Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Earthquake Risk Induction Models with Genetic Algorithm

Yuri Cossich Lavinas

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Marcelo Ladeira

Coorientador
Prof. Dr. Claus de Castro Aranha

Brasília
2016

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Earthquake Risk Induction Models with Genetic Algorithm

Yuri Cossich Lavinas

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Marcelo Ladeira (Orientador)
CIC/UnB

Prof. Dr. Donald Knuth
Stanford University

Dr. Leslie Lamport
Microsoft Research

Prof.a Dr.a Ada Lovelace
Coordenadora do Bacharelado em Ciência da Computação

Brasília, 24 de dezembro de 2016

Dedicatória

???

Agradecimentos

???

Resumo

Entender os mecanismos e padrões dos terremotos é importante para minimizar suas consequências. Neste contexto, este projeto visa desenvolver um modelo de previsão de riscos de terremotos com Algoritmos Geneticos (GA). Modelos de risco de terremotos descrevem o risco de ocorrência de atividades sísmicas in uma determinada área baseado em informações previamente obtidas de terremotos em regiões próximas da área de estudo. Nós utilizamos GA para aprender um modelo de risco usando somente informações previamente obtidas como base de treino. Baseado nos resultados obtidos, nós acreditamos que é possível obter melhores models se conhecimento do domínio da aplicação, como conhecimentos oriundos da literatura ou modelos de distribuição de terremotos, poderem ser incorporados ao processo de aprendizado do Algoritmo Evolutivo.

O objetivo principal é definir um método para estimar a probabilidade de ocorrências de terremotos no Japão usando dados históricos de terremotos para um grupo de determinadas regiões geográficas. Este trabalho se baseia no contexto do "Collaboratory for the Study of Earthquake Predictability" (CSEP), que visa padronizar os estudos e testes de modelos de previsão de terremotos.

Durante o desenvolvimento das atividades, passamos por quatro estágios. (1) Nós propusemos um método baseado em uma applicação de Algoritmos Genéticos (GA) e objetivamos gerar um método estatístico de análise de risco de terremotos. Estes foram analisados por seus valores de *log-likelihood*, como sugerido pelo *Regional Earthquake Likelihood Model* (RELM). (2) Basedos nos resultados obtidos, nós buscamos melhorar a performance do GA ao incluir uma técnica de GA chamada auto-adaptativo. (3) A seguir, modificiamos a representação do genoma, de uma representação baseada em área para uma representação baseada em ocorrências de terremotos, buscando obter uma convergência mais rápida dos valores de *log-likelihood* dos candidatos do GA e (4) usamos métodos da sismologia conhecidos (como a equação de Omori-Utsu) para refinar os candidatos gerados pelo GA.

Em todas as estapas, os modelos de risco são comparados com dados reais, com modelos gerados pela aplicação do *Relative Intensity Algorithm* (RI) e com eles próprios. Os dados utilizados foram obtidos pela *Japan Metereological Angency* (JMA) e são relativos

a atividades de terremotos no Japão entre os anos de 2000 e 2013.

Nós analisamos as contrubuições de cada modelo proposto usando metodologia descritas pelo CSEP e comparamos as performances entre (XXX method and YYY method). Os resultados apontam (XXX result, YYY result). /*terminando com os resultados obtidos e conclusoes alcancadas.*/

Palavras-chave: algoritmos genéticos, terremotos, log-likelihood

Abstract

To understand the mechanisms and patterns of the earthquakes is very important to minimize its consequences. In this context, this projects aims to develop an earthquake prevision risk model using Genetic Algorithms (GA). Earthquake Risk Models describe the risk of occurence of seismic events on a given area based on information such as past earthquakes in nearby regions, and the seismic properties of the area under study. We used EC to learn risk models using purely past earthquake occurrence as training data. Based on the results obtained, we believe that a much better model could be learned if domain knowledge, such as known theories and models on earthquake distribution, were incorporated into the Genetic Algorithm's training process.

The main goal is to define good methods to estimate the probability of earthquake occurrences in Japan using historical earthquake data of a group of given geographical regions. This work is established in the context of the "Collaboratory for the Study of Earthquake Predictability" (CSEP), which seeks to standardize the studies and tests of earthquake prevision models.

To achieve the main we passed four stages. (1) We proposed a method based in one application of Genetic Algorithms (GA) and aims to develop statistical methods of analysis of earthquake risk. The risk models generated by this application were analyzed by their log-likelihood values, as suggested by the Regional Earthquake Likelihood Model (RELM). (2) Based on the results obtained, we tried to improve the GA's performance by including the self-adaptive GA technique. (3) Then, we modify the genome representation from an area-based representation to an earthquake representation aiming to reach a faster convergency of the log-likelihood values of the GA's candidates and (4) we use known methods from seismology (such as the Omori-Utsu formula) to refine the candidates generated by the GA.

In all stages, the risk models are compared with real data, with the models generated by the application of the Relative Intensity Algorithm (RI) and with themselves. The data used was obtained from the Japan Metereological Angency (JMA) and are related with earthquake activity in Japan between the years of 2000 and 2013.

We analyze the contributions from each of each risk model using the methodologies

described in the CSEP, and compare their performance with (XXX method and YYY method). Our results indicate that (XXX result, YYY result). /*terminando com os resultados obtidos e conclusoes alcancadas.*/

Keywords: genetic algorithms, earthquake, log-likelihood

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introduction

In this chapter we present a general specification of the problem, its relevance and what are the goals of this study.

1.1 Earthquakes

Earthquakes may cause lots of damages environment and consequently may represent, directly or inderectly, a risk to human lives. They manifest themselves by shaking and moving of the ground. They may also cause tsunamis, landslides, volcano activities, etc.

There are many examples that show how devastating one large earthquake can be. In 25th of April 2015, there was a strong earthquake in Nepal, with moment magnitude of M_w 7.8 and considered the largest since 1934. It destroyed lots of buildings and infrastructure, and triggered numerous landslides and rock/boulder falls in the mountain areas [36]. Many other aftershock occurrences, which are dependent earthquakes [35], happened after it, including two major aftershocks M 6.7 and M 7.3 earthquakes that caused additional that were also very destructive [36].

Another example happened in March 2011, Japan. It was a 9.0 M_w earthquake [32], and it is considered the most powerful earthquake to ever hit Japan. It caused tsunami waves that reached more than 39 meters, moved the main island in Japan more than 2 meters east and also changed the Earth axis. It was reported that it caused more than 14 thousand deaths, made more than 244,000 people homeless and provoked a meltdown of the Fukushima Daiichi Nuclear Power Plant complex. Many large aftershocks followed the main event [19]. Also in 2011 a magnitude M_w 7.1 earthquake hit Van, Turkey and caused lots of deaths and great damages. These are only three very recent examples of

large earthquake damages of how dangerous earthquakes can be [8].

Those earthquakes, and many others that hazard the human society, have some common characteristics. They not only are powerful quakes but they happened nearby populated areas, which increase the damaged provoked. To minimize as much as possible future earthquake disaster, a lot can be done. That includes developing good urban planing, for example to build structures with techniques that can withstand the forces of earthquakes, to create earthquake warning systems, to create more precise civil enginnering codes, and such.

To be able to prevent as many casualty as possible, we need to undersatnd the patterns and mechanisms behind the occurrence of earthquakes. We need to know if there is any relationship between the earhquake locations and its time of occurrance, how they are related to each other, et cetera. With this information, it is possible to to create better seismic risk forecast models, indicating which regions show a higher probability of earthquake occurrence at certain periods in time.

Until by now, it has been difficult to clearly understand the many different seismic variables (ime of occurrence, magnitude, local, depth,...) influences the quakes and either exists a mathemathical model capable of supplying detailed and precise information about the relations and ways to estimate them. Therefore, to develop a prediction earthquake risk model can prove itself very complex.

1.2 Earthquake Prediction

In [15], Koza says that Evolutionary Computation (EC) may find, by try trial and error and based on a great amount of data, better solutions for problems that human beings may not find it easy to solve. EC is a family of subfield of artificial intelligence that aim to extract patterns and to solve problems using a great amount of historical data. We may also say that without any domain knowledge about the problem to be controlled, the EC learns about may learn and find solutions for the problem. [18].

Genetic Algorithm (GA) is the chosen EC technique that is used in this study. It constitutes a category of heuristic search, they are stochastic algorithms and their search method are based on genetic inheritance and surival of the fittest [17]. They are interesting to be used specially in cases that are difficult to understand and the knowledge

available is not sufficiently available.

Based on these information and on the difficulty to understand how earthquakes behave, we want to explore historical earthquake data using GA. It is expected that it will help to find new ideas about earthquakes, their patterns and their mechanisms behind earthquake occurrences. For doing so, we need first to outline the forecast problem, then verify the suitability of Evolutionary Computation to the problem of generating earthquake forecast models.

Earthquake prediction is a polemic subject. No research has even come close to suggesting that individual large scale earthquakes can be predicted [1] and many scientists think that earthquake prediction may not only be fully impossible, but also that the resources needed for such a prediction may be out of reach [4].

In the context of this study, we do not aim to predict any individual earthquake and its major characteristics. Our goal relies on the fact that earthquakes do cluster in time and space. We want to use computer techniques to learn and to generate risk models. There is a lot of value behind the study of earthquake mechanisms, with the goal of generating statistical models of earthquake risk [28].

Next, we will study ways to improve the generated methods using both GA and/or other computer techniques and any siesmological knowledge. We propose different representations that aim to refire the algorithm performance and to incorporate seismology methods to refine the models proposed.

1.3 Document Organization

This document is organized into 7 more chapters. The next, is about teoretical concepts there are useful for undestanding the development of this study. Then, in the chapter 4.3.2, we will discuss the current state of art regarding applications of Evolutionary Computation (EC) in the context of seismology research and earthquake risk models.

The chapter 3.2 is about the methods proposed in the context of this study and a detailed description of theirs characteristics.

The next chapter 5.1 is about the earthquake data used, also called catalogue. It is also about a statistical analysis of the data and any relevant decisions made regarding the data itself.

The chapter 6.9 is about the experiments. The following chapter 7.2 is about the analysis of the experiments and the results observed from it.

In the last chapter 8 brings the conclusion of the study as well as a little discussion of the contribuitions of this work and future works.

Capítulo 2

The Earthquake Forecasting Problem

This chapter focus on the teoretical concepts used as base for this study. The main topics are the CSEP framework and its tests.

2.1 Earthquake Likelihood Model Testing

We started studies of the earthquake forecasting problem by determining and selecting ways to build earthquake forecast models, to evaluate and to compare them, as suggested by the Collaboratory for the Study of Earthquake Predictability (CSEP). It is an international partnership to promote rigorous study of the feasibility of earthquake forecasting and predictability [1].

For that, we gathered some important information about earthquake predicting needed for this study. Most of it is based on the paper *Earthquake Likelihood Model Testing* [30]. From this paper we gathered information that guided us into how to build, evaluate and compare earthquake forecast models efficiently.

A very difficult and yet very common problem when studying earthquake models is how to compare different kinds of models, that are based on different tests protocols. The CSEP proposes a methodology for rigorous scientific testing of these many different models. This group proposed an framework called The CSEP framework. It provides a method to compare earthquakes risk models in an objectively and consistently way [1].

All forecast models proposed in this study are based in the Collaboratory for the Study of Earthquake Predictability (CSEP) framework. In the CSEP framework, a forecast model uses a gridded rate forecast [38], one common format in the literature. For evaluate and compare these models we used the likelihood based tests. They are the L-test, the N-test and the R-test, as suggested by Regional Earthquake Likelihood Model (RELM) [30].

The principle behind each consistency test is the same. One calculates a goodness-of-fit statistic for the forecast and the observed data. One then estimates the distribution of this statistic assuming that the forecast is the data-generating model (by simulating catalogues that are consistent with the forecast). One then compares the calculated statistic with the estimated distribution; if the calculated statistic falls in lower tail of the estimated distribution, this implies that the observation is inconsistent with the forecast, or that the forecast should be "rejected". For the CSEP consistency tests used here, the likelihood is the fundamental metric, but this approach would be similar for different statistical measurements [4].

2.1.1 Vector of expectations

As stated in section 2.1, The CSEP framework uses a gridded rate forecast. This gridded forecast may be structured by a vector of earthquake expectations, occurrences probabilities, that are directly related to a vector of real earthquake observations.

Based on this structure, it is possible to calculate the Log-likelihood value of a model with the real data observed. It is also possible to use comparison tests based on the calcultation of the Log-likelihood.

2.1.2 The Log-Likelihood Function

To calculate the Log-likelihood value we need both vectors cited above, in section 2.1.1. One of them is the vector of earthquake expectations and the other is the vector of real earthquake observations. On them, each element is considered a bin.

Each bin, b_n, define the set β and n is the size of the set β:

$$\beta := b_1, b_2, ..., b_n, n = |\beta|. \tag{2.1}$$

The probability values of the model j, expressed by the symbol Λ, is made of expectations λ_i^j by bin b_i. The vector is define as:

$$\Lambda^j = \left(\lambda_1^j, \lambda_2^j, \ldots, \lambda_i^j\right); \lambda_i^j := \lambda_i^j(b_i), b_i \in \beta \tag{2.2}$$

The vector of earthquake quantity expectations is defined as: earthquake by time. The Ω vector is composed by observations ω_i per bin b_i, as the Λ vector:

$$\Omega = \left(\omega_1, \omega_2, \ldots, \omega_i\right); \omega_i = \omega_i(b_i), b_i \in \beta \tag{2.3}$$

The calculation of the log-likelihood value for the ω_i observation with a given expectation λ is defined as:

$$L(\omega_i|\lambda_i^j) = -\lambda_i^j + \omega_i \log \lambda_i^j - \log \omega_i! \tag{2.4}$$

The joint probability is the product of the likelihood of each bin, so the logarithm $L(\Omega|\Lambda^j)$ is the sum of for $L(\omega_i|\lambda_i^j)$ every bin b_i:

$$\begin{aligned} L^j = L(\Omega|\Lambda^j) &= \sum_{i=1}^{n} L(\omega_i|\lambda_i^j) \\ &= \sum_{i=1}^{n} -\lambda_i^j + \omega_i \log \lambda_i^j - \log \omega_i! \end{aligned} \tag{2.5}$$

The fitness function is a coded version of the equation 2.5. It uses the probabilities of the bins of each individual of model for the λ values.

2.1.3 Uncertainties in Earthquake Parameters

It is important to say that the earthquake parameters, as the location, magnitude and focal time, cannot be estimated without uncertainties. Therefore, each parameter uncertainty has to be included in the testing [30]. Moreover, by estimating it, it is possible to judge the reliability and robustness of the forecast testing [4]. Also, each observation must be treated as independent ones. This is not the case of the aftershocks, once they are directly dependent with another stronger earthquake.

2.2 Tests for evaluating Models

In the paper *Earthquake Likelihood Model Testing* [30], it is proposed some statistical tests that are used in this study, developed by the The Regional Earthquake Likelihood Models (RELM). They were used to compare and evaluate the every forecast models. These tests are based on the log-likelihood score that compares the probability of the model with the observed events.

To evalute the data-consistency of the forecast models we used the N-Test, the Number Test, and the L-Test, or Likelihood Test. These tests fall are significance tests. Therfore, assuming a given forecast model as the null hypothesys, the distribuition of an observable test is simulated. If the observed test statistic falls into the upper or lower tail of this distribuition, the forecast is rejected [31].

To be able to compare the model that passed the N-Test and the L-test, the R-Test, the hypotheses Comparison Test, is used. It calculates the relative performance of a model, by comparing the Log-likelihood values between two forecast models.

2.2.1 L-test - Data-consistency test

The L(ikelihood)-Test considers that the likelihood value of the model is consistent with the value obtain with the simulations. The value is calculated by fowlling the formula, where $\widehat{L}_k$ is the value of the Log-likelihood of the model *j*, in the *bin i* and $\widetilde{L}$ is the value of the Log-likelihood of the simulation *j* in the *bin q*:

$$\gamma_q^j = \frac{\left|\left\{\widehat{L}_k^j | \widehat{L}_k^j \leq \widetilde{L}_q^j, \widehat{L}_k^j \in \widehat{L}^j, \widetilde{L}_q^j \in \widetilde{L}^j\right\}\right|}{|\widehat{L}^j|} \tag{2.6}$$

The analysis of the results can be splited into 3 categories, as follows:

1. Case 1: γ^j is a low value, or in other words, the Log-likelihood of the model is lower then most of the Log-likelihood of the simulations. In this case, the model is rejected.
2. Case 2: γ^j falls near the half of the values obtained from the simluations and is consistent with the data.
3. Case 3: γ^j is high. This means that the Log-likelihood of the data da is higher that the Log-likelihood of the model and no conclusion can be made what so ever.

It is important to highlight that no model should be reject in case 3, if based only on the L-Test. In this case the consistency can or cannot be real, therefore these model should be tested by the N-Test so that further conclusions can be done.

2.2.2 Number test or N-Test

The N(umber)-Test also analises the consistency of the model, but it compares the number os observations with the number of events of the simulations. This test is necessary to supply the underpredicting problem, which may pass unnoticed by the L-Test.

This mesure is estimated by the fraction of the total number of observations by the total number of observations of the model.

As the L-test, if the number of events falls near the half of the values of the distruition, then the model is consistent with the observation, nor estimating too much events nor too few of them.

2.2.3 Hypotheses Comparison Test or R-Test

The Hypotheses Comparison, or the R(atio)-Test, compares two forecast models against themselves. The log-likelihood is calculted for both models and then the difference between them is calculated, named the observed likelihood ratio. This value indicates which one of the model better fits the observations.

The likelihood ratio is calculated for each simulated catalog. If the fraction of simulated likelihood ratios less than the observed likelihood ratio is very small, the model is reject. To make this test impartial, not given an advantage to any model, this procedure is applied symmetrically [31].

2.2.4 Evaluation

This section may need to be placed elsewhere.

The evaluation process is made as follow: First, the data-consitency is tested by the L-Test and the R-test. If the model passes these tests, meaning that it was not rejected by them, they ares compared with other forecast models, which were also not reject, with

the R-Test. The model that best fits the R-Test is then chose as the best model [30].

Capítulo 3

State of Art

In this chapter we will briefly about Genetic Algorithms and then discuss some reports of the application of Evolutionary Computation and related method for Earthquake Risk Analysis.

3.1 What are Genetic Algorithms

The main goal of a Genetic Algorithm (GA) is to find approximated solutions in problems of search and optimization. Based on Koza [14], GA are mechanism of search based on natural selection and genetic. They explore historical data to find optimum search points with some performance increment, as said by Goldberg [**?**].

3.1.1 How does GA work

A GA uses those mechanisms to generate solutions to optimization and search problems. The first step is to create an initial population of possible solutions. Frequently, the initial population is randomly generated once it is common to ignore the main aspects that influence the algorithm performance.

Each possible solution of a population is called an individual. Every individual is a possible solution of a problem. Those individuals have its fitness value estimated by a fitness function. A fitness function should determine how suitable a individual is to a given problem. The most suitable individuals are graded with better values and the not so suitable ones have a lower value.

After measuring the population fitness value, some individuals are then selected by a process that takes into account each individual fitness value to influence the next population. The individuals with better values have a higher chance to be selected. The individuals selected take part in the varation process. This process may alter some of the individual characteristics using the crossover and mutation operators.

The crossover operator is a operator that is used to vary the characteristics of a group of individuals. For that a number of parents, a group of individuals from the current population, are selected. In most of the cases, the parents are chosen to compose a pair that will exchange information that will take compose the child, a new individual that will belong to the next generation.

Another important operator is called the mutation operator. It is a operator with the purpose of avoiding the loss of important information. It works by changing the characteristics of an individual, looking to add new information to the next population.

It is common to have a evolutionary operator that allows the fittest individual from the current generation to take part in the next generation. This operator is called Elitism and it is used to assure that the next generation best solution is at least as good as in the current generation.

3.2 Evolutionary Computation and Earthquake Risk Prevision

The usage of Evolutionary Computation in the field of earthquake risk models is somewhat sporadic.

Zhang and Wang [39], Zhou and Zu [40] and Sadat [27], used Artificial Neural Network (ANN) related with earthquake prediction. In all these works, they combine the ANN with some other technique, to achieve better results. They used a group of earthquake parameters, as the accumulated release energy, magnitude in a specific area, the b-value and others. Some parameters in this group are not available in our earthquake database.

Those papers use the available parameters of the earthquakes that happened in the area of study to create a risk prediction of earthquake or to propose a magnitude range for future earthquakes. They object to consider each variables influence the most the results

so that their methods can achieve higher performance. We may compare and/or evaluate our method by comparing it to the works cited before.

Nicknam et al. [21] simulated some components from a seismogram station and predicted seismograms for other stations. They combined the empirical Green's function (EGF) with GA. the EGF method is used to synthesize acceleration time histories and the GA approach is developed to optimize the seismological model. They found that this method obtained good agreement with the observed data, but are not sure that results are free from uncertainties.

In this paper, they work with more than 30 seismological model parameters. Allthough, that amount of parameters is not available to us, we can use the information from this paper to exame two options. The first, we may investigate if more earthaquake parameters will improve our method and the other option is to analyse how they dealt with so many variables. Then we may consider to do the same and observe the results.

Kennett and Sambridge [10] used GA and associated teleseisms procedures to determine the Fault Model parameters of an earthquake. By doing so, they demonstrated that non-linear inversion can be achieved for teleseismic problems without any calculation of waves travel times. They used only P-wave data and expect that if more data could be introduced, the method would accomplish better results.

Some sismological models were developed aiming to estimate parameter values by using Evolutionarry Computation. For example, Evolutionary Computation was used to estimate the peak ground acceleration (PGA) of seismically active areas [13, 2, 11, 12].

The works done by Kerh [11, 12] are basely a combination of ANN and GA to estimate or predict PGA in Taiwan. These work are based on the benefits of mixing both techniques. They state that the usage of a purely ANN method to estimate PGA may fall into a local minimum and that can be avoid by combining ANN with GA, hence GA is a good method to find global optimums.

Their goal was to decide which areas may be considered potentially hazardous areas. They focused on urban areas, these works are important to revalidate building regulations, urban development and such. The earthquake variables that were used in these work are: local magnitude, focal depth, and epicenter distance. Both magnitude and depth are already used in our work, which is not the case of the epicenter distance variable.

They also state that PGA is inversely proportional to epicenter distance, so to add data about this variable may be useful to our work, once it could provide useful information to predict risk models both direct or indirectly.

Ramos [26] used Genetic Algorithms to decide the location of sensing stations. In this work Ramos achieved, in general, better results with the GA method when compared with the seismic alert system (SAS) method and a greedy algorithm method. In some cases, the SAS has a better reponse time than the GA. They consider it to be once caused because the SAS only alerts when earthquakes with magnitude bigger than 5.0 degress in the Richter scale occurs, while the GA deals with all the earthquakes.

Ramos's work is a important work because it helps the population to avoid bigger disasters caused by earthquakes by incrensing the time response of the Seismic-Sense Stations. It has some similar feature as the one present in this document: it uses GA to prevent earthquake disasters and tries to locate targets in a given area (though the targets of this work is sensing stations and ours works target is the earthquakes location) and it proposes a methodology to do a GA parameter setting to find which combination of values for the GA parameters achieve highier results. It is interesting to state that once a solution places a station in an area that is not possible to be have sensors, this possible solution suffers some penalities.

Saeidian [29] also based on the same idea of locating sensing stations. His work differs from the work of Ramos because it makes a comparation in performance between the GA and Bees Algorithm (BA) to decide which of those techniques would performe better when chosing the location of sensing stations. He found out that the GA was faster than the BA.

Huda and Santosa [7] published a paper in which the goal is to find, via Genetic Algorithm, the speed of the waves P and S in the mantle and in the earth crust. P waves are indicated as the first fault found in seismological data and S waves are the changes caused in the phase of a P wave [7]. This research aims to obtain a structure of the japanese undergound and geographically focuses in the same region as our work, though it uses data from two kinds of waves which are not avaiable to us.

Capítulo 4

Models

As stated in the Section 2.1, all forecast models proposed for this study are based in the Collaboratory for the Study of Earthquake Predictability (CSEP) framework.

We propose four forecast model methods. The main difference between them is that they have different genome representation. The genome for each forecast model focus on different aspects of the framework, therefore their represantion vary.

The first method is the GAModel [1], a statistical method of analysis of earthquakes risk using the Genetic Algorithm technique (GA). It is a straight application of the CSEP framework. The next method is a specialization of GAModel. It focuses only on areas on which earthquakes happened already in a near past. This will lead to a faster convergence, once the amount of parameters is smaller and consequently, the search space gets smaller. We called it ReducedGAModel(bad name? suggestions?). These methods useonly computational algorithms and techniques.

Another method is the Emp-GAModel (bad name? suggestions?). This method incorpores some geophysical knowledge. It is a hybridization of the models generated by the GAModel with some empirical laws that will be discussed further, in Section 4.3. We also applied these empirical laws in the ReducedGAModel, and name it Emp-ReducedGAModel.

For all methods, the population is evolved taking into account earthquake event data for a training period, which is anterior to the target test period. After completing the evaluation stop criteria, the best individual is chosen to be the representative forecast model for that method.

4.1 1-year Forecast Models

Based on the gridded rate forecast explained in the last Chapter 2.2.4, we developed earthquake forecasts methods that will estimate the risk of earthquakes occurrence to a target region, during a time interval. Some of the methods also may estime the magnitude of these shocks. For this study we considered the target time interval of one year [1].

There is no physical measurement to identify mainshocks and its aftershocks [31], we divided the forecast models in two classes: the ones that only forecasts mainshocks, using only GA techniques, and those that forecast both mainshocks and aftershocks using both GA techniques and empirical laws, such as the modified Omori law. These laws are use to derive the aftershocks from a sintetic data of mainshocks.

Mainshocks are large and independent earthquakes. They are followed by a wave of others earthquakes, the aftershocks [31].

4.2 Mainshock Methods

The mainshock methods are considered as methods to generate space-rate-time forecasts. They could be described as:

$$\Lambda(t, x, y, M|\Upsilon_t) = \mu(x, y) \tag{4.1}$$

where the number of earthquakes forecast in all bins can denoted as $\Lambda(t, x, y)$ [38] given that Υ_t is the earthquake observation data up to time t.

4.2.1 GAModel

The GAModel is completely based on the framework suggested by the CSEP. In it, one forecast is defined as a region in a specfic time interval and is divided in bins. Each bin represents a geographical interval. The whole target area of study is covered by a group of these bins where each bin has an earthquake forecast value. This groups of bin represent the $\mu(x, y)$, the background intensity [41]. In the GAModel, each possible solution is represented as an entire forecast model.

The GAModel forecasts only earthquakes with magnitude greater than 3.0, for every scenario proposed. The space interval for the magnitude is 0.1, named as cells. That results in magnitude cells of [3.0, 3.1), [3.1, 3.2), until [9.9, 10).

Genome Representation

In the GAModel each individual represents an entire forecast model. Each gene of the individual is a real value, corresponding to one bin in the desired model. The values are sampled from the interval [0, 1). These real values are converted to a integer forecast, we use the same modification of the Poisson deviates extraction algorithm 4.2.1 used in [1]. In the algorithm x is the real value that will be converted and μ is the mean of the earthquakes observations in the real data.

Algorithm 1 Obtain a Poisson deviate from a $[0, 1)$ value

```
Parameters 0 ≤ x < 1, μ ≥ 0
L ← exp(−μ), k ← 0, prob ← 1
repeat
    increment k
    prob ← prob * x
until prob > L
return k
```

The genome is a real valued array X, where each element corresponds to one bin in the desired model (the number of bins n is defined by the problem). Each element $x_i \in X$ takes a value from $[0, 1)$. In the initial population, these values are sampled from a uniform distribution and they are randomly generated. For more details of the genome representation, please refer to [1].

To clarify how the GAModel works, we use the same example as the one used in [1]. The "Kanto"region, one of the four areas used in both studies, is divided into 2025 bins (a grid of 45x45 squares). Each bin has an area of approximately $25km^2$. The GAModel then calculates an expected number of earthquakes for every bin on a determinated time interval, so the GA searches for good values in 2025 bins.

Fitness Function

To compare the indididual data with the observated data, we use the log-likelihood calculation as fitness function. This equation allow us to compare events in the obser-

vated data with the values of occorrences obtained by a model. The models that have more similarity with to the observated data have bigger log-likelihood values. The fittest individual among all the others, is preserved inthe next generation, to make the solution of one generation as good as the its last generation.

The fitness function is a coded version of the equation 2.1.2. It uses the probabilities of the bins of each individual of model for the λ values.

Evolutionary Operators

The GAModel use a combination of operators made available by the Distributed Evolutionary Algorithms in Python (DEAP) [3]. We used the One Point Crossover for the crossover operator, the Polynomial Bounded Mutation for the mutation operator and for selection, we used Tournament selection and Elitism. The parameters are described in the Table 4.1.

Tabela 4.1: Parameters used in GAModel and Emp-GAModel

Population Size	500
Generation Number	100
Elite Size	1
Tournament Size	3
Crossover Chance	0.9
Mutation Chance (individual)	0.1
Polynomial Bounded parameters	eta = 1, low = 0, up = 1

The parameters of the Polynomial Bounded mutation funcition are: add online ref???

1. eta = 1. Crowding degree of the mutation. A high eta will produce a mutant resembling its parent, while a small eta will produce a solution much more different;
2. o low = 0. The lower bound of the search space;
3. o up = 1. The upper bound of the search space.

The chance of applying both mutation operator function and crossover operator function takes into account only their chance of occurrence. This means that it may be the case that one of them or both are not applied.

4.2.2 ReducedGAModel

The GAModel defines a expected number of earthquakes for every single bin in the target region. That could lead to exhaustive and, sometimes worthless, searches. That is caused by the number of bins in the forecast and also because in some bins there are no earthquake occurances in the observation data. That means that the GAModel has a lot of parameters and may of its bins have null values (values igual to 0). To avoid such unnecessary task we proposed the ReducedGAModel.

With this method, we aim to minimize the search space and the quantity of parameters the GA has to deal with. For that we changed the individual representation. The individuals in the ReducedGAModel only define expected number of earthquakes in bins that already had some occurance in the past, giving a direction to where the GA should search. That helps the ReducedGAModel in the search for better solutions and it makes the convergence faster once the space search is smaller.

The ReducedGAModel has a similar description of the GAModel. As said in the last paragraph, the difference is that, in the ReducedGAModel, each possible solution represents only a fraction of the forecast where we expect to find especific risk areas. To do so, this method will obtain the position of past occurances. Then it will calculate some expected number of earthquakes only for the bins related to those positions. These positions may vary during the evoluting of the method, including positions that never had earthquake events before. That is important to add some variation to the method.

The ReducedGAModel, as the GAModel 4.2.1, forecasts only earthquakes with magnitude greater than 3.0, for every scenario proposed. The space interval for the magnitude is 0.1, named as cells. That results in magnitude cells of [3.0, 3.1), [3.1, 3.2), until [9.9, 10).

Genome Representation

The genome representation in the ReducedGAModel is a simplified version of the genome of the GAModel. For the ReducedGAModel, the genome is a list of ordered pairs. The first element of the pair are the coordenates of a bin in the model. The second element of the pair is a number that indicates an earthquake occurrence estimative for this bin.

To calculate the size of the individual we use the real data from the pior 5 years and create a list of every bin that had events in it, even if only once.

In the ReducedGAModel, each individual is a list of a subregion of the forecast model. This list initially refers to bins where earthquake events happened in the past. During the develop of the ReducedGAModel, the list may refer to positions that never had occurrences before. Each element of the list, a gene, also contains one real value between [0,1). In the initial population, these values are sampled from a uniform distribution and they are randomly generated. When needed, every real value is converted to a integer forecast by the algorithm 4.2.1, as in the GAModel 4.2.1.

To generate the forecast model we need to do an intermediate step. We map every location from the list with a bin in the forecast model.

The genome size is usually smaller than the one used in the GAModel and the EmpGAModel, once the amount of subregions where earthquakes with magnitude above 3.0 happened for any given area is smaller then the total number of genes of the individual.

To examplify, we use a similar example as the one in 4.2.1. Lets consider that there are 10 bins with occurances in "Kanto"in the last 5 years, it will make the GA start searching for good values for only those 10 bins, leaving the other 2015 bins empty, representing zero occurances. It is important to highlight that in the worst case, it will make the same amount of searches as the GAModel. The final forecast model will maintain the amount of bins with occurrance, but the number of events for every bin and their location may change.

Fitness Function

The fitness function is the same as in the GAModel, 4.2.1. Here is also important to generate the forecast model by applying the map function on the individual as in the last Section, 4.2.2.

Evolutionary Operators

All operators in the ReducedGAModel are the same as the operators of the GAModel, except the the mutation fuction. We use a simple mutation operator which samples entirely two new values, both sampled from uniform distributions. The first, is a new

real value from [0,1) and the seconde one, a new integer value from [0,x), where x is the maximum position value a bin can have in the target region. For the parameters see Table 4.2.

Tabela 4.2: Parameters used in ReducedGAModel

Population Size	500
Generation Number	100
Elite Size	1
Tournament Size	3
Crossover Chance	0.9
Mutation Chance (individual)	0.1

As in the GAModel, see 4.2.1, the chance of applying both mutation operator function and crossover operator are independent and they may or may not be used.

4.3 Mainshock+Aftershock Methods

The mainshock+aftershock methods are a two-step methods. The first step is as defined for the mainshocks methods, therefore, we first use GA techniques to obtain a sintetic mainshock data. The second step is to use seismological empirical equations to obtain the aftershocks from the mainshocks.

Hence earthquakes cluster in space and inspired by the space-time epidemic-type aftershock sequence (ETAS), we proposed two methods, called Emp-GAModel and Emp-ReducedGAModel. They represent the ideia of associating the GA with seismological empirical equations. They are described as:

$$\Lambda(t, x, y, M|\Upsilon_t) = \mu(x, y)J(M) \tag{4.2}$$

That can be expanded to:

$$\Lambda(t, x, y|\Upsilon_t) = \mu(x, y) + \sum_{t_i \in t} K(M_i)g(t - t_i)P(x, y) \tag{4.3}$$

The mainshock+afterchock methods use $\mu(x, y)$ as defined for mainshock methods 4.2. It is calculated as an expected number of earthquakes for every bin in the target region, given that Υ_t is the earthquake observation data up to time t.

Empirical Equations

The Omori law, $g(t)$, which is considered one empirical formula of great success [41][34][23], is a power law that relates the earthquake occurance and its magnitude with the decay of aftershocks activity with time. For this approach we used the probabilty density function (pdf) form of the modifed Omori law [41]:

$$g(t) = \frac{(p-1)}{c(1+\frac{t}{c})^{(}-p)} \tag{4.4}$$

The variable p is a index of this equation and the variable c is a constant, given in days. In the paper [34], Utsu summarize most of the studies in Japan and described the range for these variables. For p the range is between 0.9 and 1.4 and for c, 0.003 and 0.3 days. These values were based on the Davidon-Fletcher-Powell optimization procedure and used in ETAS [34]. Also there is the variable t that is the time limit to when a mainshock may influence the cause a aftershock.

TODO: should i move this to exp?? Based on paper [37], we set the values of 1.3 for p and 0.003 for c for our experiments. We set the time interval t between a mainshock and its aftershocks at one month. In the paper, it says that if the t value is too short, the number of aftershocks is too small, but if it is too big, we may also consider background activity and suggest the use of a 30 days period.

For $K(M_i)$, the total amount of triggered events, we count aftershocks within a given area, A, using the following formula, where M_c is the magnitude threshold:

$$K(M_i) = A\ exp([\alpha(M_i - M_c)]) \tag{4.5}$$

In the paper [22], it states that α should be equal to the inverse of the magnitude of an event, or $magnitide^{-1}$. To obtain A, the following equation from [37], was used:

$$A = e^{(1.02M-4)} \tag{4.6}$$

With the $K(Mi)$ and $g(t)$, the PDF Omori, equations it is possible to calculate the total number of earthquakes. For that we must sum the product of the equations, varying t:

$$\sum_{t_i \in t} K(M_i)g(t-t_i) \tag{4.7}$$

This result will lead to a number of aftershocks realte to a single mainshock. Then, we can use the $P(x, y)$ equation to distribute the aftershock to the bins near the mainshock's position. $P(x, y)$ calculates the position of the aftershocks with base on the origin of the mainshock. It is a simple space distribuition function, that alocates the aftershocks in one of the following positions: upper, lower, left or right. It runs for a number of steps, getting futter from the origin at each step or as when there are no more events to be alocated.$P(x, y)$ can be splited into 4 equations, one for each position:

$$
\begin{aligned}
model[x + y] &= (aftershocks - [model[x] - 2 * x])/4; \\
model[x - y] &= (aftershocks - [model[x] - 2 * x])/4; \\
model[x - y * row] &= (aftershocks - [model[x] - 2 * x])/4; \\
model[x + y * row] &= (aftershocks - [model[x] - 2 * x])/4
\end{aligned}
$$

and lastly, the $J(M)$ is a simulation of the event magnitude by Gutenberg-Richter's Law, using Add SAPP

4.3.1 Emp-GAModel

The Emp-GAModel is a specialization of the GAmodel. This is achieved by the use of empirical equations after the forecast is provided. This means that the its genome representation are the same as the GAModel.

Genome Representation

The genome representation is the same as in the GAModel, 4.2.1.

Fitness Function

The fitness function is the same as in the GAModel, 4.2.1, and the ReducedGAModel.

Evolutionary Operators

The Emp-GAModel use the same combination of operators that the GAModel. For more explanation, please see 4.2.1 and table 4.1.

4.3.2 Emp-ReducedGAModel

The Emp-ReducedGAModel is a specialization of the ReducedGAmodel. This is achieved by the use of empirical equations after the forecast is provided. This means that the its genome representation are the same as ReducedGAModel.

Genome Representation

The genome representation is the same as in the ReducedGAmodel, 4.2.2.

Fitness Function

The fitness function is the same afor all methods, 4.2.1. Here is also important to generate the forecast model by applying the map function on the individual as in the last Section, 4.2.2.

Evolutionary Operators

The Emp-ReducedGAModel use the same combination of operators that the ReducedGAModel. For more explanation, please see 4.2.1 and table 4.2.2.

Capítulo 5

Análise dos Dados

5.1 Earthquake data

The goal of this research is to find existing patterns in the occurrence of earthquakes. For that it is essential to access trustful data and to explore its details. From the *Japan Metereological Agency* webpage we obtained earthquake data about quakes in Japan. In this data there are information about earthquakes that happened in or nearby Japan, with the variables: time of the occurrence, magnitude, latitude and longitude and epicenter profundity, for the years of 2000 to 2013.

From our first analysis, we discovered a higher number of occurences of earthquakes during the year of 2011, when the 9.0 M_w earthquake happened, see section 1.3. This earthquake triggered too many aftershocks in all Japan. It is considered that big earthquakes may cause others earthquakes (also called aftershocks) [41]. In Figure 5.1 it is possible to visualize a great number of quakes for the year of 2011. Because of this abnormal behavior and because we decided to focus on more stable occurances, we liminated the tranning base to earthquakes until 2010.

Based on the statement done before and considering that we want earthquakes that follow more stable patterns, we selected the ones that happened in land areas or very shallow sea areas, with maximum depth of 100 kilometers and magnitude above 3.0 in the Richter Scale.

For the experiments, the data was changed into slices for every year. Each slice is as follows: if the base contains data about a time interval of 10 years, it will be split in 10 slices.

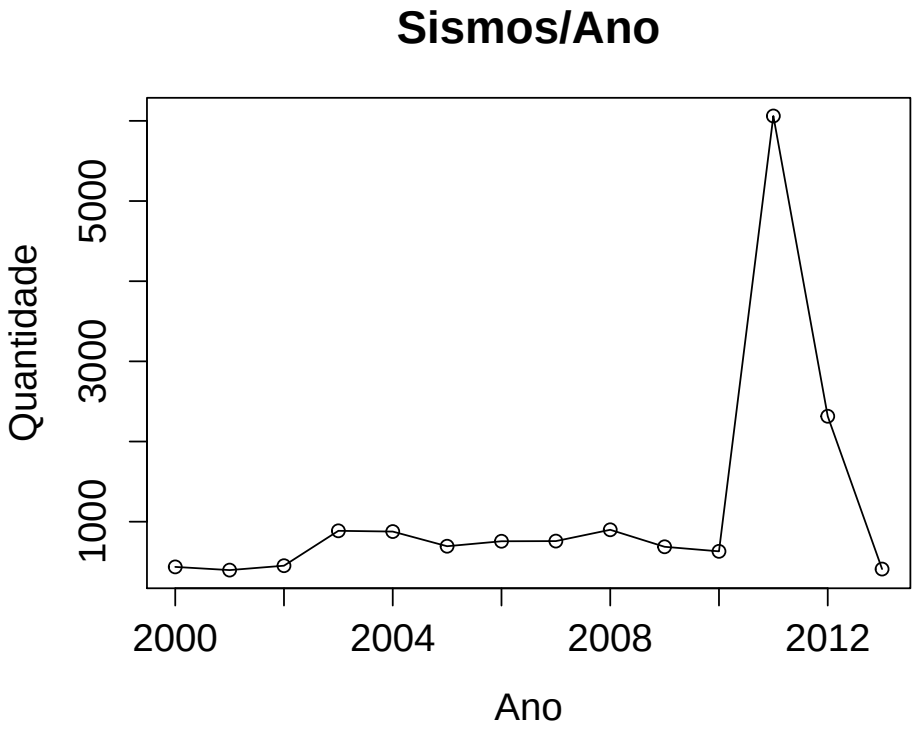

Figura 5.1: Amount of earthquake by year.

We also selected some sub-areas in Japan to better extract and understand earthquakes characteristics and patterns. Those areas are Kanto, Kansai, Touhoku and East Japan. The Figure 5.2 shows how we defined them. They are discribed as follows:

Kanto Kanto is the region around Tokyo. It is area with high simologic activity during the years we studied. Its coordenates are 34.8 North, 138.8 West, with 2025 bins. Each bin covers an area of approximately 25km^2.

Kansai Kansai is the region that includes Kyoto, Osaka and many others historical cities. In this area, rather than Kanto area, there is a small sismic activity. Its coordenates are 34 North, 134.5 West, with 1600 bins. Each bin covers an area of approximately 25km^2.

Touhoku Touhoku is the region in the North of the main japanese island. It has some clusters of sismic activities during the the years we studied. Its coordenates are 37.8 North, 139.8 West, with 800 bins. Each bin covers an area of approximately 100km^2.

East Japan Is the region that is related with the east coast of Japan. It is the most different area, because it has eathquakes that happend both in land or in the sea. It was in this region that the 9.0 M_w earthquake happened. Its coordenates are 37 North, 140 West, with 1600 bins. Each bin covers an area of approximately 100km^2.

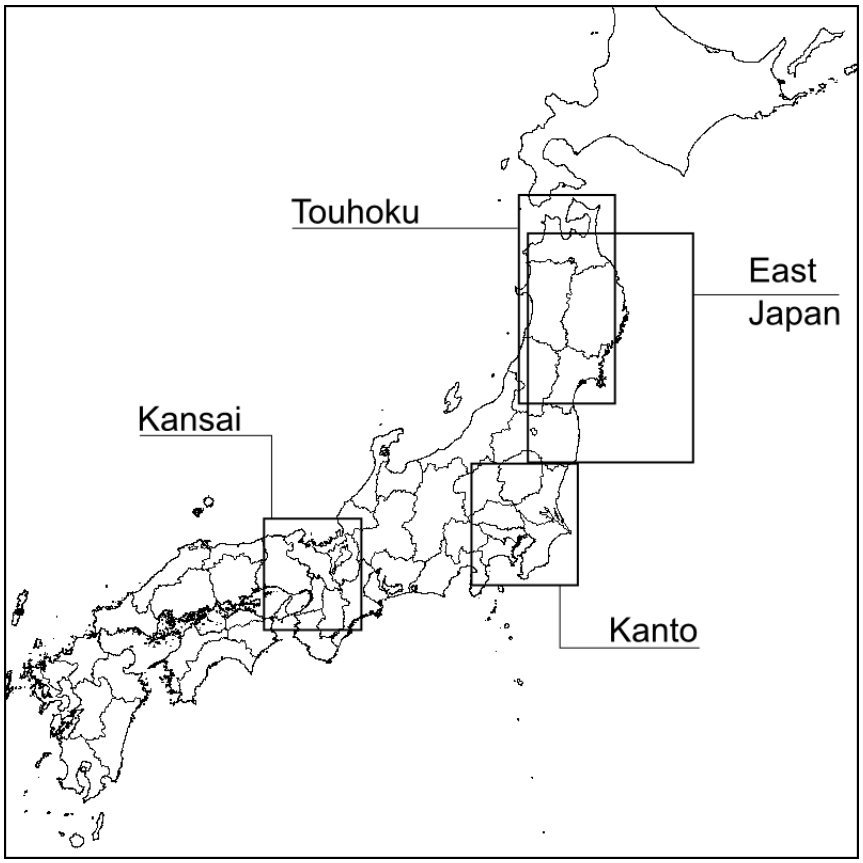

Figura 5.2: Japan and the areas used in this studied.

Capítulo 6

Metodologia Proposta

6.1 Genoma Representation

Each individual represents an earthquake risk model. It has a number of bins chose to fit the area of study and each bin has a probability value. The areas of study are: Kanto, Kansai, East Japan or Touhoku, see section 5. To obtain the number of earthquakes by the value of the bin, we use the modified Poisson distribution [24].

Algorithm 2 Obtaining values between $[0, 1)$ from the Poissonian curve.

Parameters $0 \leq x < 1, \mu \geq 0$
$L \leftarrow \exp(-\mu), k \leftarrow 0, prob \leftarrow 1$
repeat
  increment k
  $prob \leftarrow prob * x$
until $prob > L$
return k

The scale chosen for the bins of each area were defined by taking into account that the resolution of a prevision model is inversely proportional to the size of the bin. That means that the smaller the bin scale, the higher is the resolution of this model. Also, we decided that each bin would have a scale depeding of the area. Once both Kansai and Kanto regions cover an area of 25 Km^2 the scale of theirs bins is set to 0.05 degress. Once they cover a smaller area, when compared with the areas of Touhoku and East Japan, we definided, for Touhoku and East Japan, a scale of the bin a little bigger, being 0.1.

To accelerate the development of the prototype we used the *Distributed Evolutionary Algorithms in Python* - DEAP [3], a framework for Genetic Algorithms. Its design seeks to make algorithms explicit and data structures transparent. [3].

6.2 Simple L-test Fitness Function

We first intented to used the L-test, see 2.2.1, as fitness function. Once the main goal was to find if this fitness fuction would produce good results, we focus ours effort only in developing a application that would provides us fast results. Therefore, no study related with the GA paramerters and its values were made. We calculated the L-test the individuals and the one with the biggest value would be maintained into the next generation.

Once no effort was made to analyse the GA initial population and number of generations, we chose them by trial and error, until an acceptable convergency time were achieved.

We used all earthquake data available in Kanto, not taking into account the anual slices 5.

6.2.1 Crossover

The results from the executions of the Simple L-test Fitness Function were obtained by using two crossover operators the Blend and the Two Points operators.

As defined in [33], the Blend crossover is as follows:uinte forma:

1. Choose two parents, x^1 e x^2, at random.
2. A value for each element of the children x_i^c of the next population is randomly chosen of the interval $[X_i^1, X_i^2]$ of the following distribuition:

$$\begin{aligned} X_i^1 &= min(x_i^1, x_i^2) - \alpha d_i \\ X_i^2 &= max(x_i^1, x_i^2) + \alpha d_i \\ d_i &= |x_i^1, x_i^2| \end{aligned} \tag{6.1}$$

   where x_i^1 and x_i^2 are the i-th element of x_i^1 e x_i^2, respectively, and α is a positive parameter.

Herrera, Lozano and Verdegay [6] state that with an $\alpha = 0.5$ there is balance between exploitation and exploration. Based on this, the value used for α was set to 0.5.

the Two Points crossover was used to compared the effect between different kinds of crossover operators. As it is defined by Goldeberg, [**?**], it is considired a two-step operator. This crossover operator is an instance of the n-point crossover operator (where $n = 2$) which are a generalization of the simple crossover, [6].

The first step of the Two Points crossover is to choose two parents, at random. After that, each parent is splited into two parts, given a position k. This value is a number randomly chosen from a uniform distribuition, with the limits 1 to the maximum length of the individuals minus 1, $[1, l - 1]$. The children then is generating by swapping the data from the parents.

6.2.2 Mutation Operator

The mutation operator chosen was the FlipBit. It swaps the bit value from each element and uses a value between 0 to 1 to decide if an idividual will be altered by the it [5].

6.2.3 Selection Operator

For the selection operator we used the Simple Tournament and Elitism. The Simple Tournament was used to select individuals to participate on the reproduction, based on the fitness values they obtained. The elitism were used to impose that the best individual fromthe current generation would be part of the next generation, ensuring that the next generation has quality at least as good as the former one.

The parameters from the application with the Simple L-test are estão described in the next chart 6.1:

Population size	500
Number of generations	100
Blend crossver α	0.5
Mutation FlipBit	0.2
Probability of Mutation(indpb)	0.05
Tournament size	3

Tabela 6.1: Used Parameters.

6.3 Time-slice Log-Likelihood Fitness Function

After some analyses made on the last application, we realized that our model were overfitting with the data. Once this is not interesting, some changes were made.

First, we changed the fitness function from the L-test to the Log-likelihood value 2.1.2. We also started using the the anual slices, described in the chapter 5 for the region of Kanto.

6.3.1 Available Operators

In this case, a wider experiment was made. We compared all operators from the DEAP framework and selected the group that achieve greater results.

DEAP operators

The parameters used in Time-slice Log-Likelihood are the ones from the chart 6.2.

Popluation size	500
Numner of generations	100
Crossover operator	0.9
Mutation operator	0.1

Tabela 6.2: Used parameters.

The operators from DEAP are described as follows. All descriptions were obatined from the DEAP framework webapge [25].

Crossover The operators used were: *One Point*, *Uniform*, *Two Points*, *Partially Matched*, *Ordered* and *Simulated Binary*.

One Point It executes the one point crossover. Both parents are modified and the children have the same length as the parents.

Parameter (1)	An inidividual
Parameter (2)	An inidividual
Return	2 modified individuals

Tabela 6.3: Parameters and return of the One Point crossover

Uniform It executes the uniform crossover that modifies two parents. The elements are modified to a new value from an uniform distribuition according to a given probability, genneralt set to 0.5.

Parameter (1)	An inidividual
Parameter (2)	An inidividual
Parameter (3)	A probabilty value
Return	2 modified individuals

Tabela 6.4: Parameters and return of the Uniform crossover

Partially Matched It executes on the two parents a partially matched crossover. The individuals are created by matching the index pais of the parents given a interval and then exchanging their values.

Parameter (1)	An inidividual
Parameter (2)	An inidividual
Return	2 modified individuals

Tabela 6.5: Parameters and return of the Partially Matched crossover

Uniform and Partially Matched It executes a combination of the Uniform and Partially Matched crossovers. In follows the same behavior as the last one but index matching is done randomly as in the Uniform crossover.

Parameter (1)	An inidividual
Parameter (2)	An inidividual
Return	2 modified individuals

Tabela 6.6: Parameters and return of the Uniform and Partialy Matched crossover

Ordered Executes an ordered crossover on the two parents. It select a interval of indexes from a parent and swaps the values from this intervals with the values from the other parent the the same interval of indexes, in order.

Parameter (1)	An inidividual
Parameter (2)	An inidividual
Return	2 modified individuals

Tabela 6.7: Parameters and return of the Ordered

Ordęred

Simulated Binary Executes a crossover by a binary simulation. It receives two parents and a β value. With higher *beta* values the children are more similar to the parentsa nd with lower values, the children are less similar to the parents.

Mutation The mutation operators tested are: Shuffle Indexes andUniform Integer.

Shuffle Indexes It shuffles the indexes from the individual.

Parameter (1)	An inidividual
Parâmetro (2)	Probability of shuffling
Return	1 modified individuals

Tabela 6.8: Parameters and return of the Shuffle Indexes

Uniform Integer It substitutes the some elements of the individual by a value taken from an uniform distribuition in a given interval.

Parameter (1)	An inidividual
Parâmetro (2)	Probability of applying the mutation
Return	1 modified individuals

Tabela 6.9: Parameters and return of the Uniform Integer

Selection The selection operators tests are: Roulette, Random, Best and Worst.

Roulette It selects individuals by spins of a roulette. The selection is make by given priority to individuals with higher fitness value.

Parameter (1)	An list of inidividuals
Parameter (2)	A number of individual to select
Return	A list of individuals

Tabela 6.10: Parameters and return of the Roulette

Random It selects random individuals.

Parameter (1)	An list of inidividuals
Parameter (2)	A number of individual to select
Return	A list of individuals

Tabela 6.11: Parameters and return of the Random

Best It selects the best individuals.

Parameter (1)	An list of inidividuals
Parameter (2)	A number of individual to select
Return	A list of individuals

Tabela 6.12: Parameters and return of the Best

Worst It selects the worst individuals.

Parameter (1)	An list of inidividuals
Parameter (2)	A number of individual to select
Return	A list of individuals

Tabela 6.13: Parameters and return of the Worst

6.4 CEC'13 Functions

Before doing any experiments with the Log-Likehood fitness fuction on our GA application we used the CEC'13 functions. The goal here was to better understand the available operators in a optimization problem. For that, we chose to use the CEC'13 - Congress on Evolutionary Computation - suit testx. In there, there are 28 function, [16], of which 8 were considered as a group of interest.

After integrating the code made available from the CEC group with a GA application made with DEAP, we analysed the results, and against expected, these experiments did not indicated a group of operatores that lead to any better performance.

6.5 GA with Time-slice Log Likelihood Fitness Function

After the experiments with CEC'13 functions, we did some experiments within the context of our study. We splited our efforts into two directions: to test the GA with the Time-slice Log Likelihood Fitness Function and to test the GA with operators with adaptive weights.

The adaptive weights technique uses values that will vary during the an experiment. These values generally are adjusted to reflect the recent performance an operator. The major reason to apply this is because during an experiment, the influence of a operators may change and its value should also change.

Once the experiments with CEC'13 functions showed no difference between the groups of operators, we chose a group of operators arbitrarily. They were the Two Point for crossover, with 0.9 chance of occurance, the Shuffle Indexes for mutation, with 0.1, and the roulette for selection, also arbitrarily chosen. By the end of a execution of our application, the search space is already defined, meaning that we objective to specify the search. Therefore, we raised the chance of mutation operator to a higher value, 30% and lowered the chance of the crossover operator also by 30%. This was done gradually, with an equal variation per round given by the value = [30%/(numer of generations)].

By the time we analysed the resutlts from the former experiments, we realized that the code had some errors that needed to be refactored.

6.6 The GAModel Experiment

After reforming our GA application, we developed the GAModel. It is revised version and updated from our Ga application with the Time-slice Log-likelihood fitness function. Once it is viable to use this version for new experiments. We create some scenarios to compare our model with the RI - Relative Intensity Algorithm.

These scenarios are defined as space/time regions. Each scenario contains the earthquakes for the Kanto region for a given year.

As being a stochastic method we used the Power Test and estimated the number of repetitions, n, needed for detecting a significant variation. For the *Power Test* we used a pilot experiment to estimate n.

An exemple of the *Power t-test*:

Year	2006
Delta	50 (for all scenario)
Standard Deviation	25.16513
Degree of confidence	0.05
power	0.95
alternative	two.sided
n:	5.58517

Tabela 6.14: Power t-test

The pilot experiments used 10 observations for calculating the *Power Test*. The *delta* and *power* values were chosen based on the results of the pilot experiment. The standard deviation is the same as the observations. All results indicate that 10 repetitions are enough to compare the GAModel via *Student's t-test*.

We also needed the log-likelihood of the RI method. As being a deterministic method, 1 observation for each scenario is enough. This value is used as the target for the *Student's t-test*. For control, we used a random Poisson method with no data awareness.

6.6.1 The Relative Intensity Algorithm

The RI *Relative Intensity* (RI) algorithm is frequently used as reference for comparing methods [20]. The main idea behind the RI is that larger earthquakes are more likely to

occur at locations of high seismology in the past.

The log-likelihood data for the RI for each scenario were given by Aranha, [1].

6.6.2 Model Examples X Real data

The Figure 6.1 shows a model from the GAModel method for the year 2010. It indicates a low earthquake intensity as green while the more intensity areas, are shown as orange (for even higher cases, white is used). The Figure 6.2 shows a model from the RI Algorithm for the year 2010. It indicates a low earthquake intensity as white while the more intensity areas, are shown in red. For comparison reasons, we show the Figure 6.3, that shows the earthquake occurances for the same year.

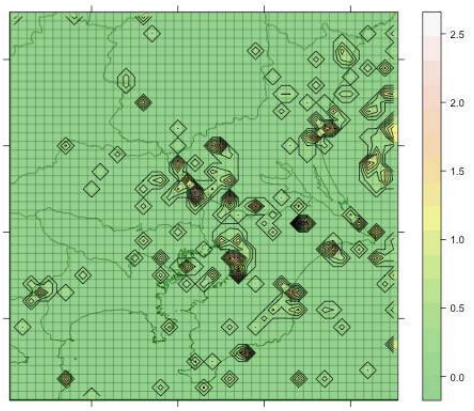

Figura 6.1: GAModel model for the year of 2010.

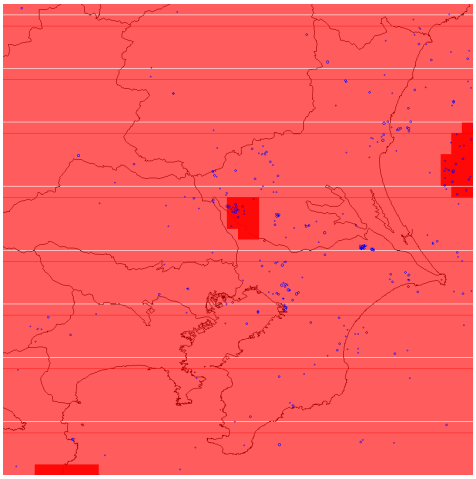

Figura 6.2: RI model for the year of 2010. Figure from [1].

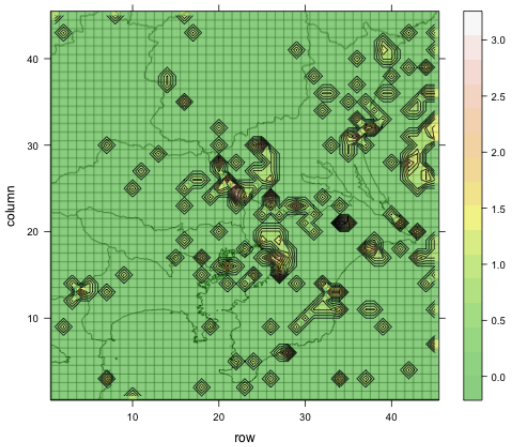

Figura 6.3: Earthquake occurances for the year of 2010

6.7 GAModel X RI Algorithm

We compared the method proposed, ReducedGAModel, 4.2.2, with the non-variated method, GAModel 4.2.1, and with the Relative Intensity Algorithm, 6.6.1, using its log-likelihood values. The values are comparece via Student's t-test, so we could understand the if there is any statistic significant difference between the methods. The data used was JMA catalog.

6.7.1 Hypothesis

There are 3 tests hypothesis for this experiment that we want to analyse.

The first is if the mean values of the log-likelihood for the ReducedGA are equal to the RI values.

$$\begin{cases} H_0 : \mu = RIlog - likelihoodvalue \\ H_1 : \mu! = RIlog - likelihoodvalue \end{cases}$$

The second is if the mean values of the log-likelihood for the ReducedGA are equal to the GAModel values.

$$\begin{cases} H_0 : \mu = GAModellog - likelihoodvalue \\ H_1 : \mu! = GAModellog - likelihoodvalue \end{cases}$$

And the last hypotheses is if the mean values of the log-likelihood for the GAModel are equal to the RI values.

$$\begin{cases} H_0 : \mu = RIlog - likelihoodvalue \\ H_1 : \mu! = RIlog - likelihood_value \end{cases}$$

6.7.2 Results

The results of the experiments are in the Table 6.18 and in the box-plots Figure 6.4 and Figure 6.5. The column labeled Random shows the result of the RandomModel, and the idea is the same for the ones labeled GAModel and RI. The "p-value" shows the significance value of the *Student's t-test* for the null hypothesis " The mean of the log-likelihood values is greater that the values for RI".

As expected, the RandomModel has lower values than the GAModel. When compared with the RI, the results show that the GAModel is competitive with the RI and that it is promising to use GA to generate earthquake forecasts.

Scenario	Log Likelihood				
Year	Random	RI	GAModel		p-value
2000	-2413.89	-2124.44	-2094.05	(8.80)	0.01
2001	-2418.14	-2103.19	-2101.65	(69.49)	0.57
2002	-2385.04	-2094.43	-2100.01	(72.62)	0.07
2003	-2401.00	-2104.65	-2100.76	(156)	0.35
2004	-2421.92	-2101.92	-2098.30	(55.28)	0.16
2005	-2643.38	-2248.40	-2114.00	(779)	0.01
2006	-2616.50	-2226.93	-2115.6	(633)	0.01
2007	-2451.68	-2109.13	-2122.03	(615)	0.13
2008	-2433.23	-2112.92	-4435.34	(657)	0.14
2009	-2884.74	-2438.10	-2113.1	(814)	0.01
2010	-2418.18	-2114.60	-2112.07	(843)	0.79

Tabela 6.15: Experiments result.

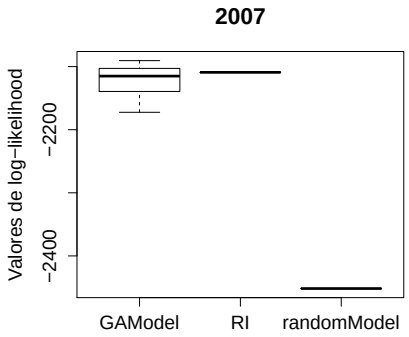

Figura 6.4: Box-plot of the values obtained by the models for the year 2007.

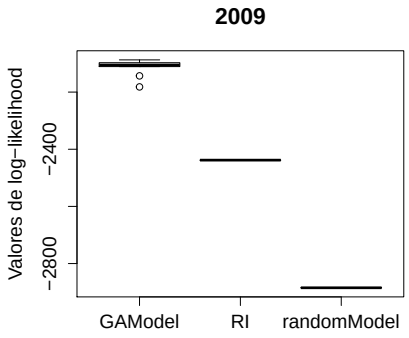

Figura 6.5: Box-plot of the values obtained by the models for the year 2009.

6.8 All Models Experiments

6.8.1 A Brief Recapitulation of the Models

Based on our promissing results and because we aim to improve them, we developed the ReducedGAModel. It a simplified version of the GAModel. We want to compare the behavior of this new method against the GAModel method.

We also wanted to explore how adding domain knowledge would improve the avarage performance of the GAModel or the ReducedGAMOdel. They are versions of GAModel/ReducedGAModel combined with emperical laws.

6.8.2 The Catalogs

The data used was JMA catalog and a version of a cluster catalog.
(JMA X métodoJanelaJMA=>clustered). How do I explain this? The EXPLANATION SHOULD BE IN THE FORMER CHAPTER here is just a recap

6.8.3 The Experiment

For this new experiment, we used even more scenarios (space/time regions) than the ones. Each scenario contains the earthquakes for the regions of Kanto, Kansai, Touhoku and East Japan for a given year (2005-2010). We wanted to explore if there exists any influence in the performance of the models that are caused by the depth of an earthquake. So, the scenarios are also composed by introducing a 3 groups depths thresholds. They are: of earthquakes with depth smaller than 25km, or between 0km and 60km or even between 0km and 100km.

These method are stochastic method and hence are variations of the GAModel, we decided to maintain the number of repetitions without redoing a Power Test.

6.8.4 ANOVA test and HSD Tukey

The goal here is to find if there is any variation between the methods and which are the most influencial variables. For achieve that, we will use the ANOVA Test.

In the ANOVA test, if a variable is out of the 5% confidence interval, with $P < 0.05$ it means that there exists a statistical significant different for that variable.

There are some tests hypothesis for this experiment that we want to analyse. They all can be generalized as follows:

$$\begin{cases} H_0 : \text{The population means are equal populacionais são iguais.} \\ H_1 : \text{The population means are different.} \end{cases}$$

Then we apply a post hoc test, the HSD Tukey test, on the results obtained from the ANOVA test to specify which groups differ.

6.8.5 Results

A one-way between subjects ANOVA was conducted to compare the effects of the models, the depths, the years and regions on the log-likelihood value. In this study there are 6 options for model: lista, gaModel, hybridgaModel, hybridlist, gaModelCluster and listaCluster. Based on the results of the test, there was a not a significant effect of the depths or years variables. For both cases at the we obtaind p>0.05 level for the depths condition [F(2) = 2.072, p = 0.126] and we also obtained p>0.05 for the years condition [F(5) = 0.050, p = 0.999]. There was a significant effect of the models condition (p>0.05 [F(5) = 9699.690, p<2e-16]) and regions condition (p>0.05 [F(3) = 764.220, p<2e-16]). Therefore, we conduct a new anova test, with only the last two variables to verify the influence of those conditions more accurately. The results only changed a little, maintaining the significant effect of both conditions, p>0.05 [F(5) = 9705.6, p<2e-16] and p>0.05 [F(3) = 764.7, p <2e-16], respectively.

The results of the experiments are in the Table 6.17.

The column labeled Random shows the result of the RandomModel, and the idea the is same for the ones labeled GAModel and RI. means of several groups are equal,e t-test to more than two groups. The “p-value” shows significance value of the *Student's t-test* for null hypothesis “ The mean of the log-likelihood values is greater that the values for RI”.

Because we found statistically significant result, we applied a Post hoc comparisons using the Tukey HSD test. It compared each condition with all others. For example, it

Variable	Degrees of Freedom	Sum Sq	Mean Sq	F Value	Pr(>F)
Model	31424058	6284812	6284812	255.0	<2e-16
Depth	2	16077491	8038746	326.2	<2e-16
Year	5	57908014	11581603	470.0	<2e-16
Region	3	878253346	292751115	11879.4	<2e-16

Tabela 6.16: ANOVA test results.

compares the values from the gaModel with the gaModelClustered, see 6.6. It indicated that the gaModelCluster and ReducedGAModelCluster, when comparared with all other models, achieve greater log-likelihood values. Furthermore, we noticed that the depths conditions show a greater influence when the depth in smaller or equal to 25 km, see Figure 6.7

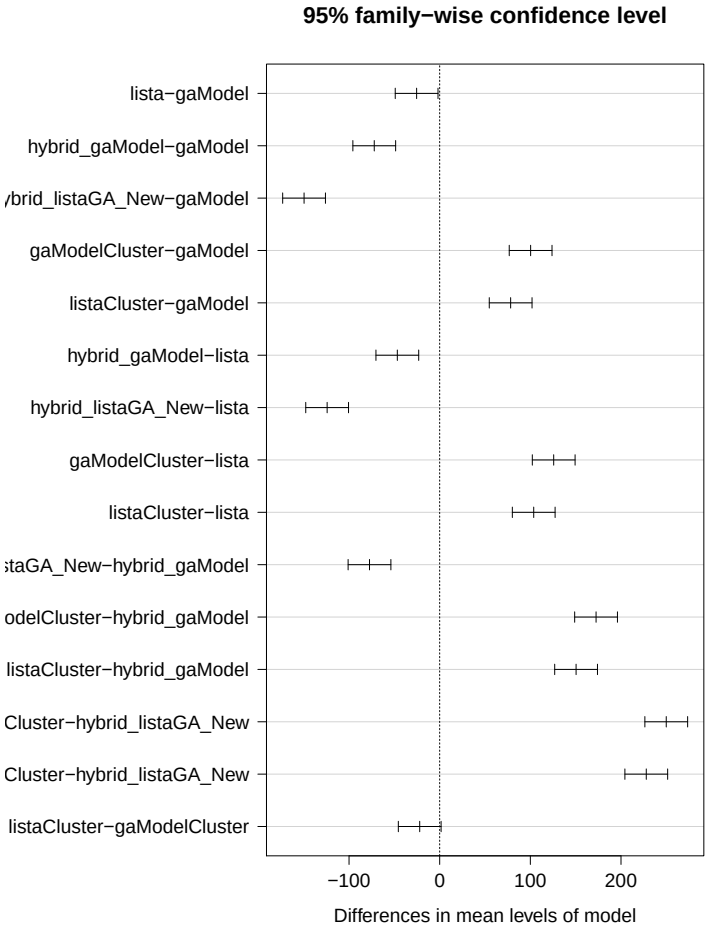

Figura 6.6: .

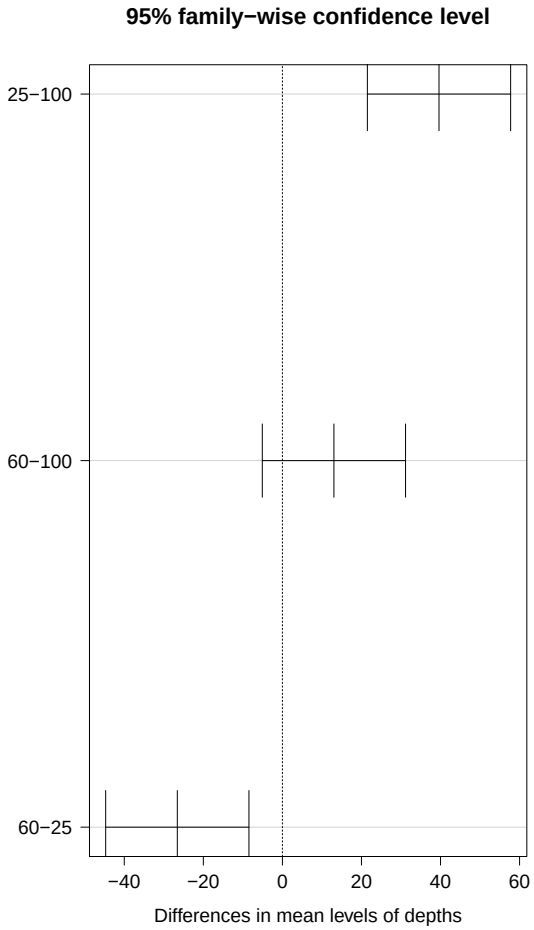

Figura 6.7: .

When comparing the models from the ReducedGAModel and from the gaModel against themselves, with or without using clustering catalogs, we found that there is no statistically significant result between the methods. That implies it can be considered that the methods are obtain statistically equal results.

Therefore, based on the result of HSD test, we performed a new AVOVA test, considering only the gaModelClustered and the listaClustered. That was meant not only to verify the previous results but also to certify if the depth influence is preserved.

Variable	Degrees of Freedom	Sum Sq	Mean Sq	F Value	Pr(>F)
Model	1	174862	174862	12.22	<0.000488
Depth	2	391370	195685	13.67	<1.32e-06
Year	5	18810831	3762166	262.82	<2e-16
Region	3	249741769	83247256	5815.53	<2e-16

Tabela 6.17: ANOVA test results.

Taken together, these results suggest that the using cluster and depth smaller or equal to 25km, see Figure 6.8 showed the best results.

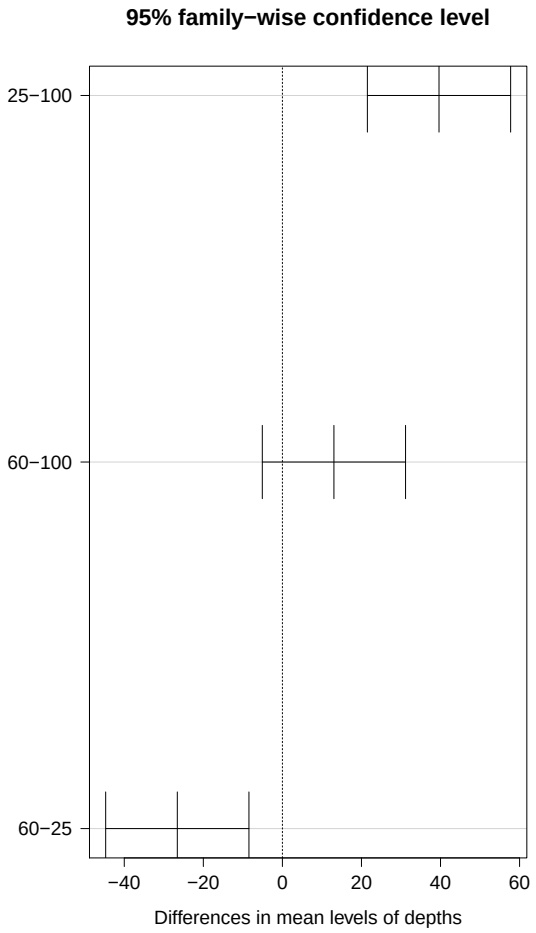

Figura 6.8: .

6.9 Paired Design

To further explore the only the variations from the models on the regions, disconsidering any other variable, we applied an paired Student's t-test. tem que fazer isso pra kanto, kansai, touhoku e EastJapan?

Scenario	Log Likelihood				
Year	Random	RI	GAModel		p-value
2000	-2413.89	-2124.44	-2094.05	(8.80)	0.01
2001	-2418.14	-2103.19	-2101.65	(69.49)	0.57
2002	-2385.04	-2094.43	-2100.01	(72.62)	0.07
2003	-2401.00	-2104.65	-2100.76	(156)	0.35
2004	-2421.92	-2101.92	-2098.30	(55.28)	0.16
2005	-2643.38	-2248.40	-2114.00	(779)	0.01
2006	-2616.50	-2226.93	-2115.6	(633)	0.01
2007	-2451.68	-2109.13	-2122.03	(615)	0.13
2008	-2433.23	-2112.92	-4435.34	(657)	0.14
2009	-2884.74	-2438.10	-2113.1	(814)	0.01
2010	-2418.18	-2114.60	-2112.07	(843)	0.79

Tabela 6.18: Experiments result.

Capítulo 7

Resultados Obtidos

7.1 Simple L-test Fitness Function

O operador escolhido como parâmetro de *crossover* foi o *Blend*, e a preferência por ele dentre os demais implementados pelo pacote DEAP foi pela observação empírica de crescimento dos valores de L-test e pelo fato de o operador ser específico para indivíduos formados por números reais, situação encontrada na aplicação. A média do resultado de 10 execuções:

Primeira geração (s)	145.6644
Última geração (s)	113.4796
Valor do L-test da primeira geração	-0.635949214
Valor do L-test da última geração	-0.009772415

Tabela 7.1: Tempo gasto e valor do L-test na média de 10 execuções com Blend.

Operador de *crossover Two Points* foi utilizado por motivos de comparação. A média do resultado de 10 execuções:

Primeira geração (s)	129.03775
Última geração (s)	97.00832
Valor do L-test da primeira geração	-0.635856248
Valor do L-test da última geração	0.042102222

Tabela 7.2: Tempo gasto e valor do L-test na média de 10 execuções com Two Points.

Tanto o modelo preferido quanto o modelo comparativo foram capazes de evoluírem, obtendo valores finais médios superiores aos valores médios aleatórios iniciais.

Surpreendemente, por [9], era esperado que representações em ponto flutuante tivessem um desempenho de maior acurácia. Porém, o desempenho do operador específico para números reais, o *Blend*, foi menor quando comparado a um operador de uso mais geral, o *Two Points*, sendo mais lento e obtendo valores de L-test menores, na média.

Todos os dados foram obtidos após a execução do algoritmo em um computador Apple MacBook Pro com processador 2.9 GHz Intel Core i7, memória RAM 8 GB 1600 MHz DDR3 com sistema operacional OS X 10.9.1 (13B42).

Para melhor visualização e aumentar o poder de comparação, quatro figuras podem ser analisadas a seguir. As Figuras **??** e **??** mostram as médias do valores do L-test para a todas as populações enquanto que as figuras **??** e **??** mostram as médias valores do tempo também para todas as populações. Figuras **??** e **??** são referentes a execuções com o *crossover Blend* e as figuras**??** e **??**, com o *crossover Two Points*.

Por observação empírica das diversas execuções realizadas o maior tempo gasto é com cálculos de L-test, sendo influenciado principalmente pela quantidade de observações e pelo tamanho escolhido para *bins*. Quanto menor for o tamanho escolhido para o *bin* maior será a resolução do terreno analisado e, conseqüentemente, mais informações sobre ele teremos e maior será o espaço de busca, aumentando o tempo total gasto.

Há um grande aumento do valor do L-test entre a vigésima geração e a quadragésima geração. Isso significa que a função de *fitness* resulta em um excessivo valor de *overfitting*, um super ajuste a base de dados.

7.2 Time-slice Log Likelihood Fitness Function

Os resultados da aplicação GA it Log Likelihood Fitness Function serão demonstrados a seguir.

Foram feitas comparações entre os resultados dos valores de *fitness* dos melhores indivíduos e o desvio padrão da população a que ele pertence. Esses resultados comparados são referentes ao estudo dos grupos de operadores e a técnica de pesos adaptativos. A

seguir, além dos resultados obtidos, algumas Figuras utilizadas para comparações serão mostradas.

As Figuras **??** e **??** demonstram o resultado das 50 execuções, mostrando dados referentes a execuções do conjunto de operadores *One Point, Worst* e *Shuffle Indexes* para os anos de 2000 (quando há um ganho de desempenho ao utilizar-se a tabela) e de 2010 (quando há uma perda de desempenho). As Figuras mostram o melhor indivíduo e o desvio padrão de sua população. Está claro que existe os resultados não são consistentes e que existe alguma falha na aplicação.

Essa falha ficou clara após a mudança do cálculo de fatorial pela tabela em memória. Até onde x = 27 (x se refere ao número da execução), os valores obtidos vieram de cálculos do fatorial, a partir disso, foi utilizada a tabela. Porém, devido a essa falha, não é possível qualificar se algum grupo gerou indivíduos mais aptos. Portanto é essencial corrigi-la para novas execuções sejam geradas e as comparações possam ser refeitas.

Foram duas as causas levantadas para essa falha. A primeira, refere-se ao valor limitante para a tabela do fatorial. Uma vez que a tabela possui valores até o fatorial de 100, quaisquer valores acima desse limitante terão seus fatoriais arredondados a 100. Há casos que essa aproximação foi vantajosa e a aplicação respondeu com modelos de melhor desempenho, mas em alguns casos, a aproximação foi desvantajosa.

A segunda, refere-se a um erro no código da execução. Após uma análise superficial do código, é provável que o mapeamento dos dados coletados esteja propagando algum erro. Esforços futuros serão direcionados para a correção deste erro.

Ainda assim, a tabela em memória do valores do fatorial resultou em um avanço em termos de performance temporal. Uma vez que a segunda possível causa seja o verdadeiro erro da aplicação, acredita-se que o uso da tabela será capaz de facilitar futuras execuções.

Os resultados obtidos pela técnica de pesos adaptativos não foram tão elevadas quanto o esperado. Para efeitos comparativos dois grupos de figuras são mostrados a seguir. As Figuras **??** e **??** mostram os valores dos melhores indivíduos e o desvio padrão da população para o ano de 2000 e compõem o primeiro grupo, já o segundo grupo é composto pelas figuras **??** e **??** e segue o mesmo princípio, porém para o ano de 2010. No primeiro grupo, a Figura **??** refere-se aos dados da técnica em questão, enquanto que a outra Figura refere-se ao mesmo conjunto de operadores utilizados (*crossover*, *Two Points* e

Shuffle Indexes), mas com pesos fixos. O segundo grupo é descrito da mesma maneira.

Não fica claro, pelos dados mostrados, quais são os benefícios que a técnica introduz para a aplicação em questão. Após os mais diversos tipos de análises, análises sobre a influência dos grupos de operadores, seja na própria aplicação GA ou mesmo nas funções da CEC'13 e análises sobre diferentes técnicas para estruturação da abordagem GA, não ficou claro qualquer tipo de variação substancial capaz de direcionar os estudos e definir uma abordagem única para os experimentos. Portanto, deve-se fazer a devida ponderação: será que as variações realmente influem pouco nos resultados da aplicação, ou a aplicação deve passar por refinamentos, que levem a resultados mais consistentes?

Para responder essa ponderação devemos analisar as escolhas feitas para a criação de nosso modelo. Pela própria estrutura definida pelo modelo proposto, cada *bin* é tratado independentemente dos seus vizinhos, não considerando a influência de sismos próximos. Porém, pela características inerentes aos sismos é claro que poucos são os casos de sismos completamente independentes entre si. Podemos considerar, logo, que a influência da independência influi consideravelmente e é ela a principal causa da mínima variação pré-citada.

Por outro lado, as funções CEC'13 são funções unicamente matemáticas e por isso a questão levantada anteriormente não é suficiente para sugerir que a resposta a nossa ponderação deva ser direcionada somente para o segundo ponto da pergunta, "a aplicação deve passar por refinamentos". Devemos deixar claro, entretanto, que essa independência entre os *bins* deve ser melhor explorada a fim de encontrar uma solução que possibilite considerar a dependência dos sismos. Nesse ponto, percebemos que a complexidade da questão é elevada.

Apesar da utilização da técnica de pesos adaptativos não ter contribuído para uma performance mais significativa, fomos capazes de observar que os valores médios do melhores indivíduos durante os anos foram muito mais coesos do que quando comparados aos valores similares dos ensaios com pesos fixos. Isso nos leva a crer que utilizar a técnica indiretamente influiu positivamente em nossos resultados. Consequentemente, a idéia de abranger uma área maior de busca e posteriormente especifica-la, mostrou-se, como esperado, bastante adequada.

Ainda em relação a técnica anterior, é possível que outras contribuições não tenham sido percebidas e que, de acordo com as devidas mudanças, sejamos capazes de percebê-las.

Essas mudanças tanto podem ser algumas das já citadas como também na porcentagem de variação dos pesos dos operadores, que é de 30% (escolhido arbitrariamente), nos valores desses pesos na situação inicial, entre outras, ou seja, variações na estrutura utilizada juntamente com a técnica.

There are 3 tests hypotheses for this experiment that we would like to check.

The first is if the mean values of the log-likelihood for the ReducedGA are equal to the RI values.

$$\begin{cases} H_0 : \mu = RI_log - likelihood_value \\ H_1 : \mu! = RI_log - likelihood_value \end{cases}$$

The second is if the mean values of the log-likelihood for the ReducedGA are equal to the GAModel values.

$$\begin{cases} H_0 : \mu = GAModel_log - likelihood_value \\ H_1 : \mu! = GAModel_log - likelihood_value \end{cases}$$

And the last hypotheses is if the mean values of the log-likelihood for the GAModel are equal to the RI values.

$$\begin{cases} H_0 : \mu = RI_log - likelihood_value \\ H_1 : \mu! = RI_log - likelihood_value \end{cases}$$

Summary

O objeto é descobrir se existem variações ente os métodos e quais são as variáveis mais influentes.

STAtistical Analysis ANOVA test and HSD Tukey

Vou utilizar o ANOVA para nos dados obtidos para verificar qual composição de variáveis e métodos mais influênciam no resultado final.

Aplico um teste post hoc nos resultados do ANOVO oara especificar quais são os grupos que diferem. O teste utilizado foi o Tukey teste.

É importante resaltar que para todos os casos, aplico uma função de limite, que altera os valores do bins com mais que 12 ocorrências para 12.

Começo a análise carregando o data.frame com os dados, seguindo para a aplicação do teste ANOVA e finalizando com o uso do Tukey teste.

Faço o ANOVA somente para os modelos "clusterizados"

Aplico o anova, com a regressão para modelos, profundidades, anos e regiões.

Agora faço o Paired Design t.test aplicando para todas as combinações possíveis de modelos, em todas as regiões e profundidades, para todos os anos.

Baseado nos arquivos que explicam o Paired Desing, escrevi o código a seguir. Porém não entendi porque ao fazer desta forma pode ser considerado um teste pareado. Os slides comparam duas formas de realizar este tipo de teste. Uma delas tem *seta* um parametro da função com **True**, explicitando que é um teste pareado. Já para o outra forma, esse parametro fica com **False**.

When comparing the models from the lista method and from the gaModel against themselves, with or without using clustering techniques, we found that there is no statistically significant result between the methods. That implies that it can be considered that the methods are obtain statistically equal results.

Therefore, based on the result of the HSD test, we performed a new AVOVA test, considering only the gaModelClustered and the listaClustered. That was meant not only to verify the previous results but also to certify if the depth influence is preserved.

Taken together, these results suggest that the using cluster and depth smaller or equal to 25km showed the best results.

Capítulo 8

Conclusão

Este projeto apresentou uma proposta de desenvolvimento de um modelo de previsão de probabilidades elaborado a partir de uma implementação simples de algoritmos genéticos. Foi possível perceber uma evolução do modelo em relação ao modelo completamente aleatório, caracterizado pela população inicial.

Foram muitos os desafios enfrentados ao utilizar os testes propostos pelo RELM como função de *fitness* pelo algoritmo genético. O primeiro está relacionada ao tempo de execução e o uso desses a cada geração como função de *fitness*. Por se tratar de testes com uma grande quantidade de cálculos em grande quantidades de observações, o tempo gasto para analisar as informações poderia ser demasiado grande para ser viável continuar a utilizar os testes junto ao algoritmo. O segundo problema estava vinculado ao comportamento desses testes em uma aplicação de algoritmos genéticos, pois não havia conhecimento anterior sobre a aplicabilidade desses com Computação Evolutiva e se aplicável, se o resultado seria promissor.

Pelos resultados finais do L-test, foi verificado que tanto o primeiro desafio quanto o segundo, não foram suficientes para impossibilitar o uso dos testes e que as gerações de populações resultaram em indivíduos mais aptos para a previsão de sismos. Pelos resultados mostrados o trabalho mostra que aplicar Computação Evolutiva para prever ocorrências de sismos é minimamente promissor, uma vez que há uma evolução do modelo em relação ao modelo pseudo-aleatório.

Foi possível, portanto, vincular os testes proposto pelo RELM com algoritmos genéticos. A partir disso, agora, deve-se explorar as características da aplicação, tornar os testes implementados mais completos e estruturados, definir os operadores genéticos visando um comportamento adequado em relação aos indivíduos como para as populações evoluídas.

Posteriormente, o primeiro desafio foi minimizado, pelo uso de uma tabela em memória do fatorial. Alguns estudos deverão ser realizados para compreendermos melhor o porquê da alteração dos valores após a introdução da tabela.

O uso de operadores de peso adaptativos mostrou-se interessante e é provável que, sua inclusão na GA estudada traga ainda mais benefícios assim que outros problemas forem resolvidos. Problemas, tais quais, a independência dos *bins*, que influenciaram negativamente a performance da aplicação e serão alvos de maiores esforços.

Algumas propostas de melhorias e trabalhos futuros podem ser listadas:

• Compará-lo com um outro modelo de previsão de terremotos, como por exemplo, o *Relative Intensity* (RI). Para melhores efeitos de comparação, é interessante acrescentar cálculos de confiabilidade estatísca, como *p-value*, e criar um mapa demonstrativo da previsão relativa de cada área representada pelos *bins*;

• Implementar o operador de mutação específico para números reais e que tenha um comportamento direcionado a aplicação, capaz de alterar seus valores no decorrer das gerações capaz de equilibrar *explotaition* e *exploration*;

• Aplicar alguns testes sugeridos por Zechar[**?**] afim de definir a qualidade do modelo desenvolvido;

• Fazer analises de complexidade do algoritmo executado;

• Criar múltiplas observações para cada indivíduo para que o cálculo de incertezas possa ser aplicado e, assim, aumentarmos a qualidade dos modelos;

• Realizar experimentos em outras áreas do Japão, além de Kanto;

• Especializar a abordagem de algoritmo genético (*crossover* mais apropriado, mutação específica, ...) ou utilizar soluções híbridas, entre algoritmo genéticos e outras técnicas de aprendizado de máquina;

• Incrementar os cálculos estatísticos com as ferramentas adequadas, desvio padrão, variância, etc, que compõe os cálculos dos testes demonstrados;

• A fim de evitar o *over fitting*, podemos inserir dados sismológicos, como sobre de magnitude e hora da ocorrência, aumentando a área de busca da aplicação;

• Cada *bin* é trato e considerado individualmente. Portanto a ocorrência de sismos em um *bin* não influência ocorrências de sismos em seus vizinhos, o que sabemos não ser verdade;

• Contrariando expectativas, execuções com operadores reais obtiveram desempenho pior que com operadores tradicionais pode estar relacionado ao fato de não existir influência entre os *bins*, uma vez que pode haver áreas de altíssima probabilidade próximas a áreas de baixa probabilidade.

Referências

[1] Claus Aranha, Yuri Cossich Lavinas, Marcelo Ladeira, e Bogdan Enescu. Is it possible to generate good earthquake risk models using genetic algorithms? In *Proceedings of the International Conference on Evolutionary Computation Theory and Applications*, pages 49–58, 2014. xi, 3, 5, 15, 16, 17, 37, 38

[2] Ali Firat Cabalar e Abdulkadir Cevik. Genetic programming-based attenuation relationship: An application of recent earthquakes in turkey. *Computers and Geosciences*, 35:1884–1896, October 2009. 13

[3] François-Michel De Rainville, Félix-Antoine Fortin, Marc-André Gardner, Marc Parizeau, e Christian Gagné. Deap: A python framework for evolutionary algorithms. In *Proceedings of the Fourteenth International Conference on Genetic and Evolutionary Computation Conference Companion*, GECCO Companion '12, pages 85–92, New York, NY, USA, 2012. ACM. 18, 28

[4] David Eberhard. Multiscale seismicity analysis and forecasting: Examples from the western pacific and iceland. 2014. 3, 6, 7

[5] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989. 30

[6] Francisco Herrera, Manuel Lozano, e Jose L. Verdegay. Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. *Artificial intelligence review*, 12(4):265–319, 1998. 29, 30

[7] A. M. Huda e Bagus Santosa. Subsurface structure in japan based on p and s waves travel time analysis using genetic algorithm in japan seismological network. *International Journal of Science and Engineering*, 6(1), 2014. 14

[8] T Serkan Irmak, Bülent Doğan, e Ahmet Karakaş. Source mechanism of the 23 october, 2011, van (turkey) earthquake (m w= 7.1) and aftershocks with its tectonic implications. *Earth, Planets and Space*, 64(11):991–1003, 2012. 2

[9] Cezary Z Janikow e Zbigniew Michalewicz. An experimental comparison of binary and floating point representations in genetic algorithms. In *ICGA*, pages 31–36, 1991. 48

[10] B. L. N. Kennet e M. S. Sambridge. Earthquake location — genetic algorithms for teleseisms. *Physics of the Earth and Planetary Interiors*, 75(1–3):103–110, December 1992. 13

[11] Tienfuan Kerh, David Gunaratnam, e Yaling Chan. Neural computing with genetic algorithm in evaluating potentially hazardous metropolitan areas result from earthquake. *Neural Comput. Appl.*, 19(4):521–529, June 2010. 13

[12] Tienfuan Kerh, Yu-Hsiang Su, e Ayman Mosallam. Incorporating global search capability of a genetic algorithm into neural computing to model seismic records and soil test data. *Neural Computing and Applications*, pages 1–12, 2015. 13

[13] E. Kermani, Y. Jafarian, e M. H. Baziar. New predictive models for the v_{max}/a_{max} ratio of strong ground motions using genetic programming. *International Journal of Civil Engineering*, 7(4):236–247, December 2009. 13

[14] John R Koza, Martin A Keane, e Matthew J Streeter. Genetic programming's human-competitive results. *IEEE Intelligent Systems*, pages 25–31, 2003. 11

[15] John R. Koza, Martin A. Keane, e Matthew J. Streeter. What's ai done for me lately? genetic programming's human-competitive results. *IEEE Intelligent Systems*, 18(3):25–31, 2003. 2

[16] JJ Liang, BY Qu, PN Suganthan, e Alfredo G Hernández-Díaz. Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization. *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report*, 201212, 2013. 35

[17] Zbigniew Michalewicz. Heuristic methods for evolutionary computation techniques. *Journal of Heuristics*, 1(2):177–206, 1996. 2

[18] D. Michie, D. J. Spiegelhalter, e C.C. Taylor. Machine learning, neural and statistical classification, 1994. 2

[19] Nobuo Mimura, Kazuya Yasuhara, Seiki Kawagoe, Hiromune Yokoki, e So Kazama. Damage from the great east japan earthquake and tsunami-a quick report. *Mitigation and Adaptation Strategies for Global Change*, 16(7):803–818, 2011. 1

[20] K. Z. Nanjo. Earthquake forecasts for the csep japan experiment based on the ri algorithm. *Earth Planets Space*, 63:261–274, 2011. 36

[21] Ahmad Nicknam, Reza Abbasnia, Yasser Eslamian, Mohsen Bozorgnasab, e Ehsan Adeli Mosabbeb. Source parameters estimation of 2003 bam earthquake mw 6.5 using empirical green's function method, based on an evolutionary approach. *J. Earth Syst. Sci.*, 119(3):383–396, June 2010. 13

[22] Yosihiko Ogata e Jiancang Zhuang. Space–time etas models and an improved extension. *Tectonophysics*, 413(1):13–23, 2006. 22

[23] Fusakichi Omori. On the after-shocks of earthquakes. 1895. 22

[24] William H. Press, Saul A. Teukolsky, Willian T. Vetterling, e Brian P. Flannery. *Numerical Recipes, The Art of Scientific Computing*. Cambridge University Press, third edition, 2007. 28

[25] Deap Project. Evolutionary tool. `http://deap.gel.ulaval.ca/doc/default/api/tools.html#deap.tools.mutPolynomialBounded`, July 2015. [Online; acessado: 08-07-2015]. 31

[26] Josafath I. Espinosa Ramos e Roberto A. Vázques. Locating seismic-sense stations through genetic algorithms. In *Proceedings of the GECCO'11*, pages 941–948, Dublin, Ireland, July 2011. ACM. 14

[27] Negar Sadat, Soleimani Zakeri, e Saeid Pashazadeh. Application of neural network based on genetic algorithm in predicting magnitude of earthquake in north tabriz fault (nw iran). *Current Science (00113891)*, 109(9), 2015. 12

[28] A. Saegusa. Japan tries to understand quakes, not predict them. Nature 397, 284, 1999. 3

[29] Bahram Saeidian, Mohammad Saadi Mesgari, e Mostafa Ghodousi. Evaluation and comparison of genetic algorithm and bees algorithm for location-allocation of earthquake relief centers. *International Journal of Disaster Risk Reduction*, 2016. 14

[30] D. Schorlemmer, M. Gerstenberger, S. Wiemer, D. Jackson, e D. A. Rhoades. Earthquake likelihood model testing. *Seismological Research Letters*, 78(1):17–29, 2007. 5, 6, 7, 8, 10

[31] Danijel Schorlemmer, J Douglas Zechar, Maximilian J Werner, Edward H Field, David D Jackson, Thomas H Jordan, e RELM Working Group. First results of the regional earthquake likelihood models experiment. *Pure and Applied Geophysics*, 167(8-9):859–876, 2010. 8, 9, 16

[32] Mark Simons, Sarah E Minson, Anthony Sladen, Francisco Ortega, Junle Jiang, Susan E Owen, Lingsen Meng, Jean-Paul Ampuero, Shengji Wei, Risheng Chu, et al. The 2011 magnitude 9.0 tohoku-oki earthquake: Mosaicking the megathrust from seconds to centuries. *science*, 332(6036):1421–1425, 2011. 1

[33] Masato Takahashi e Hajime Kita. A crossover operator using independent component analysis for real-coded genetic algorithms. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 1, pages 643–649. IEEE, 2001. 29

[34] Tokuji Utsu e Yosihiko Ogata. The centenary of the omori formula for a decay law of aftershock activity. *Journal of Physics of the Earth*, 43(1):1–33, 1995. 22

[35] Thomas van Stiphout, Jiancang Zhuang, e David Marsan. Seismicity declustering. *Community Online Resource for Statistical Seismicity Analysis*, 10, 2012. 1

[36] S Wilkinson, G Chiaro, Rama Mohan Pokhrel, T Kiyota, Toshihiko Katagiri, Keshab Sharma, e K Goda. The 2015 gorkha nepal earthquake: Insights from earthquake damage survey. 2015. 1

[37] Yoshiko Yamanaka e Kunihiko Shimazaki. Scaling relationship between the number of aftershocks and the size of the main shock. *Journal of Physics of the Earth*, 38(4):305–324, 1990. 22

[38] J Douglas Zechar. Evaluating earthquake predictions and earthquake forecasts: A guide for students and new researchers. *Community Online Resource for Statistical Seismicity Analysis*, pages 1–26, 2010. 6, 16

[39] Qiuen Zhang e Cheng Wang. Using genetic algorithms to optimize artificial neural network: a case study on earthquake prediction. In *Second International Conference on Genetic and Evolutionary Computing*, pages 128–131. IEEE, 2012. 12

[40] Feiyan Zhou e Xiaofeng Zhu. Earthquake prediction based on lm-bp neural network. In Xiaozhu Liu e Yunyue Ye, editors, *Proceedings of the 9th International Symposium on Linear Drives for Industry Applications, Volume 1*, volume 270 of *Lecture Notes in Electrical Engineering*, pages 13–20. Springer Berlin Heidelberg, 2014. 12

[41] Jiancang Zhuang, Yosihiko Ogata, e David Vere-Jones. Analyzing earthquake clustering features by using stochastic reconstruction. *Journal of Geophysical Research: Solid Earth (1978–2012)*, 109(B5), 2004. 16, 22, 25