.30	0.82
TOTAL	**3.38**	**0.39**	**1.38**	**181.21**	**62.59**	**10.76**	**93.35**	**2.82**	**8.08**

Minimum and maximum time per fold, and total time for the 10 folds (in hours). Please note: these times are highly dependent on the hardware on which the model was trained, and in particular how many other models were trained at the same time. During this study, for which many models were developed in a short time, this was a significantly important factor.

In addition to the required physical infrastructure, it is also necessary to work with other software packages. Analyses in the risk adjustment are often performed in SAS software (although other packages are also an option). Machine learning, however, is mainly developed within R and Python. Parties involved, e.g. National Health Care Institute, universities and research agencies may need to go through a learning curve to implement this software for this purpose, train staff etc.

All things considered, we score the aspect of 'infrastructure' as 'some consequences' for models that require the most computing power in our study. It will definitely have an impact, but in principle it is easy to resolve. The learning curve aspects are indeed important, but not of a lasting nature and therefore only weigh in to a limited extent in the scores in Table 12.

6.2 Improvement cycle

This section describes the situation in which machine learning models would *replace* the current OLS. Perhaps unnecessarily, we point to the fact that this is not an estimate of the *added value* that these models may have in the *current* improvement cycle (see Chapter 7).

Implementation improvement cycle

The objective of studies into the improvement cycle is to test possible improvements to the risk adjustment model. Improvement ideas are substantiated, which can lead, for example, to new or revised risk features. These features are then tested by applications in a risk adjustment model,

[37] These models make use of the hyper parameters as reported in the appendices to this document. Thus differences in run time between M3, M4, M5 and M3a, M4a and M5a, respectively, are a consequence of a different dataset as well as different model settings

using the WOR871 testing framework to test the improvement for aspects such as adjustment effect, measurability, stability and validity. [19].

If one of the models M1-M5 were to replace the current adjustment model, performing improvement studies would become more complex, because hyperparameters would have to be optimized as well. In this study too, we also found that, as with the selection of features, there is a certain degree of 'art' involved that will differ between different modelers: for many models, 'optimal' selection of hyperparameters is (almost) impossible. Increasingly, this will lead to the question whether finding a better model is the result of better features or of 'better' modelers who find better sets of hyperparameters. Especially in the more complex models, in particular model M4, the choice of hyperparameters is highly decisive for the result.

A more limited, but evident, consequence is the fact that some models require more from infrastructure than others. This will have to be taken into account when designing improvement studies, especially when many different model variants have to be tested. This aspect has already been mentioned above and we will not include it in the score again.

To summarize, the fact that hyperparameters must also be optimized when making machine learning models has some consequences for conducting improvement studies. Because there are many hyperparameters, with model M4 in particular, and the choice of these parameters also seemed to be greatly decisive for the result in this study, we especially scored the consequences of applying model M4 as great.

Interpretation improvement studies

In addition to conducting improvement studies, the use of machine learning models also has consequences for the interpretation of improvement studies. Obviously, it is still possible perform evaluations on the basis of a substantiated hypothesis whether the addition of a particular feature leads to an adjusting model with a better adjusting effect. But because the more complex models, especially M3-M5, factor in interaction more and because hyperparameters can be changed with every model change, it may be more difficult to pinpoint whether an improvement is actually caused by adding that feature. In addition, it may be more difficult to demonstrate aspects from the testing framework, like stability (phrased in the testing framework as: 'The adjustment criterion shows stability if the systematic link between the adjustment criterion and subsequent costs recurs annually') and validity (phrased in the testing framework as: 'The adjustment criterion is valid if it shows a systematic connection with the costs for insurers to be expected on the basis of the feature) according to the going definition.'

To summarize, we think that particularly the application of the more complex models (M3-M5) will have drastic consequences for the interpretation of improvement studies.

6.3 Maintenance cycle

The use of the models tested in this study does not necessarily lead to a different timeline of maintenance cycle activities. During the cycle, however, some activities will have a significantly different course:

- Data phase:
    - Machine learning models, more so than OLS, carry a risk of overfitting. It may be that a model is developed that is very capable of summarizing random variation in a data set, but that does not perform well at all when predicting a new, not formerly seen data set. To prevent this, it is important to train the model using a technique such as 10-fold cross-validation. And to be able to draw conclusions about how well

the model will work on a data set not formerly seen, it is important to test the final model for an unseen test set. In this data phase, therefore, a test set and a training set must be constructed after construction of the new total data set. The test set may not be viewed or used until the end of the OT, to avoid any appearance that modelers could be unwittingly influenced by features of the test set when designing their model (a principle similar to 'double-blind' working in randomized clinical trials).

- OT: in this phase the model is developed based on the training set.
    - The previously separated test set can be used at the end of the OT when several model variants have to be tested simultaneously, for example when multiple variants are possible after the pre-OT.
    - After the OT, the hyperparameters are fixed and will not be further tightened up, because it is not feasible to further explore hyperparameters in the short time available for the standard amounts phase. To ensure that this will not lead to problems during the standard amounts phase, one could consider making multiple model options in the OT that factor in multiple scenarios for what the macro budget will ultimately look like in September. In this way, there is still some freedom in the standard amounts phase to choose the set of hyperparameters that is most suitable.
    - Also, no standard amounts (per criterion or per combination of criteria) can be determined for models M3-M5 anymore. Instead, the model as a whole will be published open source. It is also possible to make the trained model available as a simple application to the stakeholders, so that, for example, insurers can easily determine the adjustment contribution for their policyholders.
    - Models M3 and M4 in particular require a relatively great amount of computing time. The pressure of time may play an important role where the use of these models is concerned.
- Standard amounts phase:
    - In this phase the model's final training takes place. Only the final weights within the model are determined now, the hyperparameters have already been determined in the OT phase.
    - The total OT set is scaled to the relevant macro amounts, and policyholder numbers are re-weighted to be aligned with the National Health Care Institute's policyholder estimate. The level of detail at which this estimate can still take place meaningfully does not depend on specific machine learning techniques, but on the level of detail of the *features* that are included in the model. In this study, models M4 and M5 use detailed underlying data sets, for which an accurate policyholder estimate probably is not possible. The importance of a good policyholder estimate at this stage will be obvious – otherwise conclusions must be based on data that is strongly outdated – but the effect of the level of detail at which this estimate is made is not entirely clear. Based on interviews and literature research, we found no clear evidence that very detailed estimates are more valuable than slightly less detailed estimates. Additional research in this context would be helpful to draw more precise conclusions. Not performing a policyholder estimate is probably only an option if the adjustment can be performed on much more recent data. As yet, this does not seem realistic.

In summary, the use of machine learning models has major implications for the OT phase and the standard amounts phase. The fact that hyperparameters cannot be tightened up in the standard

amounts phase leads to uncertainty, the impact of which is yet to be studied. As for models that also make use of more source data, it is important to realize that a highly detailed policyholder estimate is probably no longer possible, and it will first have to be investigated better whether the slightly less accurate estimate still leads to reliable results.

6.4 Implementation

The use of alternative models also has consequences for the actual implementation of risk adjustment by the National Health Care Institute. We will render the main aspects here.

Adjustment

For the different adjustment moments (from ex-ante to final settlement) insurers, the Tax Department, Employee Insurance Agency UWV and Vektis provide the National Health Care Institute with source data. The National Health Care institute checks these source files, and uses them for classification into features, estimating the standard amounts, determining ex-post corrections, et cetera. Models M4-M5 use significantly more source data, which will make the settlement output as described here more complex.

When actually using the final adjustment model for payment, it is important that this leads to the best possible ex-ante determination per insurer, which subsequently does not fluctuate inexplicably over the various moments. The stability of the settlement results across the various determinations *within an adjustment year* is therefore very important. It is undesirable for large shifts to occur, especially when this is due to model properties.

In machine learning models, as with OLS, the model is fixed after the standard amounts phase – the model itself does not change anymore. Shifts in budgets (with the exception of the ex-post corrections) are then caused by increasingly definite policyholder numbers and features. However, because the more complex machine learning models, especially models M3-M5, model extensive interactions, it cannot be claimed with certainty that the effect of changing inputs will not lead to greater fluctuations in outcomes than when using OLS. This has not been tested in this study, and we recommend that this be done in the future.

For health insurers it is important that they are able to calculate the amounts themselves. In essence, this will be no different is machine learning models are used. With the final trained model, insurers can make an exact calculation of the amounts using of the features of their policyholders. It is especially important that there is sufficient confidence in the workings of the model - through a combination of transparency, understanding and experience.

The National Health Care Institute regularly has to answer questions not only about the amount of the contributions, but particularly about the classification of policyholders into features as well. There continues to be a great need for the reliable, explicable establishing of features per policyholders - regardless of the model in which they are used.

In sum, we think that the successful use of machine learning models must primarily be based on trust. Stability, especially of the more complex models, has not yet been demonstrated and it is important that both insurers and the National Health Care Institute get confidence in the operation of the model. For this reason, we are assessing the consequences of using the more complex machine learning models in particular as major, especially those that use more data as well. Model M2 is the exception, as this model is very similar to the current OLS.

Ex-post mechanisms

Criterion neutrality, where the National Health Care Institute redetermines a number of weights afterwards because the number of policyholders is not found to be predictable beforehand, cannot be applied when a model is used that does not lead to standard amounts per criterion or per group of criteria (models M3-M5). A different ex-post correction mechanism would then have to be found and tested for these criteria. We therefore consider the consequences of the use of these models to be major in this respect Other ex-post mechanisms that do not depend on model outcomes at the feature level, such as high cost compensation and supporting policies, do remain technically feasible.

6.5 Interpretation and incentive effect

The use of a different model will also have consequences for the way in which results can be interpreted, and for the incentive effect of the model. In this section we summarize the main consequences.

Validity

According to the current testing framework, validity means that there is a logical, systematic relationship between the features on which the model is based and the costs of healthcare. Based on the analyses performed in this study, in particular in Chapter 5, we can conclude that the main features in the OLS model play a more or less similar role in the models M1-M5: the top 10 features are more or less the same across the different models. The models are therefore largely based on the same features that have been previously found to be valid.

What has not been tested, though, is whether the features still show a valid relationship with costs of healthcare in all cases. In the case of OLS and M2 this can easily be established. After all, OLS simply applies a standard amount per criterion, the (relative) height of which can be assessed. In case of a decision tree, the effect of an individual criterion is also still fairly easy to determine. With more complex models, this is probably (largely) the case, but no longer easy to test, due to the extensive interaction possibilities between features. It may be, for example, that a given DCG will generally lead to high modeled costs, but not for some combinations of insurance features.

The question is whether this is actually important. After all, one may decide that features are valid if there is a clear substantive (medical) basis for it, and for example in exploratory multivariate regression a clear relationship can be demonstrated between the feature and costs of healthcare. If a machine learning model based on this feature then leads to a good, stable adjustment effect, but does not always show the expected relationship between input variable and outcome, the latter may only be of secondary importance.

It is clear, however, that the *consequences* of using such models are major in this respect: it is either necessary to conduct comprehensive validity tests or to use a different definition of the term validity.

Stability and homogeneity

Stability and homogeneity, both within an adjustment year and over several years, have not been tested in this study. As a result, we cannot draw any conclusions about stability based on quantitative finding.

Conceptually it can be concluded that model M1 may be less stable with changes in data. For input changes can lead to a different order and combination of branches, which will easily lead to a

completely different model. In other models, where features work in parallel instead of serially, this effect is likely to be less substantial.

This effect does not apply within an adjustment year - after all, the adjustment model is fixed after the standard amounts phase. However, it may be that the model developed in the standard amounts phase is sub-optimal for the final dataset at the final determination. Potentially this uncertainty is an issue with any model, but perhaps to a slightly greater degree with models that are more prone to overfitting. That is why working with a training set vs. a test set so important.

The fact that models M3-M5 use randomness may also affect stability. For example, the neural network algorithm uses a *random number generator* to choose initial values. An easy way to deal with this for reproducing results is to use a 'random seed' value. This value helps to reproduce the same results within the same data set. Choosing a different random seed will lead to subtly different results. This solution does not help from year to year, but in practice the effect will be very limited - the effect of using other features and newer data from year to year is likely to be much greater. This is why we assess the consequences of the use of these models on the aspect of ‘stability’ as very limited.

In summary, the use of model M1 in particular will lead to some stability issues that one has to be mindful of. For the other models we think the consequences will be limited.

Transparency and simplicity

Explicability of models is of great importance for the confidence that parties have in them. Every year, the National Health Care Institute receives many, very detailed questions from insurers about the established adjustment contribution: about the amount of the contribution and about changes to this amount over the years. A transparent, simple model helps when answering these question. Although it is not always necessary for a model to be ‘simple’. As expressed in the assessment framework, there must ultimately be a balanced consideration with other aspects: a more complex model that does have a clearly better adjusting effect can still be highly desirable.

Models M1 and M2 are easy to grasp intuitively. M2 is as easy to explain as the OLS model but does lead to more standard amounts because the OLS is carried out on various age segments. M1 can even be drawn as a decision tree and contains fewer combinations of features than the OLS model.

Where models allow more room for modeling interactions between features, they also become more complex to understand. Model M3 can still be explained quite easily in terms of the underlying concept, but the effect of the model is less intuitive. This applies to an even greater extent to models M4 and M5. It is important to note that it is in fact possible to establish a relationship between a feature and an outcome - but the precise relationship between combinations of features and the exact prediction can no longer be comprehended without an additional analysis of the results.

Although these models score lower on simplicity, they can certainly be very transparent, which also goes for models M3-M5. Not only the source code, but also the trained model can be published, analogous to the standard amounts of the OLS. This allows any insurer to easily determine the adjustment contribution for its own population. As stated, it is also possible to make the trained model available to the stakeholders as a simple application, so that, for example, insurers can easily determine the adjustment contribution for their policyholders.

All in all we estimate that models M3, M4 and M5 in particular will have 'some consequences'. It is possible to provide the same transparency as with other models, but this requires more analyses because relationships are not always intuitive anymore. The fact that models are less simple weighs in only a little in our opinion, because simplicity is desirable, but it is not a goal in and of itself.

Incentive effect

All models use the same features as the current model. Models M2 and M3 use continuous age, which does not have a significant impact on the incentive effect. Models M4 and M5 also use more detailed features, for example, the number of ddds underlying the PCGs, the number of surgical procedures and the number of patient days.

These features my lead to an inefficiency incentive. This certainly applies to ddds: the exploratory analysis in section 5.1 also shows that a higher number of ddds may lead to a higher adjustment contribution in both model M4 and model M5. In practice it is therefore preferable to work with more detailed limit values than to actually include the continuous variable. The number of procedures and the number of patient days are based on standard numbers per healthcare product, which limits the incentive somewhat. Here too, the use of limit values is an obvious choice.

All models with the exception of model M2, allow for certain features to not be 'picked up' because they do not contribute significantly enough to the optimization of the target function, in the case of this study a better R^2. This may lead to an unforeseen and/or undesired incentive effect. For example, the 'higher educated' group has been intentionally added to the OLS to limit the incentive to risk selection, but as an extra feature within NOI only has a limited influence on the R^2. Nevertheless, this is a problem that can be solved: it is possible to provide models with constraints such that the result for certain groups is explicitly set to zero. This has not been further examined in this study.

Also, the potential negative incentive effect of models is offset by the fact that the potential value continues to be leading. If the best possible techniques are not used at a national level, it cannot be ruled out that individual insurers will apply these anyway. If this allows one insurer to predict the profitability of a group scheme better than another insurer, *failure to implement* techniques can actually lead to risk selection.

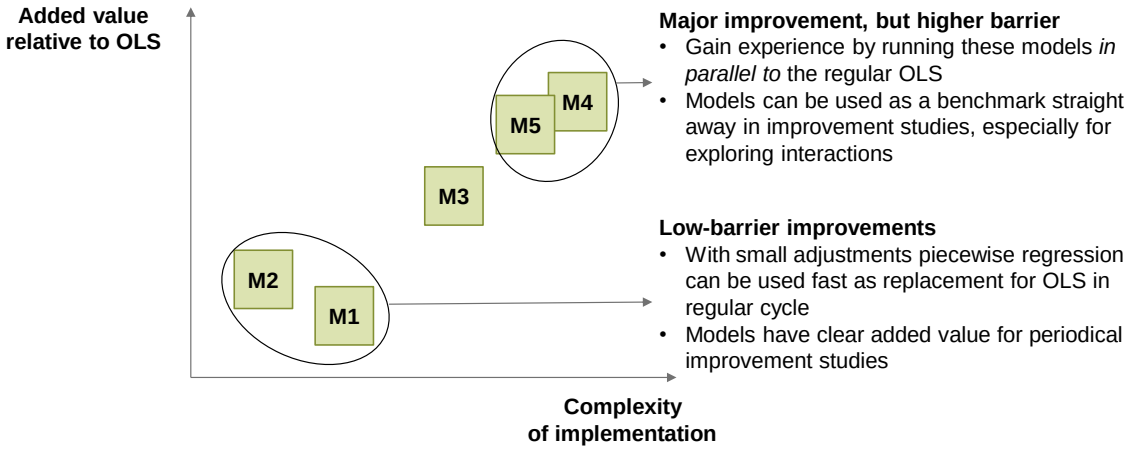
7 Conclusions and recommendations

This exploratory research provides many leads for the application of machine learning in risk adjustment. In section 7.1 we will describe where, based on this research, there are low-barrier opportunities for application, and where there are opportunities for major, but less accessible, improvements. In section 7.2 we will give concrete recommendations for additional research, and in section 7.3 we will summarize the main 'by-catch' recommendations for regular risk adjustment.

7.1 How to continue with machine learning in risk adjustment?

In this chapter, based on a weighting of the impact on adjustment on the one hand, and consequences of implementation on the other, we will provide accessible opportunities for improvement and opportunities with great added value but with a higher barrier for implementation. The recommendations have been summarized in Figure 15.

Figure 15: How to continue with machine learning in risk adjustment?



A number of models tested in this study are easily applicable and could offer added value soon:

- Model M2 (Piecewise Linear Regression) is a logical extension of the current OLS and, with limited further development, could potentially be suitable to replace the OLS soon, in particular to better deal with interactions between age and regular adjustment features such as PCG or DCG. For this same purpose, the model is also very useful in the improvement cycle.
- Model M1 (Decision Tree) is easy to implement and has led to the identification of large, cost-homogeneous groups in this study. Periodically conducting a study like this can help provide insight into interactions between features in a simple manner, and determine whether, for example, interaction terms can provide added value.

Models M4 (Gradient Boosting Machine) and M5 (Artificial Neural Network) both are very promising models. Even on exactly the same data, these models show slightly better results than OLS. It is important to note that the machine learning models have not been extensively optimized on the limited data sets, and that further improvement may still be possible. Even if these models will not *replace* the OLS as yet, they can soon prove to be of added value as an additional research technique by way of a benchmark for the 'best achievable' adjustment result (for example, as part of the OT), and as part of partial studies into the improvement cycle.

Especially when more data will be added, these models clearly show added value. The ability of these models to extract predictive power from interactions seems even more apparent. Note also that this was only an exploratory study, and even better results may well be achieved in the future with further optimization. On the other hand, these are also the models whose implementation will have greater consequences for the current way of working, which means that the barrier for implementation is higher.

7.2 Recommendations for further research

In addition to the specific recommendations for implementation of the different models, this study also leads to a number of more general recommendations for further research into machine learning used in risk adjustment:

- Our main recommendation is to run these machine learning models *in parallel* with the regular OLS for several years in order to gradually gain experience with the robustness and implementation aspects of these models, and also to address the main uncertainties. In addition, the best performing models could be tested retrospectively over several years.
- It would also be interesting to study how machine learning models perform when less desirable aspects of the current model are omitted. In this study, models were trained on *at least the same features* as available in the current model, but not on a *more limited* set of features. How valuable are these models if, for example, MHC or MHCN is not included as a feature?
- In this study only the somatic model has been explored. In future research it would also be useful to explore the use of machine learning for other risk adjustment models, in particular the mental healthcare model. It would also be useful to explore the extent to which machine learning can help achieve a reduction in number of required models in general.
- In further studies, it would be useful to look at other target functions. The main purpose of a risk adjustment model is to set up incentives the right way. Good target functions may not only take R^2 into account.
- Regularization and dimensionality reduction are used in machine learning to prevent overfitting on the data. Through a penalty on the number of variables to be included or a principal component analysis, high-noise variables are weighted less heavily in the model – or are deleted, even -- than high-predictive variables. It follows from literature that this can increase the predictive power of algorithms on new records. For example in the OLS model with additional data (e.g. M0b and M0d) in this study, this technique has not been used, so this may still be a starting point for future research.

7.3 Recommendations for the regular risk adjustment cycle

7.3.1 Lessons learned from the machine learning paradigm

Machine learning is not just a collection of tools and techniques, it is also a different paradigm. Within this paradigm some matters are self-evident while outside of this field they are not that obvious. Below, we will make a number of recommendations that were initially derived from the

machine learning paradigm. We think, however, that they can be of use in a general sense, and also within risk adjustment.[38]

- It is important to test the total of all models and operations: from compiling the data set to determining the amounts in the final determination. For this, all manual actions must be automated or at least captured in an unambiguous procedure with strict terms and definitions. All steps that are taken, including, for example, the implementation of the RAS method and the making of policyholder estimate, should be executed (i.e. according to the machine learning paradigm) in accordance with a protocol and tested in its entirety on the data from 2 or more earlier years. This way, the total prediction can be tested.[39]
- Force the use of a single outcome measure. During this study, the modelers regularly had to deal with the fact that wherever they were asked to optimize R^2, other outcomes, such as results at a subgroup or insurer level, implicitly turned out to be important as well. Discussions about the relative importance take place afterwards, and it may be that ultimately a model is developed that does not optimally match the *implicitly* desired mix of outcomes. Making the outcome measure explicit first, forces researchers to have a sufficient grip on the problem beforehand and for them to define optimal outcomes as precisely as possible. If multiple outcome metrics are important, it is best to combine these based on a previously determined metric. Another option would be to determine a lower limit for important but not-optimized outcome metrics.
- Keep a test set separate and secret and use it for a one-off test. This is important because otherwise there is always a risk of overfitting, with random variation being modeled. In practice, this risk is not that high with OLS, but, as becomes apparent from the working method in this study, it would be perfectly feasible in regular adjustment studies to work according to this best practice.

7.3.2 By-catch: possible leads for regular follow-up studies

Finally, this study has led to insights that may be relevant for further research within the regular risk adjustment cycle.

- When evaluating risk adjustment as is, it is useful to test other steps and not just the model development itself. In this study, for example, the question was raised to what extent the scaling of the OT data set to the policyholder estimate has an impact on the quality of the adjustment. How bad would it be if we do this less precisely or even not at all? After all, the estimate could be wrong, and the RAS procedure in and of itself is an assumption about the actual number of policyholders per subgroup. Besides, the combining of different models (somatic, mental healthcare and deductible) for a total result per policyholder can be useful to test separately, for example. Even smaller steps that are used in the adjustment do not always seem to have been fully lived through. For example, an exploratory analysis shows that the practice of scaling up costs for policyholders who are insured for only part of the year may lead to bias in the analysis (see Appendix N). Obviously

[38] Any readers who are interested in the differences in paradigm we can recommend Leo Breiman's very accessible paper: "Statistical Modeling: The Two Cultures (2001)"

[39] This would amount to the following: starting with the features of 2014 and the costs of 2015. Next, go through all the steps according to a recorded protocol (including the RAS method and the supplementing of missing values). Finally, the total costs for 2018 (including deductible and mental health care) would be predicted based on the features of 2017. These predicted costs should then be compared with the costs actually incurred in 2018.

there are good, perfectly logical reasons for doing this upscaling, however, the historical reference [20]–[22] in which this upscaling was first applied has not extensively explored the introduction of such a bias.

- The Decision Tree analysis identified a few major groups of healthy policyholders for whom feeding more data into the model seems unfavorable. It is worth studying whether adjusting the OLS so that it accounts for the groups identified in the Decision Tree analysis can lead to a better adjusting result in the current adjustment model.
- Piecewise Regression shows that MHC in particular shows possible interaction with age. Whether this is in fact the case cannot be determined with certainty on the basis of this study. What we do see, though, is that for the majority of policyholder groups model M2 better predicts the results than the baseline MHC, based on MHC and age segment. Further research could look into this observation in more detail and determine whether segmentation of features such as MHC by age could improve the adjustment result.
- Breakdown of PCG based on the number of ddds seems to add value to the predictive power, also in the case of an OLS model. It is useful, for example in the case of major maintenance, to explore whether a more fine-grained model, or at least even more specific threshold values per PCG, can lead to a better adjustment model.
- Age in years, number of patient days, number of healthcare products, number of diagnoses and number of procedures are all possible *features* that may increase the predictive power of the OLS model and are therefore worth looking into. It is especially important to include the incentive effect in the evaluation.
- Splitting DCG into dx groups seems to improve the predictive power. This possibility has already been explored in the recent overhaul of the DCG (WOR 988) and in the pre-OT 2020 (WOR 990), and a more detailed diagnosis categorization has been included. This is why we will not make any further recommendations for this now.

References

[1] I. Ismail, "Improving risk adjustment through machine learning: a comparative evaluation of Random Forests and Gradient Boosted Machines to OLS regression," 2018.

[2] S. Rose, "A Machine Learning Framework for Plan Payment Risk Adjustment," *Health Services Research*, vol. 51, no. 6, pp. 2358–2374, 2016.

[3] "WOR 856: Onderzoek 'gezonde verzekerden': verbetering van de compensatie voor chronisch zieken in het somatisch vereveningsmodel," 2017.

[4] "WOR 902: Huisartsenzorg in de risicoverevening," 2018.

[5] "WOR 973: Onderzoek risicoverevening 2020: Overall Toets," 2019.

[6] W. van de Ven *et al.*, "Preconditions for efficiency and affordability in competitive healthcare markets: Are they fulfilled in Belgium, Germany, Israel, the Netherlands and Switzerland?," *Health Policy*, vol. 109, no. 3, pp. 226–245, 2013.

[7] "WOR 929: Onderzoek risicoverevening 2019: Overall Toets," 2018.

[8] I. Duncan, M. Loginov, and M. Ludkovski, "Testing Alternative Regression Frameworks for Predictive Modeling of Health Care Costs," *North American Actuarial Journal*, vol. 20, no. 1, pp. 65–87, 2016.

[9] M. Morid, K. Kawamoto, T. Ault, J. Dorius, and S. Abdelrahman, "Supervised Learning Methods for Predicting Healthcare Costs: Systematic Literature Review and Empirical Evaluation," *AMIA Annual Symposium proceedings*, vol. 2017, pp. 1312–1321, 2017.

[10] S. Sushmita *et al.*, "Population cost prediction on public healthcare datasets," *ACM International Conference Proceeding Series*, vol. 2015-May, pp. 87–94, 2015.

[11] D. Bertsimas *et al.*, "Algorithmic prediction of health-care costs," *Operations Research*, vol. 56, no. 6, pp. 1382–1392, 2008.

[12] C. Yang, C. Delcher, E. Shenkman, and S. Ranka, "Machine learning approaches for predicting high cost high need patient expenditures in health care," *BioMedical Engineering Online*, vol. 17, no. S1, pp. 1–20, 2018.

[13] B. Panay, "Predicting Health Care Costs Using Evidence," pp. 1–13, 2019.

[14] C. Yang, C. Delcher, E. Shenkman, and S. Ranka, "Machine learning approaches for predicting high utilizers in health care," *International Conference on Bioinformatics and Biomedical Engineering*, pp. 382–395, 2017.

[15] H. Kan, H. Kharrazi, H. Chang, D. Bodycombe, K. Lemke, and J. Weiner, "Exploring the use of machine learning for risk adjustment : A comparison of standard and penalized linear regression models in predicting health care costs in older adults," pp. 31-12-2019

[16] L. Breiman, J. Friedman, R. Olsehn, and C. Stone, *Classification and Regression Trees*. Wadsworth, 1984.

[17] F. Buchner, J. Wasem, and S. Schillo, “Regression Trees Identify Relevant Interactions: Can This Improve the Predictive Performance of Risk Adjustment?,” *Health Economics (United Kingdom)*, vol. 26, no. 1, pp. 1/24/2017

[18] S. van Veen, R. van Kleef, W. van de Ven, and R. van Vliet, “Exploring the predictive power of interaction terms in a sophisticated risk adjustment model using regression trees,” *Health Economics (United Kingdom)*, vol. 27, no. 2, pp. 31-12-2019

[19] “WOR 871: Toetsingskader 2017,” 2017.

[20] R. P. Ellis, B. Martins, and S. Rose, “Risk adjustment for health plan payment,” *Risk Adjustment, Risk Sharing and Premium Regulation in Health Insurance Markets: Theory and Practice*, pp. 55–104, 2018.

[21] A. Ash, F. Porell, L. Gruenberg, E. Sawitz, and A. Beiser, “Adjusting Medicare capitation payments using prior hospitalization data.,” *Health Care Financing Review*, vol. 10, no. 4, pp. 17–29, 1989.

[22] R. P. Ellis and A. Ash, “Refinements to the diagnostic cost group (DCG) model,” *Inquiry*, vol. 32, no. 4, pp. 418–429, 1995.

Appendix A: Available data for the models

Table 14 gives an overview of the available data sources and the fields used for the different models. M0 and M1 only use the OT dataset. M2 and M3 also use ages in years. M4 and M5 use further details on morbidity criteria from source files.

Table 14: List of available data files.

Source	Name	Description	Used in model M4 and M5
Personal data 2016 and 2017	Leeftijd	Age in years on reference date 30 June of concerning year	yes, also in models M2 and M3 for 'leeftijd_continu'
	BTL_ident	Identification resident (1 = Netherlands / 2 = abroad)	no; only policy holders in OT-file are in model
Claims file pharmaceutical care 2016	ATC_code	ATC-code in 2016 (xxxxxxx = unknown) based on G-Standaard Z=Index	yes, for feature 'PCG_DDD'
	PRGR	Product code in 2016 (xx = unknown) based on G-Standaard Z-Index	no
	HPK_ddd	DDD factor per HPK unity	yes, for feature 'PCG_DDD'
	HPK_eh	HPK unity (xx = unknown) based on G-Standaard Z-Index	yes, for feature 'PCG_DDD'
	IK_eh	Unity of supplied quantity (xx = unknown)	yes, for feature 'PCG_DDD'
	HOEVEEL	Supplied quantity	yes, for feature 'PCG_DDD'
	ddd_aantal	*construct DDD with HPK_ddd and HOEVEEL*	yes, for feature 'PCG_DDD'
	ZI_nr	ZI number based on G-Standaard Z-Index	no
	AFL_datum	Date of delivery (EEJJMMDD)	no
	schade_FAR	amount of invoice	no
	Dcind	Indicator debet (= D) / credit (= C)	no
Claims file add-on medication 2016	diag_code_add	Diagnose code, only for nivolumab (decl_code_add = 194610)	yes, for feature 'PCG_DDD'
	decl_code_add	DBC invoice code, only for ADD-ON orphan drugs	no
	hoeveel	Number of units	yes, for feature 'PCG_DDD'
	schade_ADD	Invoice amount	no
	Dcind	Indicator debet (= D) / credit (= C)	no
Claims file DTC somatic care 2016	DBC_code	Invoice code	no
	ZP_code	Product code	yes, for features 'Dx-groups', 'Patient_days_number', 'Procedures_number' and 'Healthcare_products_number'
	DIAG_code	Diagnose code	yes, for features 'Dx-groups' and 'Diagnoses_number'
	SPEC_code	Specialty code	yes, for features 'Dx-groups', 'Diagnoses_number' and 'Specialities_number'
	INST_code	AGB code organization	no
	MND_open	Month of start traject	no
	schade_DBC	Invoice amount	no
	DCind	Indicator debet (= D) / credit (= C)	no

Appendix B: Decision Tree

For this study, the *rpart* implementation of the CART algorithm in R, version 4.1-15 was chosen. The following hyperparameters have been applied for the risk adjustment model (hyperparameters not shown are in accordance with the default settings of the relevant R implementation)

Hyperparameter	Setting	Explanation
Method:	'Anova'	Standard splitting criteria for a regression tree.
Control.minsplit	60 (default = 20)	Only in sets > 60 the algorithm will look for a new split
Control.minbucket	9 (default = minsplit/3))	Each split contains at least 9 policyholders.
Control.cp	0.000015 (def = 0.01)	A group is only split if with the split R^2 increases by at least 0.000015 point.
Control.xval	0 (default = 10)	We do not use internal cross-validation
Control.maxdepth	Default (30)	No need to define – with 2^30 possibilities control.cp and control.minsplit are leading in terms of the depth of the decision tree.

Appendix C: Piecewise Regression.

For this study, the *partykit* implementation in R, version 1.2-5 was chosen. The following hyperparameters have been applied for the risk adjustment model (hyperparameters not shown are in accordance with the default settings of the relevant R implementation)

Hyperparameter	Setting	Explanation
Alpha	Default = 0.05	The algorithm will split a dataset if this leads to a significant improvement ($p < 0.05$).
Minsize	10,000 (Default = 10)	Each split contains at least 20,000 policyholders.
Maxdepth	Default = inf	There is no limit to the depth of the tree. The tree is used to make the data split into segments.
Model	Model = adjusted OLS regression	The algorithm used to determine the splits is an edited version of the regular OLS regression

Appendix D: Random Forest

For this study, the *ranger* implementation of the Random Forest algorithm in R, version 1.2-5 was chosen. The following hyperparameters have been applied for the risk adjustment model (hyperparameters not shown are in accordance with the default settings of the relevant R implementation).

Hyperparameter	Setting	Explanation
Num.trees	200	The algorithm trains 200 independent decision trees
Mtry	20	Each tree has 20 risk features at its disposal
Importance	'impurity'	Each split created minimizes the *impurity index*.
Min.node.size	16 (on OT data) 30 (on OT data with age)	Each tree only will consider a split if the relevant subset contains data from at least 16 or 30 policyholders.
Max.depth	NULL	The trees have no limit on the number of consecutive splits.
Replace	TRUE	Policyholders can be selected multiple times to train the same tree.
Sample.fraction	1	Every tree uses as many randomly selected policyholders as the number of policyholders included in the training set.

Appendix E: Gradient Boosting Machine

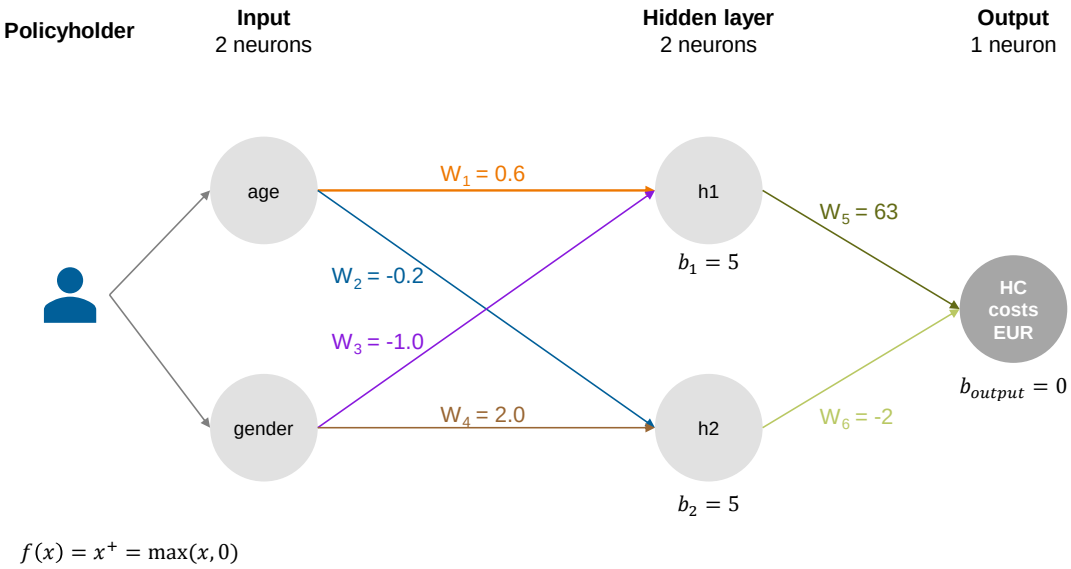
For this study, the *XGBoost* implementation of the Gradient Boosting Machine algorithm in R, version 0.90.0.2 was chosen. The following hyperparameters have been applied for the risk adjustment model (hyperparameters not shown are in accordance with the default settings of the relevant R implementation)

Hyperparameter	Setting	Explanation
Alpha	128	The regularization parameter reduces the influence of risk features with little predictive power.
Base_score	0	The base value of the algorithm equals 0 Euro.
Colsample_bytree	0.9	The algorithm randomly selects 90% of the available adjustment features per tree to train the decision trees.
Colsample_bynode	0.16	Each split of the tree randomly uses 16% of the adjustment features already selected for that *layer*.
Eta	0.5 (on OT data) 0.4 (on OT and source data)	Every tree on the algorithm has a weighting factor of 0.5 or 0.4 on the total prediction.
Gamma	1	Every split reduced the *loss function* by at least 4.8.
Lambda	1	The regularization parameter reduces the influence of risk features with little predictive power.
Max_depth	200 (on OT data) 10,000 (on OT and source data)	Each tree contains a maximum of 200 or 10,000 consecutive splits. (this is definitely not achieved)
Min_child_weight	16 (on OT data) 1,5-32 (on OT and source data)	Simply put: nodes that contain OT data on less than 16 policyholders are not split any further. The model for OT and source data varies this number between 1.5 and 32, depending on how many successive trees have already been created.
Subsample	0.632	Per tree, the algorithm uses half of the policyholders to train the tree in question.
Tree_method	'Exact'	The algorithm considers all available risk features to make a split.
Nround	9	The algorithm uses 9 consecutive decision trees, each of which minimizes the residual error of the previous tree.

Appendix F: Simple example Artificial Neural Network

Suppose we have a neural network with one *hidden layer* containing two neurons and the *input layer* also has two neurons. We model a simple risk adjustment model with only two features: gender and age. Figure 16 shows the already trained model.

Figure 16: Example of a simple neural network that has already been trained.



The weights between the input layer and the hidden layer use a function to translate the input values into a signal for the neurons in the hidden layer. This can be rendered as follows:

$$h_1 = f(x_1 * w_1 + x_2 * w_3 + b_1)$$

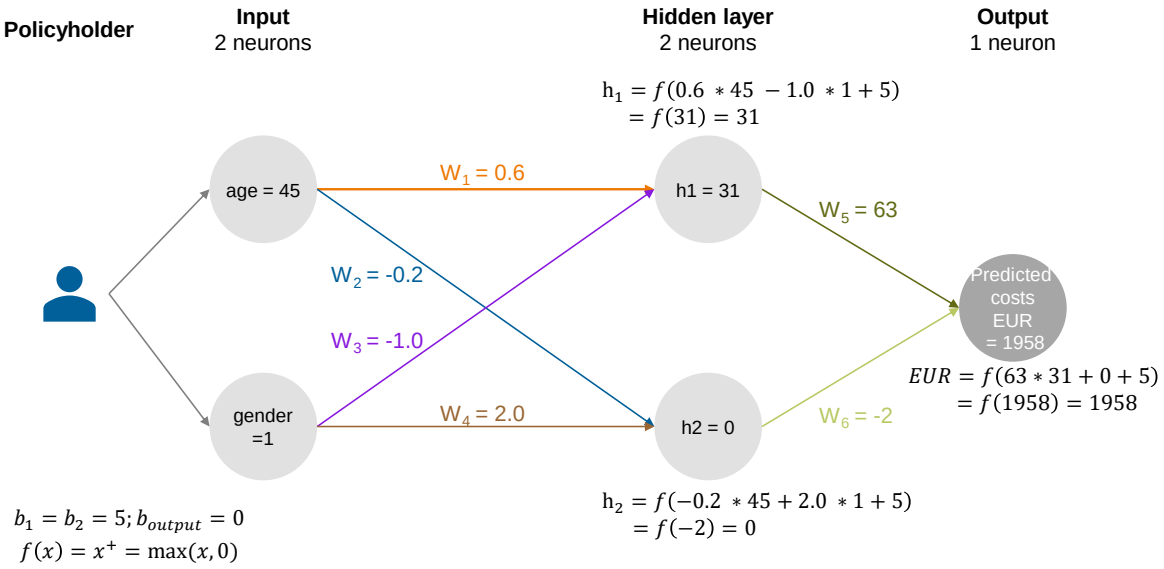
$$h_2 = f(x_1 * w_2 + x_2 * w_4 + b_2)$$

Where (x_1, x_2) are the features (gender, age) of a policyholder; (w_1, w_2, w_3, w_4) the weighting factors between the neurons, b_1, b_2, b_{output} are the bias of the neurons and $f(x)$ the activation function. The activation function is used to translate the input into an output that can also learn *nonlinear* relationships - without such an activation function, a neural network would not be able to do any more than OLS. In this example we use a bias of 5 for each neuron, although of course is not necessarily true in reality; $b_1 = b_2 = 5; b_{output} = 0$. Furthermore, we use the *rectifier* as activation function for the example (and also in the study) $f(x) = x^+ = \max(0, x)$.

This process is reiterated for the translation from the hidden layer to the output layer. For more complex neural networks, multiple hidden layers with a large number of neurons can be used. The principle will stay the same, though.

Suppose we want to predict costs of healthcare for a 45-year-old male policyholder. According to the model, these are 1958 euro; see Figure 17.

Figure 17: Example of predicting healthcare costs with a neural network for an individual policyholder (45-year-old man).



When training the model, the weighting factors (w_1, w_2, w_3, w_4) are established in such a way that for this training set of policyholders, of which the features (x_1, x_2) and the costs of healthcare O are known, the R^2 is maximized. To achieve this, the policyholders from the training set are given to the model in batches several times, to further optimize the weighting factors.

Appendix G: Artificial Neural Network

For this study, the *keras* implementation of the Artificial Neural Network algorithm in R, version 2.2.5.0 was chosen. The following hyperparameters have been applied for the risk adjustment model (hyperparameters not shown are in accordance with the default settings of the relevant R implementation).

Hyperparameter	Setting	Explanation
Layers 1, 2 and 3	256, 128 and 64 neurons (on OT data) 512 and 256 neurons (on OT and source data)	In the model on OT data, the first *hidden layer* contains 256 neurons, the second 128 and the third 64 (on OT data). In the model on OT and source data, the first hidden layer contains 512 neurons and the second 256.
Dropout	0 (on OT data) 0.35 (on OT and source data)	The model deactivates 0% and 35% of neurons to prevent overfitting.
Activation	'relu' [40]	The neurons (excluding the output neuron) use a non-linear activation function to convert input into output. The translated negative values to 0 – positive values are forwarded to the next layer without processing.
Optimizer	'Adam'	The *adam* algorithm adjusts the weighting factors and biases after every batch.
Batch_size	1.024 (on OT data) 2.048 (on OT and source data)	Each batch comprises the data of 1,024 or 2,048 policyholders. After each batch the weighting factors are adjusted.
Epochs	20 (on OT data) 45 (on OT and source data)	The complete dataset is used 20 or 45 times to train the algorithm.
Shuffle	True	After each *epoch*, the data is randomly distributed over new batches.

[40] Rectified linear unit

Appendix H: Comparison metrics M1-M5 with OLS on the same datasets

In this appendix, we compare the metrics of all calculated models with the metrics of OLS on the exact same data set. This way, the effect of applying a different prediction model can be properly assessed, without the outcomes being blurred by the effect of enriching the data that models have at their disposal to make the prediction. All metrics in this appendix have be computed on the test set.

With the exception of M1 (decision tree) and M2 (Piecewise Regression), all models score better on the individual metrics than the OLS model on the same data. On a subgroup and insurer level the picture shows more variation. M4 (gradient boosting machine) and M5 (artificial neural network) outperform OLS on the same data on almost all metrics. What stands out in particular is the substantial improvement of R^2 for models M4 and M5 compared to OLS. The additional data improves OLS by only 1.0 percentage point (M0 vs M0d1 and M0d2), while the same data improves M4 and M5 by 2.2 percentage point and 1.9 percentage point, respectively.

Table 15: Metrics of relevant models in test set on OT data. The values marked green have improved relative to M0; the values marked orange have deteriorated

Level	Measure		M0 OLS	M1 Decision Tree	M3a Random Forest	M4a Gradient Boosting Machine	Artificial Neural Network
Individual	R^2 x 100%		35.1%	35.0%	36.3%	36.3%	36.3%
	CPM x 100%		33.6%	33.6%	34.1%	34.0%	34.1%
	MAPE		1,984	1,983	1,969	1,971	1,968
	Standard deviation outcomes		6,909	6,915	6,845	6,843	6,843
	# with negative standard costs		4,412	-	-	19	-
Subgroups	MAPE on all subgroups		1,132	1,124	1,080	1,141	1,099
	Res. 15% lowest costs in t-3		112	171	159	159	150
	Res. 15% highest costs in t-3		-129	-147	-191	-124	-158
Insurer	R^2 x 100%		98.9%	98.3%	98.8%	98.3%	98.9%
	MAPE		29	38	31	32	30
	Bandwidth of results	All	310	347	315	434	323
		Excl. 2[b]	180	191	167	165	148
		Small	279	319	265	409	259
		Medium	157	192	172	168	178
		Great	63	99	79	60	67
		Not-concern	186	209	167	183	148
		Concern	310	347	315	434	323
	MARS[c]		-	14	6	6	6

[a] Per model the subgroups have been defined on the basis of the baseline model (1.85 million subgroups)
[b] This line shows the bandwidth of the results at the insurer level, excluding the two risk bearers that always determine the actual bandwidth
[c] The MARS for all models was compared with M0 – OLS

Table 16: Metrics of models in test set on OT data including age in years The values marked green have improved relative to M0; the values marked orange have deteriorated

Level	Measure		M0b OLS	M2 Piecewise Regression	M3 Random Forest
Individual	R^2 x 100%		35.1%	35.3%	36.3%
	CPM x 100%		33.6%	32.9%	34.2%
	MAPE		1,984	2,004	1,965
	Standard deviation outcomes		6,909	6,898	6,843
	# with negative standard costs		4,411	62,137	-
Subgroups	MAPE on all subgroups		1,131	1,176	1,091
	Res. 15% lowest costs in t-3		113	140	144
	Res. 15% highest costs in t-3		-132	-105	-131
Insurer	R^2 x 100%		98.9%	98.6%	98.8%
	MAPE		29	31	30
	Bandwidth of results	All	311	394	302
		Excl. 2[b]	180	176	138
		Small	280	335	258
		Medium	158	179	160
		Great	63	64	77
		Not-concern	186	176	138
		Concern	311	394	302
	MARS[c]		-	3	4

[a] Per model the subgroups have been defined on the basis of the baseline model (1.85 million subgroups)
[b] This line shows the bandwidth of the results at the insurer level, excluding the two risk bearers that always determine the actual bandwidth
[c] The MARS for all models was compared with M0 – OLS

Table 17: Metrics of relevant models in test set on OT data including source data. The values marked green have improved relative to M0; the values marked orange have deteriorated

Level	Measure		M0d1 OLS	M4 Gradient Boosting Machine	M0d2 OLS	M5 Artificial Neural Network
Individual	R^2 x 100%		36.1%	38.5%	36.1%	38.2%
	CPM x 100%		34.1%	35.7%	33.8%	35.7%
	MAPE		1,969	1,920	1,977	1,920
	Standard deviation outcomes		6,854	6,726	6,855	6,741
	# with negative standard costs		4,390	58	3,176	544
Subgroups	MAPE on all subgroups[a]		1,114	1,047	1,108	1,058
	Res. 15% lowest costs in t-3		108	107	206	111
	Res. 15% highest costs in t-3		-149	-118	-457	-90
Insurer	R^2 x 100%		98.9%	99.0%	97.8%	99.0%
	MAPE		28	26	44	21
	Bandwidth of results	All	294	302	353	235
		Excl. 2[b]	185	138	228	96
		Small	274	258	333	230
		Medium	160	160	231	138
		Great	84	77	135	57
		Not-concern	185	138	250	146
		Concern	294	302	353	235
	MARS[c]		-	4	-	23

[a] Per model the subgroups have been defined on the basis of the baseline model (1.85 million subgroups)

[b] This line shows the bandwidth of the results at the insurer level, excluding the two risk bearers that always determine the actual bandwidth

[c] The MARS for all models was compared with M0 – OLS

Appendix I: Metrics on training set via 10-fold cross validation

Table 18 shows the metrics calculated through 10-fold cross validation on the training set. These outcomes are subordinate to the outcomes on the test set but provide additional insight into the stability of the outcomes.

The 2020 baseline set was trained and tested on the complete dataset, which is why it is not representative of the comparison with models M1-M5. The metrics in column M0 - OLS have been calculated using 10-fold cross validation and are therefore comparable to the metrics of M1-M5.

A number of conclusions can be drawn from the calculated criteria of the five developed models and the current model. The most striking outcomes of this training set are:

- Model M1 - Decision Tree scores slightly lower than the current model on nearly all metrics. The metrics are negatively impacted on an individual, subgroup and insurer level.
- Model M2 – Piecewise linear regression achieves an improvement on R2 relative to the current model. On the other hand, however, there are substantially more policyholders with a negative standard amount. This high number is caused by the model assuming a linear relationship between age in years and healthcare costs. However, for the age group 0-8 yrs, policyholders born in the adjustment year are significantly more expensive, so this relationship does not exist. The modeled healthcare costs for part of the 8-year-olds are therefore negative. At subgroup and insurer level, a small deterioration can be observed on the majority of metrics.
- Model M3 – Random Forest shows a superior results compared with the current model on all individual metrics. The R^2 in particular show a substantial improvement, from 34.0% to 35.3%. At the subgroup and insurer levels, different metrics show both an improvement and a deterioration. The addition to the dataset of age in years slightly improves most metrics (M3 over M3a). In particular, it significantly improves the financial result on the subgroup of the 15% most-expensive policyholders in t-3. This is offset, however, by a deterioration in the bandwidth of insurers - also compared to M0.
- Model M4 - Gradient Boosting Machine, achieves the most favorable result on the different standards, with a strong improvement compared to M0, especially at the level of individuals. Remarkably, this is accompanied by a strong undercompensation on the subgroup 15% highest costs in t-3. It may be that the model shifts attention to more recent information, causing metrics to improve over the next year, while metrics looking further back deteriorate. There is a clear positive effect of adding additional source data (M4 vs. M4a) on almost all metrics, with a strong improvement of R^2 from 35.3% to 37.3%, for example.
- Model M5 – Artificial Neural Network, also achieves mainly favorable results on all metrics. Positive developments are visible at the individual, subgroup and insurer level. What is particularly striking, is that the bandwidth of the financial result at the insurer level is the best of all the models examined on almost all metrics. There is a clear positive effect of adding additional source data (M5 vs. M5a) with a strong improvement on almost all metrics. The predictive power goes up from 35.4% to 37.0%, but the result on the subgroups shows an equally strong improvement.

Table 18: Metrics of models in training set through 10-fold cross validation. The values marked green have improved relative to M0; the values marked orange have deteriorated

Level	Metric		2020 Baseline model[a]	M0 OLS	M1 Decision Tree	M2 Piecewise Regression	M3 Random Forest	M4 Gradient Boosting Machine	M5 Artificial Neural Network	M3a[b] Random Forest	M4a[b] Gradient Boosting Machine	M5a[b] Artificial Neural Network
Individual	R^2 x 100%		34.4%	34.0%	34.0%	34.2%	35.3%	37.3%	37.0%	35.3%	35.3%	35.4%
	CPM x 100%		33.5%	33.5%	33.5%	33.0%	33.9%	35.9%	35.8%	33.8%	34.0%	34.4%
	MAPE		1,980	1,979	1,979	1,994	1,967	1,907	1,912	1,969	1,966	1,953
	Standard deviation outcomes		6,906	6,907	6,909	6.897	6,840	6,735	6,748	6,842	6,842	6,834
	# with negative standard costs[c]		15,054	10,773	-	113,548	-	113	836	-	78	-
Subgroups	MAPE on all subgroups[d]		1,011	1,042	1,038	1,079	944	922	947	933	1,026	960
	Res. 15% lowest costs in t-3		107	107	167	138	142	93	97	155	153	124
	Res. 15% highest costs in t-3		-124	-122	-141	-113	-84	-183	-83	-153	-123	-145
Insurer	R^2 x 100%		99.1%	99.1%	98.5%	98.7%	99.1%	99.1%	99.1%	99.0%	98.6%	99.1%
	MAPE		26	26	34	30	25	33	21	26	27	28
	Bandwidth of results	All	302	297	340	388	320	296	235	292	417	286
		Excl. 2nd	115	103	182	97	82	102	96	89	104	99
		Small	284	286	336	348	289	274	230	268	415	258
		Medium	154	153	185	174	150	163	138	158	159	158
		Great	64	65	91	65	66	70	73	77	58	63
		Not-concern	167	160	191	151	135	150	146	145	161	145
		Concern	302	297	340	388	320	296	235	292	417	286
	MARS[d]		3	-	13	6	13	23	17	9	6	9

[a] As reported in WOR 973 and reproduced for this study; the small difference on the outcome 15% lowest costs t-3 may result from rounding up or down or very limited differences in data

[b] Per model the subgroups were defined based on the subgroups from the baseline model (1.85 million subgroups)

[c] This line shows the bandwidth of the outcomes at the insurer level, excluding the two risk bearers that always determine the actual bandwidth

[d] The MARS has been compared with M0 – OLS for all models, including the 2020 baseline model

Table 19 describes the R^2 per model and per fold. This ranges from 32.5% (OLS on fold 4) to 38.4% (GBM and Artificial Neural Network on fold 10). The Gradient Boosting Machine achieves the best prediction on 8 out of 10 folds. The Artificial Neural Network algorithm gives the prediction on the other two folds, but on these folds too the Gradient Boosting Machine only scores 0.2 percent point lower. It demonstrates that models M4 and M5 show a stable better performance than the other models. This is not absolute proof of the 'superiority' of these models, though.

Table 19: R^2 of prediction per model and per fold; the best prediction on a fold is given in bold green; the worst prediction is given in bold red

	R^2 x 100% of predicted vs actual costs of healthcare; per model and per fold					
Fold	**M0 – OLS**	**M1 - DT**	**M2 - PLR**	**M3 - RF**	**M4 - GBM**	**M5 - ANN**
1	33.6%	**33.3%**	33.6%	34.9%	**36.7%**	36.3%
2	**33.4%**	33.6%	33.9%	34.8%	**36.9%**	36.5%
3	**34.4%**	34.7%	34.5%	35.9%	**37.8%**	37.3%
4	**32.5%**	32.9%	32.9%	34.0%	**35.9%**	35.8%
5	34.2%	34.2%	34.4%	35.2%	**37.2%**	37.1%
6	34.4%	34.5%	**34.4%**	35.7%	**37.6%**	37.1%
7	33.8%	**33.0%**	33.1%	34.2%	36.3%	**36.5%**
8	**35.0%**	35.0%	35.4%	36.2%	**38.1%**	37.9%
9	**34.4%**	34.9%	35.2%	35.8%	37.9%	**38.1%**
10	35.0%	**34.9%**	35.4%	36.3%	**38.4%**	38.4%

Appendix J: Permutation feature importance M3-M5 on OT data

Table 20 gives the 10 main risk categories per model on the OT data, ranked from most to least important. These have been determined by means of permutation feature importance. That is: one risk category at a time was always mixed in dataset to determine the effect this has on predictive power (R^2).

Table 20: Ten main risk categories per model, rendered as *x*

Risk category	**M0** OLS	**M3a** Random Forest	**M4a** Gradient Boosting Machine	**M5a** Artificial Neural Network
PCG	x	x	x	x
MHCN	x	x	x	x
MHC	x	x	x	x
pDCG	x	x	x	x
Age and gender	x	x	x	x
sDCG	x	x	x	x
PPA	x	x	x	x
MACG	x	x	x	x
PDG	x	x		
NOI	x	x	x	x
SES			x	x

Appendix K: Piecewise Regression vs OLS – results per segment

Segment	R^2 M0 (OLS)	R^2 M2 (Piecewise Regression)	Weight (policyholders in test set)
0 – 8 yrs	16.4%	16.0%	490,607
9 – 16 yrs	38.8%	40.4%	467,937
17 – 26 yrs	46.9%	49.6%	607,153
27 – 33 yrs	34.1%	34.1%	428,693
34 – 41 yrs	38.9%	39.4%	474,640
42 – 53 yrs	38.9%	39.0%	874,041
54+ yrs	36.8%	37.0%	1,708,976
TOTAL	35.1%	**35.3%**	**5,052,047**

Please note: the overall result of the models is presented for reference – this is not a weighted average of the individual segments

Appendix L: Metrics OLS – Piecewise Regression and OLS on age segments

The table below provides the metrics at the individual level within the test set of the regular OLS model (M0), Piecewise Regression (M2) and the adjusted model that determine standard amounts according to regular OLS within the segments that follow from M2.

The model with OLS on segments outperforms the Piecewise Regression model on all individual metrics.

Level	Metric	M0 OLS	M2 Piecewise Regression	M2 OLS on PLR segments
Individual	R^2 x 100%	35.1%	35.3%	35.4%
	CPM x 100%	33.6%	32.9%	33.8%
	MAPE	1,982	2,004	1,979
	Standard deviation outcomes	6,909	6,898	6,888
	# with negative standard costs	4,411	62,137	6,713

Appendix M: Standard amounts morbidity criteria per age segment

Important: Both for the baseline model and the Piecewise Regression model, the results relate to the training set. The standard amounts per age segment are based on the segments as determined by the Piecewise Regression algorithm and calculated with the regular OLS model per age segment.

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No pDCG	-212	-79	-43	-40	-57	-69	-134	-478
1	603	711**	456*	640	759	725	625	366
2	1,259	1,069*	784*	1,174	1,086	861	1,050	1,117
3	1,336	1,125	754	1,030	938	882	992	1,287
4	1,689	2,860	2,135	1,678	1,757	1,545	1,473	1,443
5	2,659	2,336*	1,569*	2,218	2,609	2,529	2,335	2,449
6	2,132	3,641	1,900	1,927	2,176	1,829	1,888	1,802
7	4.790	4,996	2,037	1,815	2,760*	4,267	4,935	4,993
8	5,790	6,617	3,247	1,633	3,071*	3,898	5,950	6,573
9	5,863	10,061*	9,485*	2,964*	437*	642*	4,305	5,247
10	7,427	19,045**	2,640**	5,675*	9,499*	7,307*	5,734*	7,687
11	11,862	N/A	N/A	9,509*	5,950*	11,335*	11,344	12,314
12	13.869	15,917*	2,468*	8,959*	13,569*	10,954*	13,657*	16,192
13	6,987	75,091*	27,564*	8,357*	3,798*	3,133*	3,808*	5,414
14	67.149*	22,815*	67,992*	61,107*	89,398**	43,433**	93,876*	58,056*
15	48,298	31,056***	N/A	39,075**	41,803**	39,563*	50,166*	51,953

** Marked standard amounts are based on fewer than 1,000 policyholder years*

*** Marked standard amounts are based on fewer than 100 policyholder years*

**** Marked standard amounts are based on fewer than 20 policyholder years*.

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No sDCG	-89	-18	-13	-10	-19	-33	-68	-203
1	932	2,909	952	957	1,097	1,139	1,109	751
2	2.369	3,140*	1,184**	1,077*	3,582*	2,939	2,342	2,204
3	3.892	-242*	5,937*	-1,471*	3,578*	4,185	4,428	4,155
4	7.189	3,958*	3,862*	4,098*	6,802*	10,371*	8,999	6,732
5	13,458	8,137*	7,151*	16,814*	12,225*	15,768*	12,908*	13,654
6	17.150	10,462*	716*	9,546*	15,170**	14,289**	19,656*	19,493
7	67,221*	28,107*	82,936**	52,083**	35,252***	62***	74,519***	33,709**

** Marked standard amounts are based on fewer than 1,000 policyholder years*

*** Marked standard amounts are based on fewer than* 100 policyholder years

**** Marked standard amounts are based on fewer than* 20 policyholder years.

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No PCG	-272	-42	-58	-56	-84	-127	-210	-588
1	25	746*	166	244	366	241	87	-202
2	250	349*	2,487*	1,359*	356*	453	611	1
3	117	693***	465	178	96	72	86	-8
4	388	1,351**	320	415	454	304	328	323
5	565	4,075	1,132	666	607	582	330	406
6	773	6,484**	-321**	891*	801	624	1,000	481
7	1.373	8,133*	1,363*	1,365	1,020	918	1,078	1,470
8	2,022	239***	-1,117***	-1,267**	2,950**	1,748*	1,689	1,814
9	1.745	31,945**	14,536**	976*	3,418*	3,344*	2,422	1,446
10	835	9,250***	3,510*	1,521	831	911	815	629
11	1.488	16,223***	3,981***	3,178*	2,088*	2,598	1,443	1,273
12	399	N/A	349**	463*	502*	710	517	95
13	761	N/A	-456***	5.347**	627*	1,080*	787	506
14	1.579	6,466*	2,958	1,622	1,886	1,731	1,448	1,115
15	1.927	20,736***	-5,201**	2,305*	3,571*	2,121	2,171	1,644
16	3.781	15,336*	22,225*	13,166*	9,733*	8,910*	4,127*	1,511
17	2,496	6,707*	3,414	-2,406*	1,214*	1,573*	1,211*	624*
18	2.797	19,255***	7,974**	1,277*	800	1,837*	3,312	2,949
19	4.009	N/A	5,886***	6,195*	6,017*	5,576	4,116	2,150
20	4.191	5,500**	4,673**	5,794*	5,698	5,389	4,316	3,280
21	524	-2,288**	359*	707*	746*	804	646	266
22	663	-2,849***	1,984*	1,321	1,185	691	679	424
23	684	-838**	-1,475*	1,272*	1,211	1,138	502	483
24	4.823	14,310**	8,567*	5,097	5,340	5,820	4,919	4,285
25	7.329	20,113***	20,540***	1,442**	1,004**	9,458*	8,306*	7,265
26	11,491	25,015***	1,391***	12,706***	24,161**	20,033**	14,897*	10,111
27	8.079	-2,677**	20,823**	8,341**	6,398**	6,630**	7,704*	7,955*
28	405	565	362	374	319	385	393	306
29	1.583	13,592**	3,950**	2,097*	2,397*	2,057	1,658	1,288
30	11.673*	N/A	14,612***	11,484**	11,582**	8,642**	12,245*	12,119*
31	743	-3,290**	1,073*	1,878**	376*	1,139	-123	608
32	1.469	-19,289**	-9,497**	1,699*	2,657*	1,688*	1,816	1,097
33	11.390	51,976*	46,502*	22,112*	9,565*	8,927	9,676	10,441
34	19.845	11,151**	17,471**	26,887**	11,380**	22,314**	25,084*	18,969*
35	120.671*	114,344***	101,367***	93,169***	75,206***	130,923***	121,094**	125,891**
36	242.336**	42,744***	107,595***	305,349***	228,917***	259,941***	202,539**	318,950***
37	386,189**	320,395***	591,782***	427,178***	367,875***	402,387***	348,732***	297,490**

** Marked standard amounts are based on fewer than 1,000 policyholder years*

*** Marked standard amounts are based on fewer than 100 policyholder years*

**** Marked standard amounts are based on fewer than 20 policyholder years.*

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No MACG	48%	-30	-17	-10	-13	-21	-32	-94
1	219	-8,511***	910**	206	191	455	320	103
2	428	100 %	1,531**	203	64	369	248	383
3	1.267	2.173*	833*	1,220*	1,822*	1,895*	2,374	1,032
4	1.276	257	635	1,524	1,307*	1,405	1,042	1,398
5	1.808	3,565*	684*	537	1,379	2,191	2,094	1,833
6	2.169	3,479	6,278*	1,988	1,819	1,759	1,606	2,059
7	4.164	14,226*	4,065**	4,316*	2,364*	3,084*	4,187	4,165
8	6.987	7,809	7,579*	4,242*	4,743*	7,388*	7,248*	5,897
9	17.471*	52,209*	13,573**	4,928**	3,771***	-5,052***	7,564**	9,661*
10	9.099	21,455*	19,499*	8,839*	8,574*	10,480*	12,331	6,692

** Marked standard amounts are based on fewer than 1,000 policyholder years*

*** Marked standard amounts are based on fewer than 100 policyholder years*

**** Marked standard amounts are based on fewer than 20 policyholder years.*

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No PDG	-20	-43	-13	-4	-4	-4	-9	-36
1	580	686	367	381	1,679	809	636	735
2	1.903	851*	246*	1,245*	2,385*	829*	1,755	1,962
3	1.253	2,577*	-291*	2,215	919*	1,149	1,307	1,294
4	8,302*	4,399**	-431*	-92*	820**	5,386**	6,215**	10,637*

** Marked standard amounts are based on fewer than 1,000 policyholder years*
*** Marked standard amounts are based on fewer than 100 policyholder years*
**** Marked standard amounts are based on fewer than 20 policyholder years.*

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No MHCN	-185	-35	-20	-11	-14	-21	-46	-491
1	1.226	1,467*	-616*	2,340*	1,120*	1,298*	1,760	871
2	1.860	11,551*	626*	1,302*	1,900*	2,009*	2,345	1,469
3	3.213	5,514*	2,844*	2,170*	3,256*	3,478*	3,610	2,880
4	5.706	5.769*	967*	2,946*	4,477*	5,075*	6,003	5,459
5	8.541	14,321*	13,063*	5,735*	7,762*	8,784*	8,155	8,254
6	12,196	6,314*	11,193*	9,665*	8,283*	12,171*	13,538	11,977
7	17.533	-2,493**	23,724*	11,756*	20,439*	14,122*	18,494*	17,353
8	29.665	-2,643*	5,173*	21,660*	29,212*	31,823*	33,253*	29,910
9	60.336*	54,412*	42,813*	48,439**	N/A	N/A	N/A	N/A

** Marked standard amounts are based on fewer than 1,000 policyholder years*
*** Marked standard amounts are based on fewer than 100 policyholder years*
**** Marked standard amounts are based on fewer than 20 policyholder years.*

Standard amounts in Euros per policyholder year

	2020 Baseline model	Piecewise Regression with the following age segments: 0-8	9-16	17-26	27-33	34-41	42-53	54+
No MHC	-578	-268	-243	-268	-424	-422	-467	-1,052
1	143	235	302	384	423	321	240	-211
2	2.400	2,972	3,142	3,198	1,885	2,313	2,511	1,912
3	2,218	2,297	2,730	2,757	2,216	2,374	2,529	1,835
4	3,562	3,068	4,624	4,371	3,192	3,482	3,602	3,248
5	5,470	4,859	6,245	6,113	5,055	4,728	5,498	5,239
6	8,685	8,004	10,942	10,239	8,514	7,632	8,314	8,612
7	18.003	16,539*	22,548*	20,664*	20,575*	20,010*	18,173	17,142
8	43,292	44,202*	81,732*	74,790*	63,756*	48,686*	47,950*	34,722

** Marked standard amounts are based on fewer than 1,000 policyholder years*
*** Marked standard amounts are based on fewer than 100 policyholder years*
**** Marked standard amounts are based on fewer than 20 policyholder years.*

Appendix N: Exploratory analysis of the effect of upscaling of costs for policyholders who were not insured for the full year

The current practice within risk adjustment is to always work with full policyholder years, where a policyholder year represents one policyholder with one insurer for an entire calendar year. For various reasons, however, people may not have been insured for a whole calendar year, or they have not been insured with the same insurer the entire year. In these cases, the costs incurred, before they are used to create the model, are linearly extrapolated to a full year. For example, if a person has been registered with an insurer for 4 months, the costs incurred with that insurer will be tripled.

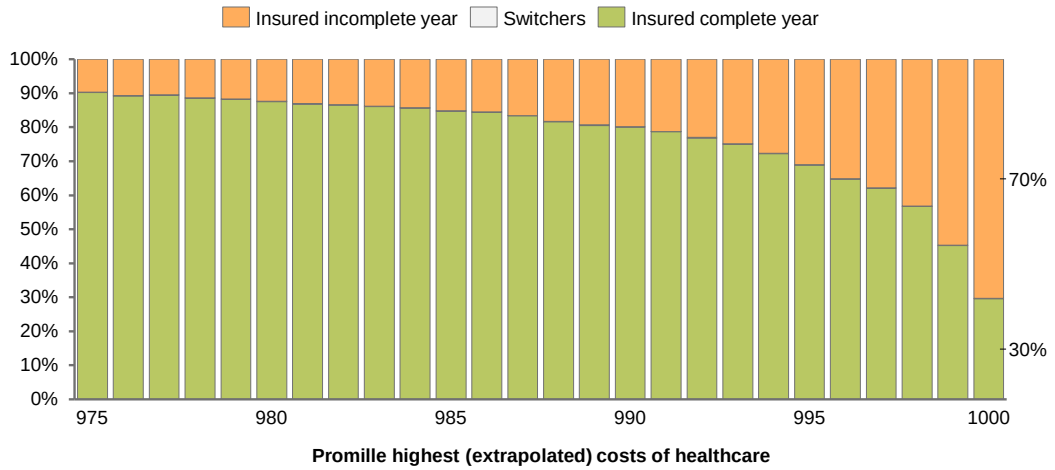
For the sake of recognizability of the results, we will use this current practice of extrapolation in this study. For the interested reader, this appendix contains a description of an exploratory analysis that shows that this methodology may lead to unwanted bias.

In the OT dataset we distinguish 3 types of records:

- Records of full policyholder years:
- Records of persons who switched and for whom the total of all records present does constitute a whole year. This category mainly consists of people who have turned 18, because most other people can only switch at the start of a new calendar year.
- Other records. This last category mainly comprises newborns, deceased persons and migrants. These are records that are extrapolated to a full year.

The total data set (after the ultimate test of the machine learning algorithms) appears to contain strong outliers in terms of costs (up to 11 million Euros). What is very remarkable is that the extrapolated datapoints are severely overrepresented among these outliers. Figure 18 below shows how record types are distributed over the highest costs (switchers are all but invisible because they rarely occur and hardly incur high costs).

Figure 18: Share of insurance term for the 25 per thousand insured with the highest (extrapolated) healthcare costs



This leads us to suspect that the extrapolation to policyholder years can lead to unrealistically high costs. For example, when a person undergoes major surgery early in the year, then spends a week in intensive care, and then dies, this leads to extremely costs when scaled to a full year. This is

very different from, for example, the costs of medication for chronic conditions, for which it *does* seem reasonable to assume that a person who lives twice as long in a year also incurs twice the costs.

The question is whether situations such as the ones described above, of (disproportionately) rising costs in the last few months, are exceptions with a negligible effect or not. To study this further, we looked at records of the type 'other', of people aged 70 years and older. We suppose that the vast majority of these people passed away.[41] If the effect of the disproportionately rising costs of healthcare were negligible, a linear regression of actual costs on the number of days insured would have to go approximately through the origin. In our exploratory analysis this appears not to be the case, but we already found approximately € 4,500 in healthcare costs with an insurance term of 0 days. This indicates that the costs of the deceased can indeed not be extrapolated in the way that is now customary.

[41] This workaround is necessary because death is not a feature included in our dataset.