Genetic Programming for Time Series Forecasting: a Reuma.pt Study

Frederico Borges Costa Coelho Nunes
frederico.coelho.nunes@ist.utl.pt

Instituto Superior Técnico, Lisboa, Portugal

April 2018

Abstract

Genetic Programming (GP) is a set of machine learning techniques and algorithms based on genetic and evolutionary concepts. It consists on the evolution of a population of computer programs, through some set of genetic operators, in order to find the program which best achieves a given task. What sets GP apart from other machine learning algorithms is that the form of the solution is extracted directly from the data, making it a very versatile approach with applications in many areas, from economics to the study of biological processes. In this thesis, GP is used to forecast the response of Rheumatoid Arthritis patients to their treatments, using data from Reuma.pt, the Rheumatic Diseases Portuguese Register [3].
Keywords: Genetic Programming, Time Series, Forecasting, Machine Learning, Rheumatoid Arthritis

1. Introduction

Rheumatic diseases are characterized by inflammation that affects joints, tendons, ligaments, bones and muscles. These diseases, although not directly fatal, have severe effects on quality of life, causing discomfort, pain and impairing the majority of daily activities. They also carry a substantial economic burden – their cost is estimated at more than 200 billion Euro per year in Europe [6].

The focus of this work is Rheumatoid Arthritis (RA), which is an inflammatory auto-immune disease that primarily affects joints. According to [9], RA affects about 24.5 million people as of 2015, and the usual treatments are associated with very high costs [7]. In Portugal, it is estimated that 0.8 to 1.5% of the population suffers from this disease [2].

When traditional methods of treatment fail, RA patients transition to biological treatments. It is desirable to develop methods that can reliably predict patients' responses to these treatments. Such methods would allow doctors to switch a patient to a different treatment earlier, in the case of a negative response prediction. This would save a lot of time and money, and increase the patient's quality-of-life.

This work is therefore motivated by both the financial and quality-of-life aspects, and aims to make a contribution to the study of RA and to the use of Genetic Programming as a tool for forecasting treatment effectiveness.

2. Genetic Programming

Genetic Programming consists in the evolution of a population of computer programs, through the use of genetic operators, with the goal of finding the program which best performs a certain task.

In order to use Genetic Programming, the following information is required:

- The goal: specified through a properly designed fitness function;
- The variables and constants: specified through the terminal set;
- The building blocks of the solution: specified through the function set;
- The genetic operators.

The fitness function is used to evaluate program performance. This function is to be minimized (or maximized) by the programs, and must be very well defined in order to correctly express the goal of the evolutionary process. The most commonly used fitness function is the root-mean-squared error.

The variables and constants related to our problem are specified through the terminal set. A set of allowed functions for the algorithm to use must also be chosen - the function set. The function set must obey the closure assumption, which consists of *type consistency* and *evaluation safety*. This is the assumption that every function returns a value which can be used as an argument for any other

function within the function set. This is a requirement in order to guarantee that every program is valid, regardless of the combinations made by the genetic operators.

An individual in Genetic Programming is a program composed in some way by combinations of these functions. The most commonly used structure for a program is the tree structure. An example can be seen in Figure 1.

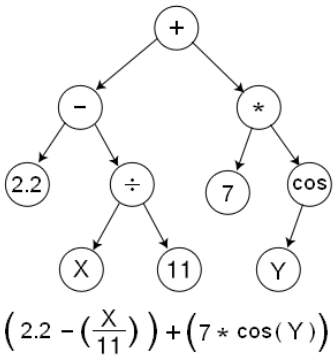

Figure 1: Tree representing a GP individual, and its corresponding program.

In order to generate the initial population, the maximum starting tree depth and the population size must be specified. There are several different generation methods, with the most common being the "Full", "Grow" and "Ramped Half-and-Half".

The user must also specify the number of generations for which to run the evolutionary algorithm. In each generation, new offspring is generated by applying the genetic operators to suitably selected parents. The most common parent selection process is a tournament. This consists of randomly picking a certain number of individuals from the population, and evaluating their fitness (through the fitness function). The fittest individuals are then chosen to be acted upon by the genetic operators. This method is suitable for two reasons. Firstly, because it ensures that even less fit programs have a chance to generate offspring, maintaining genetic diversity. Secondly, because it allows the user to control the selection pressure - the larger the tournament, the less probability there is of a less fit program passing on its genetic material.

On every generation, this tournament process is repeated many times, generating a new population. The new population can be comprised by offspring only, but an elitist approach can also be taken, by keeping one or more of fittest parents, or by including only offspring which are fitter than their parents.

There are many types of genetic operators, the two main kinds being sub-tree crossover and mutation. An example of sub-tree crossover can be seen in Figure 2, and an example of sub-tree mutation can be seen in Figure 3. The probability of each genetic operator is set by the user, and can be set to change as the GP run progresses.

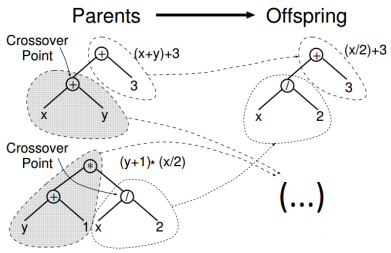

Figure 2: Sub-tree Crossover (taken from "A Field Guide to Genetic Programming" [8]).

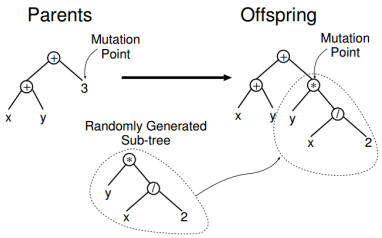

Figure 3: Sub-tree Mutation (taken from "A Field Guide to Genetic Programming" [8]).

Lastly, it is important to specify when the evolution process should stop. It can stop after a certain number of generations, or when a desired fitness value has been reached. If the last option is selected, there is no guarantee that the process will come to an end.

3. Data

The data that was used in this work comes from the Rheumatic Diseases Portuguese Register (RNDR) [3], a database developed by the Portuguese Society of Rheumatology (SPR). This is an extensive dataset containing information from several appointments of patients suffering from RA, who are under biological treatments.

There is a total of 424 patients in the database. Some of the patients have switched between different treatments at some point, resulting in 742 patient-treatment time series.

For each time series, there are 6 time instants, corresponding to the start of the treatment, and 3, 6, 12, 18 and 24 months after the start. There are 61 independent variables measured at each time instant, which are briefly described in Table 1.

4. Methodology

4.1. Treatment Response

The target variable of this forecasting problem is a response code based on the Disease Activity Score

Table 1: Variables' names and descriptions.

Variable	Name	Description
X1	n_dmards_previos_ini	Number of synthetic DMARDs treatments started before that time
X2	n_dmards_previos_fim	Number of synthetic DMARDs treatments finished before that time
X3	eva_doente	Visual analogue scale pain evaluation according to the patient
X4	eva_dor	Visual analogue scale pain evaluation
X5	eva_medico	Visual analogue scale pain evaluation according to the doctor
X6	VS	Sedimentation Rate
X7	PCR	C-reactive protein
X8	cdai	CDAI (Clinical Disease Activity Index) score
X9	SDAI	SDAI (Simple Disease Activity Index) score
X10-X29	haq1 - haq20	HAQ (Health Assessment Questionnaire) score for questions 1-20
X30	HAQ	HAQ score
X31	DAS44_CALC	Calculated DAS44 value
X32	I_ENV_PUNHO	Binary indicator of fist related disease
X33	I_ENV_ANCA	Binary indicator of hip related disease
X34	I_ENV_TIBIO_TAR	Binary indicator of tibiotarsal related disease
X35	I_ENV_COL_CERVICAL	Binary indicator of cervical spine related disease
X36	DAS_28_4V	DAS28 VS 4 variables
X37	DAS28_3V	DAS28 VS 3 variables
X38	DAS28_4V_PCR	DAS28 PCR 4 variables
X39	DAS_28_3V_PCR	DAS28 PCR 3 variables
X40	DELTA_DAS	DAS28 variation between starting time instant and current time instant
X41	Abatacept_i_terap	Drug_i_terap: Drug was being taken in that time instant
X42	Adalimumab_i_terap	
X43	Anacinra_i_terap	
X44	Etanercept_i_terap	
X45	Infliximab_i_terap	
X46	Rituximab_i_terap	
X47	Tocilizumab_i_terap	
X48	Golimumab_i_terap	
X49	Metotrexato_i_terap	
X50	Azatioprina_i_terap	
X51	Ciclosporina_i_terap	
X52	Hidroxicloroquina_i_terap	
X53	Leflunomida_i_terap	
X54	Sulfassalazina_i_terap	
X55	AurotiomalatoSodio_i_terap	
X56	Betametasona_i_terap	
X57	Deflazacorte_i_terap	
X58	Prednisolona_i_terap	
X59	Prednisona_i_terap	
X60	Metilprednisolona_i_terap	
X61	cod_resposta_das	Treatment Response Class in that time instant

28 variable (DAS28) [1]. It was defined by the European League Against Rheumatism (EULAR) and it measures the evolution of the disease in a patient. A response criteria, based on the current DAS28 value and on its improvement from the start of the treatment, can be seen in Table 2.

The patients can then be separated into three classes according to how well they respond to a certain treatment: C0 (no response), C1 (moderate response) and C2 (good response).

A good classification model must identify as many patients belonging to C0 as possible, while maintaining a close to perfect accuracy for patients belonging to classes C1 and C2. This is necessary to guarantee that:

1. No patients who would respond well to a given treatment are misclassified as 'non-respondent' (false negatives) and changed to a different treatment;

2. As many 'non-respondent' patients as possible switch to a different treatment early.

4.2. Experimental Setup

Several forecasting experiments can be carried out, by varying the forecasting horizon (time instant for which the treatment response is predicted) and what previous time instants are used as input data.

In the case of this thesis, two forecasting horizons were considered: 12 and 24 months. In both these experiments, data from the start of the treatment, as well as the data from 6 months after the start of the treatment, was used as input.

For each experiment, the data was first analysed to decide which data instances are suitable. Only instances whose target variable value is not missing were used; additionally, all instances which have more than a certain percentage of missing values, at any time instant relevant to the experiment, were discarded. For both the 12 and 24 month forecasting experiments, a 50% missing value threshold was chosen, which means 229 and 242 instances of data were used, respectively.

Table 2: Treatment Response Criteria.

DAS28 Improvement → / **Present DAS28 ↓**	> 1.2	> 0.6 and ≤ 1.2	≤ 0.6
≤ 3.2	Good response	Moderate response	No response
> 3.2 and ≤ 5.1	Moderate response	Moderate response	No response
> 5.1	Moderate response	No response	No response

An overfitting analysis, consisting of 10 test runs, was performed for each case, to evaluate at which generation overfitting started to occur. A suitable maximum number of generations (N_{Gen}) was then set for the following runs. The overfitting analysis resulted in: a value of $N_{Gen} = 40$, for both the binary and non-binary case of the 12 month forecasting experiment; values of $N_{Gen} = 15$ for the binary case and $N_{Gen} = 20$ for non-binary case of the 24 month forecasting experiment.

For every experiment, two types of forecasting are considered: binary, and non-binary. In the binary case, patients are divided in two classes – responder and non-responder. If the model's output is less than 0.5, then the instance is classified as C0 (non-respondent); otherwise, it is classified as C1. In the non-binary case, all three response classes are considered. If the output is less than 0.5, the instance is classified as C0; if the output is in the interval $[0.5, 1.5[$, the instance is classified as C1; otherwise, it is classified as C2.

30 GP runs are performed for each case, using random train/test splits of 70%/30%. The data was also pre-processed by replacing the missing values for each variable are by the mean value of that variable, for every time instant, and by normalizing the data between 0 and 1. The GP parameters that were used can be seen in Table 3. These parameters were set after an initial round of exploratory runs, with the idea of minimizing the computational time of the experiments without compromising the results. The parameters remained constant throughout all the experiments.

Table 3: Table with the GP parameters.

Number of Generations	**Parameter**
Population Size	800
Initialization Method	Ramped half-and-half
Crossover Probability (%)	95
Mutation Probability (%)	5
Terminal Set	1,2,3,rand(0,1)
Function Set	+,-,x,/,$\sqrt{}$,power,sin,cos,log
Maximum Depth	17

The fitness function that was used during the evolutionary process was the root-mean-squared error. In order to compare the models, the metrics that were used were their accuracy and recall. Finally, regarding the tree representations of the models: variables X1-X61 correspond to the variables at the start of the treatment, and X62-X122 correspond to the same variables 6 months after the start of the treatment. In the tree representation of the models, functions *mysqrt*, *mypower* and *mylog* correspond to the square root, power (a^b) and natural logarithm functions, respectively. These names indicate that the functions were modified, in order to guarantee that the closure assumption was satisfied: for the *mysqrt* function, the output value is set to zero for negative arguments; for the *mylog* function, the output is set to zero if the argument is zero, and for negative arguments, the absolute value of the argument is used instead; the output of the *mydivide* function is equal to the numerator if the denominator is zero; and finally, the output of the *mypower* function is set to zero if it is imaginary, or NaN (not a number).

5. Results

5.1. 12 Month Response Prediction

The results for the 12 month response prediction experiment are now presented. In Table 4, the results for 30 GP runs are presented, for the binary case. All results that are presented come from applying the models to previously unseen test data.

It can be seen that the results have mixed quality. The only model which achieves a perfect recall classifies every instance as C1, rendering it useless. Models 10 and 26 are considered the best models, having only one false negative while correctly identifying 9 non-responders.

The models' accuracies were then evaluated as a function of their output's distance to the nearest class, for each data instance. If, on average, the outputs corresponding to the correctly classified instances are close to the true class value, as opposed to being close to the midpoint between two classes, then the models' output can be likened to their classification "certainty".

More precisely, this was done by considering five intervals of length 0.1: [0,0.1[, [0.1,0.2[, [0.2,0.3[, [0.3,0.4[, [0.4,0.5]. These are identified in the figure by their upper bound. Then, for each model, the data instances are distributed into each of the intervals, according to their output's distance to the nearest class (0 or 1). The model's accuracy is then calculated separately for each of the intervals.

The boxplot in Figure 4 illustrates the average results for the 30 models. The natural output of the models is already very close to the true class, so

Table 4: Results for the binary case, with a 12 month forecasting horizon.

Run	TN	FN	TP	FP	Accuracy (%)	Recall (%)
1	7	4	50	7	83.8	92.6
2	2	7	38	21	58.8	84.4
3	6	5	42	15	70.6	89.4
4	7	2	48	11	80.9	96.0
5	8	4	43	13	75.0	91.5
6	11	10	36	11	69.1	78.3
7	9	4	47	8	82.4	92.2
8	5	1	48	14	77.9	98.0
9	2	2	48	16	73.5	96.0
10	9	1	46	12	80.9	97.9
11	17	9	36	6	77.9	80.0
12	8	7	46	7	79.4	86.8
13	5	2	44	17	72.1	95.7
14	11	16	31	10	61.8	66.0
15	9	7	44	8	77.9	86.3
16	10	9	37	12	69.1	80.4
17	8	3	45	12	77.9	93.8
18	10	3	46	9	82.4	93.9
19	10	2	45	11	80.9	95.7
20	6	5	44	13	73.5	89.8
21	6	2	43	17	72.1	95.6
22	9	2	45	12	79.4	95.7
23	8	11	35	14	63.2	76.1
24	0	0	42	26	61.8	100.0
25	0	2	48	18	70.6	96.0
26	9	1	37	21	67.6	97.4
27	7	2	42	17	72.1	95.5
28	7	16	29	16	52.9	64.4
29	2	3	48	15	73.5	94.1
30	3	5	47	13	73.5	90.4

there were very few instances for which the output was more than 0.1 units away from the closest class. However, some likeness can be drawn between the output of the models and their certainty, for this experiment.

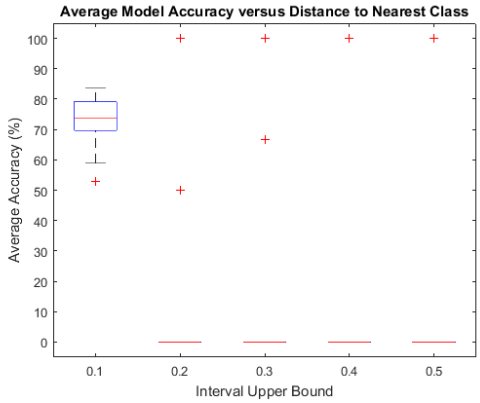

Figure 4: Evaluation of classifier "certainty", for 12 months treatment response forecasting.

A tree representations of model 10 can be seen in Figure 5.

The procedures were repeated for the non-binary case. For 30 runs, the average accuracy was ≈ 43.6% and the average recall was ≈73.6%.

The results are very unsatisfactory, with only a few models achieving over 50% accuracy. However, upon closer analysis, it could be seen that these low values were mostly due to the models' inability to effectively distinguish between C1 and C2. Models 9 and 14 are considered to be the best ones out of the 30. Model 9 has the highest recall (96.4%), while allowing for the correct classification of 5 non-responders. However, it has a low accuracy (47.1%). Model 14 , on the other hand, has a slightly lower recall (93.3%) but a higher accuracy (51.5%), and correctly classifies 7 non-responders.

The model "certainty" analysis was also performed for this case, and it can be seen in Figure 6. It can be seen that the results are much worse than in the binary case, and that the average classification accuracy is below 50%.

The tree representation of model 14 can be seen in Figure 7.

5.2. 24 Month Response Prediction

In this section the results for a more ambitious 24 month forecasting experiment are presented. The same time instants as in the 12 month forecasting were used as input data. In Table 5, the results for 30 GP runs are presented, for the binary case.

Table 5: Results for the binary case, with a 24 month forecasting horizon.

Run	TN	FN	TP	FP	Accuracy (%)	Recall (%)
1	0	0	48	24	66.7	100.0
2	0	1	50	21	69.4	98.0
3	1	2	50	19	70.8	96.2
4	0	3	57	12	79.2	95.0
5	0	2	51	19	70.8	96.2
6	2	0	51	19	73.6	100.0
7	1	2	55	14	77.8	96.5
8	2	4	50	16	72.2	92.6
9	0	1	46	25	63.9	97.9
10	0	0	52	20	72.2	100.0
11	0	0	51	21	70.8	100.0
12	1	1	50	20	70.8	98.0
13	0	0	46	26	63.9	100.0
14	0	1	49	22	68.1	98.0
15	0	1	52	19	72.2	98.1
16	1	1	51	19	72.2	98.1
17	1	2	52	17	73.6	96.3
18	0	3	50	19	69.4	94.3
19	1	3	50	18	70.8	94.3
20	0	0	47	25	65.3	100.0
21	1	3	54	14	76.4	94.7
22	0	5	48	19	66.7	90.6
23	1	3	55	13	77.8	94.8
24	0	0	46	26	63.9	100.0
25	0	0	47	25	65.3	100.0
26	0	0	53	19	73.6	100.0
27	1	1	48	22	68.1	98.0
28	0	0	53	19	73.6	100.0
29	1	0	54	17	76.4	100.0
30	0	2	51	19	70.8	96.2

It can be seen that many models achieve a recall of 100% simply by classifying every instance as belonging to C1. Only models 6 and 29 are able to correctly identify non-responders – 2 and 1, respectively – while achieving perfect recall.

The model "certainty" analysis was again per-

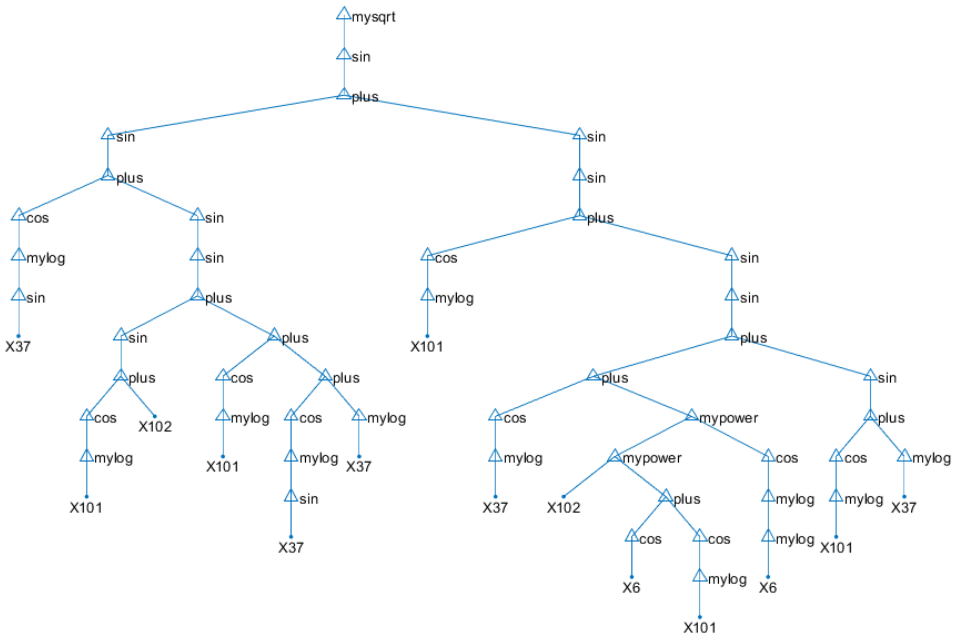

Figure 5: Tree representation of model 10 for the 12 month binary forecasting task.

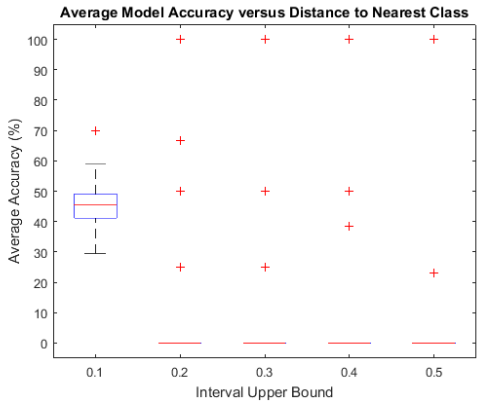

Figure 6: Evaluation of classifier "certainty", for 12 months non-binary treatment response forecasting.

formed and it can be seen in Figure 8. The results are similar than those obtained in the 12 month binary case, but with lower accuracy. This makes sense due to the higher difficulty of performing a 24 month response forecasting, using the same input data.

5.2.1. Non-Binary Case

The procedures were repeated for the non-binary case. In 30 runs, the average accuracy was ≈47.4% and the average recall was ≈81.4%. Only model 19 achieved a recall of 100%. Its accuracy (52.8%) is considered very low, but once again this seems to relate to a difficulty in differentiating between classes C1 and C2. However, it correctly identifies 7 non-responders, which is considered a good result given the distant forecasting horizon of 24 months.

The results for the model "certainty" analysis can be seen in Figure 9. This figure indicates that there is less "certainty" in the results for some of the models, compared to the previous experiments. This is likely due to the harder difficulty of performing a 24 month forecast, when compared to a 12 month forecast.

5.3. Classification Threshold Optimization

Since the Genetic Programming algorithm has a numeric (not a categorical) output, the results have to be rounded up or down according to some threshold, in order to match one of the classes, as mentioned in Section 4.2.

However, given the goal of minimizing the amount of false negatives, an attempt was made to find a more suitable threshold value separating C0 from C1, for each of the models. This is done with the idea of decreasing the amount of false negatives, possibly at the expense of a higher false positive rate.

For each of the models, classification was again performed on the training data, using threshold values between 0.01 and 0.5 (with a 0.01 step). The highest threshold that achieved the best recall on the training data was then saved, for each of the models; in case of a tie, the threshold resulting in the highest accuracy was saved. Afterwards, the models were again used to predict the test instances, using the new threshold values.

This process was performed for both the binary and non-binary cases in the 12 and 24 month forecasting experiments. The models for which a better

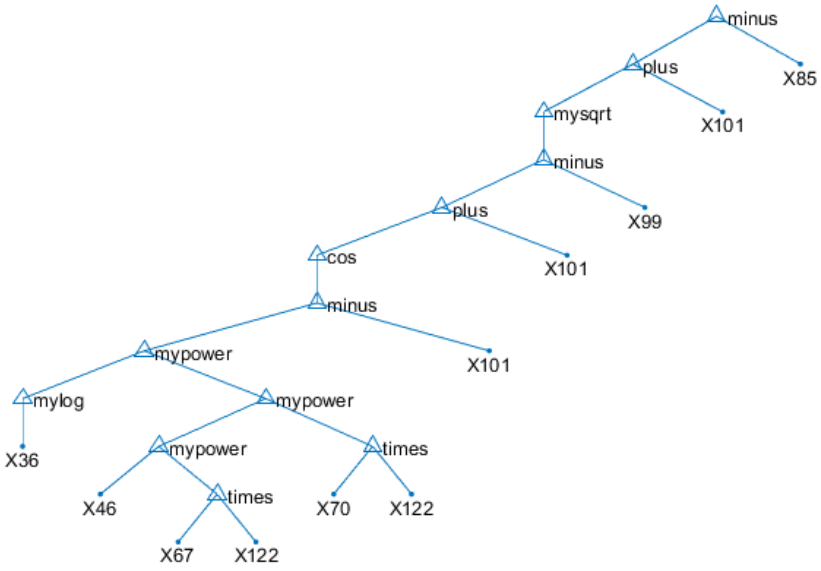

Figure 7: Tree representation of Model 14 for the 12 month non-binary forecasting task.

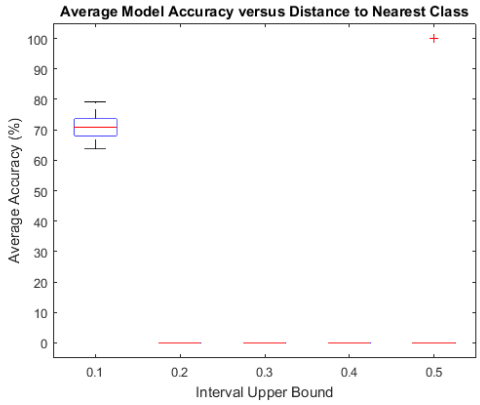

Figure 8: Evaluation of classifier "certainty", for 24 months treatment response forecasting.

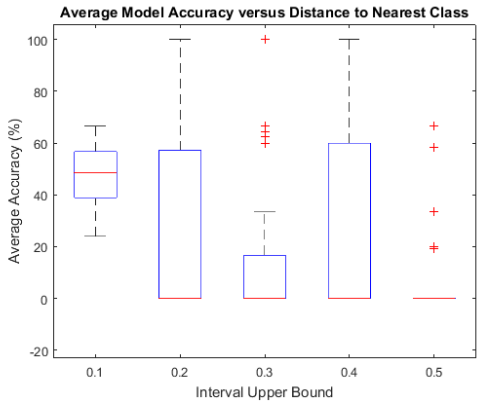

Figure 9: Evaluation of classifier "certainty", for 24 months non-binary treatment response forecasting.

threshold was found (according to the aforementioned metrics) are presented in Tables 6 and 7, where the values for the default 0.5 threshold are in parenthesis.

The results obtained vary significantly. Some models achieve a higher recall with a different threshold, but lose the ability to identify any non-responders. However, it seems that in general performing a threshold analysis can improve the recall and accuracy of certain models, and the quality of forecasting.

5.4. Variable Frequency Analysis

In this section, a study of variable frequency is presented, in order to assess the relative importance of the different variables in the forecasting task. Tables 8 to 11 contain, for each forecasting task, the five most frequently appearing variables, as well as the corresponding frequency of appearance. At the end of each variable name, ${}_0m$ or ${}_6m$ indicates the corresponding time instant (0 or 6 months). The models were simplified by removing introns: functional branches that are irrelevant for the model, or that simply pass on the values that are passed to them. This was done because variables appearing in introns are not meaningful, and such appearances should not contribute to the assessment of that variable's importance.

It can be seen that for the 12 month forecasting case, the variable **cod_resposta_das_6m**, corresponding to the patients' response class 6 months after the start of their treatment, is the variable that appears most frequently in the models, which seems to indicate its importance especially in the non-binary case. In the binary case, there are sev-

Table 6: Threshold analysis for the 12 and 24 month binary forecasting.

Experiment	Model	TN	FN	TP	FP	Accuracy (%)	Recall (%)	Threshold
12 Months	3	4 (6)	2 (5)	45 (42)	17 (15)	72.1 (70.6)	95.7 (89.3)	0.04
	21	3 (6)	1 (2)	44 (43)	20 (17)	69.1 (72.0)	97.7 (95.6)	0.03
24 Months	21	0 (1)	2 (3)	55 (54)	15 (14)	76.4 (76.4)	96.5 (94.7)	0.07

Table 7: Threshold analysis for the 12 and 24 month non-binary forecasting.

Experiment	Model	Accuracy (%)	Recall (%)	Threshold
12 Months	3	39.7 (48.5)	100 (96.4)	0.02
	14	45.6 (51.5)	96.7 (93.3)	0.10
24 Months	13	44.4 (50)	90.0 (83.3)	0.19
	14	52.7 (51.4)	97.1 (94.1)	0.43
	21	59.7 (62.5)	79.5 (77.3)	0.35
	26	40.3 (51.4)	100 (86.2)	0.19

Table 8: Study of variable occurrence frequency for the binary 12 month forecasting task.

Variable	Occurrences	Frequency (%)
cod_resposta_das_6m	94	12.129
DAS_28_4V_0m	82	10.5806
VS_6m	40	5.1613
I_ENV_PUNHO_0m	35	4.5161
DELTA_DAS_6m	28	3.6129

Table 9: Study of variable occurrence frequency for the non-binary 12 month forecasting task.

Variable	Occurrences	Frequency (%)
cod_resposta_das_6m	42	35
I_ENV_PUNHO_0m	8	6.6667
I_ENV_COL_CERVICAL_0m	7	5.8333
Prednisona_i_terap_0m	7	5.8333
Rituximab_i_terap_6m	7	5.8333

Table 10: Study of variable occurrence frequency for the binary 24 month forecasting task.

Variable	Occurrences	Frequency (%)
Azatioprina_i_terap_6m	28	11.2903
Azatioprina_i_terap_0m	18	7.2581
Golimumab_i_terap_6m	18	7.2581
I_ENV_COL_CERVICAL_6m	14	5.6452
I_ENV_ANCA_0m	12	4.8387

Table 11: Study of variable occurrence frequency for the non-binary 24 month forecasting task.

Variable	Occurrences	Frequency (%)
cod_resposta_das_6m	46	28.0488
DELTA_DAS_6m	10	6.0976
Anacinra_i_terap_6m	9	5.4878
DAS_28_4V_0m	6	3.6585
Azatioprina_i_terap_0m	6	3.6585

eral other variables which also appear frequently. Given the relatively good results obtained for this case, it might be worth investigating further how these variables relate to the effectiveness of RA treatments.

For the 24 month forecasting case, the variable **cod_resposta_das_6m** is again the dominant variable for the non-binary case. However, for the binary case, the two most important variables were **Azatioprina_i_terap_6m** and **Azatioprina_i_terap_0m**, that indicate whether or not an Azatioprina treatment was active at 0 and 6 months. This might indicate a relationship between this treatment and the effectiveness of biological RA treatments.

5.5. Random Forest Classification

The same forecasting tasks – binary and non-binary – are performed for both the 12-month and 24-month forecasting horizon using Random Forests. A Random Forest (Ho [5]) is a commonly used ensemble learning method that operates by constructing several decision trees and outputting, for a given data instance, a class that is the mode of the output of the individual trees. The parameters used for the decision tree were the default WEKA parameters, with the exception of the following:

- Number of trees in the ensemble: 700
- Number of features used per tree: 12

The number of trees used in the ensemble was set to a higher value than the default (100), because preliminary experiments indicated that this value was too low.

Regarding the number of features used per tree: this indicates that each tree in the ensemble could only use 12 randomly selected features. This limitation is imposed to increase diversity in the structures of the trees in the ensemble, preventing a single feature or set of features from dominating the main nodes in every tree. In both experiments there are two input time instances, each with 61 features, resulting in p=122 total features. A commonly used value is $\sqrt{p}$ (Friedman et al. [4]), which in this case is approximately 12.

The results for the binary cases are presented in Tables 12 and 13.

It can be seen that the Random Forest results are, in general, better than the GP results, both in average accuracy and recall. However, these models are not interpretable, as opposed to the GP models. It is concluded that, if the only concern is classification accuracy and recall, then a Random Forest is a suitable method to approach this forecasting problem.

Table 12: Results of the Random Forest method, for the 12-month horizon binary forecasting.

Run	TN	FN	TP	FP	Accuracy (%)	Recall (%)
1	9	6	48	5	83.8	88.9
2	13	1	44	10	83.8	97.8
3	8	4	43	13	75.0	91.5
4	8	1	49	10	83.8	98.0
5	10	4	43	11	77.9	91.5
6	9	3	43	13	76.5	93.5
7	7	2	49	10	82.4	96.1
8	9	3	46	10	80.9	93.9
9	6	1	49	12	80.9	98.0
10	6	1	46	15	76.5	97.9
11	14	2	43	9	83.8	95.6
12	10	7	46	5	82.4	86.8
13	10	2	44	12	79.4	95.7
14	8	3	44	13	76.5	93.6
15	8	4	47	9	80.9	92.2
16	9	1	45	13	79.4	97.8
17	9	5	43	11	76.5	89.6
18	7	4	45	12	76.5	91.8
19	8	4	43	13	75.0	91.5
20	8	0	49	11	83.8	100.0
21	7	4	41	16	70.6	91.1
22	9	2	45	12	79.4	95.7
23	8	2	44	14	76.5	95.7
24	8	0	42	18	73.5	100.0
25	6	4	46	12	76.5	92.0
26	10	0	38	20	70.6	100.0
27	10	4	40	14	73.5	90.9
28	8	5	40	15	70.6	88.9
29	8	1	50	9	85.3	98.0
30	6	8	44	10	73.5	84.6

Table 13: Results of the Random Forest method, for the 24-month horizon binary forecasting.

Run	TN	FN	TP	FP	Accuracy (%)	Recall (%)
1	1	1	47	23	66.7	97.9
2	2	4	47	19	68.1	92.2
3	1	1	51	19	72.2	98.1
4	3	5	55	9	80.6	91.7
5	2	1	52	17	75.0	98.1
6	1	4	47	20	66.7	92.2
7	2	2	55	13	79.2	96.5
8	3	4	50	15	73.6	92.6
9	2	0	47	23	68.1	100.0
10	4	5	47	16	70.8	90.4
11	2	3	48	19	69.4	94.1
12	1	3	48	20	68.1	94.1
13	1	1	45	25	63.9	97.8
14	2	1	49	20	70.8	98.0
15	1	3	50	18	70.8	94.3
16	1	4	48	19	68.1	92.3
17	1	1	53	17	75.0	98.1
18	1	2	51	18	72.2	96.2
19	1	1	52	18	73.6	98.1
20	3	8	39	22	58.3	83.0
21	2	7	50	13	72.2	87.7
22	3	9	44	16	65.3	83.0
23	0	4	54	14	75.0	93.1
24	1	0	46	25	65.3	100.0
25	0	5	42	25	58.3	89.4
26	1	1	52	18	73.6	98.1
27	1	1	48	22	68.1	98.0
28	1	3	50	18	70.8	94.3
29	0	3	51	18	70.8	94.4
30	1	3	50	18	70.8	94.3

6. Conclusions

The goal of this thesis was to forecast Rheumatoid Arthritis patients' responses to their biological treatments using Genetic Programming. Two forecasting horizons (12 and 24 months) were considered, and for each of these experiments, both binary and non-binary classification cases were tested. Random Forests were also used to a lesser extent in these problems, as a measure of comparison.

Overall, the results were mixed. The 12 month forecasting binary case was the most successful experiment, with several models achieving a high accuracy while maintaining an almost perfect recall. Model 10 of this experiment was the best overall model, achieving an accuracy of 80.9% and a recall of 97.9%.

For the 24 month forecasting binary case, the results were worse. This seems to indicate that there is a lower correlation between the data at 0 and 6 months and the data at 24 months. There were, however, some results worthy of attention. Model 6, in particular, was able to correctly forecast two non-responders, achieving a perfect recall and an accuracy of 73.6%.

The non-binary experiments were largely unsuccessful. Only one model – model 19 of the 24 month forecasting experiment – achieved perfect recall, and although its accuracy was only 52.8%, it was still able to correctly identify 7 non-responders. Most models have accuracies below 50%. This is due to an apparent difficulty in differentiating between C2 and C1 responders. These results indicate that binary experiments with shorter forecasting horizons are more promising when it comes to RA treatment effectiveness forecasting.

Regarding the classifier "certainty" analysis: in most experiments, the GP models' outputs were naturally very close to 0, 1 and 2, the true class values. However, a relationship can be drawn between the output's distance to the nearest class and the likelihood of it being correct. In the 24 month non-binary case, the classifier "certainty" boxplot presents a different result: in this case, this "certainty" is not a good estimator of the real accuracy of the models.

The proximity of most models' outputs to the true class values can be explained by the variable frequency analysis. It can be seen that several of the most frequently occurring variables are either binary or take only the true class values. It is therefore more likely that operations over these variables will result in values close to the true class values. The most frequent variable overall was variable **cod_resposta_das_6m**, which represents the patients' responses 6 months after the start of their treatments. However, there are other variables worthy of attention, mainly related to the presence of

other treatments such as Azatioprina, Golimumab, Leflunomida and AurotiomalatoSodio. The high frequency of these variables might indicate a relationship between these treatments and the effectiveness of the biological treatments, and should be investigated further.

A classification threshold optimization process was performed in order to improve the models. The results seem to indicate that this process is a suitable way to improve the quality of forecasting, particularly in situations where a high recall is a priority. However, in the case of this thesis, no model suffered a significant improvement.

Finally, the Random Forests were used in the same classification tasks, as a benchmark for the GP models. The results were better overall, both in terms of recall and accuracy. In the 12 month binary case, the average results were better for GP, but a non-parametric Kruskal Wallis test showed that this difference had no statistical significance. However, the best Random Forest models were only slightly superior to the best GP models, and due to the amount of trees in the ensemble, these models are uninterpretable. Furthermore, no Random Forest model was able to achieve a perfect recall in a non-binary classification task, as opposed to model 19 of the 24 month non-binary forecasting experiment. These results validate GP as a suitable forecasting tool for this problem, especially when interpretable models are desirable.

7. Future Work

Some lines of research arising from this thesis can hopefully be pursued in future works.

First of all, an improvement can be achieved by increasing the amount of data available, but most importantly, by reducing the number of missing values in the data. There is also the possibility of experimenting with different techniques for replacing the missing data that allow a better representation of the data.

It is also possible to experiment with different forecasting horizons and using data from different time instants as input. If more data was available, more than two time instants could be used as input, which could improve the quality of the forecasting. A better segmentation of the data instances would also be possible (for example, analysing different treatments separately).

Another idea would be to perform forecasting directly on the DAS28 variable, and converting the output to the respective response class based on the obtained DAS28 value and on the DAS28 improvement.

It could also be interesting to evaluate the accuracy and recall of an ensemble of GP models, making it a fairer comparison to a Random Forest.

Acknowledgements

I would first like to thank my thesis supervisors, Prof. Alexandra Carvalho and Prof. Sara Silva, for all the help they have provided. I am thankful to the members of the Systems Engineering in Life Sciences group, for helping me with my presentations. I am grateful to my family, for supporting me in every aspect of life in general. And finally, I would like to thank Turno da Noite and all of my friends, for motivating me and for making life fun.

References

[1] Handout on health: Rheumatoid arthritis. `https://www.das-score.nl/das28/en/`. Accessed: 2018-01-12.

[2] Artrite reumatide - o que . `http://www.spreumatologia.pt/doencas/artrite-reumatoide`. Accessed: 2018-01-03.

[3] Reuma.pt - the rheumatic diseases portuguese register. acta reumatol. port., (36):45-56, 2011. `http://reuma.pt/pt_PT/Default.aspx`. Accessed: 2017-06-10.

[4] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[5] Tin Kam Ho. Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 1, pages 278–282. IEEE, 1995.

[6] Jesper Johansson. 10 Things You Should Know About Rheumatic Disease. *European League Against Rheumatism (EULAR)*, 440:1–3, 2005.

[7] Luis Cunha Miranda, Helena Santos, Júlia Ferreira, Paulo Coelho, Catarina Silva, and Jose Saraiva-Ribeiro. Finding rheumatoid arthritis impact on life (frail study): economic burden. *Acta reumatologica portuguesa*, 37:134–142, 2012.

[8] Riccardo Poli, William B Langdon, Nicholas F McPhee, and John R Koza. *A field guide to genetic programming*. Lulu. com, 2008.

[9] Theo Vos, Christine Allen, Megha Arora, Ryan M Barber, Zulfiqar A Bhutta, Alexandria Brown, Austin Carter, Daniel C Casey, Fiona J Charlson, Alan Z Chen, et al. Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: a systematic analysis for the global burden of disease study 2015. *The Lancet*, 388(10053):1545–1602, 2016.