

ARISTOTLE UNIVERSITY OF THESSALONIKI  
FACULTY OF SCIENCES  
SCHOOL OF INFORMATICS  
DEPARTMENT OF COMPUTER SCIENCE  
«KNOWLEDGE, DATA AND SOFTWARE TECHNOLOGIES»



*Master Thesis*

# **Machine learning methods for the analysis of data of an Electricity Distribution Network Operator**

**by Ioannis Mamalikidis (UID: 633)  
for the degree of Master of Science**

**Thesis Committee**

**Supervisor:** Eleftherios Angelis

**Members:** Grigorios Tsoumakas  
Ioannis Vlahavas

**THESSALONIKI  
MARCH 2017**



ΑΡΙΣΤΟΤΕΛΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ  
«ΤΕΧΝΟΛΟΓΙΕΣ ΓΝΩΣΗΣ, ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΛΟΓΙΣΜΙΚΟΥ»  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



# **Machine learning methods for the analysis of data of an Electricity Distribution Network Operator**

**Διπλωματική Εργασία του Ιωάννη Μαμαλικίδη**

**Εξεταστική Επιτροπή**

**Επιβλέπων:** Ελευθέριος Αγγελής

**Μέλη:** Γρηγόριος Τσουμάκας  
Ιωάννης Βλαχάβας

**ΘΕΣΣΑΛΟΝΙΚΗ  
ΜΑΡΤΙΟΣ 2017**

# Abstract

Once every few decades an invention changes the landscape of some aspects of our life. Industrial revolutions improved our everyday lives whilst medical revolutions expanded our lifespans. In the path we're leading, most of sciences will be reduced to computer science, enabling faster and more accurate results. Machine learning is a vast field whose use spans over a plethora of tasks like optical character recognition, search engines and computer vision, to applications on other fields, such as the medical one. Here, two of the three main categories of machine learning are being used; namely unsupervised learning to cluster the data into geographical groups, and an array of different types of supervised learning to make predictions.

The data which are subject to machine learning originate from the Hellenic Electricity Distribution Network Operator (HEDNO S.A.), the largest company for the operation, maintenance and development of the power distribution network in Greece. Working with real data is bound to come with solid hurdles, such as a substantial amount of noise in the data, erroneous entries, missing values and incomplete data. To this end, a considerable amount of time was devoted to pre-processing; cleaning up the data, retrieving or extrapolating from it and transforming it into a suitable form for the next steps.

Since the dataset consists of projects dealing with construction or repair on actual locations, it has a geographical aspect to it. As the data themselves do not come with associated longitudes and latitudes, a method was devised to retrieve them and in turn use them to cluster the projects into geographical groups.

The Last step was to apply 10 machine learning algorithms to predict which future projects are going to be approved and which are not, hence enabling the company to be better prepared in terms of needed items availability. Statistical analysis on the trained machine learning algorithms themselves was also important in order to identify the best model for this dataset.

Ioannis Mamalikidis

10/03/2017

Keywords: Machine Learning, Big Data, Forecasting, Classification, Clustering

# Περίληψη

Μια φορά κάθε μερικές δεκαετίες μια εφεύρεση αλλάζει εντελώς το τοπίο ορισμένων πτυχών της ζωής μας. Η βιομηχανική επανάσταση βελτίωσε την καθημερινή μας ζωή, ενώ οι ιατρικές επαναστάσεις επέκτειναν το προσδόκιμο ζωής μας. Στην κατεύθυνση που ακολουθούμε οι περισσότερες επιστήμες θα υπάγονται στην επιστήμη των υπολογιστών, επιτρέποντας έτσι ταχύτερα και πιο ακριβή αποτελέσματα. Η μηχανική μάθηση είναι ένα ευρύ πεδίο του οποίου η χρησιμότητα άπτεται πληθώρας εργασιών. Από την οπτική αναγνώριση χαρακτήρων, τις μηχανές αναζήτησης και την όραση των υπολογιστών, μέχρι και εφαρμογές σε άλλους τομείς, όπως σε αυτόν της ιατρικής. Εδώ χρησιμοποιούνται δύο από τις τρεις βασικές κατηγορίες της μηχανικής μάθησης, η μάθηση χωρίς επίβλεψη για συσταδοποίηση των δεδομένων σε γεωγραφικές ομάδες και διάφοροι αλγόριθμοι μάθησης με επίβλεψη για παραγωγή προβλέψεων.

Για τη βέλτιστη ταχύτητα και κλιμάκωση των αλγορίθμων πάνω στο μέγεθος της βάσης δεδομένων έγινε χρήση προγραμμάτων, μεθόδων και τρόπου γραφής πηγαίου κώδικα κατάλληλων για Μεγάλα Δεδομένα (Big Data). Χρησιμοποιήθηκαν οι τεχνολογίες SQL Server για σχεσιακή βάση δεδομένων, R Language για μηχανική μάθηση, Microsoft ScaleR για άκρως κλιμακώσιμες υλοποιήσεις αλγορίθμων μηχανικής μάθησης αλλά και επικοινωνίας, ανάκτησης, επεξεργασίας και αποθήκευσης δεδομένων με πολυνηματικό (multi-threaded) και παράλληλο (concurrent) τρόπο ξεπερνώντας ταυτόχρονα περιορισμούς μνήμης (RAM) και επεξεργαστή (CPU) και VB.NET για ανάπτυξη προγράμματος με γραφικό περιβάλλον το οποίο θα αξιοποιεί τα προαναφερθέντα και θα δίνει δυνατότητα παραμετροποίησης στον τελικό χρήστη μέσα από ένα εύχρηστο, διαισθητικό περιβάλλον όπου κάθε εντολή συνοδεύεται από λεπτομερή περιγραφή της λειτουργίας της.

Τα δεδομένα τα οποία υπόκεινται σε μηχανική μάθηση είναι της ΔΕΔΔΗΕ Α.Ε. (Διαχειριστής του Ελληνικού Δικτύου Διανομής Ηλεκτρικής Ενέργειας), της μεγαλύτερης εταιρίας για τη λειτουργία, τη συντήρηση και την ανάπτυξη του δικτύου διανομής ηλεκτρικής ενέργειας στην Ελλάδα. Η βάση δεδομένων περιλαμβάνει περίπου

450.000 έργα, τα οποία απαρτίζονται από 2.533.079 βαριάντες και έχουν χρησιμοποιηθεί 17.134.977 υλικά.

Δουλεύοντας με πραγματικά δεδομένα είσαι σίγουρο ότι θα προκύψουν εμπόδια όπως σημαντικά μεγάλος θόρυβος στα δεδομένα, εσφαλμένες εγγραφές, ελλιπείς τιμές και ελλιπή δεδομένα. Για αυτό τον σκοπό, ένα σεβαστό ποσοστό χρόνου αφιερώθηκε στην προ-επεξεργασία των δεδομένων, τον καθαρισμό τους, και στην ανάκτηση ή εξαγωγή από αυτά και τη μετατροπή τους σε μία κατάλληλη για τα επόμενα βήματα μορφή.

Δεδομένου ότι το σύνολο δεδομένων αποτελείται από έργα που έχουν να κάνουν με την κατασκευή ή επισκευή πάνω στη γη, το σύνολο δεδομένων έχει μία γεωγραφική πτυχή. Καθώς τα δεδομένα δεν περιέχουν εγγενώς πεδία γεωγραφικού μήκους και πλάτους, αναπτύχθηκε ένα πρόγραμμα για να τα ανακτήσει και με τη σειρά τους να χρησιμοποιούν για τη συσταδοποίηση των έργων σε γεωγραφικές ομάδες.

Το τελευταίο βήμα ήταν η εφαρμογή 10 αλγορίθμων μηχανικής μάθησης για πρόβλεψη ως προς το ποια μελλοντικά έργα θα εγκριθούν και ποια θα απορριφθούν, ως εκ τούτου, επιτρέποντας στην εταιρεία να είναι καλύτερα προετοιμασμένη όσον αφορά στη διαθεσιμότητα των απαιτούμενων αντικειμένων. Η στατιστική ανάλυση των μοντέλων μηχανικής μάθησης ήταν επίσης σημαντική για τον προσδιορισμό του καλύτερου μοντέλου για το συγκεκριμένο σύνολο δεδομένων. Για τον σκοπό αυτό χρησιμοποιήθηκαν οι μετρικές Accuracy, Balanced Accuracy, Detection Rate, Misclassification Rate, Sensitivity (ή Recall ή True Positive Rate), Specificity (ή True Negative Rate), False Negative Rate, Precision (ή Positive Predictive Value), Alternative Positive Predictive Value with Prevalence, Negative Predictive Value, Alternative Negative Predictive Value with Prevalence, False Discovery Rate, Null Error Rate, Prevalence, F1 Score, G-measure, Matthews correlation coefficient, Cohen's kappa coefficient, Youden's J statistics, Receiver Operating Characteristic Curves, και Area Under the Curve. Ο εκάστοτε αλγόριθμος ενδέχεται να παράγει και δικές μετρικές ή γραφήματα για διάφορους λόγους, όπως στην περίπτωση της οπτικοποίησης του δέντρου απόφασης του αλγορίθμου rxDTree με την "plot()" συνάρτηση της R γλώσσας.

Ιωάννης Μαμαλικίδης

10/03/2017

Keywords: Μηχανική Μάθηση, Μεγάλα Δεδομένα, Πρόβλεψη, Κατηγοριοποίηση, Συσταδοποίηση

# Acknowledgements

I would like to wholeheartedly thank my parents for their unwavering support, trust and love they've been continually providing throughout my life; to say that without you I would have never made it this far, is, frankly, an understatement. I have also been fortunate enough to meet the brilliant professors, Dr. Eleftherios Angelis, Dr. Grigorios Tsoumakas, and Dr. Ioannis Vlahavas, which cemented my way here. Each, in their own way, rekindled my interest in computer science as they put their hearts into their jobs, never reluctant to go the extra mile. I'm also incredibly grateful for all their help with the thesis itself. I'd also like to thank Mr. Christos Karapiperis, Head of the Regulatory Adjustment Sector, for his cooperation, time and effort regarding this thesis. Lastly, I'd be remiss should I not mention my English teacher, Enie Matsangou Sfyaki, who is all but solely responsible for my having a firm grasp of the English language, opening new roads for me, cascading into this very moment. I'm immensely thankful to Dr. Nikolaos Mittas and Nikolas Vordos for introducing me to, and affording me the chance, to glimpse into how an academic life and work in a research lab, is. I'll forever remain grateful to you all.

# Table of Content

<b>ABSTRACT.....</b>	<b>IV</b>
<b>ΠΕΡΙΛΗΨΗ .....</b>	<b>V</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>VII</b>
<b>TABLE OF CONTENT .....</b>	<b>VIII</b>
<b>LIST OF FIGURES.....</b>	<b>XI</b>
<b>1 INTRODUCTION .....</b>	<b>1</b>
<b>2 MACHINE LEARNING .....</b>	<b>3</b>
<b>3 PROGRAMMES AND TOOLS.....</b>	<b>5</b>
3.1 SQL SERVER.....	5
3.2 R LANGUAGE .....	6
3.3 MICROSOFT R & SCALER .....	6
3.4 VB.NET .....	7
<b>4 HEDNO S.A. ....</b>	<b>8</b>
4.1 COMPANY’S GOAL, VISION AND MISSION .....	8
4.2 ORGANIZATIONAL STRUCTURE.....	9
4.3 ACTIVITIES .....	10
4.4 PROJECT TIMELINE .....	11
<b>5 DATA AND PREPROCESSING .....</b>	<b>13</b>
5.1 SQL VIEW CREATION .....	13
5.1.1 Variables used as is from the “Εργα” table.....	13
5.1.2 Variables engineered from the “Εργα” table.....	15
5.1.3 Clauses applied to the “Εργα” table .....	19
5.2 THE ORIGINAL DATABASE .....	29
5.2.1 General Information.....	29
5.2.2 Descriptive Statistics .....	30
5.2.3 Column Selection.....	37
<b>6 APPLYING MACHINE LEARNING .....</b>	<b>57</b>



6.1	UNSUPERVISED LEARNING.....	57
6.2	SUPERVISED LEARNING .....	61
6.2.1	<i>Logistic Regression</i> .....	62
6.2.2	<i>Decision Trees</i> .....	62
6.2.3	<i>Naïve Bayes</i> .....	62
6.2.4	<i>Random Forest</i> .....	63
6.2.5	<i>Stochastic Gradient Boosting</i> .....	63
6.2.6	<i>Stochastic Dual Coordinate Ascent</i> .....	64
6.2.7	<i>Boosted Decision Trees</i> .....	64
6.2.8	<i>Ensemble of Decision Trees</i> .....	64
6.2.9	<i>Neural Networks</i> .....	65
6.2.10	<i>Fast Logistic Regression</i> .....	66
6.3	MODEL EVALUATION .....	67
6.3.1	<i>Accuracy</i> .....	67
6.3.2	<i>Balanced Accuracy</i> .....	67
6.3.3	<i>Detection Rate</i> .....	68
6.3.4	<i>Misclassification Rate</i> .....	68
6.3.5	<i>Sensitivity / Recall / True Positive Rate</i> .....	68
6.3.6	<i>False Positive Rate (FPR)</i> .....	69
6.3.7	<i>Specificity / True Negative Rate (TNR)</i> .....	69
6.3.8	<i>False Negative Rate (FNR)</i> .....	69
6.3.9	<i>Precision / Positive Predictive Value (PPV1)</i> .....	69
6.3.10	<i>Positive Predictive Value (PPV2)</i> .....	70
6.3.11	<i>Negative Predictive Value (NPV1)</i> .....	70
6.3.12	<i>Negative Predictive Value (NPV2)</i> .....	70
6.3.13	<i>False Discovery Rate (FDR)</i> .....	71
6.3.14	<i>Null Error Rate</i> .....	71
6.3.15	<i>Prevalence</i> .....	71
6.3.16	<i>F1 Score</i> .....	71
6.3.17	<i>G-measure</i> .....	72
6.3.18	<i>Matthews correlation coefficient, <math>\phi</math> (PhiMCC)</i> .....	72
6.3.19	<i>Cohen's kappa coefficient, <math>\kappa</math> (CohensK)</i> .....	72
6.3.20	<i>Youden's J statistic</i> .....	73
6.3.21	<i>Receiver Operating Characteristic (ROC) Curve</i> .....	73

6.3.22 Area Under the Curve (AUC).....	74
<b>7 IMPLEMENTATION .....</b>	<b>75</b>
7.1 SETTINGS .....	76
7.1.1 R Settings.....	77
7.1.2 Geolocation Settings.....	78
7.1.3 SQL Views Settings.....	79
7.2 PRE-PROCESSING .....	80
7.2.1 Create Geolocation SQL Columns: .....	80
7.2.2 Create Needed SQL Views: .....	81
7.2.3 Geo-Locate:.....	82
7.2.4 Geo-Location Status:.....	84
7.2.5 Export List of Problematic Addresses:.....	85
7.2.6 Reset Invalid Geolocation Entries:.....	86
7.2.7 Pre-Process the Data:.....	88
7.3 CLUSTERING .....	92
7.3.1 Step 0: Process Data .....	93
7.3.2 Step 1: Apply Unsupervised Learning .....	95
7.4 CLASSIFICATION.....	98
7.4.1 Form Train and Test sets.....	98
7.4.2 Logistic Regression.....	100
7.4.3 Decision Trees.....	106
7.4.4 Naïve Bayes.....	110
7.4.5 Random Forest.....	111
7.4.6 Stochastic Gradient Boosting.....	114
7.4.7 Stochastic Dual Coordinate Ascent.....	116
7.4.8 Boosted Decision Trees.....	117
7.4.9 Ensemble of Decision Trees.....	119
7.4.10 Neural Networks.....	120
7.4.11 Fast Logistic Regression.....	122
7.4.12 Model Evaluation.....	125
<b>8 CONCLUSIONS.....</b>	<b>129</b>
<b>REFERENCES .....</b>	<b>131</b>

# List of Figures

Figure 1: Organizational structure of HEDNO S.A. <sup>[24]</sup> .....	10
Figure 2: Project Flowchart <sup>[24]</sup> .....	12
Figure 3: SQL Database ER Diagram .....	29
Figure 4: Visualisation of original data (Iteration 0) .....	59
Figure 5: Greece's Map.....	59
Figure 6: Data Visualisation of Iteration 1 .....	60
Figure 7: Sum of Squared Error Scree Plot .....	60
Figure 8: Clusters Visualisation after K-Means .....	61
Figure 9: R Settings.....	77
Figure 10: Geolocation Settings.....	78
Figure 11: SQL Views Settings .....	79
Figure 12: Create Geolocation SQL Columns Mouse Hover .....	80
Figure 13: Create Geolocation SQL Columns Pushed .....	81
Figure 14: Create Needed SQL Views Mouse Hover .....	81
Figure 15: Create Needed SQL Views Pushed .....	82
Figure 16: Geo-Locate Mouse Hover .....	83
Figure 17: Geo-Locate Pushed .....	84
Figure 18: Geo-Location Status Mouse Hover .....	84
Figure 19: Geo-Location Status Pushed .....	85
Figure 20: Export List of Problematic Addresses Mouse Hover .....	86
Figure 21: Export List of Problematic Addresses Pushed .....	86
Figure 22: Reset Invalid Geolocation Entries Mouse Hover.....	87
Figure 23: Reset Invalid Geolocation Entries Pushed .....	87
Figure 24: Pre-Process The Data Mouse Hover .....	88
Figure 25: Pre-Process The Data Mouse Pushed.....	88
Figure 26: Data Summary Form.....	89
Figure 27: Variables Information Form.....	90
Figure 28: Pre-Processing Geolocation Visualisation.....	91
Figure 29: Step 0: Process Data Mouse Hover .....	93
Figure 30: Step 0: Process Data Mouse Pushed .....	93
Figure 31: Step 0: Process Data Geolocation Graph.....	94

Figure 32: Step 1: Apply Unsupervised Learning Mouse Hover.....	95
Figure 33: Step 1: Process Data Mouse Pushed .....	95
Figure 34: Step 1: Optimal k Value Selection.....	96
Figure 35: Step 1: Optimal k Value Selection.....	97
Figure 36: Form Train and Test Sets Mouse Hover.....	98
Figure 37: Form Train and Test Sets Pushed.....	98
Figure 38: Class Imbalance Plot.....	99
Figure 39: Logistic Regression Mouse Hover.....	100
Figure 40: Logistic Regression Pushed.....	101
Figure 41: Model Statistics .....	102
Figure 42: Single-Model ROC Curve.....	104
Figure 43: ROC Curves of Multiple Models .....	105
Figure 44: Decision Trees Mouse Hover.....	106
Figure 45: Decision Trees Mouse Pushed.....	107
Figure 46: Decision Tree Model Plot .....	108
Figure 47: Plot of Optimal Pruning Based on Complexity .....	109
Figure 48: Naïve Bayes Mouse Hover.....	110
Figure 49: Naïve Bayes Pushed .....	110
Figure 50: Random Forest Mouse Hover.....	111
Figure 51: Random Forest Pushed.....	112
Figure 52: Variables Importance Plot .....	112
Figure 53: OOB Error vs nTree .....	113
Figure 54: Stochastic Gradient Boosting Mouse Hover.....	114
Figure 55: Stochastic Gradient Boosting Pushed.....	115
Figure 56: Stochastic Dual Coordinate Ascent Mouse Hover.....	116
Figure 57: Stochastic Dual Coordinate Ascent Pushed.....	116
Figure 58: Boosted Decision Trees Mouse Hover.....	117
Figure 59: Boosted Decision Trees Pushed .....	118
Figure 60: Ensemble of Decision Trees Mouse Hover .....	119
Figure 61: Ensemble of Decision Trees Mouse Hover .....	119
Figure 62: Neural Networks Mouse Hover.....	120
Figure 63: Neural Networks Pushed.....	121
Figure 64: Fast Logistic Regression Mouse Hover .....	122
Figure 65: Fast Logistic Regression Pushed.....	123

# 1 Introduction

Since time immemorial, the human race has been striving for optimisation. An unbroken chain of people, iterating time and again over certain procedures with one goal in mind, one perpetual struggle; to maximise the gain whilst keeping the cost at a minimum. For an Electricity Distribution Network Operator, such as the Greek one, HEDNO S.A. (also known as ΔΕΔΔΗΕ), it means a foresight as to which items are to be needed in the near future, therefore not having to fetch for them in the eleventh hour as new deals for projects are being sealed. Instead of being vulnerable to the risk of whole projects being held back as a result of having a limited window to assemble the items and materials needed, every so often, barely in the nick of time, or worse, overdue, the preparation procedure can benefit from predicting which projects are going to be sealed next. This translates into more revenue for the company, and less time and man-hours spent in what, at the end of the day, would seem time wasted, and for naught. To this end, statistics and machine learning are employed in the pursuit of analysis of previous data, extracting useable information and using it to inaugurate a predictive procedure, supervised learning, which is used to classify the company's new projects in advance, to either 'Cancelled' or 'Approved'. In its essence, it is akin to inventory optimisation and management, for within it, lies the opportunity to balance capital investment objectives over the company's vast number of stock-keeping units, taking demand and supply volatility into account. Far from just an inventory optimisation measure, the information amassed can aid in setting the strategy of the organisation, coordinating the efforts of its employees.



## 2 Machine Learning

Machine learning is a field which borrows from and relies upon the natural world for inspiration, as much as for mechanism, and in such it reflects the mechanics of evolution, adaptation and learning represented by the very species we are a member of.<sup>[1]</sup>

Learning, in and of itself, is a very broad concept, incorporating notions such as the development of skills, the acquisition of declarative knowledge, the generalisation of specific knowledge to render it useful and applicable to a broader field or to solve various tasks, and the discovery of new facts resulting in new hypotheses and theories. The underlying logic behind it all, is, instead of using static programme instructions, an algorithm whose purpose is to create a model from various input data is constructed, which in turn is used to make data-driven decisions and predictions.<sup>[5]</sup>

With its first appearance in mid-seventies, and its first manifesto in the “First Machine Learning Workshop” documents at Carnegie-Mellon University, it all seems to have started with an effort to model self-organisation and self-stabilisation along with an ability to comprehend shapes.<sup>[2]</sup> There is no shortage of things that are immensely difficult, or downright impossible to programme by hand, including, but not limited to spam filtering, face, object and speech recognition, accurate language translation, data mining, robot motion, or common sense, for that matter.<sup>[7]</sup> To this end, the field of machine learning was created so that data can be collected and subsequently be used by a machine to learn how to robustly solve complex tasks using learning methods, such as supervised learning, where both inputs and outputs are available, unsupervised learning where only inputs are available for some data, or reinforcement learning where we lack a direct access to the output but we can get a sort of measure of its quality.<sup>[3]</sup>

Unbeknownst to us or not, as humanity relies more and more in technology, machine learning is ever more common in our lives and we have people like George Mason University’s Professor Ryszard Michalski to thank for that.<sup>[6]</sup> For it was research like his handwriting recognition models, and software modelling a form of reasoning, that revolutionised and helped shape the field of machine learning to become what it is today, facilitating our efforts to solve problems in an ever expanding span of fields.<sup>[4]</sup>





## 3 Programmes and Tools

This endeavour required several technologies, programming languages and tools be used. HEDNO company's database is a "Microsoft SQL Server" one, so commands passed to it for things like SQL View creation, or functions were done so using "TSQL". Interoperability between the programming languages and the Microsoft SQL Server was handled by built-in functions on the programming languages themselves ("RxDboData" for R and "SqlConnection" for VB.NET). The ever-popular with statisticians and data miners, R language, with its vast array of packages along with Microsoft's RevoScaleR was employed for Machine Learning purposes, graphs, plots, and statistics. Manipulating and running R code is garden-variety for data miners, but unlike them, most end-users do not find it that commonplace a phenomenon. With that in mind, as well as user-friendliness, a programme with a Graphical User Interface was also developed to complement the SQL and R code. It allows for error prevention, effortless customisation, easy access to straightforward tips and information, and needless to say, execution of said machine learning algorithms by means of a click of a button.

### 3.1 SQL Server

SQL Server is a computer programme used by other computer programmes as a means of delivering database services as defined by the client-server model. Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it might run either on the same computer or on a remote computer across a network, including the internet itself and it serves as a way of storing and retrieving data as requested by other software applications. From in-memory performance and advanced security to in-database analytics, SQL Server's built in functionality allows for building intelligent, mission-critical applications using a scalable, hybrid database platform. It allows for transformation of data into actionable insights delivered on any device, online or offline. Its integration with R is such that data analysis can be achieved directly within the SQL Server database without the need to move the data. <sup>[10]</sup>

## 3.2 R Language

R is an Open Source mathematical programming language and software environment for statistical computing and graphics that is supported by the R Foundation for Statistical Computing. It is a GNU project, akin to the S mathematical programming language and software environment, so much so that it can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R. It provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, etc.) and graphical techniques through its command line interface or third-party graphical front-ends, and it is highly extensible. [8]

## 3.3 Microsoft R & ScaleR

Parallel computing is the process of breaking a given job into computationally independent components and running those independent components on separate threads, cores, or computers and then combining the results into a single returned result. RevoScaleR is able to perform parallel computing on any computer with multiple computing cores. Distributed computing is often used as a synonym for parallel computing, but a distinction is drawn: distributed computing refers to computations distributed over more than one computer, whilst parallel computing can occur on one computer or many. Distributed computing capabilities are built into RevoScaleR, meaning that developing complex analysis scripts on a local computer, creating one or more compute contexts for use with distributed computing resources, and then seamlessly move between executing scripts on the local computer and in a distributed context is feasible. Microsoft R is a broadly deployable analytics platform for R, supporting a variety of big data statistics, predictive modelling and machine learning capabilities. The in-memory limitations of open source R, are addressed by adding parallel and chunked processing of data, enabling users to run analytics on datasets much bigger than what fits in main memory (RAM) via ScaleR. ScaleR is a collection of functions in Microsoft R that are used for practicing data science in Big Data. Although ScaleR works on both small and large datasets, what ScaleR enables is analysis of very large data sets that would otherwise exceed the memory and processing capabilities of any one machine. It delivers enterprise class performance and scalability for R-based applications with libraries that allow for writing once and deploying across multiple platforms with minimal effort,

whether on-premises or in the cloud. The supported platforms are R Server for Hadoop, R Server for Teradata DB, R Server for Linux, R Server for Windows, and SQL Server R Services.<sup>[14, 11]</sup>

### **3.4 VB.NET**

VB.Net comes from the BASIC language which first appeared some 52 years ago, on May 1, 1964. It is a programming language which supports multiple programming paradigms, enabling targeted use for most efficient writing on each problem. It is object-oriented and implemented on the .NET framework. The integrated development environment (IDE) for developing in VB.NET is Visual Studio, which is also the IDE that has been used for the R code.

## **4 HEDNO S.A.**

The Hellenic Electricity Distribution Network Operator (HEDNO) S.A. is an anonymous company which was formed in May 2012 after the separation of the Distribution Department from the PPC (Public Power Corporation) S.A., the biggest power producer and electricity supply company in Greece, according to L.4001/2011 and in compliance with 2009/72/EC EU Directive relative to the electricity market organisation. In spite of its being a subsidiary of PPC S.A., it remains independent in operation and management.

HEDNO's mission is to ensure the proper operation, maintenance and development of the Distribution Network and the unhindered access to it. HEDNO S.A. is responsible for the operation of the Hellenic Electricity Distribution Network and, as such, is responsible for the uninterruptible electricity supply throughout the country. In terms of number of consumers served, HEDNO maintains that it is the fifth largest Distribution Company in EU. In terms of volume, HEDNO's network lines span over 236,000km in total length. HEDNO delivers Medium and Low Voltage electricity to 7.4 million customers through its network, whilst it also handles High Voltage networks in Attiki and in the non-interconnected islands. The Company employs about 7,000 individuals (regular and temporary staff) directly and through the cooperating contractors. about 5,000 individuals indirectly. <sup>[24]</sup>

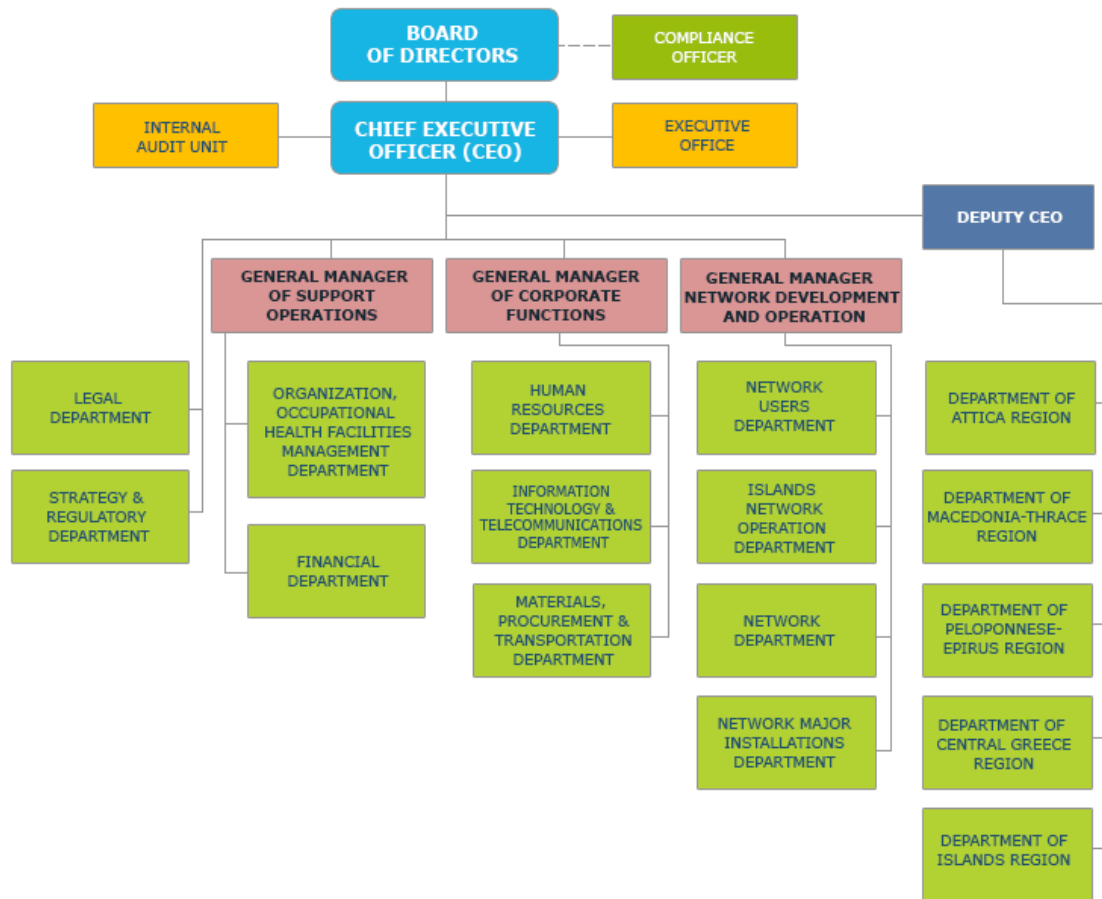
### **4.1 Company's Goal, Vision and Mission**

HEDNO S.A. aims at making meaningful contributions to the development of Greece as a whole, as well as improving the welfare and quality of life of its residents. This is achieved through providing reliable and economically efficient power whilst keeping a healthy respect for the people and the environment, reflected in the process undertaken to fulfil the promise. The company's vision is to establish a company-model in the field of power supply with the intention of providing prime services to citizens, to operate and develop the network conforming with Advanced Countries standards and to assure that the Network users, employees, associates, shareholders and society in general find the services satisfactory. The company's mission is the development and operation of the

electricity distribution network and systems of the non-interconnected islands as well as the assurance of equal access to them by all consumers, producers and suppliers with transparency and objectivity. In terms of service quality, this means meeting users' high standards by constantly upgrading itself to provide state-of-the-art services, by maintaining a swift pace in granting requests like power connections, and by improving the level of service provision where applicable. In terms of Energy quality, it means modernising the distribution installations, maintaining high reliability and efficiency rates, decreasing the scheduled and unscheduled power interruptions rate, reducing the number of outages, and ultimately holding the quality of the product (electricity/voltage) to an ever-higher calibre. In terms of Operating costs, it entails minimising them through a series of examinations and informed decisions having considered all applicable factors. The company intends to ensure a reliable and safe network under which all users are granted equal access permission given the current regulatory framework and society. <sup>[24]</sup>

## **4.2 Organizational Structure**

The organisational structure of HEDNO S.A., as illustrated in the figure below, puts the Board of Directors on the top of the chain of command, followed by the Chief Executive Officer (CEO). Under the CEO's supervision, are: The Deputy CEO, responsible for the Regional Departments, the "Legal" and "Strategy & Regulatory" departments, and the three General Managers (that of "Support Operations", "Corporate Functions" and "Network Development and Operation"), responsible for their respective departments. <sup>[24]</sup>



**Figure 1:** Organisational structure of HEDNO S.A. [24]

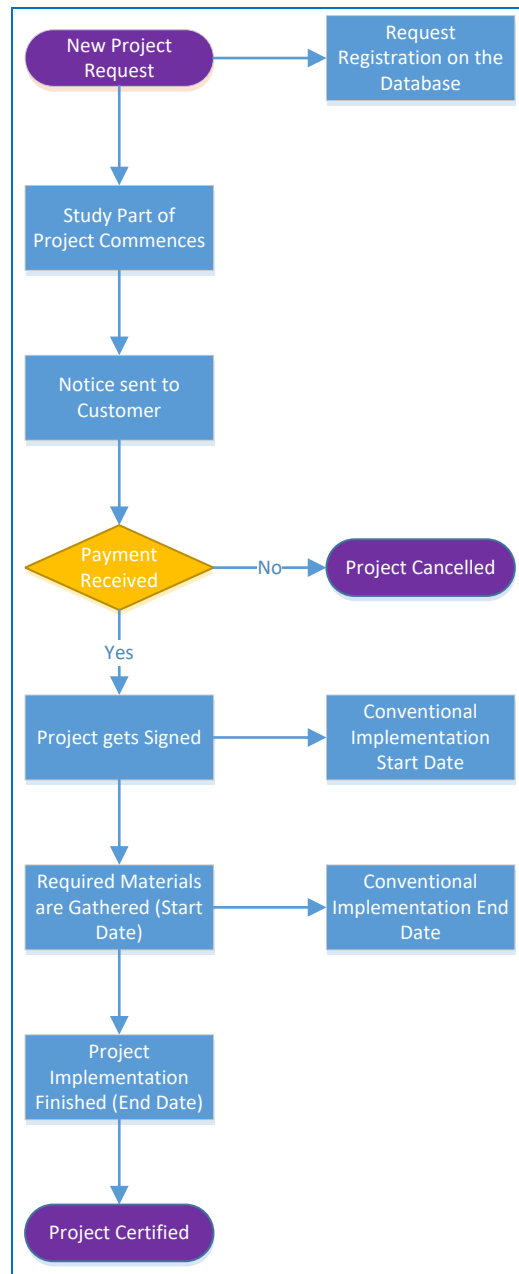
### 4.3 Activities

The company-tasks include the operation, maintenance and development of the power distribution network in Greece, as well as the assurance of a transparent and impartial access for consumers and all network users in general. HEDNO S.A. aims at providing reliable power supply to the customers, quality of electricity voltage and constant improvement of quality in services. HEDNO S.A.'s activity spans over users' request fulfilment regarding new consumer connections, upgrading older supplies, increasing their power, and providing Network rearrangement services. In addition, the company strives to improve and update its distribution network and continue a steady construction of Distribution Centres and Power Lines. Amongst the projects executed by the company, are, the operation of Distribution Networks, inspections, and maintenance performed on them, power failure restoration, customer service, and consumption metering. With regards to networks, it maintains the efficient operation of the electricity market and a

reliable and financially viable operation for the autonomous electricity systems on the islands. [24]

## 4.4 Project Timeline

Each step of the process for a project's completion is reflected in the figure below and has a corresponding date field on the database. A project request may originate from inside the company itself, or by a customer. When a new project request arrives, the request is registered on the database and the study part of it commences. A notice is subsequently sent to the customer, informing them of the cost and awaiting payment. Each project falls under a predefined set of categories, namely EA for "Consumer Electrification", EB for "Producer Electrification", EC "Variations", ED for "Aesthetic improvement", EE for "Enhancement", EF for "Space reformation", SA for "Localized Maintenance", SB for "Preventive Maintenance", SC for "Repair damaged network", SD for "Repair damages of third parties", SE for "Measurements", SF "Network Elimination / Dismantling", SG for "Column Maintenance", SH for "Pruning / Deforestation" for "Remedy for theft", SJ for "Disconnections / Reconnections due to debt", SK for "Disconnections / Reconnections due to customer request", and SL for "Technical procedures in counters". The customers have 3 months at their disposal to complete the transaction before the project gets cancelled. Should they opt to pay, the project is signed and a conventional start date is set. The required materials get assembled and the implementation commences, setting a conventional end-of-project date. As soon as the project has been successfully constructed, the certification process starts taking place. [24]



**Figure 2: Project Flowchart**



# 5 Data and Preprocessing

As is almost always the case when working with real data, they are organised in such a way that it makes it easier for a company to retrieve the information it needs, but in spite of its being ideal for the company, when it comes to data analysis, it's a whole other story. A substantial part of the total amount of time had to be spent on this part, getting to know the data, their 'quirks' and all; unearthing the patterns, the distributions, and acquiring enough insight to genuinely comprehend its structure, underlying logic, and meaning.

To do so, there were certain enquiries that had to occur, namely, 'what is the relationship between the SQL Tables?', 'which are the variables/columns and cases/rows of each table, and what do they represent?', as well as an enquiry on the summary of each table, revealing key information, such as the range, mean and standard deviation of values for quantitative variables; the factor levels for qualitative variables; and the percentage of missing data and variable type for either one.

All the SQL and R source code is publicly available through the github repository: <https://github.com/N1h11sT/Thesis-CSD-AUTH-UID-633>; however, the dataset itself, be it in a SQL Database format, XDF files or otherwise, cannot be made public as a result of its proprietary nature and rules and regulations that come with it.

## 5.1 SQL View Creation

Using the variables picked above and passing them through certain clauses (mentioned below), we're getting two kinds of new variables: Those used as is, only with clauses applied to them, and those who were engineered by transforming existing variables.

### 5.1.1 Variables used as is from the "Εργα" table

The variables in the table below which are part of the picked variables are marked with an asterisk (\*) and they are useful absent of any need for transformation. That being said, their respective value ranges declared above were based on their original distribution. The database does include many a project unrelated with the kind of projects we are to predict, so clauses (mentioned later) had to be enforced, changing the distribution. As a result, a new value range will be mentioned for each variable, post clause imposition. Their respective description does remain the same and shall not be restated.

- [ID\_Erga]: is a Categorical numerical integer field with a range of [104517, 105563, 105677, ..., 563829, 563842, 563907].
- [GrafioEktelesisErgou]: is a Categorical numerical integer field with a range of [0506000600, 0506000700, 0506000800, ..., 0509011204, 0509011206, 0509011300].
- [Katigoria]: is a Categorical string field with a range of ['X', ',', '^', ..., '999', 'P', 'Q'].
- [Meres\_Meletis]: is a scale numerical integer field with a range of [-3029, -1950, -986, ..., 5545, 21682, 30000]
- [Sinergio\_Meletis]: is a Categorical string field with a range of [ΣΥΝΕΡΓΕΙΑ ΔΕΔΔΗΕ, ΣΥΝΕΡΓΕΙΑ ΕΡΓΟΛΑΒΟΥ, ΣΥΝΕΡΓΕΙΑ ΤΡΙΤΩΝ].
- [Kostos\_Ergatikon\_Kataskevis]: is a scale numerical decimal field with a range of [1.7134, 2, 2.7144, ..., 1229880.6889, 1409426.9387, 1489345.8301].
- [Kostos\_Il原因\_Kataskevis]: is a scale numerical decimal field with a range of [0.01, 0.0165, 0.024, ..., 594815, 792868.475, 2587043].
- [Kostos\_Kataskevis]: is a scale numerical decimal field with a range of [1, 3, 5, ..., 2027147.8186, 15013244, 16797610].
- [Kostos\_Ergolavikon\_Epidosis]: is a scale numerical decimal field with a range of [1.416, 2, 2.2433, ..., 1174579.8841, 1245709.0566, 11176571].
- [Ektasi\_Ergou]: is a Categorical numerical integer field with a range of [1, 2]
- [Anagi\_YS]: is a binary field.
- [SAP\_Typos\_Pelati]: is a Categorical string field with a range of [NULL, E, I].
- [SAP\_Eidos\_Aitimatos]: is a Categorical string field with a range of [NULL, 6121215016, 719, ..., WTINC10, WTINC11, WTINC12]

**Table 1:** Variables used as is from the “Εργα” table

Variable Name	Short Description
[ID_Erga]	Primary identification of each project
[GrafioEktelesisErgou]	General understanding of the geographical area the project took place on
[Katigoria]	Generalised project categories
[Meres_Meletis]	Number of days the ‘study’ part of the project lasted (DEDDHE-delay inclusive)
[Sinergio_Meletis]	The study workshop used
[Kostos_Ergatikon_Kataskevis]	The amount of money that construction workers cost the company
[Kostos_Il原因_Kataskevis]	The cost of the materials to be used for the project
[Kostos_Kataskevis]	The amount of money the construction costs to the company (inclusive of [ΚΟΣΤΟΣ_ΥΛΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ] and [ΚΟΣΤΟΣ_ΕΡΓΑΤΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ])
[Kostos_Ergolavikon_Epidosis]	The cost of service contractors
[Ektasi_Ergou]	The project’s scale (small or big)
[Anagi_YS]	Whether or not there’s a need for a substation

[SAP_Typos_Pelati]	5 codes for 5 delineating types of customers
[SAP_Eidos_Aitimatots]	41 codes for 41 delineating types of requests

### 5.1.2 Variables engineered from the “Epya” table

Apart from these variables, it was deemed of paramount importance that some feature engineering occur, as well. To that end, the variables below were implemented:

- [Onoma\_Polis]: Because the city that a project takes place on is thought to be of critical importance as the rates could very well differ amongst different cities, this measure was taken to enforce the ‘city’ information availability. It takes the value of [ΠΟΛΗ] when it is available, otherwise the value of [ΠΟΛΗ\_Y\_Σ], as it should be the same or near enough. Whilst it is not used itself, this variable is in turn used to retrieve the geographical longitude and latitude via a Google’s Geolocation API. It’s a Categorical string variable with a range of [3, 40 ΕΚΚΛΗΣΙΕΣ, 50200, ..., ΩΡΟΛΟΓΙΟ, ΩΡΟΛΟΓΙΟΝ, ΩΡΩΠΙΟΣ].
- [Xaraktirismos\_Ergou]: separates projects into 2 main categories: An investment or a utilisation. When [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] or [SAP\_XΑΡΑΚΤΗΡΙΣΜΟΣ\_ΕΡΓΟΥ] have a value of either ‘41’ or ‘D’ or when [ΚΑΤΗΓΟΡΙΑ] has a value of ‘9300’ or ‘9900’ or ‘9500’ or ‘9700’ or ‘9600’, then its value becomes the former (‘EPENDISI’), whereas if [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] or [SAP\_XΑΡΑΚΤΗΡΙΣΜΟΣ\_ΕΡΓΟΥ] is either ‘42’ or ‘M’ or [ΚΑΤΗΓΟΡΙΑ] has a value of ‘250’, its value becomes the latter (‘EKMETALEFSI’). Any other values are not of interest to us, and hence, are left as NULL so that their corresponding rows get dropped. Depending on its category, a project could be more or less likely to happen as some categories could have significantly higher percentages of completions than others. It’s a Categorical string field with a range of [EKMETALEFSI, EPENDISI].
- [Skopos\_Ergou]: encompasses the 4 different sub-categories of the Investment category and the 12 sub-categories of the Utilisation category. Its value becomes the value of [SAP\_ΣΚΟΠΟΣ\_ΕΡΓΟΥ] if [SAP\_ΣΚΟΠΟΣ\_ΕΡΓΟΥ] takes one of these values: EA, EB, EC, ED, EE, EF, SA, SB, SC, SD, SE, SF, SG, SH, SI, SJ, SK, SL, or it becomes the value of [ΚΩΔ\_ΑΝΑΛΥΣΗΣ] if [ΚΩΔ\_ΑΝΑΛΥΣΗΣ] takes one of these values: EA, EB, EC, ED, EE, EF, SA, SB, SC, SD, SE, SF, SG, SH, SI, SJ, SK, SL, or it becomes ‘EA’ if [ΚΑΤΗΓΟΡΙΑ] = ‘9300’, or it becomes ‘EB’ if [ΚΑΤΗΓΟΡΙΑ] = ‘9900’, or it becomes ‘EC’ if [ΚΑΤΗΓΟΡΙΑ] = ‘9500’, or it becomes ‘ED’ if [ΚΑΤΗΓΟΡΙΑ] = ‘9700’, or it becomes ‘EF’ if [ΚΑΤΗΓΟΡΙΑ] = ‘9600’, or it becomes ‘EE’ if [ΚΑΤΗΓΟΡΙΑ] = ‘9400’ and also [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is either ‘41’ or ‘D’, or [SAP\_XΑΡΑΚΤΗΡΙΣΜΟΣ\_ΕΡΓΟΥ] is either ‘41’ or ‘D’. Any other values are not of interest to us, and hence, are left as NULL so that their corresponding rows get dropped. Depending on its sub-category, a project could be more or less likely to happen as some sub-categories could have significantly higher percentages of completions than others. It’s a Categorical string field with a range of [EA, EB, EC, ED, EE, EF, SA, SB, SC, SD, SE, SF, SG, SH, SI].

- [TimeSeriesDate]: Since the data can be thought of as time-series, this variable separates the projects into different groups based on the year and month of [HMEP\_AITHΣHΣ]. Each value is the year and the quarter of [HMEP\_AITHΣHΣ], e.g. “2017 Q1”. It’s an ordinal string field with a range of [1998 Q2, 1998 Q4, 1999 Q2, ..., 2015 Q4, 2016 Q1, 2016 Q2].
- [Label]: is a Categorical binary field which serves as the data label for the Supervised Learning. It takes a value of ‘0’ when [HMEP\_ΥΠΟΓΡΑΦΗΣ] is NULL, and a value of ‘1’ when [HMEP\_ΥΠΟΓΡΑΦΗΣ] is not NULL.
- [Mel\_Kathisterisi\_Pelati]: This variable was created by taking [MEΛ\_KΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]’s value when available, or else [MEΛ\_XYK\_KΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]’s value was taken instead. This way data availability is maximised with a negligible error. It’s a continuous numerical integer field representing the number of days that the ‘study’ part of the project was delayed by, due to the client. It is likely that patterns will emerge where, for instance, high delay on the client’s part could be indicative of a project to occur as the customer doesn’t cancel it and they instead try to tackle the problems. Its range is [0, 1, 2, ..., 2991, 3253, 4138].
- [MelClientDelay]: [MEΛ\_KΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ] and [MEΛ\_XYK\_KΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ] along with [MEΛ\_ΕΝΔ\_ΚΑΘ\_ΠΕΛΑΤΗ] are used to create this variable for whether or not there was a delay. If [MEΛ\_KΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ] is NULL or its value is 0, and [MEΛ\_XYK\_KΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ] is NULL or its value is 0, and [MEΛ\_ΕΝΔ\_ΚΑΘ\_ΠΕΛΑΤΗ] is 0, then this variable becomes ‘0’, otherwise it becomes ‘1’. It’s a categorical binary field reflecting whether or not there was a delay in the ‘study’ part of the project caused by the client. A reason for the delay could be that the client is to provide certain documents which they have yet to assemble. A pattern that could potentially come to light is projects with a delay have a higher cancellation rate.
- [Mel\_Kathisterisi\_DEH]: This variable was created by taking [MEΛ\_KΑΘΥΣΤΕΡΗΣΗ\_ΔΕΗ]’s value when available, or else [MEΛ\_XYK\_KΑΘΥΣΤΕΡΗΣΗ\_ΔΕΗ]’s value was taken instead. This way data availability is maximised with a negligible error. It’s a continuous numerical integer field representing the number of days that the ‘study’ part of the project was delayed by, due to the company. It is likely that patterns will emerge where, for instance, high delay on the company’s part could be indicative of a project being cancelled as the customer refuses to wait that long a time. Its range is [0, 1, 2, ..., 1332, 1795, 2167].
- [MelDEHDelay]: [MEΛ\_KΑΘΥΣΤΕΡΗΣΗ\_ΔΕΗ] and [MEΛ\_XYK\_KΑΘΥΣΤΕΡΗΣΗ\_ΔΕΗ] along with [MEΛ\_ΕΝΔ\_ΚΑΘ\_ΔΕΗ] are used to create this binary variable for whether or not there was a delay. If [MEΛ\_KΑΘΥΣΤΕΡΗΣΗ\_ΔΕΗ] is NULL or its value is 0, and [MEΛ\_XYK\_KΑΘΥΣΤΕΡΗΣΗ\_ΔΕΗ] is NULL or its value is 0, and [MEΛ\_ΕΝΔ\_ΚΑΘ\_ΔΕΗ] is 0, then this variable becomes ‘0’, otherwise it becomes ‘1’. It’s a categorical binary field reflecting whether or not there was a delay in the ‘study’ part of the project caused by the company. A reason for the

delay could be that some aspect of the project is conflicting with a protocol or another project.

- [Mel\_Kathisterisi\_Triton]: This variable was created by taking [MEA\_KAΘYΣTEPHΣH\_TPITΩN]'s value when available, or else [MEA\_XYK\_KAΘYΣTEPHΣH\_TPITΩN]'s value was taken instead. This way data availability is maximised with a negligible error. It's a continuous numerical integer field representing the number of days that the 'study' part of the project was delayed by, due to external factors. It is likely that patterns will emerge where, for instance, high delay due to third parties could be indicative of a project being cancelled as the customer refuses to wait that long a time. Its range is [0, 8, 9, ..., 1779, 1793, 3023].
- [MelOthersDelay]: [MEA\_KAΘYΣTEPHΣH\_TPITΩN] and [MEA\_XYK\_KAΘYΣTEPHΣH\_TPITΩN] along with [MEA\_ENΔ\_KAΘ\_TPITΩN] are used to create this binary variable for whether or not there was a delay. If [MEA\_KAΘYΣTEPHΣH\_TPITΩN] is NULL or its value is 0, and [MEA\_XYK\_KAΘYΣTEPHΣH\_TPITΩN] is NULL or its value is 0, and [MEA\_ENΔ\_KAΘ\_TPITΩN] is 0, then this variable becomes '0', otherwise it becomes '1'. It's a categorical binary field reflecting whether or not there was a delay in the 'study' part of the project caused by external factors. A reason for the delay could be that certain documents which are vital to the project are behind schedule due to external factors, such as the Forestry.
- [GeoLocX]: Stores the geographical Longitude value given by the geolocation API for the value of the corresponding field in [Onoma\_Polis]. This field will be used for unsupervised learning purposes. It's a categorical numerical integer field with a range of [-122, -118, -112, ..., 130, 138, 153].
- [GeoLocY]: Stores the geographical Latitude value given by the geolocation API for the value of the corresponding field in [Onoma\_Polis]. This field will be used for unsupervised learning purposes. It's a categorical numerical integer field with a range of [-35, -28, -27, ..., 50, 56, 62].
- [Kathisterisi\_AitisisKataxoris]: is a scale continuous numerical integer variable containing the delay in days between the date of the application for the project and the day of its registration. It is calculated as the difference between [HMEP\_AITHΣHΣ] and [HMEP\_KATAXΩPHΣHΣ]. It is likely that patterns will emerge where, for instance, high delay could be indicative of a project being cancelled as the customer refuses to wait that long a time. Its range is [NULL, -2449, -2330, ..., 5545, 21975, 30001].
- [Kathisterisi\_Meletis]: is a continuous numerical integer variable containing the delay in days between the date of the project's registration and the date the its study part finished. It is calculated as the difference between [HMEP\_KATAXΩPHΣHΣ] and [HMEP\_MEΛETHΣ]. It is likely that patterns will emerge where, for instance, high delay could be indicative of a project being cancelled as the customer refuses to wait that long a time. Its range is [NULL, -5140, -4635, ..., 2398, 2410, 2632].

- [Kathisterisi\_Anagelias]: is a continuous numerical integer variable containing the delay in days between the date the project's study part finished and the date the letter informing the customer of its price is sent. It is calculated as the difference between [HMEP\_MEΛETHΣ] and [HMEP\_ANAΓΓΕΛΙΑΣ]. It is likely that patterns will emerge where, for instance, high delay could be indicative of a project being cancelled as the customer refuses to wait that long a time. Its range is [NULL, 0, 1, ..., 996, 1227, 1792].
- [DayOfYearSine]: This feature has very close values for days in a year that are close to each other, for instance January 1<sup>st</sup> and December 31<sup>st</sup>. The variable is created by taking the sine of day of year from [HMEP\_AITHΣHΣ]. It's a continuous numerical decimal variable with a range of [-0.999990339506171, -0.999990339506171, -0.999990206550703, -0.999755839901149, ..., 0.99952109184891, 0.999911860107267, 0.999912259871926].
- [DayOfYearCosine]: This feature has very close values for days in a year that are close to each other, for instance January 1<sup>st</sup> and December 31<sup>st</sup>. The variable is created by taking the cosine of day of year from [HMEP\_AITHΣHΣ]. It's a continuous numerical decimal variable with a range of -0.999999999545659, -0.999960826394637, -0.999843841806507, ..., 0.999843308647691, 0.999961092757309, 1].
- [DayOfYearCartesX]: This feature has very close values for days in a year and years themselves that are close to each other, for instance January 1<sup>st</sup> 2016 and December 31<sup>st</sup> 2015. The variable is created from [HMEP\_AITHΣHΣ] by using the year as a polar coordinate's length taking the sine of day of year as the angle, and then converting them to Cartesian, keeping the X. It's a continuous numerical decimal variable with a range of [-2015.92102601159, -2015.28927122816, -2014.9999990845, ..., 2014.29123559727, 2014.73653992937, 2014.92160190598].
- [DayOfYearCartesY]: This feature has very close values for days in a year and years themselves that are close to each other, for instance January 1<sup>st</sup> 2016 and December 31<sup>st</sup> 2015. The variable is created from [HMEP\_AITHΣHΣ] by using the year as a polar coordinate's length taking the cosine of day of year as the angle, and then converting them to Cartesian, keeping the Y. It's a continuous numerical decimal variable with a range of [-2015.98025640622, -2015.50642949098, -2014.98053410493, ..., 2014.82320364193, 2015.03263969875, 2015.82230997625].

**Table 2:** Variables engineered from the “Εργα” table

Variable Name	Short Description
[Onoma_Polis]	The city that a project takes place on
[Xaraktirismos_Ergou]	Separates projects into categories: investment, utilisation
[Skopos_Ergou]	Separates projects into the 4 Investment sub-categories and the 12 Utilisation sub-categories: EA, EB, EC, ED, EE, EF, SA, SB, SC, SD, SE, SF, SG, SH, SI, SJ, SK, SL

[TimeSeriesDate]	Separates the projects into quarters of their respective years
[Label]	The label of the data for the Supervised Learning (Dependent Variable)
[Mel_Kathisterisi_Pelati]	Number of days the study was held back due to the client
[MelClientDelay]	Binary variable for whether or not there was a delay in the 'study' due to the customer/client
[Mel_Kathisterisi_DEH]	Number of days the study was held back due to the organisation itself
[MelDEHDelay]	Binary variable for whether or not there was a delay in the 'study' due to the organisation itself (DEDDHE)
[Mel_Kathisterisi_Triton]	Number of days the study was held back due to other factors
[MelOthersDelay]	Binary variable for whether or not there was a delay in the 'study' due to other external factors
[GeoLocX]	The geographical Latitude of the corresponding field in [Onoma_Polis]
[GeoLocY]	The geographical Longitude of the corresponding field in [Onoma_Polis]
[Kathisterisi_AitisisKataxorisis]	The delay in days between the date of the project's application and the day of its registration
[Kathisterisi_Meletis]	The delay in days between the date of the project's registration and the date its study part finished
[Kathisterisi_Anagelias]	The delay in days between the date the project's study part finished and the date the letter informing the customer of its price is sent
[DayOfYearSine]	Gives very close values for days in a year that are close to each other
[DayOfYearCosine]	Gives very close values for days in a year that are close to each other
[DayOfYearCartesX]	Gives very close values for days in a year and years themselves that are close to each other
[DayOfYearCartesY]	Gives very close values for days in a year and years themselves that are close to each other

### 5.1.3 Clauses applied to the “Εργα” table

There are certain rules that must be enforced so that the only rows retrieved are those relevant to our ends. This very feature is implemented through SQL Server's *where* clause. The clause envelops the following rules for Row Restriction:

- [ΔΕΗ\_ΠΕΛΑΤΗΣ] is not '1' or NULL: Because if it's '1' then it is DEH's project and the rule states those fields should not be taken into account. If the field is NULL, then it's a client's project and SQL Server requires it is explicitly stated before it pushes rows with NULL values on the clause.

- [AKYPΩΘEN] is '0' or NULL: On any other case the row is erroneous and therefore has to be dropped.
- [Onoma\_Polis] is NOT NULL: For this value is needed for the Clustering procedure.
- [Label] is '1' OR the current date is greater than the date in [HMEP\_AITHΣHΣ] plus 2 months: If the Label is '1', it means that the customer has paid for the project and the date is of little consequence; however, if the Label is '0', it means that the customer has not paid for the project which in turn raises a conundrum: "Is the project cancelled, and the '0' for the Label is actually valid, or does the customer still have time left to pay, rendering the '0' for the Label potentially wrong?" The way to overcome this obstacle is to drop those rows containing projects in which the customer is still undecided or has simply yet to make a payment.
- [Xaraktirismos\_Ergou] is NOT NULL: The only rows relevant to our cause are the ones containing values on this field (as previously discussed); it so follows that this rule be applied to actually retrieve only relevant rows.
- [Skopos\_Ergou] is NOT NULL: The only rows relevant to our cause are the ones containing values on this field (as previously discussed); it so follows that this rule be applied to actually retrieve only relevant rows.
- [HMEPEΣ\_MEΛETHΣ]  $\geq 0$ : A study cannot have ended before it even began, hence erroneous data (noise) are being cleaned.
- [ΚΟΣΤΟΣ\_ΕΡΓΑΤΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ] is NOT NULL: This variable is considered critical for the prediction of the dependent variable. Perhaps only rows that do have a value for this should be retrieved.
- [ΚΟΣΤΟΣ\_ΥΛΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ] is NOT NULL: This variable is considered critical for the prediction of the dependent variable. Perhaps only rows that do have a value for this should be retrieved.
- [ΚΟΣΤΟΣ\_ΚΑΤΑΣΚΕΥΗΣ] is NOT NULL: This variable is considered critical for the prediction of the dependent variable. Perhaps only rows that do have a value for this should be retrieved.
- [ΚΟΣΤΟΣ\_ΕΡΓΟΛΑΒΙΚΩΝ\_ΕΠΙΔΟΣΗΣ] is NOT NULL: This variable is considered critical for the prediction of the dependent variable. Perhaps only rows that do have a value for this should be retrieved.
- [ΚΟΣΤΟΣ\_ΕΡΓΑΤΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]  $> 0$ : A negative cost points to a dismantlement, which are irrelevant projects.
- [ΚΟΣΤΟΣ\_ΥΛΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]  $> 0$ : A negative cost points to a dismantlement, which are irrelevant projects.
- [ΚΟΣΤΟΣ\_ΚΑΤΑΣΚΕΥΗΣ]  $> 0$ : A negative cost points to a dismantlement, which are irrelevant projects.
- [ΚΟΣΤΟΣ\_ΕΡΓΟΛΑΒΙΚΩΝ\_ΕΠΙΔΟΣΗΣ]  $> 0$ : A negative cost points to a dismantlement, which are irrelevant projects.



- [MONADA] is NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [KATHΓΟΡΙΑ] is NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [HMEPEΣ\_MEΛETHΣ] is NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [ΣΥΝΕΡΓΕΙΟ\_MEΛETHΣ] is NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [ΕΚΤΑΣΗ\_ΕΡΓΟΥ] is NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [ΑΝΑΓΚΗ\_ΥΣ] is NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [HMEP\_AITHΣHΣ] IS NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [HMEP\_KATAXΩPHΣHΣ] IS NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [HMEP\_ANATTEΛIAΣ] IS NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.
- [HMEP\_MEΛETHΣ] IS NOT NULL: Missing values on this could potentially lower the predictive accuracy of the machine learning algorithms.

The table below shows the drop rate in rows that each clause caused. The first fifteen clauses are essential, for their purpose is to rid the dataset from project unrelated to the task at hand. The following four clauses were important as they ensure that the most prevalent variables are filled in, especially since they are guaranteed to be present in all future projects. The rest of the clauses are to minimise the missing values whilst making sure to keep the drop rate in check.

The “# Rows” column is fairly self-explanatory; it is the number of remaining rows on the dataset after the clause is applied.

The “Dropped by % Since” column shows the drop rate since the last clause was applied and is calculated as  $\frac{\# \text{ Rows from previous clause} - \# \text{ Rows after current clause}}{\# \text{ Rows from previous clause}} \cdot 100$ .

The “% Left since Original” column shows the percentage of rows left to the dataset compared to the original number of rows (432646) after the new clause is applied, and it's calculated as  $\frac{\# \text{ Rows after current clause}}{\text{Original \# rows}} \cdot 100$ .

**Table 3:** Drop Rate per Clause

Clause:	Rows #:	Dropped by % Since	% Left since Original
Before any clauses	432646	0	100
[ΔΕΗ_ΠΕΛΑΤΗΣ] is not '1' or NULL	249528	42.32513	57.67487
[ΑΚΥΡΩΘΕΝ] is '0' or NULL	239555	3.996746	55.36975
[Onoma_Polis] is NOT NULL	216171	9.761433	49.96487
[Label] is '1' OR the current date is greater than the date in [ΗΜΕΡ_ΑΙΤΗΣΗΣ] plus 2 months	213535	1.219405	49.35559
[Χαρακτηρισμος_Ergou] is NOT NULL	204771	4.104245	47.32992
[Skopos_Ergou] is NOT NULL	190401	7.017595	44.0085
[ΗΜΕΡΕΣ_ΜΕΛΕΤΗΣ] >= 0	182691	4.049348	42.22644
[ΚΟΣΤΟΣ_ΕΡΓΑΤΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ] is NOT NULL	159326	12.78935	36.82595
[ΚΟΣΤΟΣ_ΥΛΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ] is NOT NULL	155915	2.140894	36.03755
[ΚΟΣΤΟΣ_ΚΑΤΑΣΚΕΥΗΣ] is NOT NULL	155740	0.112241	35.9971
[ΚΟΣΤΟΣ_ΕΡΓΟΛΑΒΙΚΩΝ_ΕΠΙΔΟΣΗΣ] is NOT NULL	153739	1.284834	35.53459
[ΚΟΣΤΟΣ_ΕΡΓΑΤΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ] > 0	153577	0.105373	35.49715
[ΚΟΣΤΟΣ_ΥΛΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ] > 0	150520	1.990532	34.79057
[ΚΟΣΤΟΣ_ΚΑΤΑΣΚΕΥΗΣ] > 0	150486	0.022588	34.78271
[ΚΟΣΤΟΣ_ΕΡΓΟΛΑΒΙΚΩΝ_ΕΠΙΔΟΣΗΣ] > 0	150485	0.000665	34.78248
[ΜΟΝΑΔΑ] is NOT NULL	150485	0	34.78248
[ΚΑΤΗΓΟΡΙΑ] is NOT NULL	146759	2.475994	33.92127
[ΗΜΕΡΕΣ_ΜΕΛΕΤΗΣ] is NOT NULL	146759	0	33.92127
[ΣΥΝΕΡΓΕΙΟ_ΜΕΛΕΤΗΣ] is NOT NULL	146722	0.025211	33.91271
[ΕΚΤΑΣΗ_ΕΡΓΟΥ] is NOT NULL	134618	8.249615	31.11505
[ΑΝΑΓΚΗ_ΥΣ] is NOT NULL	134618	0	31.11505
[ΗΜΕΡ_ΑΙΤΗΣΗΣ] IS NOT NULL	134480	0.102512	31.08315
[ΗΜΕΡ_ΚΑΤΑΧΩΡΗΣΗΣ] IS NOT NULL	134479	0.000744	31.08292

[HMEP_ΑΝΑΓΓΕΛΙΑΣ] IS NOT NULL	133122	1.009079	30.76927
[HMEP_ΜΕΛΕΤΗΣ] IS NOT NULL	133098	0.018029	30.76372

The tables below show how the dataset we're working with looks like post-clauses and post-transformations.

**Table 4:** SQL View, General Information.

Columns:	33
Rows:	133098

A summary of each of SQL View's Quantitative variables, including their type, minimum value, maximum value, number of values present, number of missing values, percentage of values present, mean value and standard deviation is illustrated on the table below.

The minimum and maximum values give us the variable's range, which is essential to understanding the nature of the data. These values can vastly differ from the original ones as the distribution has changed. Take "Meres\_Meletis" for instance; its values used to range from -37944 to 37924, whilst now their minimum number is 0 as something having lasted a negative amount of days makes no sense and nonsensical values have been cleared; its range is 0 to 30000.

The Valid, Missing, and Valid% can be used as an indicator of how beneficial the state of a variable's data is, where, for instance, remember the "ΚΟΣΤΟΣ\_ΜΕΛΕΤΗΤΗ" one; an otherwise critical variable, was deemed unfit for use solely on the grounds of too many missing values. It is now no more part of the dataset.

The Mean and Standard Deviation try to quantify the knowledge of how values are distributed in each variable. Take "Mel\_Kathisterisi\_Pelati" for example; we know that, on average, the 'study' part of a project (which is inclusive of projects with 0 delay), was delayed by 12 days (mean value of 1.197817e+01) by the client, whilst the low standard deviation means that in large, the delay on each case revolved around its mean value.

**Table 5:** Summary of Picked Quantitative Variables

Name	Type	Min	Max	Valid	Missing	Valid%	Mean	Std. Dev.
Mel_Kathisterisi_Pelati	Integer	0	4138	133098	0	100	1.190948e+01	7.519879e+01
Mel_Kathisterisi_DEH	Integer	0	2167	133098	0	100	1.155089e-01	9.123177e+00
Mel_Kathisterisi_Triton	Integer	0	3023	133098	0	100	7.112053e-01	2.323592e+01
Meres_Meletis	Integer	0	30000	133098	0	100	3.241717e+01	1.428256e+02
Kostos_Ergatikon_Kataskevis	Decimal	1.7	1489345.8	133098	0	100	3.397392e+03	1.554327e+04
Kostos_Ilikon_Kataskevis	Decimal	0.01	2587043	133098	0	100	2.715408e+03	1.060648e+04
Kostos_Kataskevis	Decimal	1	16797610	133098	0	100	6.349360e+03	6.549886e+04
Kostos_Ergolavikon_Epidosis	Decimal	1.4	11176571	133098	0	100	3.073315e+03	3.361315e+04
DayOfYearSine	Decimal	-1.0	1.0	133098	0	100	6.386165e-03	7.090093e-01
DayOfYearCosine	Decimal	-1.0	1.0	133098	0	100	-1.257044e-02	7.050635e-01
DayOfYearCartesX	Decimal	-2015.9	2016	133098	0	100	-2.528794e+01	1.417915e+03
DayOfYearCartesY	Decimal	-2016	2015.8	133098	0	100	1.283772e+01	1.425859e+03
Kathisterisi_AitisisKataxorisis	Integer	-2449	30001	133098	0	100	4.620410e+01	1.953177e+02
Kathisterisi_Meletis	Integer	-5140	2632	133098	0	100	-4.112939e+00	1.237617e+02
Kathisterisi_Anagelias	Integer	0	1792	133098	0	100	4.573998e-01	1.069004e+01

A summary of each of SQL View's Qualitative variables, including their type, number of factors, number of values present, number of missing values and percentage of values are highlighted on the table below.

The type depicts how the values are saved, and what we'd expect the SQL Server to return to a variable, so that an Integer means that we're only expecting whole (integer) numbers, a Double means that fractions of a number can also be returned, a binary means that either a '0/False' or an '1/True' is returned, and a String means that a text is returned.

The number of factors is basically how many different values are in each respective variable.

The Valid, Missing, and Valid% can be used as an indicator of how beneficial the state of a variable's data is. Should the valid percentage be significantly low, for instance, a variable will, in most cases, be disregarded.

**Table 6:** Summary of Picked Qualitative Variables

Name	Type	Factor Levels	Valid	Missing	Valid%
Label	Binary	2	133098	0	100
ID_Erga	Integer	N/A*	133098	0	100
TimeSeriesDate	String	N/A*	133098	0	100
GrafioEktelesisErgou	Integer	127	133098	0	100
Onoma_Polis	String	N/A*	133098	0	100
GeoLocX	Integer	N/A*	133098	0	100
GeoLocY	Integer	N/A*	133098	0	100
Kategoria	String	77	133098	0	100
Xaraktirismos_Ergou	String	2	133098	0	100
Skopos_Ergou	String	15	133098	0	100
MelClientDelay	Binary	2	133098	0	100
MelDEHDelay	Binary	2	133098	0	100
MelOthersDelay	Binary	2	133098	0	100
Sinergio_Meletis	String	3	133098	0	100
Ektasi_Ergou	Binary	2	133098	0	100
Anagi_YS	Binary	2	133098	0	100
SAP_Typos_Pelati	String	2	28207	104891	~26.9
SAP_Eidos_Aitimatatos	String	33	21428	111670	~19.2

The N/A\* notation on the 'Factor Levels' column suggests that the number of factors are potentially changing with each entry, and as such, their current value does not matter.

The SQL Views described above were created with SQL code, a sample of which is:

```
USE YLIKA_KOSTOL
GO

CREATE VIEW [dbo].[v4Erga]
ALTER VIEW [dbo].[v4Erga]
AS
SELECT [Label]
      ,[ID] AS [ID_Erga]
      ,[TimeSeriesDate]
      ,[MONADA] AS [GrafioEktelesisErgou]
      ,[Onoma_Polis]
      ,[GeoLocX]
      ,[GeoLocY]

      [ . . . ]

      INNER JOIN (SELECT [ID] AS tmp_ID1,
                        (SELECT CASE
                          WHEN
UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))) = 'SG' OR
UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))) = 'SH' OR
UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))) = 'SI' OR
UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))) = 'SJ' OR
UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))) = 'SK' OR
UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))) = 'SL' THEN

                        (UPPER(LTRIM(RTRIM([ΚΩΔ_ΑΝΑΛΥΣΗΣ]))))
                          WHEN ([ΚΩΔ_ΛΟΓΑΡΙΑΣΜΟΥ] = '41'
OR UPPER(LTRIM(RTRIM([ΚΩΔ_ΛΟΓΑΡΙΑΣΜΟΥ]))) = 'D') AND [ΚΩΔ_ΑΝΑΛΥΣΗΣ]
LIKE '324%' THEN

                        'EB'

                        [ . . . ]

      WHERE (([ΔΕΗ_ΠΕΛΑΤΗΣ] <> 1 OR [ΔΕΗ_ΠΕΛΑΤΗΣ] IS NULL) AND --249528
--Getting only Clients --1=DEH
      ([AKYPOΘEN] = 0 OR [AKYPOΘEN] IS NULL) AND --239555 --
Akyrothen <> 0 = false record, ergo non needed
      [Onoma_Polis] IS NOT NULL AND --216171 --Needed for the
Clustering
      --([Hmerominia] IS NOT NULL) AND --NOT Null so that we
can have an accurate Label --The difference between those dates is
negligible, so if the one is NULL, the other is used
      ([Label] = 1 OR DATEADD(MONTH,2,[HMEP_ΑΙΤΗΣΗΣ]) <
'2016-07-06') AND --213535 --Getting the results with actual label
of 0 or 1, not as of yet undecided
      [Xaraktirismos_Ergou] IS NOT NULL AND --204771 --The
database encompasses a wide range of things but we only care for
projects. Those are the 41,D,42,M. The NOT NULL means that that's
all the query retrieves
      [Skopos_Ergou] IS NOT NULL AND

      [ . . . ]

)
```

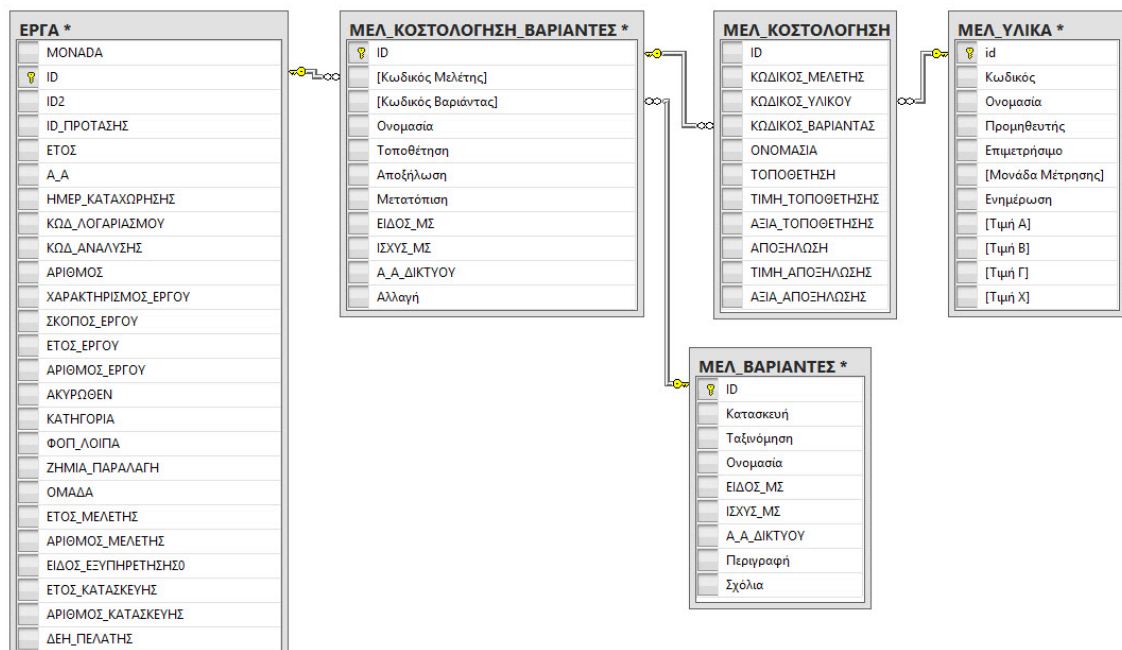




## 5.2 The Original Database

### 5.2.1 General Information

The data provided came in the form of a SQL Server Database, in which there is a total of 5 SQL tables, namely, “ΕΡΓΑ”, “ΜΕΛ\_ΚΟΣΤΟΛΟΓΗΣΗ\_ΒΑΡΙΑΝΤΕΣ”, “ΜΕΛ\_ΒΑΡΙΑΝΤΕΣ”, “ΜΕΛ\_ΚΟΣΤΟΛΟΓΗΣΗ\_ΑΝΑΛΥΣΗ\_ΥΛΙΚΑ”, and “ΜΕΛ\_ΥΛΙΚΑ”. The data represent all projects of DEDDHE from the beginning till July 6<sup>th</sup>, 2016. The relationships between these tables are depicted on the figure below.



**Figure 3:** SQL Database ER Diagram

The “ΕΡΓΑ” table has more columns than shown in the figure above; in fact, it has 104 ones, making it difficult to show them all in one picture. Their names are mentioned on following chapters. Each row represents a project that has been, hasn’t been, or will be undertaken, adding up to a total of 432,646 different projects.

The “ΜΕΛ\_ΚΟΣΤΟΛΟΓΗΣΗ\_ΒΑΡΙΑΝΤΕΣ” consists of 11 columns and 2,533,079 rows, with each row representing basic information for a set of items used for the project. Each row in “ΕΡΓΑ” can be connected with many rows in “ΜΕΛ\_ΚΟΣΤΟΛΟΓΗΣΗ\_ΒΑΡΙΑΝΤΕΣ” as each project can use more than one set of items, and as a matter of fact, it can also partially use sets, as well. There is a finite, explicitly declared number of said sets, each with its own descriptive name, contained in the “ΜΕΛ\_ΒΑΡΙΑΝΤΕΣ” table.

The “MEA\_BAPIANTEΣ” table is comprised of 9 columns and 3,385 rows, with each row representing a different set of items that projects can use. Each row in “MEA\_BAPIANTEΣ” is connected with many rows in “MEA\_ΚΟΣΤΟΛΟΓΗΣΗ\_BAPIANTEΣ”.

The “MEA\_ΚΟΣΤΟΛΟΓΙΣΗ\_ΑΝΑΛΥΣΗ\_ΥΛΙΚΑ” has a set of 11 columns and 17,134,977 rows, with each row representing an item used for the project which was part of a specific set of items, along with relevant information, such as, how much of it or how many of them were used, or how much it cost. Since each project points to potentially many sets of items and each set of items points to many items, it comes as no surprise that this table’s row count far outnumbers the row count of any other table.

The “MEA\_ΥΛΙΚΑ” table consists of 11 columns and 3,753 rows, with each row referencing a single item along with information about it, such as its name, price or SKU.

## 5.2.2 Descriptive Statistics

The original number of columns, the number of columns used for the SQL View and the number of rows of the “ΕΠΓΑ” table is as shown in the table below.

**Table 7:** “ΕΠΓΑ”, General Information.

<b>Columns/Variables/Features:</b>	104
<b>Columns of Interest:</b>	35 [Marked with ‘*’]
<b>Rows:</b>	432646

A summary of each of ΕΠΓΑ’s Quantitative variables, including their type, minimum value, maximum value, number of values present, number of missing values, percentage of values present, mean value and standard deviation, is illustrated on the table below.

The minimum and maximum values give us the variable’s range, which is essential to understanding the nature of the data. Take “ΗΜΕΡΕΣ\_ΜΕΛΕΤΗΣ” for instance; its values range from -37944 to 37924, which is quite baffling as it gives us the total number of days that the ‘study’ part of the project lasted. Taking it literally, it would mean that had I just started the ‘study’ part today, I would have finished about 104 years before I began. Information such as this illuminates something about the nature of the data which begs for further investigation.

The Valid, Missing, and Valid% can be used as an indicator of how beneficial the state of a variable’s data is, where, for instance, the “ΚΟΣΤΟΣ\_ΜΕΛΕΤΗΣ” one, an

otherwise critical variable, was deemed unfit for use solely on the grounds of too many missing values. To wit, only about 0.2% of its data are filled in.

The Mean and Standard Deviation try to quantify the knowledge of how values are distributed in each variable. Take “ΜΕΛ\_ΚΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ” for example; we know that, on average, the ‘study’ part of a project (of those projects who did experience a delay due to the client, as witnessed by the 94.2% missing values), was delayed by 117 days (mean value of 1.168564e+02) by the client, whilst the low standard deviation means that the delay of each case revolved predominantly around its mean value.

**Table 8:** “ΕΡΓΑ”, Quantitative Variables Summary

Name	Type	Min	Max	Valid	Missing	Valid%	Mean	Std. Dev.
ΜΕΛ_ΚΑΘΥΣΤΕΡΗΣΗ_ΠΕΛΑΤΗ*	Integer	0	4138	25123	407523	~5.8	1.168564e+02	2.379374e+02
ΜΕΛ_ΧΥΚ_ΚΑΘΥΣΤΕΡΗΣΗ_ΠΕΛΑΤΗ*	Integer	0	2875	25148	407498	~5.8	7.990842e+01	1.627119e+02
ΜΕΛ_ΚΑΘΥΣΤΕΡΗΣΗ_ΔΕΗ*	Integer	0	3122	639	432007	~0.1	2.016463e+02	4.384191e+02
ΜΕΛ_ΧΥΚ_ΚΑΘΥΣΤΕΡΗΣΗ_ΔΕΗ*	Integer	0	2141	643	432003	~0.1	1.371089e+02	2.994333e+02
ΜΕΛ_ΚΑΘΥΣΤΕΡΗΣΗ_ΤΡΙΤΩΝ*	Integer	0	3023	821	431825	~0.2	2.522704e+02	3.252805e+02
ΜΕΛ_ΧΥΚ_ΚΑΘΥΣΤΕΡΗΣΗ_ΤΡΙΤΩΝ*	Integer	0	2072	830	431816	~0.2	1.712277e+02	2.223619e+02
ΗΜΕΡΕΣ_ΜΕΛΕΤΗΣ*	Integer	-37944	37924	417194	15452	~96.4	-7.141975e+00	1.316398e+03
ΕΡΓ_ΗΜΕΡΕΣ_ΜΕΛΕΤΗΣ	Integer	-3003	21326	414881	17765	~95.9	1.801832e+01	8.295884e+01
ΚΟΣΤΟΣ_ΜΕΛΕΤΗΤΗ	Decimal	-91	612131500	984	431662	~0.2	6.228202e+05	1.951402e+07
ΚΟΣΤΟΣ_ΕΡΓΑΤΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ*	Decimal	-2745.5	60840609	350209	82437	~80.9	3.613953e+03	1.333233e+05
ΚΟΣΤΟΣ_ΥΛΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ*	Decimal	-301792	133449073	339522	93124	~78.5	2.947045e+03	2.493976e+05
ΚΟΣΤΟΣ_ΚΑΤΑΣΚΕΥΗΣ*	Decimal	-299734	194289682	351260	81386	~81.2	6.507491e+03	3.648899e+05
ΚΟΣΤΟΣ_ΕΡΓΟΛΑΒΙΚΩΝ_ΕΠΙΔΟΣΗΣ*	Decimal	-3572	694214876	350224	82422	~80.9	5.187676e+03	1.178611e+06
ΚΑΤ_ΚΑΘΥΣΤΕΡΗΣΗ_ΠΕΛΑΤΗ	Integer	0	2595	18163	414483	~4.2	1.328172e+02	2.179043e+02
ΚΑΤ_ΧΥΚ_ΚΑΘΥΣΤΕΡΗΣΗ_ΠΕΛΑΤΗ	Integer	0	1774	18185	414461	~4.2	9.082425e+01	1.492141e+02
ΚΑΤ_ΚΑΘΥΣΤΕΡΗΣΗ_ΔΕΗ	Integer	0	1920	3339	429307	~0.8	1.438428e+02	2.127396e+02
ΚΑΤ_ΧΥΚ_ΚΑΘΥΣΤΕΡΗΣΗ_ΔΕΗ	Integer	0	1317	3340	429306	~0.8	9.839790e+01	1.455905e+02
ΚΑΤ_ΚΑΘΥΣΤΕΡΗΣΗ_ΤΡΙΤΩΝ	Integer	0	274	149	432497	~0.0	1.996644e+01	3.803811e+01
ΚΑΤ_ΧΥΚ_ΚΑΘΥΣΤΕΡΗΣΗ_ΤΡΙΤΩΝ	Integer	0	189	152	432494	~0.0	1.334868e+01	2.600618e+01
ΗΜΕΡΕΣ_ΕΚΤΕΛΕΣΗΣ	Integer	-1910	4542	318895	113751	~73.7	4.958026e+01	1.291759e+02
ΕΡΓ_ΗΜΕΡΕΣ_ΕΚΤΕΛΕΣΗΣ	Integer	-1910	3108	318897	113749	~73.7	3.400196e+01	8.854809e+01
ΑΠΟΛ_ΚΟΣΤΟΣ_ΕΡΓΑΤΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ	Decimal	-8569	23217393	30944	401702	~7.2	3.651256e+03	1.325670e+05
ΑΠΟΛ_ΚΟΣΤΟΣ_ΥΛΙΚΩΝ_ΚΑΤΑΣΚΕΥΗΣ	Decimal	-28554	30602010	27945	404701	~6.5	2.813482e+03	1.835455e+05
ΑΠΟΛ_ΚΟΣΤΟΣ_ΚΑΤΑΣΚΕΥΗΣ	Decimal	-14410	25002008	28188	404458	~6.5	5.630270e+03	1.507156e+05

A summary of each of EPTA's Qualitative variables, including their statistical type, programming type, number of factors, number of values present, number of missing values and percentage of values are highlighted on the table below.

The statistical type portrays the different notions between Categorical, Ordinal, Binary, and Date values.

The programming type depicts how the values are saved, and what we'd expect the SQL Server to return to a variable, so that an Integer means that we're only expecting whole (integer) numbers, a Double means that fractions of a number can also be returned, a binary means that either a '0/False' or an '1/True' is returned, a Date means that we're expecting a date with or without a time, and a String means that a text is returned.

The number of factors is basically how many different values are in each respective variable.

The Valid, Missing, and Valid% can be used as an indicator of how beneficial the state of a variable's data is. Should the valid percentage be significantly low, for example, a variable will most likely be disregarded.

**Table 9:** “ΕΡΓΑ”, Qualitative Variables Summary

Name	Type	Variable	Factors	Valid	Missing	Valid%
MONADA*	Categorical	Integer	161	432646	0	100
ID*	Categorical	Integer	432646	432646	0	100
ID2	Categorical	Integer	3755	3755	428891	~0.9
ID_ΠΡΟΤΑΣΗΣ	Categorical	Integer	7173	201731	230915	~46.6
ΕΤΟΣ	Date	Date	90	432131	515	~99.9
Α_Α	Categorical	Integer	9820	429442	3204	~99.3
ΗΜΕΡ_ΚΑΤΑΧΩΡΗΣΗΣ*	Date	Date	424720	428067	4579	~98.9
ΚΩΔ_ΛΟΓΑΡΙΑΣΜΟΥ*	Categorical	String	4	404347	28299	~93.5
ΚΩΔ_ΑΝΑΛΥΣΗΣ*	Categorical	String	1163	396426	36220	~91.6
ΑΡΙΘΜΟΣ	Categorical	Integer	4457	374670	57976	~86.6
ΧΑΡΑΚΤΗΡΙΣΜΟΣ_ΕΡΓΟΥ	N/A	N/A	N/A	0	432646	0
ΣΚΟΠΟΣ_ΕΡΓΟΥ	N/A	N/A	N/A	0	432646	0
ΕΤΟΣ_ΕΡΓΟΥ	N/A	N/A	N/A	0	432646	0
ΑΡΙΘΜΟΣ_ΕΡΓΟΥ	N/A	N/A	N/A	0	432646	0
ΑΚΥΡΩΘΕΝ*	Categorical	Binary	2	432646	0	100
ΚΑΤΗΓΟΡΙΑ*	Categorical	String	266	384144	48502	~88.8
ΦΟΠ_ΛΟΠΙΑ	Categorical	Integer	3	175050	257596	~40.8
ΖΗΜΙΑ_ΠΑΡΑΛΑΓΗ	Categorical	Integer	3	203613	229033	~47.1
ΟΜΑΔΑ	Categorical	String	23220	291180	141466	~67.3
ΕΤΟΣ_ΜΕΛΕΤΗΣ	Date	String	355	417079	15567	~96.4
ΑΡΙΘΜΟΣ_ΜΕΛΕΤΗΣ	Categorical	Integer	10243	375627	57019	~86.8
ΕΙΔΟΣ_ΕΞΥΠΗΡΕΤΗΣΗΣ0	Categorical	Integer	2	428512	4134	~99.0
ΕΤΟΣ_ΚΑΤΑΣΚΕΥΗΣ	Date	String	1216	285107	147539	~65.9
ΑΡΙΘΜΟΣ_ΚΑΤΑΣΚΕΥΗΣ	Categorical	Integer	9827	286301	146345	~66.2

ΔΕΗ_ΠΕΛΑΤΗΣ*	Categorical	Integer	3	416164	16482	~96.2
ΑΡ_ΠΡΩΤΟΚΟΛΟΥ_ΠΕΛΑΤΗ	Categorical	String	71238	86869	345777	~20.1
ΑΡ_ΠΡΩΤΟΚΟΛΟΥ_ΔΕΗ	Categorical	String	130247	150089	282557	~34.7
ΝΕΟ_ΠΑΡΑΛΛΑΓΗ	Categorical	Integer	2	753	431893	~0.17
ΑΡ_ΠΑΡΟΧΗΣ	Categorical	String	5406	6226	426420	~1.44
ΠΕΡΙΓΡΑΦΗ	Categorical	String	412442	412442	20204	~95.3
ΗΜΕΡ_ΑΙΤΗΣΗΣ*	Date	Date	4865	424123	8523	~98.0
ΑΡΧΗ_ΠΑΡΑΤΗΡ_ΜΕΛΕΤΗΣ	Date	Date	148	209	432437	~0.05
ΤΕΛΟΣ_ΠΑΡΑΤΗΡ_ΜΕΛΕΤΗΣ	Date	Date	128	208	432438	~0.05
ΜΕΛ_ΕΝΔ_ΚΑΘ_ΠΕΛΑΤΗ*	Binary	Binary	2	432646	0	100
ΜΕΛ_ΕΝΔ_ΚΑΘ_ΔΕΗ*	Binary	Binary	2	432646	0	100
ΜΕΛ_ΕΝΔ_ΚΑΘ_ΤΡΙΤΩΝ*	Binary	Binary	2	432646	0	100
ΗΜΕΡ_ΜΕΛΕΤΗΣ*	Date	Date	4673	398446	34200	~92.1
ΗΜΕΡ_ΑΝΑΓΓΕΛΙΑΣ*	Date	Date	4403	392357	40289	~90.7
ΣΥΝΕΡΓΕΙΟ_ΜΕΛΕΤΗΣ*	Categorical	String	3	432551	95	~100
ΜΕΛΕΤΗΤΗΣ	Categorical	String	1913	417610	15036	~96.5
ΑΩ_ΚΑΤΑΣΚΕΥΗΣ	Categorical	Double	79	3445	429201	~0.8
ΑΩ_ΜΕΛΕΤΗΣ	Categorical	Double	143	228	432418	~0.05
ΗΜΕΡ_ΥΠΟΓΡΑΦΗΣ*	Date	Date	4504	301564	131082	~69.7
ΗΜΕΡ_ΠΑΡΑΛΑΒΗΣ	N/A	N/A	N/A	0	432646	0
ΕΙΔΟΣ_ΕΞΥΠΗΡΕΤΗΣΗΣ	Categorical	Integer	2	431269	1377	~99.7
ΤΙΤΛΟΣ_ΕΡΓΟΥ	Categorical	String	271493	386946	45700	~89.4
ΣΥΜΒ_ΗΜΕΡ_ΕΝΑΡΞΗΣ	Date	Date	4271	301887	130759	~69.8
ΣΥΜΒ_ΗΜΕΡ_ΕΚΤΕΛΕΣΗΣ	Date	Date	4078	301328	131318	~69.6
ΑΡΧΗ_ΠΑΡΑΤΗΡ_ΕΚΤΕΛΕΣΗΣ	Date	Date	17	20	432626	~0.0
ΤΕΛΟΣ_ΠΑΡΑΤΗΡ_ΕΚΤΕΛΕΣΗΣ	Date	Date	20	20	432626	~0.0
ΚΑΤ_ΕΝΔ_ΚΑΘ_ΠΕΛΑΤΗ	Binary	Binary	2	432646	0	100
ΚΑΤ_ΕΝΔ_ΚΑΘ_ΔΕΗ	Binary	Binary	2	432646	0	100

ΚΑΤ_ΕΝΔ_ΚΑΘ_ΤΡΙΤΩΝ	Binary	Binary	1	432646	0	100
ΗΜΕΡ_ΕΝΑΡΞΗΣ	Date	Date	4669	263620	169026	~60.9
ΗΜΕΡ_ΕΚΤΕΛΕΣΗΣ	Date	Date	4490	292571	140075	~67.6
ΠΟΣ_ΕΚΤΕΛΕΣΗΣ	Categorical	Integer	83	295674	136972	~68.3
ΠΙΣΤΟΠΟΙΗΣΗ	Binary	Binary	2	432646	0	100
ΗΜΕΡ_ΠΙΣΤΟΠΟΙΗΣΗΣ	Date	Date	4214	159561	273085	~36.9
ΣΥΝΕΡΓΕΙΟ_ΚΑΤΑΣΚΕΥΗΣ	Categorical	String	3	432388	258	~99.9
ΚΑΤΑΣΚΕΥΑΣΤΗΣ	Categorical	String	641	419055	13591	~96.9
ΕΚΤΥΠΩΣΗ_ΠΡΩΤΟΚΟΛΟΥ	Binary	Binary	2	432646	0	100
ΟΝΟΜΑΤΕΠΩΝΥΜΟ	Categorical	String	432646	432646	0	100
ΔΙΕΥΘΥΝΣΗ	Categorical	String	432646	432646	0	100
ΠΟΛΗ*	Categorical	String	10932	371353	61293	~85.8
ΠΟΛΗ_Υ_Σ*	Categorical	String	7155	300145	132501	~69.4
Υ_Σ	Categorical	String	30874	223714	208932	~51.7
ΤΗΛΕΦΩΝΟ	N/A	N/A	N/A	82810	349836	~80.7
ΠΑΡΑΤΗΡΗΣΕΙΣ	Categorical	String	93066	122364	310282	~28.3
ΠΑΡΑΤΗΡΗΣΕΙΣ2	Categorical	String	27033	42124	390522	~9.7
ΕΚΤΑΣΗ_ΕΡΓΟΥ*	Categorical	Integer	2	403064	29582	~93.2
ΑΝΑΓΚΗ_ΥΣ*	Binary	Binary	2	432646	0	100
ΔΙΚΤΥΟ_XT_MT	Categorical	String	3	149664	282982	~34.6
SAP_ΑΡΙΘΜΟΣ_ΕΡΓΟΥ	Categorical	String	89051	92317	340329	~21.3
SAP_ΧΑΡΑΚΤΗΡΙΣΜΟΣ_ΕΡΓΟΥ*	Categorical	String	2	81927	350719	~18.9
SAP_ΤΥΠΟΣ_ΠΕΛΑΤΗ*	Categorical	String	5	95551	337095	~22.1
SAP_ΕΙΔΟΣ_ΑΙΤΗΜΑΤΟΣ*	Categorical	String	41	73910	358736	~17.1
SAP_ΣΚΟΠΟΣ_ΕΡΓΟΥ*	Categorical	String	25	81622	351024	~18.9
UserName	N/A	N/A	N/A	426478	6168	~98.4
UpDate	N/A	N/A	N/A	426481	6165	~98.6
Test	Categorical	String	432646	432646	432646	100



### 5.2.3 Column Selection

There are 104 variables on the “Εργα” table, 13 of which are of use for the task at hand as is, 20 of which are used for feature engineering purposes, 2 of which are used solely to set clauses (restriction) on row retrieval, the rest of which are simply proven fruitless. The variables, their description and the reason each was or was not picked, are as follows.

- [MONADA]: is picked (renamed [GrafioEktelesisErgou]), for it provides a general understanding of the geographical area the project took place on. Each region is coded as a numerical integer field of varied length with a range of [5, 505, 5050000, ..., 528, 5280006, 528000606], making it a Categorical variable. If one were to enquire into Central Thessaloniki in particular, one would retrieve rows where [MONADA] equals 5060006. The percentage per region of whether or not a project gets the go could very well be subject to its whereabouts. That being said, the 161 factors of this variable are very many indeed, and under no circumstances would I be taken aback should it turn out it contributes little to nil.
- [ID]: is the primary identification of each project which was picked (renamed [ID\_Erga]). Whilst it is utterly useless for any prediction purposes, it is vital for data viewing, manipulation, and interconnectivity reasons. It is a numerical integer field with a range of [12, 13, 14, ..., 564033, 564034, 564035], serving as a Categorical variable.
- [ID2]: is not picked, as not only is its use unknown, but it's also mostly missing with only ~0.9% data availability. It is a numerical integer field, referencing a secondary identification number with a range of [NULL, 2882, 2883, ..., 117522, 117523, 117524], serving as a Categorical variable.
- [ID\_ΠΠΟΤΑΣΗΣ]: is not picked, as it provides no useful data or relationships whatsoever; it's also mostly missing with ~46.6% data availability. It is a numerical integer field, reflecting the proposal's identification number, with a range of [NULL, 19204, 19289, ..., 509011104, 509011200, 509011300], serving as a Categorical variable.
- [ΕΤΟΣ]: is not picked, as its essence, the project's year, is subsumed under [TimeSeriesDate]'s paradigm, which will be mentioned below. It is a string field, with a range of [NULL, /, 0, ..., X70, X70Σ, X70X], serving as a Categorical variable. It should be noted that, although it's not immediately visible looking at

its *noisy* range, somewhere after the 0 and before the X70, there are actual year values as well. Since the dataset is a time series, the notion of a date-time is integral to it.

- [A\_A]: is not picked, as its use is unknown and therefore yields no insight. It is a numerical integer field with a range of [NULL, 0, 1, ..., 32362, 32363, 32364], serving as a Categorical variable. Taking the fact that it only has 9820 different factors into account, from 0 to 32364, whereas the projects have a count of 432646, it's hard to conclude that it represents a project's serial number like its name seems to hints (in Greek, that is).
- [HMEP\_KATAXΩPHΣHΣ]: is not picked, for there's no use for dates in the classification. However, it is used for feature engineering, creating new variables with time differences. It is a date field with a range of [NULL, 2001-01-01 04:16:13.000, 2004-02-25 13:06:33.000, ..., 2016-05-28 09:13:42.000, 2016-05-28 10:33:51.000, 2016-05-28 10:38:48.000], serving as a Categorical variable. It is the 2<sup>nd</sup> date in line, part of the 11 date variables that describe a project's full cycle; it refers to the date the project was registered to the system, and the 2 new features engineered by it are [Kathisterisi\_AitisiKataxoris], the delay between the application and actual registration of the project, and [Kathisterisi\_Meletis], the delay between the registration and the date of completion of the 'study' part.
- [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ]: is not picked, as it holds only part of the information intended for it, which has been fragmented into old and new variables. It is, however, amongst the set of variables used to feature engineer the information originally designated for this one. It is a string variable, portraying the project's main category, whilst its range is [NULL, 41, 42, D, M]. In this Categorical field, Null designates missing values, 41 and D stand for Investment, and 42 and M stand for Utilisation.
- [ΚΩΔ\_ΑΝΑΛΥΣΗΣ]: is not picked, as it holds only part of the information intended for it, which has been fragmented into old and new variables. It is, however, amongst the set of variables used to feature engineer the information originally designated for this one. It is a string field whose range is [NULL, '335', '33645', ..., 'M/sdenys01', 'M/sfenys01', 'Msamten01']. This Categorical variable reflects the project's sub-category via the following rules: If it starts with '32' and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 41, then the sub-

category is Electrification for Consumers. If it starts with 324 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 41, then the sub-category is Electrification for Producers. If it starts with 33 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 41, then the sub-category is Variant. If it starts with 316 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 41, then the sub-category is Aesthetical Upgrade. If it starts with 34 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 41, then the sub-category is Amplification. If it starts with 336 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 41, then the sub-category is Layout Reconfiguration on a Substation. If it starts with 321 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 42, then the sub-category is Localised Maintenance. If it starts with 322 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 42, then the sub-category is Preventative Maintenance. If it starts with 33 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 42, then the sub-category is Network Damage Maintenance. If it starts with 36 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 42, then the sub-category is Network Removal. If it starts with 325 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 42, then the sub-category is Electric Pillar Maintenance. If it starts with 326 and its respective value on [ΚΩΔ\_ΛΟΓΑΡΙΑΣΜΟΥ] is 42, then the sub-category is Pruning.

- [ΑΡΙΘΜΟΣ]: is not picked, as its use is unknown and therefore, yields no insight. It is a numerical integer variable with a range of [NULL, -595, -582, ..., 900954, 900955, 900956], serving as a Categorical field.
- [ΧΑΡΑΚΤΗΡΙΣΜΟΣ\_ΕΡΓΟΥ]: is not picked, for it is completely empty. By extension, it lacks any value range or semantics.
- [ΣΚΟΠΟΣ\_ΕΡΓΟΥ]: is not picked, for it is completely empty. By extension, it lacks any value range or semantics.
- [ΕΤΟΣ\_ΕΡΓΟΥ]: is not picked, for it is completely empty. By extension, it lacks any value range or semantics.
- [ΑΡΙΘΜΟΣ\_ΕΡΓΟΥ]: is not picked, for it is completely empty. By extension, it lacks any value range or semantics.

- [AKYPΩΘEN]: is not picked, but the insight it grants is used for the SQL View construction. Rows cannot be deleted off the SQL Server Tables, hence if the need arises due to an erroneous entry, it's reflected here. This variable is a binary Categorical one, with a value of '0' meaning that the entry is not cancelled, and a value of '1' that it is. Taking this under account, for each row, when its value on this variable is 1, the row is dropped, otherwise it's kept.
- [KATHΓΟΙΑ]: is picked under the alias of [Katigoria]. It holds generalised project categories and is likely to correlate with the dependent variable as some categories might have a lower chance of being cancelled due to the nature of the category. It is a string variable with a range of [NULL, ',', 'X', ..., 'P', 'Q', 'EKM'], serving as a Categorical field.
- [ΦΟΠ\_ΛΟΙΠΑ]: is not picked, as not only is its use unknown, but it's also mostly missing with only ~40.8% data availability. It is a numerical integer variable with a range of [NULL, 0, 1, 2], serving as a Categorical field.
- [ZHΜΙΑ\_ΠΑΡΑΛΑΓΗ]: is not picked, for, ultimately, the objective is to predict whether or not a new project will come to completion, which has nothing to do with this variable's content, rendering it useless to our end. It's also mostly missing with only ~47.1% data availability. It is a numerical integer variable which is about the type of maintenance performed and has a range of [NULL, 0, 1, 2], serving as a Categorical field.
- [ΟΜΑΔΑ]: is not picked because of its sheer amount of factors, surpassing 23000 different ones. It is a string variable with a range of [NULL, ',', '1', ..., 'M.SD.2015.00002', 'M.SD.2015.00005', 'Σ'], which is basically yet another way of categorisation, serving as a Categorical field.
- [ΕΤΟΣ\_ΜΕΛΕΤΗΣ]: Is not picked. The information is already accessible from the variable "[ΗΜΕΡ\_ΜΕΛΕΤΗΣ]". In addition, the content itself contributes nothing to the model creation. It is a string variable created with the intention of containing the year of the project's 'study' part and it has a range of [NULL, '-', '--', ..., 'ΥΣ29', 'ΥΣ8B', 'Φ200' ], serving as a Categorical field.
- [ΑΡΙΘΜΟΣ\_ΜΕΛΕΤΗΣ]: is a mere protocol number, which is of no usefulness to our goal, and as such, is not picked. It is a numerical integer field with a range

of [NULL, -2008, -142, ..., 95542, 95629, 857178], serving as a Categorical variable.

- [ΕΙΔΟΣ\_ΕΞΥΠΗΡΕΤΗΣΗΣ0]: is a binary field whose meaning is unknown and has been characterised as inconsequential and “a field for DEDDHE’s employees eyes only” by the SQL Server holders/experts, and is hence, not picked. It is a numerical integer field with a range of [NULL, 1, 2], serving as a Categorical variable.
- [ΕΤΟΣ\_ΚΑΤΑΣΚΕΥΗΣ]: Is not picked. The information is already accessible from [ΗΜΕΡ\_ΕΚΤΕΛΕΣΗΣ]. In addition, the content itself contributes nothing to the model creation. It is a string field, intended to contain the project’s construction year and it has a range of [NULL, ‘-’, ‘200’, ..., ‘ΦΠ68’, ‘ΦΠ69’, ‘ΧΖΡΟ’], serving as a Categorical variable.
- [ΑΡΙΘΜΟΣ\_ΚΑΤΑΣΚΕΥΗΣ]: is a mere protocol number, which is of no interest to our goal, and as such, is not picked. It is a numerical integer field with a range of [NULL, -451, -318, ..., 1401805, 8792885, 34299300], serving as a Categorical variable.
- [ΔΕΗ\_ΠΕΛΑΤΗΣ]: is not picked, but the insight it grants is used for the SQL View construction. A value of ‘1’ means that the project is for the DEDDHE itself, whilst any other value means it is for one of its customers. As we’ve been told that a prediction is only necessary for projects for DEDDHE’s customers, the View is comprised only by rows whose value in this variable does not equal 1. It is a numerical integer field with a range of [NULL, 0, 1, 2], serving as a Categorical variable. It should be noted that there are 183,118 projects with a value of 1, hence being DEDDHE’s ones, whilst 249,528 projects are for clients.
- [ΑΡ\_ΠΡΩΤΟΚΟΛΟΥ\_ΠΕΛΑΤΗ]: is a mere protocol number, which is of little usefulness to our goal; in addition, it is mostly missing with ~20.1% data availability, and as such, is not picked. It is a string field with a range of [NULL, “-”, “ /26-08-2013”, ..., “ΧΡΗΣΤΙΑΣ ΣΩΤ”, “ΧΩΡΙΣ ΑΡΙΘ. ΠΡΩΤ./3/5/2011”, “ΧΩΡΙΣ/29469/26/9/2014”], serving as a Categorical variable.
- [ΑΡ\_ΠΡΩΤΟΚΟΛΟΥ\_ΔΕΗ] is a mere protocol number, which is of no interest to the classification; in addition, it is mostly missing with ~34.7% data availability, and as such, is not picked. It is a string field with a range of [NULL,

“-”, “--”, ..., “ΧΩΡΙΣ ΣΑΒ.”, “ΧΩΡΙΣ ΣΗΜ.”, “ΧΩΡΙΣ ΣΗΜΕΙΩΜΑ”], serving as a Categorical variable.

- [NEO\_ΠΑΡΑΛΛΑΓΗ]: is a binary field whose meaning is unknown and has been characterised as inconsequential and “a field for DEDDHE’s employees eyes only” by the SQL Server holders/experts. In addition, it is mostly missing with ~0.17% data availability, and is, hence, not picked. It is a numerical integer field with a range of [NULL, 1, 2], serving as a Categorical variable.
- [AP\_ΠΑΡΟΧΗΣ]: is a mere protocol number, which is of no interest to our goal; in addition, it is mostly missing with ~1.44% data availability, ergo, it is not picked. It is a string field with a range of [NULL, 00000100, 1000/05, ..., Σ8014590, Σ8014591, ΨΑΡΡΑ ΑΡ], serving as a Categorical variable.
- [ΠΕΡΙΓΡΑΦΗ]: is a description of the project that a human can easily read, holding no value for the classification process. It is a string field with a range of [NULL, ΑΠ, ΠΑΡΑΛΛΑΓΗ., ΕΠΕΚΤΑΣΗ ΔΧΤ ΓΙΑ ΦΟΠ., ..., ΜΕΤΑΤΟΠΙΣΗ ΣΤΥΛΩΝ Ν.ΕΦΕΣΟΥ, ΜΕΤΑΤΟΠΙΣΗ ΔΜΤ/ΔΧΤ ΑΠΟΣΤΟΛΙΔΗ ΘΕΟΔ., ΗΛ/ΣΗ ΕΡΓ.ΠΑΡΟΧΗΣ], serving as a Categorical variable.
- [ΗΜΕΡ\_ΑΙΤΗΣΗΣ]: is not picked, for there’s no use for dates in the classification. However, it is used for feature engineering, creating new variables with time differences. It is a date Categorical field, holding the date the project’s application was first received, and its range is [NULL, 1930-02-12 00:00:00, 1936-02-01 00:00:00, ..., 2016-05-26 00:00:00, 2016-05-27 00:00:00, 2016-10-21 00:00:00]. It is the 1<sup>st</sup> date in line, part of the 11 date variables that describe a project’s full cycle; it refers to the date the application for the project was filed. Many a thing depend on it:
  - [TimeSeriesDate] which represents a project’s order in the time series.
  - [Kathisterisi\_AitisisKataxis] reflecting the delay between the application for a project and the time it took for it to be registered to the system.
  - [DayOfYearSine] and [DayOfYearCosine] which are ways of simulating a close value for project whose day of year of commencement is close to one another

- [DayOfYearCartesX] and [DayOfYearCartesY] which are ways of simulating a close value for projects that commenced in a close temporal proximity.
- [APXH\_ΠΑΡΑΘΗΡΙΑ\_ΜΕΛΕΤΗΣ]: is not picked, for there's no use for dates in the classification process. In addition, it is mostly missing with ~0.05% data availability. It is a date field with a range of [NULL, 2001-03-21 00:00:00, 2002-09-10 00:00:00, ..., 2004-06-29 00:00:00, 2004-07-13 00:00:00, 2004-08-16 00:00:00], and it refers to the beginning date of the study's delay. It's unfortunate that the missing values rate is so dreadfully high, because it, along with [ΤΕΛΟΣ\_ΠΑΡΑΘΗΡΙΑ\_ΜΕΛΕΤΗΣ], could provide a new delay variable.
- [ΤΕΛΟΣ\_ΠΑΡΑΘΗΡΙΑ\_ΜΕΛΕΤΗΣ]: is not picked, for there's no use for dates for the classification process. In addition, it is mostly missing with ~0.05% data availability. It is a date field with a range of [NULL, 2001-04-01 00:00:00, 2002-11-24 00:00:00, ..., 2004-08-02 00:00:00, 2004-10-19 00:00:00, 2004-11-01 00:00:00], and it refers to the ending date of the study's delay. It's unfortunate that the missing values rate is so dreadfully high, because it, along with [APXH\_ΠΑΡΑΘΗΡΙΑ\_ΜΕΛΕΤΗΣ], could provide a new delay variable.
- [ΜΕΛ\_ΚΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]: is picked. It is a numerical integer field portraying the total amount of days that a project was delayed by, due to the client. A reason for the delay could be that the client is to provide certain documents which they have yet to assemble. This delay *is not* reflected in [ΗΜΕΡΕΣ\_ΜΕΛΕΤΗΣ]. Its ~5.8% data availability does not pose a problem as it can be interpreted as only ~5.8% of the projects having been delayed by the customer. Its range is [NULL, 0, 1, ..., 3674, 3776, 4138], serving as a Continuous variable. Although a delay of 4138 days could strike one as odd, it is in fact something that could happen. These are rare, isolated cases, where not only is the project not a priority, but there are other problems as well, for instance, the forestry or archaeology department got involved, impeding the process.
- [ΜΕΛ\_ΧΥΚ\_ΚΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]: is not picked because it's incorporated into [ΜΕΛ\_ΚΑΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]. It is a numerical integer field portraying the total amount of business days that a project was delayed by, due to the client. This delay *is not* reflected in [ΗΜΕΡΕΣ\_ΜΕΛΕΤΗΣ]. Its ~5.8% data availability does not pose a problem as it can be interpreted as only ~5.8% of

the projects having been delayed by the customer. Its range is [NULL, 0, 1, ..., 2537, 2639, 2875], serving as a Continuous variable.

- [MEA\_ENA\_KAΘ\_PIEΛATH]: is not picked as is, but it's used for feature engineering. It is a binary categorical field which takes a value of '0' when there's not been a delay, and a value of '1' otherwise. This delay *is not* reflected in [HMEPEΣ\_MEΛETHΣ].
- [MEA\_KAΘYΣTEPHΣH\_ΔEH]: is picked. It is a numerical integer field portraying the total amount of days that a project was delayed by, due to the organisation itself (DEDDHE). A reason for the delay could be that some aspect of the project is conflicting with a protocol or another project. Very high delays could correlate with a project's implementation never commencing as the client's frustration could outweigh their will to see it through. This delay *is* reflected in [HMEPEΣ\_MEΛETHΣ]. Its ~0.1% data availability does not pose a problem as it can be interpreted as only ~0.1% of the projects having been delayed by the organisation. Its range is [NULL, 0, 1, ..., 2536, 2842, 3122], serving as a Continuous variable.
- [MEA\_XYK\_KAΘYΣTEPHΣH\_ΔEH]: is not picked because it's incorporated into [MEA\_KAΘYΣTEPHΣH\_ΔEH]. It is a numerical integer field portraying the total amount of business days that a project was delayed by, due to the organisation (DEDDHE). This delay *is* reflected in [HMEPEΣ\_MEΛETHΣ]. Its ~0.1% data availability does not pose a problem as it can be interpreted as only ~0.1% of the projects having been delayed by the customer. Its range is [NULL, 0, 1, ..., 1741, 1954, 2141], serving as a Continuous variable.
- [MEA\_ENA\_KAΘ\_ΔEH]: is not picked as is, but it's used for feature engineering. It is a categorical binary variable which takes a value of '0' when there's not been a delay, and a value of '1' otherwise. This delay *is* reflected in [HMEPEΣ\_MEΛETHΣ].
- [MEA\_KAΘYΣTEPHΣH\_TPITΩN]: is picked. It is a numerical integer field portraying the total amount of days that a project was delayed by, due to any other factor. A reason for the delay could be that certain documents which are vital to the project are behind schedule due to external factors, such as the Forestry. This delay *is not* reflected in [HMEPEΣ\_MEΛETHΣ]. Its ~0.2% data availability does



not pose a problem as it can be interpreted as only ~0.2% of the projects having been delayed by other factors. Its range is [NULL, 0, 1, ..., 1806, 1862, 3023], serving as a Continuous variable.

- [MEΛ\_XYK\_KAΘYΣTEPHΣH\_TPITΩN]: is not picked because it's incorporated into [MEΛ\_KAΘYΣTEPHΣH\_TPITΩN]. It is a numerical integer field portraying the total amount of business days that a project was delayed by, due to any other factor. Very high delays could correlate with a project never commencing as the client's frustration could outweigh their will to see it through. This delay is not reflected in [HMEPEΣ\_MEΛETHΣ]. Its ~0.2% data availability does not pose a problem as it can be interpreted as only ~0.2% of the projects having been delayed by other factors. Its range is [NULL, 0, 2, ..., 1236, 1276, 2072], serving as a Continuous variable.
- [MEΛ\_ENΔ\_KAΘ\_TPITΩN]: is not picked as is, but it's used for feature engineering. It is a categorical binary field which takes a value of '0' when there's not been a delay, and a value of '1' otherwise. This delay *is not* reflected in [HMEPEΣ\_MEΛETHΣ].
- [HMEP\_MEΛETHΣ]: is the date that the 'study' part of the project concluded on. It is not picked, for there's no use for dates in the classification. However, it is used for feature engineering, creating new variables with time differences. It's a date field with a range of [NULL, 1974-08-07 00:00:00, 1992-07-22 00:00:00, ..., 2017-08-12 00:00:00, 2018-03-18 00:00:00, 2020-11-17 00:00:00]. Given that the data were collected on July 2016, a study having been completed on 2020 is an impossibility pointing to erroneous data. This noise is partially handled by a clauses imposed.
- [HMEP\_ANAΓΓΕΛΙΑΣ]: is the date that the letter informing the customer of the project's price, which remains valid for 2 months, is sent. It is not picked, for there's no use for dates in the classification. However, it is used for feature engineering, creating new variables with time differences. This variable is also used to ensure that only rows with valid and correct "Label" are retrieved given that a customer has a maximum of 2 months on their disposal to pay the money before the project is cancelled. This variable, whilst used for the SQL View creation as a means for comparison, is ultimately not picked as one of View's columns. It is a date field with a range of [NULL, 1966-09-19 00:00:00, 1992-07-

22 00:00:00, ..., 2017-08-12 00:00:00, 2018-03-18 00:00:00, 2020-11-17 00:00:00].

- [HMEPEΣ\_MEΛETHΣ]: is picked under the alias of [Meres\_Meletis]. It reflects the number of days the ‘study’ part of the project lasted, and is DEDDHE-delay inclusive. It’s a numerical integer field with a range of [NULL, -37944, -37943, ..., 37915, 37917, 37924], serving as a Continuous variable. It could correlate with the dependent variable for reasons such as run-of-the-mill high values could potentially mean a big project with high probability taking place whereas excessively high values could indicate potential problems, and thus having a low probability.
- [EPΓ\_HMEPEΣ\_MEΛETHΣ]: is not picked, for the information is already included in [HMEPEΣ\_MEΛETHΣ]. It is numerical integer field containing the number of business days the project’s ‘study’ part lasted, and its range is [NULL, -3003, -2640, ..., 15439, 17386, 21326], serving as a Continuous variable.
- [ΣYNEPΓEIO\_MEΛETHΣ] is picked as it could potentially correlate with the dependent variable, in that, for example, the projects of a certain Study Workshop could have, for reasons spanning beyond what meets the eye, a significantly lower percentage of completion. It is a string field consisting of 3 factors whose range is [NULL, ΣYNEPΓEIA ΔEΔΔHE, ΣYNEPΓEIA EPΓOΛABOY, ΣYNEPΓEIA TPITΩN], serving as a Categorical variable.
- [MEΛETHTHΣ]: is not picked, as it has 1932 factors, rendering it unhelpful. It is a string field which contains the researcher’s name, and its range is [NULL, -----, ΦPAΓKANΔPEΑΣ, ..., ΨAXOYΛIAC A & K OE, ΨAXOYΛIAC A & K OE, ΨHΛOΠANAGΩTHΣ], serving as a Categorical variable.
- [AΩ\_KATAΣKEYHΣ]: is a numerical decimal field whose meaning is unknown and has been characterised as inconsequential and “a field for DEDDHE’s employees eyes only” by the SQL Server holders/experts. In addition, it is mostly missing with ~0.8% data availability, and therefore, is not picked. It’s a Categorical variable with a range of [NULL, 0, 0.01, ..., 4580.3, 4590.55, 6031].
- [KOΣTOΣ\_MEΛETHTH]: is a numerical decimal field portraying the researcher’s cost to the company. It is, however, not picked as a consequence of its mere ~0.2% data availability. It’s a Continuous variable with a range of

[NULL, -91, -6, ..., 119156, 268096, 612131500]. This could have potentially been a critical value where patterns could have emerged; for instance, the bigger the cost, the most likely for a project to succeed, until a critical point where the pattern gets reversed. Such a pattern could show a tendency for more expensive and usually more esteemed professionals to have a higher success rate, until a point where the cost is too much to afford.

- [ΚΟΣΤΟΣ\_ΕΡΓΑΤΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]: is a numerical decimal field reflecting the amount of money that construction workers cost the company. It is one of the 4 most critical independent variables, and is picked. It's a Continuous variable with a range of [NULL, -2745.512, -2084, ..., 31132970.2847, 33895228, 60840608.818].
- [ΚΟΣΤΟΣ\_ΥΛΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]: is a numerical decimal field stating the cost of the materials to be used for the project. It is one of the 4 most critical independent variables, and is picked. It's a Continuous variable with a range of [NULL, -301792.437222764, -162520.25, ..., 13290497, 55571115.5156, 133449073.0268].
- [ΚΟΣΤΟΣ\_ΚΑΤΑΣΚΕΥΗΣ]: is a numerical decimal field reflecting the amount of money the construction costs to the company; it is inclusive of the values of [ΚΟΣΤΟΣ\_ΥΛΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ] and [ΚΟΣΤΟΣ\_ΕΡΓΑΤΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]. It is one of the 4 most critical independent variables, and is picked. It's a Continuous variable with a range of [NULL, -299734.136359997, -160924.2548, ..., 25341675, 86704085.8003, 194289681.8448].
- [ΚΟΣΤΟΣ\_ΕΡΓΟΛΑΒΙΚΩΝ\_ΕΠΙΔΟΣΗΣ]: is a Numerical decimal field reflecting the cost of service contractors. It is one of the 4 most critical independent variables, and is picked. It's a Continuous variable with a range of [NULL, -3572, -2269.0181, ..., 33895228, 50282662.3368, 694214876].
- [ΑΩ\_ΜΕΛΕΤΗΣ]: is a Categorical variable whose meaning is unknown and has been characterised as inconsequential and “a field for DEDDHE’s employees eyes only” by the SQL Server holders/experts; in addition, it is mostly missing with ~0.05% data availability, and as such, is not picked. It's a numerical integer field with a range of [NULL, 0, 1, ..., 4185, 4968, 32345].

- [HMEP\_ΥΠΟΓΡΑΦΗΣ]: is the date the customer pays so that the project can commence. This is the dependent variable; however, it is not picked in this configuration, instead, a new feature is engineered reflecting whether or not the customer paid, in a binary manner. It's a date field with a range of [NULL, 1997-09-17 00:00:00, 1998-02-16 00:00:00, ..., 2020-05-03 00:00:00, 2022-10-02 00:00:00, 2029-02-13 00:00:00].
- [HMEP\_ΠΑΡΑΛΑΒΗΣ]: is the date the contractor took on the project. It is not picked, for there's no use for dates in the classification, as well as the fact that it is a completely empty column. By extension, it lacks any value range or semantics.
- [ΕΙΔΟΣ\_ΕΞΥΠΗΡΕΤΗΣΗΣ]: is a categorical binary field whose meaning is unknown and has been characterised as inconsequential and “a field for DEDDHE's employees eyes only” by the SQL Server holders/experts, and is hence, not picked. It is a numerical integer variable with a range of [NULL, 1, 2].
- [ΤΙΤΛΟΣ\_ΕΡΓΟΥ]: is a descriptive Title for the project that a human can easily read, holding no value for the classification process. It's a string field with a range of [NULL, ‘’, ‘-’, ‘ΩΣΗΦ ΚΑΛΔΕΡΩΝ ΚΑΙ Σ’, ‘ΩΣΤΟΠΟΥΛΟΥ Β’, ‘ΩΣΥΝΤΗΡΗΣΗ ΔΜ.Τ.ΧΤ ΟΣΜΟΣΕΣΥΝΤΗΡΗΣΗ ΔΜ.Τ.ΧΤ ΟΣΜΟΣ’], serving as a Categorical variable.
- [ΣΥΜΒ\_ΗΜΕΡ\_ΕΝΑΡΞΗΣ]: is the date the construction should begin as set by the contract, e.g. the contract requires that the project commence within 10 days of said date. Subtracting this date from the actual date yields the delay. It is a date variable with a range of [NULL, 1919-06-01 00:00:00, 1930-02-01 00:00:00, ..., 2055-09-03 00:00:00, 2066-04-07 00:00:00, 2066-09-01 00:00:00]. It is not picked since as far as the classification is concerned, the construction part is irrelevant.
- [ΣΥΜΒ\_ΗΜΕΡ\_ΕΚΤΕΛΕΣΗΣ]: is the date reflecting the upper limit set by the contract as the deadline for the project, e.g. the contract requires that the project be completed within 10 days after said date. Subtracting this date from the actual date yields the delay. It is a date field with a range of [NULL, 1919-06-01 00:00:00, 1930-02-01 00:00:00, ..., 2055-09-03 00:00:00, 2066-04-07 00:00:00,

2066-09-01 00:00:00]. It is not picked since as far as the classification is concerned, the construction part is irrelevant.

- [APXH\_ΠΑΡΑΤΗΡ\_ΕΚΤΕΛΕΣΗΣ] is not picked, for there's no use for dates in the classification; in addition, it is mostly missing with ~0.0% data availability, and as such, is not picked. It's a date field with a range of [NULL, 2003-06-05 00:00:00, 2003-08-14 00:00:00, ..., 2004-01-03 00:00:00, 2004-02-05 00:00:00, 2004-11-02 00:00:00] and it refers to the beginning date of the construction's delay.
- [ΤΕΛΟΣ\_ΠΑΡΑΤΗΡ\_ΕΚΤΕΛΕΣΗΣ]: is not picked, for there's no use for dates in the classification; in addition, it is mostly missing with ~0.0% data availability, and as such, is not picked. It's a date field with a range of [NULL, 2003-09-07 00:00:00, 2003-10-04 00:00:00, ..., 2004-02-25 00:00:00, 2004-06-24 00:00:00, 2004-11-24 00:00:00] and it refers to the ending date of the construction's delay.
- [KAT\_KAΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]: is not picked for any information regarding a project's construction phase is deemed irrelevant as for it to have already gone to that phase it must have already been approved. It is a numeric integer field portraying the total amount of days that a project was delayed by, due to the client. This delay *is not* reflected in [HMEPEΣ\_ΕΚΤΕΛΕΣΗΣ]. Its ~4.2% data availability does not pose a problem as it can be interpreted as only ~4.2% of the projects having been delayed by the customer. It has a range of [NULL, 0, 1, ..., 2576, 2593, 2595], serving as a Continuous variable.
- [KAT\_XYK\_KAΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]: is not picked because it's incorporated into [KAT\_KAΘΥΣΤΕΡΗΣΗ\_ΠΕΛΑΤΗ]. It is a numeric integer field portraying the total amount of business days that a project was delayed by, due to the client. This delay *is not* reflected in [HMEPEΣ\_ΕΚΤΕΛΕΣΗΣ]. Its ~4.2% data availability does not pose a problem as it can be interpreted as only ~4.2% of the projects having been delayed by the customer. It has a range of [NULL, 0, 1, ..., 1768, 1773, 1774], serving as a Continuous variable.
- [KAT\_ENΔ\_KAΘ\_ΠΕΛΑΤΗ]: is not picked as is, but it can be used for feature engineering. It is a categorical binary field which takes a value of '0' when there's not been a delay, and a value of '1' otherwise. This delay *is not* reflected in [HMEPEΣ\_ΕΚΤΕΛΕΣΗΣ].

- [KAT\_KAΘΥΣΤΕΡΗΣΗ\_ΔΕΗ]: is not picked. It is a numerical integer field portraying the total number of days that a project was delayed by, due to the organisation (DEDDHE). This delay *is* reflected in [HMEPEΣ\_EKTEΛΕΣΗΣ]. Its ~0.8% data availability does not pose a problem as it can be interpreted as only ~0.8% of the projects having been delayed by the organisation. Its range is [NULL, 0, 1, ..., 1581, 1595, 1920], serving as a Continuous variable.
- [KAT\_XYK\_KAΘΥΣΤΕΡΗΣΗ\_ΔΕΗ]: is not picked because it's incorporated into [KAT\_KAΘΥΣΤΕΡΗΣΗ\_ΔΕΗ]. It is a numerical integer field portraying the total number of business days that a project was delayed by, due to the organisation (DEDDHE). This delay *is* reflected in [HMEPEΣ\_EKTEΛΕΣΗΣ]. Its ~0.8% data availability does not pose a problem as it can be interpreted as only ~0.8% of the projects having been delayed by the customer. Its range is [NULL, 0, 1, ..., 1078, 1095, 1317], serving as a Continuous variable.
- [KAT\_ENΔ\_KAΘ\_ΔΕΗ]: is not picked as is, but it can be used for feature engineering. It is a categorical binary field which takes a value of '0' when there's not been a delay, and a value of '1' otherwise. This delay *is* reflected in [HMEPEΣ\_EKTEΛΕΣΗΣ].
- [KAT\_KAΘΥΣΤΕΡΗΣΗ\_ΤΡΙΤΩΝ]: is not picked. It is a numerical integer field portraying the total number of days that a project was delayed by, due to any other factor. This delay *is not* reflected in [HMEPEΣ\_EKTEΛΕΣΗΣ]. Its ~0.0% data availability does not pose a problem as it can be interpreted as only ~0.0% of the projects having been delayed by other factors. Its range is [NULL, 0, 1, ..., 206, 237, 274], serving as a Continuous variable.
- [KAT\_XYK\_KAΘΥΣΤΕΡΗΣΗ\_ΤΡΙΤΩΝ]: is not picked because it's incorporated into [KAT\_KAΘΥΣΤΕΡΗΣΗ\_ΤΡΙΤΩΝ]. It is a numerical integer field portraying the total number of business days that a project was delayed by, due to any other factor. This delay *is not* reflected in [HMEPEΣ\_EKTEΛΕΣΗΣ]. Its ~0.0% data availability does not pose a problem as it can be interpreted as only ~0.0% of the projects having been delayed by other factors. Its range is [NULL, 0, 1, ..., 142, 162, 189], serving as a Continuous variable.
- [KAT\_ENΔ\_KAΘ\_ΤΡΙΤΩΝ]: is not picked as is, but it's used for feature engineering. It is a categorical binary value which takes a value of '0' when there's

not been a delay, and a value of '1' otherwise. This delay *is not* reflected in [HMEPEΣ\_EKTEΛEΣHΣ].

- [HMEP\_ENAPEHΣ]: is the actual date that the construction part of the project commenced. It is not picked, for there's no use for dates in the classification. It's a date field with a value range of [NULL, 1931-04-08 00:00:00, 1936-09-01 00:00:00, ..., 2029-02-10 00:00:00, 2029-02-14 00:00:00, 2044-05-05 00:00:00]
- [HMEP\_EKTEΛEΣHΣ]: is the actual date that the construction part of the project finished. It is not picked, for there's no use for dates in the classification. It's a date field with a value range of [NULL, 2000-02-02 00:00:00, 2001-01-10 00:00:00, ..., 2016-12-31 00:00:00, 2017-03-17 00:00:00, 2023-01-23 00:00:00].
- [ΠΟΣ\_EKTEΛEΣHΣ]: is a numerical integer field holding the percentage the project's construction phase is at. It is not picked as it has no predictive potential over the dependent variable. Its range is [Null, 0, 1, ..., 1100, 2006, 2007], serving as a Categorical variable.
- [ΠΙΣΤΟΠΟΙΗΣΗ]: is a categorical binary field whose meaning is unknown and has been characterised as inconsequential and "a field for DEDDHE's employees eyes only" by the SQL Server holders/experts, and is hence, not picked.
- [HMEP\_ΠΙΣΤΟΠΟΙΗΣΗΣ]: is not picked, for there's no use for dates in the classification; in addition, it is mostly missing with ~36.9% data availability, ergo, it is not picked. It's a date field whose meaning is unknown and has been characterised as inconsequential and "a field for DEDDHE's employees eyes only" by the SQL Server holders/experts, with a value range of [NULL, 1930-09-29 00:00:00, 1931-02-01 00:00:00, ..., 2019-04-05 00:00:00, 2020-02-23 00:00:00, 2020-06-28 00:00:00].
- [HMEPEΣ\_EKTEΛEΣHΣ]: is not picked. It reflects the number of days the construction part of the project lasted, and is DEDDHE-delay inclusive. It's a numerical integer field with a range of [NULL, -1910, -1089, ..., 3951, 4410, 4542], serving as a Continuous variable.
- [ΕΡΓ\_HMEPEΣ\_EKTEΛEΣHΣ]: is not picked, for the information is already included in [HMEPEΣ\_EKTEΛEΣHΣ]. It's a numerical integer field about the number of business days the construction part lasted, and its range is [NULL, -1910, -729, ..., 2705, 3058, 3108].

- [ΣΥΝΕΡΓΕΙΟ\_ΚΑΤΑΣΚΕΥΗΣ]: is not picked. It is a Categorical string variable with a mere 3 factors and a range of [NULL, ΣΥΝΕΡΓΕΙΑ ΔΕΔΔΗΕ, ΣΥΝΕΡΓΕΙΑ ΕΡΓΟΛΑΒΟΥ, ΣΥΝΕΡΓΕΙΑ ΤΡΙΤΩΝ]. It portrays the name of the construction workshop used for the project, which is unnecessary information for predicting whether or not a project will be approved as it only gets filled in after the fact.
- [ΚΑΤΑΣΚΕΥΑΣΤΗΣ]: is not picked. It is a Categorical string variable with 641 factors and a range of [NULL, ‘ ΤΣΕΓΚΟΣ Γ’, ‘ ΤΣΕΓΚΟΣ Γ ΤΣΙΠΑΣ Δ’]. It reflects the project’s constructor name, which is unnecessary information for predicting whether or not a project be approved as it only gets filled in after the fact.
- [ΑΠΟΛ\_ΚΟΣΤΟΣ\_ΕΡΓΑΤΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]: is not picked for it only has ~7.2% of its data filled in and available. Its name suggest that it contains the absolute (positive only) number reflecting the amount of money that construction workers cost the company and it’s a decimal field with a range of [NULL, -8569, -2779, ..., 652025.61, 785405, 23217393], serving as a Continuous variable.
- [ΑΠΟΛ\_ΚΟΣΤΟΣ\_ΥΛΙΚΩΝ\_ΚΑΤΑΣΚΕΥΗΣ]: is not picked for it only has ~6.5% of its data filled in and available. Its name suggests that it contains the absolute (positive only) number stating the cost of the materials to be used for the project and it’s a decimal field with a range of [NULL, -28554, -17732.8, ..., 747046, 1835053, 30602010], serving as a Continuous variable.
- [ΑΠΟΛ\_ΚΟΣΤΟΣ\_ΚΑΤΑΣΚΕΥΗΣ]: is not picked for it only has ~6.5% of its data filled in and available. Its name suggests that it contains the absolute (positive only) number reflecting the amount of money the construction costs to the company and it’s a decimal field with a range of [NULL, -28554, -17732.8, ..., 747046, 1835053, 30602010], serving as a Continuous variable.
- [ΕΚΤΥΠΩΣΗ\_ΠΡΩΤΟΚΟΛΟΥ]: is a categorical binary field whose meaning is unknown and has been characterised as inconsequential and “a field for DEDDHE’s employees eyes only” by the SQL Server holders/experts, and as such it has not been picked. Its name suggests that it reflects whether or not a protocol has been printed, or is to be printed, but it’s probably a proxy for something else altogether.



- [ONOMATEΠΩNYMO]: is a classified descriptive field of the client's name to which we have no access, and is, therefore, not picked. By extension, it lacks any value range and variable type.
- [ΔΙΕΥΘΥΝΣΗ]: is a classified descriptive field of the client's address to which we have no access, and is, therefore, not picked. By extension, it lacks any value range and variable type.
- [ΠΟΛΗ]: is a Categorical field portraying the city the project is to take place on. As there are many missing values, a new feature is engineered by the combination of this one and [ΠΟΛΗ\_Y\_Σ], and hence it is not picked by itself. It's a string variable with a range of [NULL, ΑΔΕΝΔΡΟ, ΑΜΠΛΙΑΝΗ, ..., ΩΡΟΛΟΓΙΟ, ΩΡΟΛΟΓΙΟΝ, ΩΡΩΠΟΣ].
- [ΠΟΛΗ\_Y\_Σ]: is a Categorical field portraying the city that the substation providing electricity to the project's construction is. As there are many missing values, a new feature is engineered by the combination of this one and [ΠΟΛΗ], so it is not picked by itself. It's a string variable with a range of [NULL, 40 ΕΚΚΛΗΣΙΕΣ, 4η ΜΟΝΑΔΑ, ..., ΩΡΙΑ, ΩΡΟΛΟΓΙΟ, ΩΡΩΠΟΣ].
- [Y\_Σ]: is not picked, as not only is it a Categorical field whose use is unknown, but there's also a data availability of about half. It's a string variable with a range of [NULL, -, --, ..., Ω-981X, Ω-982X, Ω-984].
- [ΤΗΛΕΦΩΝΟ]: is a classified Categorical field of the client's telephone number to which we have no access, and is, therefore, not picked. By extension, it lacks any value range and variable type.
- [ΠΑΡΑΤΗΡΗΣΕΙΣ]: is a description of the project that a human can easily read, holding no value for the classification process, and as such, it is not picked. It's a string field with a range of [NULL, ΑΔΥΝΑΜΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΜΕ ΙΔΙΟΚΤΗΤΗ.- , ΑΝΑΜΟΝΗ ΑΠΟ ΔΗΜΟ, ..., ΕΝΑΡΞΗ Ε/Ε 30/06/04, ΔΕΝ ΒΡΕΘΗΚΕ.-, ΥΠΑΡΧΟΥΝ ΟΡΙΑ ΑΣΦΑΛΕΙΑΣ.-], serving as a Categorical Variable.
- [ΠΑΡΑΤΗΡΗΣΕΙΣ2]: is a description of the project that a human can easily read, holding no value for the classification process, and as such, it is not picked. It is a string field with a range of [195, ΑΚΥΡΟΝ ΛΟΓΩ ΜΗ ΕΦΑΡΜΟΓΗΣ ΤΗΣ ΥΠΑΡΧΟΥΣΑΣ ΜΕΛΕΤΗΣ.ΘΑ ΣΥΝΕΧΙΣΤΗ ΜΕ ΆΛΛΟ ΕΡΓΟ ,

ΔΙΕΥΚΡΗΝΙΣΗ ΣΗΜΕΙΟΥ Φ/Σ ΑΠΟ ΔΗΜΟ, ΕΓΙΝΕ ΑΝΤΙΚΑΤΑΣΤΑΣΗ ΤΟΥ ΣΤΥΛΟΥ, ΤΟΠΟΘΕΤΗΣΗ, ..., ΤΟΠΟΘΕΤΗΣΗ 2 Φ/Σ ΣΕ ΥΠΑΡΧΟΝΤΕΣ ΣΤΥΛΟΥΣ, ΣΑΒ 3966/178 Η ΑΝΤΙΚΑΤΑΣΤΑΣΗ ΤΟΥ ΣΤΥΛΟΥ ΕΓΙΝΕ ΣΤΙΣ 18-11-2004, ΣΑΒ 4135/73 ΑΝΑΓΓΕΛΙΑ ΒΛΑΒΗΣ 11/12/04 ΑΠΟΚΑΤΑΣΤΑΣΗ ΒΛΑΒΗΣ 13/12/04 (ΑΝΤΙΚΑΤΑΣΤΑΣΗ Μ/Σ)], serving as a Categorical Variable.

- [ΕΚΤΑΣΗ\_ΕΡΓΟΥ]: is a numerical integer field illustrating the project's scale (small or big). It is picked as it's expected that it's correlated with the dependent variable as big projects are probably less likely to be cancelled. Its range is [NULL, 1, 2], serving as a Categorical Variable.
- [ΑΝΑΓΚΗ\_ΥΣ]: is a categorical binary field reflecting whether or not there's a need for a substation. Since it's likely to be correlated with the dependent variable, for instance, projects that do need a substation are big and less likely to be cancelled, it's picked.
- [ΔΙΚΤΥΟ\_XT\_MT]: is not picked, for not only is it mostly missing with only ~31.4% data availability (after converting blank entries to NULL), but it's also been described as “a field for DEDDHE's employees eyes only” by the SQL Server holders/experts. It is a string field with a range of [NULL, ‘’, ‘MT’, ‘XT’], serving as a Categorical Variable.
- [SAP\_ΑΡΙΘΜΟΣ\_ΕΡΓΟΥ]: is not picked as it's composed of over 89,000 factors, rendering the variable useless to our cause. Chances are this variable is but another protocol number. It's a string field with a range of [NULL, ‘’, ‘61416150099’, ..., ‘Α|Α 331|2012’, ‘ΑΝΕΥ’, ‘ΠΡΟΥΠΟΛΟΓΙΣΜ’], serving as a Categorical Variable.
- [SAP\_ΧΑΡΑΚΤΗΡΙΣΜΟΣ\_ΕΡΓΟΥ]: is not picked, as it holds only part of the information intended for it, which has been fragmented into old and new variables. It is, however, amongst the set of variables used to feature engineer the information originally designated for this one. It's a string field whose values represent the project's main category, and its range is [NULL, D, M], serving as a Categorical Variable. Rows with a “D” in this variable refer to an Investment, whilst “M” refers to a Utilisation.

- [SAP\_TYΠΟΣ\_ΠΕΛΑΤΗ]: is picked, as its 5 categories could potentially correlate with the dependent variable. The meaning behind the categories remains a mystery and its data availability of just ~22.1% does raise some flags, however, its potentiality for being detrimental to the classification will be put to the test. The reason behind the majority of values being missing is that it is a newly introduced variable which has only been there on the SQL Server over the last few years. It's a string field with a range of [NULL, ',', '1', '6', 'E', 'I'], serving as a Categorical Variable.
- [SAP\_ΕΙΔΟΣ\_ΑΙΘΜΑΤΟΣ]: is picked, as its 41 categories could potentially correlate with the dependent variable. This is a lower (more specific) level of categorisation compared to [ΚΩΔΙΚΟΣ\_ΑΝΑΛΥΣΗ], with "WT03", for instance, referring to a Wattage Increase, but its data availability of a mere ~17.1% does raise some flags, however, its capacity for being detrimental to the classification will be put to the test. The reason behind the majority of values being missing is that it is a newly introduced variable which has only been there on the SQL Server over the last few years. It's a string field with a range of [NULL, '6121215016', '719', ..., 'wtnc04', 'AYE. IΞXYO'], serving as a Categorical Variable.
- [SAP\_ΣΚΟΠΟΣ\_ΕΡΓΟΥ]: is not picked, as it holds only part of the information intended for it, which has been fragmented into old and new variables. It is, however, amongst the set of variables used to feature engineer the information originally designated for this one. It's a string field with a range of [NULL, '', '0', ..., 'EA', 'EI', 'Σ'], serving as a Categorical Variable. This field reflects the project's sub-category; 'EA' refers to an Electrification for Consumers, 'EB' refers to an Electrification for Producers, 'EC' refers to a Variant, 'ED' refers to an Aesthetical Upgrade, 'EE' refers to an Amplification, 'EF' refers to a Layout Reconfiguration on a Substation, 'SA' refers to a Localised Maintenance, 'SB' refers to Preventative Maintenance, 'SC' refers to Network Damage Maintenance, 'SD' refers to a Repair of damage caused by a Third Party, 'SE' refers to Measurements, 'SF' refers to a Network Removal, 'SG' refers to an Electric Pillar Maintenance, 'SH' refers to a Pruning, 'SI' refers to Maintenance due to Theft, 'SJ' refers to an Electrification Discontinuation/Reconnection due to Debt, 'SK'

refers to an Electrification Discontinuation/Reconnection Because the Customer Asked for it, and 'SL' refers to Technical Interference to Measurement.

- [UserName]: is not picked. This field holds the User Name of the person creating the entry for the project, a classified variable. By extension, it lacks any value range and variable type.
- [UpDate]: is not picked, for there's no use for dates in the classification. It's a date field depicting the last time the row was accessed and changed and its value range is [NULL, 2001-01-01 04:16:00, 2003-01-02 12:08:00, ..., 2016-05-28 10:26:00, 2016-05-28 10:36:00, 2016-05-28 10:39:00].
- [Test]: is not picked. The variable seems to be a hexadecimal one whose meaning is unknown. It's value range is [0x000000000001EED85, 0x000000000001EED87, 0x000000000001EED88, ..., 0x00000000000293D55, 0x00000000000293D56, 0x00000000000293D57].

# 6 Applying Machine Learning

## 6.1 Unsupervised Learning

The k-means algorithm has been used for the clustering purposes. There are many clustering algorithms, arguably better than this, and the reason behind picking it is because the code as a whole has been written in a multi-threaded, cluster-ready manner, and k-means is the sole algorithm in Microsoft R supporting such a feature.

k-means was discovered by many a researcher across different disciplines, is an iterative method, part of the hill-climbing algorithms, which partitions a dataset into clusters. The initial ones are picked either at random from the dataset or by perturbing the global mean of the data  $k$  times and then it's just a matter of iteration between Data Assignment and Relocation of “means” until convergence is achieved. The k-means, too, has its drawbacks like an unwanted sensitivity to initialisation, outliers and the accuracy of the data's description. <sup>[22]</sup>

There are many metrics and criteria to evaluate and measure clustering characteristics by, but the one we use and try to minimise in this implementation is the Sum-of-Squared-Error. It is a basic criterion summing over the squared distances between the clustering objects and their cluster representatives (i.e. the respective cluster centroids. The evaluation defines a measure for the homogeneity of the clustering results with respect to the object description data. The sum-of-squared-error “E” originally refers to Euclidean distance between each object “o” and its own cluster's centre, but is considered now applicable to further distance measures. The definition was given in the following equation with the cluster centroid of cluster  $C_i$  abbreviated as  $cen_i$ . <sup>[23]</sup>

$$[EQ0] E(C) = \sum_{i=1}^k \sum_{o \in C_i} d(o, cen_i)^2$$

Plunging into, and browsing through hundreds of thousands of numbers is no way to get a deep understanding of the data, which is why before continuing with the actual algorithms, the data should be visualised.

Google's GeoLocation API was used to get the longitude and latitude of each project's city. The way the API works, addresses and zip codes being retracted for privacy reasons

(as, of course, they should), as well as the way the information was gathered, introduced inaccuracies and altogether erroneous entries to the geolocation data.

Methods written below were implemented with R Code, a sample of which is:

```
#####
## Iteration 0 ##
#####
#Peaking at the database as it is in the SQL View
rxLinePlot(formula = GeoLocY ~ GeoLocX,
            data = vErga_DS,
            type = "p"
)
#Many entries are outside Greece's rectangle

[ . . . ]

rxFactors(inData = paste(strXDF, "tmp2.xdf", sep = ""),
          outFile = paste(strXDF, "Clustering_DS.xdf", sep = ""),
          factorInfo = c("TimeSeriesDate"),
          sortLevels = TRUE,
          overwrite = TRUE
)
Clustering_DS <- RxXdfData(paste(strXDF, "Clustering_DS.xdf", sep =
""))

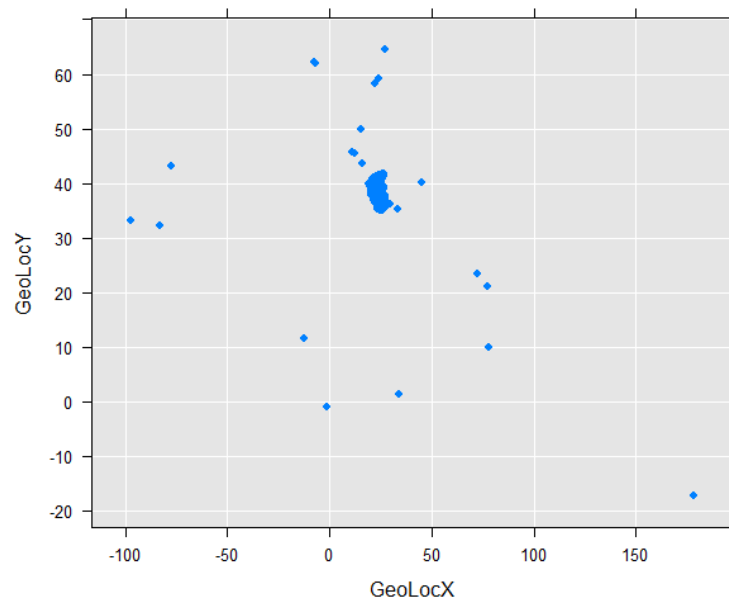
[ . . . ]

for (i in 2:30) {
  WithinGroupsSquaredError[i] <- sum(rxKmeans(formula = formula(~
GeoLocX + GeoLocY),
                                             data =
unsupervisedLocationData1,
                                             numClusters = i,
                                             algorithm = "lloyd"
                                             )$withinss
)
}

[ . . . ]

rxLinePlot(GeoLocY ~ GeoLocX,
            groups = .rxCluster,
            data = paste(strXDF, "Clustering_DS.xdf", sep = ""),
            type = "p"
)
```

The figure below is a visualisation of said data immediately after the clauses are applied.



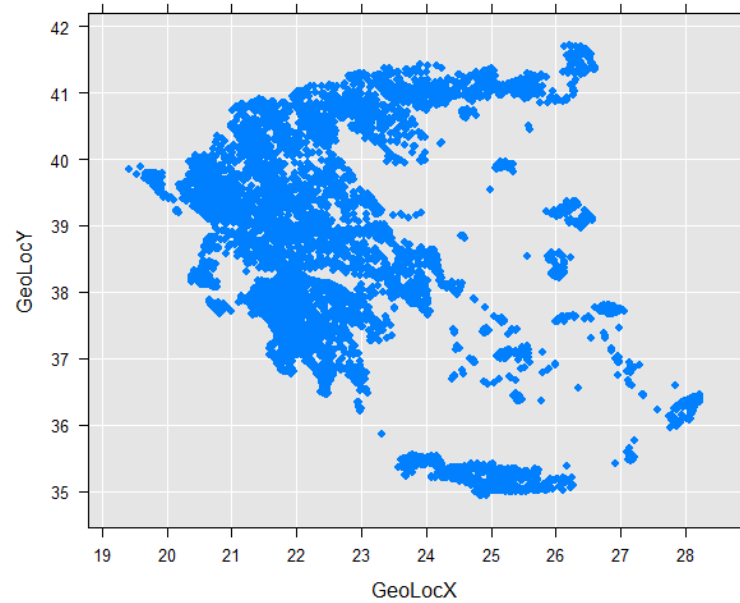
**Figure 4:** Visualisation of original data (Iteration 0)

Looking at the plot, it quickly becomes apparent that the vast majority of locations are concentrated in the upper-mid section of the graph, whilst few others are scattered here and there. What this reveals, is that in all probability, that tight all-encompassing rectangle is Greece. To verify the hypothesis and to impose applied rules to cross out non-Greek longitudes/latitudes, the picture below is juxtaposed.



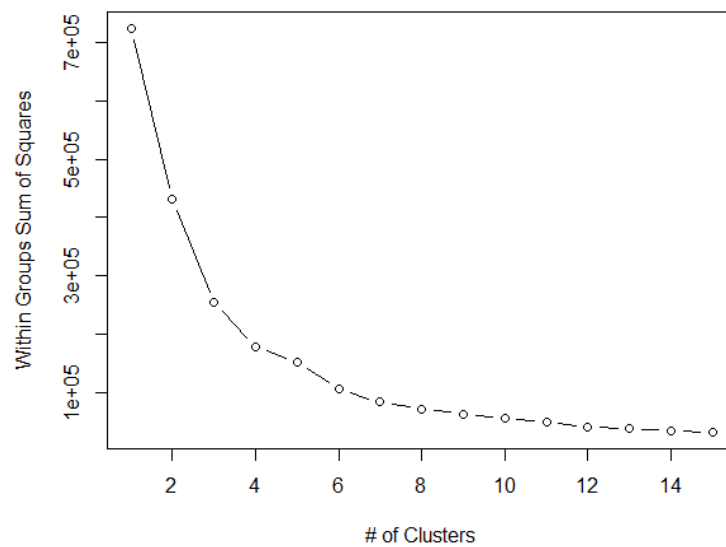
**Figure 5:** Greece's Map

What we witness is that Greece's Latitude (Y) spans from 34 to 42, and its Longitude (X) from 18 to 29. Accounting for that, we're coming to the visualisation of the 1<sup>st</sup> iteration, by eliminating invalid entries, as shown in the picture below.



**Figure 6:** Data Visualisation of Iteration 1

To determine the optimal number of clusters, the Sum of Squared Error was computed and the number where the SSE starts becoming linear, as witnessed at the respective Scree Plot, is the optimal number.

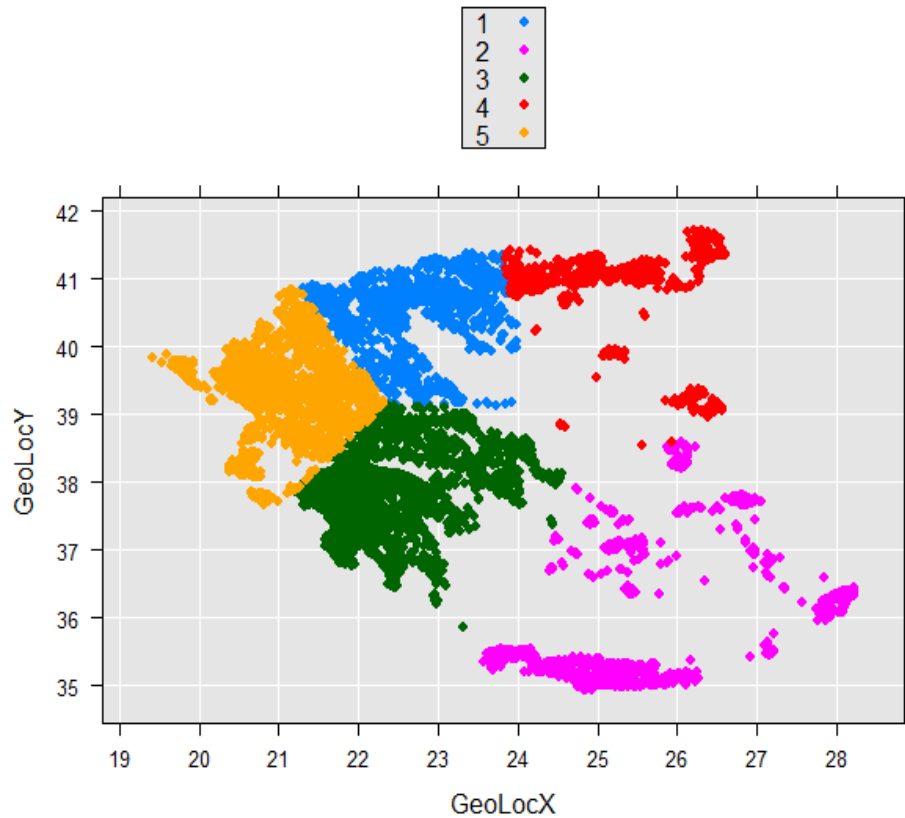


**Figure 7:** Sum of Squared Error Scree Plot



According to the Scree Plot, an appropriate number of clusters for the dataset is 5 as at that number the rate at which the Within Groups Sum of Squares drops is starting to turn into a straight line. Whilst maintaining the ability to override the value by viewing the Scree Plot, the computation of the optimal number by the percentage drop in the SSE is done autonomously by the code as will be stated later in the ‘Implementation’ chapter.

Applying the RevoScaleR’s K-Means, implemented with the Lloyd algorithm, to the dataset, given the number of clusters being five, yielded the result shown in the figure below.



**Figure 8:** Clusters Visualisation after K-Means

The five clusters are comprised of 37724, 15789, 39210, 14096, and 23138 valid observations (HENDO projects) respectively. Their Within Cluster Sum of Squares are 20600.88, 42544.32, 37476.60, 14300.08, 17029.28 respectively.

## 6.2 Supervised Learning

Having processed the data, they are fed to 10 classification methods, each with its own strengths and weaknesses, for two main reasons. For starters, predicting which algorithm is going to produce the most desirable results beforehand for such a dataset is not easy.

Statistic measures explained in later sections are used to rank each algorithm's performance in terms of what is most important to the company. Ultimately, turning the statistics mode off, the algorithms are used to predict the truly as-of-yet future cases for which no label exists.

### **6.2.1 Logistic Regression**

Logistic regression is handled by rxLogit, and is computed using the Iteratively Reweighted Least Squares (IRLS) algorithm, which is equivalent to full maximum likelihood.<sup>[11]</sup>

### **6.2.2 Decision Trees**

Decision Trees is handled by rxDTree, which is a parallel external memory decision tree algorithm targeted for very large data sets. It is modelled after R's rpart (Version 4.1-0) and inspired by the algorithm proposed by Yael Ben-Haim and Elad Tom-Tov (2010). It uses a histogram as the approximate compressed representation of the data and builds the tree in a breadth-first fashion using horizontal parallelism. maxNumBins specifies the maximum number of bins for the histogram of each continuous independent variable and thus controls the accuracy of the algorithm. Also, rxDTree builds histograms with roughly equal number of observations in each bin and checks only the boundaries of the bins as candidate splits to find the best split. So it is possible that a suboptimal split is chosen if maxNumBins is too small. This may cause the tree to be different from one constructed by a standard algorithm. Increasing maxNumBins allows more accurate results but with increased time and memory usage. Surrogate splits may be used to assign observations for which the primary split variable is missing. Surrogate splits compare the groups produced by the remaining predictor variables to the groups produced by the primary split variable, and the predictors are ranked by how well their groups match the primary predictor. The best match is used as the surrogate split.<sup>[11]</sup>

### **6.2.3 Naïve Bayes**

Naïve Bayes, studied extensively since the 1960s, is a highly scalable simple probabilistic classifier based on Bayes' theorem describing the probability of an event, based on prior knowledge of conditions that might be related to the event. The theorem is applied with strong independence assumptions between the features. Conditional probabilities are calculated for all of the factor variables, and means and standard deviations are calculated

for numeric variables. It follows the standard practice of assuming that variables follow Gaussian distributions.

#### **6.2.4 Random Forest**

Random Forest is handled by rxDForest, which is a parallel external memory decision forest algorithm targeted for very large data sets. It is modelled on the random forest ideas of Leo Breiman and Adele Cutler and the randomForest package of Andy Liaw and Matthew Weiner. In a decision forest, a number of decision trees are fit to bootstrap samples of the original data. Observations omitted from a given bootstrap sample are termed “out-of-bag” observations. For a given observation, the decision forest prediction is determined by the result of sending the observation through all the trees for which it is out-of-bag. For classification, the prediction is the class to which a majority assigned the observation, and for regression, the prediction is the mean of the predictions. For each tree, the out-of-bag observations are fed through the tree to estimate out-of-bag error estimates. The reported out-of-bag error estimates are cumulative (that is, the *i*th element represents the out-of-bag error estimate for all trees through the *i*th).<sup>[11]</sup>

#### **6.2.5 Stochastic Gradient Boosting**

Stochastic Gradient Boosting is handled by rxBTrees, which is a parallel external memory algorithm for stochastic gradient boosted decision trees targeted for very large data sets. It is based on the gradient boosting machine of Jerome Friedman and Trevor Hastie and Robert Tibshirani and modeled after the gbm package of Greg Ridgeway with contributions from others. In a decision forest, a number of decision trees are fit to bootstrap samples of the original data. Observations omitted from a given bootstrap sample are termed “out-of-bag” observations. For a given observation, the decision forest prediction is determined by the result of sending the observation through all the trees for which it is out-of-bag. For classification, the prediction is the class to which a majority assigned the observation, and for regression, the prediction is the mean of the predictions. For each tree, the out-of-bag observations are fed through the tree to estimate out-of-bag error estimates. The reported out-of-bag error estimates are cumulative (that is, the *i*th element represents the out-of-bag error estimate for all trees through the *i*th).<sup>[11]</sup>

### 6.2.6 Stochastic Dual Coordinate Ascent

Stochastic Dual Coordinate Ascent is handled by the `rxFastLinear()` algorithm which is based on the Stochastic Dual Coordinate Ascent (SDCA) method, a state-of-the-art optimisation technique for convex objective functions. The algorithm can be scaled for use on large out-of-memory data sets due to a semi-asynchronised implementation that supports multithreaded processing. Several choices of loss functions are also provided and elastic net regularisation is supported. The SDCA method combines several of the best properties and capabilities of logistic regression and SVM algorithms. <sup>[11, 12]</sup>

### 6.2.7 Boosted Decision Trees

Boosted Decision Trees is handled by the `rxFastTrees()` algorithm which is a high performing, state of the art scalable boosted decision tree that implements FastRank, an efficient implementation of the MART gradient boosting algorithm. MART learns an ensemble of regression trees, which is a decision tree with scalar values in its leaves.

Traditional optimisation algorithms, such as stochastic gradient descent (SGD), optimise the empirical loss function directly. The SDCA chooses a different approach that optimises the dual problem instead. The dual loss function is parametrised by per-example weights. In each iteration, when a training example from the training data set is read, the corresponding example weight is adjusted so that the dual loss function is optimised with respect to the current example. No learning rate is needed by SDCA to determine step size as is required by various gradient descent methods. <sup>[11, 12]</sup>

### 6.2.8 Ensemble of Decision Trees

Ensemble of Decision Trees is handled by the `rxFastForest()` algorithm which is a random forest that provides a learning method for classification by constructing an ensemble of decision trees at training time, and outputting the class that is the mode of the classes of the individual trees. Random decision forests can correct for the over-fitting to training data sets to which decision trees are prone. Decision trees have several advantages:

- They are efficient in both computation and memory usage during training and prediction.
- They can represent non-linear decision boundaries.
- They perform integrated feature selection and classification.
- They are resilient in the presence of noisy features.

Fast forest regression is a random forest and quantile regression forest implementation using the regression tree learner in rxFastTrees. The model consists of an ensemble of decision trees. Each tree in a decision forest outputs a Gaussian distribution by way of prediction. An aggregation is performed over the ensemble of trees to find a Gaussian distribution closest to the combined distribution for all trees in the model.

This decision forest classifier consists of an ensemble of decision trees. Generally, ensemble models provide better coverage and accuracy than single decision trees. Each tree in a decision forest outputs a Gaussian distribution by way of prediction. An aggregation is performed over the ensemble of trees to find a Gaussian distribution closest to the combined distribution for all trees in the model. <sup>[11, 12]</sup>

### **6.2.9 Neural Networks**

Neural Networks is handled by the rxNeuralNet() algorithm which supports a user-defined multilayer network topology with GPU acceleration. In the spirit of the human brain, neural network, a class of prediction models was developed. The neurons in the graph are arranged in layers, where neurons in one layer are connected by a weighted edge (weights can be 0 or positive numbers) to neurons in the next layer. The first layer is called the input layer, and each neuron in the input layer corresponds to one of the features. The last layer of the function is called the output layer. In the case of binary neural networks it contains two output neurons, one for each class, whose values are the probabilities of belonging to each class. The remaining layers are called hidden layers. The values of the neurons in the hidden layers and in the output layer are set by calculating the weighted sum of the values of the neurons in the previous layer and applying an activation function to that weighted sum. A neural network model is defined by the structure of its graph (namely, the number of hidden layers and the number of neurons in each hidden layer), the choice of activation function, and the weights on the graph edges. Using the training data, the neural network algorithm tries to learn the optimal weights on the edges.

Although neural networks are widely known for use in deep learning and modelling complex problems such as image recognition, they are also easily adapted to regression problems. Any class of statistical models can be considered a neural network if they use adaptive weights and can approximate non-linear functions of their inputs. Neural

network regression is especially suited to problems where a more traditional regression model cannot fit a solution. [11, 12]

### **6.2.10 Fast Logistic Regression**

Fast Logistic Regression is handled by the `rxLogisticRegression()` algorithm which is used to predict the value of a categorical dependent variable from its relationship to one or more independent variables assumed to have a logistic distribution. The difference between this model and the first one (Logistic Regression) is that this is typically faster and allows for a wider customisation.

The optimisation technique used for `rxLogisticRegression` is the limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS). Both the L-BFGS and regular BFGS algorithms use quasi-Newtonian methods to estimate the computationally intensive Hessian matrix in the equation used by Newton's method to calculate steps. But the L-BFGS approximation uses only a limited amount of memory to compute the next step direction, so that it is especially suited for problems with a large number of variables. The `memorySize` parameter specifies the number of past positions and gradients to store for use in the computation of the next step.

This learner can use elastic net regularisation: a linear combination of L1 (lasso) and L2 (ridge) regularisations. Regularisation is a method that can render an ill-posed problem more tractable by imposing constraints that provide information to supplement the data and that prevents overfitting by penalising models with extreme coefficient values. This can improve the generalisation of the model learned by selecting the optimal complexity in the bias-variance trade-off. Regularisation works by adding the penalty that is associated with coefficient values to the error of the hypothesis. An accurate model with extreme coefficient values would be penalised more, but a less accurate model with more conservative values would be penalised less. L1 and L2 regularisation have different effects and uses that are complementary in certain respects. `l1Weight`: can be applied to sparse models, when working with high-dimensional data. It pulls small weights associated features that are relatively unimportant towards 0. `l2Weight`: is preferable for data that is not sparse. It pulls large weights towards zero.

## 6.3 Model Evaluation

An ideal model would discriminate unerringly between approved and disapproved projects, scoring perfectly on all measures, however, since that is next to impossible, different measures evaluate different things, and how well a model does on a particular measure reveals a desirable or non-desirable aspect of its nature, an achievement of partial success. Thus, there is a need to rank models in an objective manner, providing a means of deciding by which amount distinct models differ in their capacity to discriminate between approved and cancelled projects. The most critical rate for this particular application is how good a job the model does on identifying True Positives, even at the expense of having a higher false positive rate. That said, all measures play a role to one extend or another, and hitting all the true positives is not a panacea, or else the classifier could simply predict everything as positive. The measures by which the model is evaluated follow below.

### 6.3.1 Accuracy

When the performance of a classifier is in question, accuracy is considered the most straightforward and intuitive metric as it basically tells us, overall, how often it is correct. Unfortunately, accuracy alone is insufficient under certain circumstances, like when there's significant class imbalance. To aid in conceptualisation, an extreme example is a model for diagnosing a disease which only ails one in a thousand people. Even if the model is a sham, always predicting the majority class, its accuracy is still going to be 99.9%. This is backed up by the accuracy paradox for predictive analytics which states that a model with lower accuracy can have higher predictive power over a model with higher accuracy. Which is why, other measures, such as the AUC, are going to be prioritised and given more attention.

$$[EQ1] Acc = \frac{(TN + TP)}{(TN + FN + FP + TP)}$$

Accuracy's equation is given by EQ1.

### 6.3.2 Balanced Accuracy

An alternative accuracy measure that does not lead to an optimistic estimate when a biased classifier is tested on an imbalanced dataset is the Balanced Accuracy. This is

achieved by replacing the conventional point estimate of accuracy by an estimate of the posterior distribution of the balanced accuracy <sup>[15]</sup>

$$[EQ2] BA = \left( \frac{TP}{(TP + FN)} \right) + \left( \frac{TN}{(TN + FP)} \right)$$

Balanced Accuracy's equation is given by EQ2.

### 6.3.3 Detection Rate

Detection rate is the rate at which the classifier identifies the True Positive cases, i.e. the true positives over all cases (n).

$$[EQ3] Detection Rate = \frac{TP}{(TN + FN + FP + TP)}$$

Detection Rate's equation is given by EQ3.

### 6.3.4 Misclassification Rate

Misclassification Rate is the answer to the question “overall, how often is the classifier wrong?”. Put another way, it is the rate at which the model predicts wrongly.

$$[EQ4] Misclassification Rate = 1 - Accuracy = \frac{(FP + FN)}{(TN + FN + FP + TP)}$$

Misclassification Rate's equation is given by EQ4.

### 6.3.5 Sensitivity / Recall / True Positive Rate

Sensitivity or Recall (as it is called in Psychology), or True Positive Rate is the proportion of Real Positive cases that are Correctly Predicted Positive, answering the question: “when the project is actually approved, how often does the classifier predict approved?”. A fundamental flaw in this is that it overlooks how well negative examples are handled, thus propagating the underlying marginal prevalences and biases, failing to take the chance level performance into account. It is, however, considered primary for it is one of the two legs that the ROC analysis stands on. With regards to Bayesian statistics, Sensitivity is a conditional probability <sup>[17, 21]</sup>

$$[EQ5] TPR = \frac{TP}{(TP + FN)}$$

Sensitivity's equation is given by EQ5.



### 6.3.6 False Positive Rate (FPR)

False Positive Rate is the proportion of Real Negatives that occur as Predicted Positives, answering the question: when the project is actually cancelled, how often does the classifier predict approved? It is also considered primary for it is the second of the two legs which ROC analysis is based on. <sup>[17]</sup>

$$[EQ6] FPR = \frac{FP}{(FP + TN)}$$

False Positive Rate's equation is given by EQ6.

### 6.3.7 Specificity / True Negative Rate (TNR)

There's nothing inherently unique about the positive cases, however. Inversing positives and negatives can predict the opposite case. Inverse Recall, for instance, is the proportion of Real Negative cases that are correctly Predicted Negative, and is also known as the True Negative Rate. this answers the question: "when the project is actually cancelled, how often does the classifier predict cancelled?" With regards to Bayesian statistics, Specificity is a conditional probability <sup>[17, 21]</sup>

$$[EQ7] TNR = 1 - FPR = \frac{TN}{(TN + FP)}$$

Specificity's equation is given by EQ7.

### 6.3.8 False Negative Rate (FNR)

False Negative Rate is the proportion of Real Positives that are Predicted Negatives, answering the question: "when the project is actually approved, how often does the classifier predict cancelled"? <sup>[17]</sup>

$$[EQ8] FNR = \frac{FN}{(FN + TP)}$$

False Negative Rate's equation is given by EQ8.

### 6.3.9 Precision / Positive Predictive Value (PPV1)

Positive Predictive Value is the proportion of projects with positive prediction results which are correctly identified, answering the question: when the classifier predicts approved, how often is it correct? Precision is a measure of statistical variability, a description of random errors, but a fundamental flaw in this is that it overlooks how well negative examples are handled, thus propagating the underlying marginal prevalences and

biases, failing to take the chance level performance into account. With regards to Bayesian statistics, Precision is a posterior probability. <sup>[17, 21]</sup>

$$[EQ9] \frac{TP}{(TP + FP)}$$

Precision's equation is given by EQ9.

### 6.3.10 Positive Predictive Value (PPV2)

PPV2 is similar to PrecisionPPV1, except that it takes prevalence into account. In PPV1, the less projects were being approved, the surer we could be that a negative prediction indicated a cancelled project, and the less sure that a positive prediction indicated an approved project. Under no class imbalance, PPV2 equals PPV1 (Precision). <sup>[18]</sup>

$$\begin{aligned} [EQ10] PPV2 &= \frac{Sensitivity \cdot Prevalence}{Sensitivity \cdot Prevalence + (1 - Specificity) \cdot (1 - prevalence)} \\ &= \frac{\left( \frac{TP}{(TP + FN)} \cdot \frac{(FP + TP)}{(TN + FN + FP + TP)} \right)}{\left( \frac{TP}{(TP + FN)} \cdot \frac{(FP + TP)}{(TN + FN + FP + TP)} \right) + \left( 1 - \left( \frac{TN}{(TN + FP)} \right) \right) \cdot \left( 1 - \left( \frac{(FP + TP)}{(TN + FN + FP + TP)} \right) \right)} \end{aligned}$$

Positive Predictive Value's equation is given by EQ10.

### 6.3.11 Negative Predictive Value (NPV1)

Negative Predictive Value is the proportion of projects with negative predictions who were correctly classified, in other words, when the classifier predicts cancelled, how often is it correct? With regards to Bayesian statistics, Precision is a posterior probability. <sup>[21]</sup>

$$[EQ11] NPV1 = \frac{TN}{(TN + FN)}$$

Positive Predictive Value's equation is given by EQ11.

### 6.3.12 Negative Predictive Value (NPV2)

NPV2 is similar to NPV1, except that it takes prevalence into account. In NPV1, the less projects were being approved, the surer we could be that a negative prediction indicated a cancelled project, and the less sure that a positive prediction indicated an approved project. Under no class imbalance, NPV2 equals NPV1 <sup>[18]</sup>

$$\begin{aligned} [EQ12] NPV2 &= \frac{Specificity \cdot (1 - Prevalence)}{(1 - Sensitivity) \cdot Prevalence + Specificity \cdot (1 - Prevalence)} \\ &= \frac{\left( \frac{TN}{(TN + FP)} \cdot \left( 1 - \frac{(FP + TP)}{(TN + FN + FP + TP)} \right) \right)}{\left( \left( 1 - \frac{TP}{(TP + FN)} \right) \cdot \frac{(FP + TP)}{(TN + FN + FP + TP)} \right) + \frac{TN}{(TN + FP)} \cdot \left( 1 - \frac{(FP + TP)}{(TN + FN + FP + TP)} \right)} \end{aligned}$$

Positive Predictive Value's equation is given by EQ12.

### 6.3.13 False Discovery Rate (FDR)

False Discovery Rate is a way to measure the rate at which the classifier does Type I Error, given by the amount of false positive prediction out of all predicted positives.

$$[EQ13] FDR = \frac{FP}{(FP + TP)}$$

False Discovery Rate's equation is given by EQ13.

### 6.3.14 Null Error Rate

Null Error Rate shows how often would the classifier be wrong if it only predicted the majority class (in this case, "Approved"), serving as a baseline to compare a classifier against. It should be noted that the accuracy paradox mentioned on the "Accuracy" metric applies in Null Error Rate as well.

$$[EQ14] NER = \frac{MinClass}{MaxClass}$$

Null Error Rate's equation is given by EQ14.

### 6.3.15 Prevalence

Prevalence is essentially the probability before any attempt at prediction is made, that a project is going to be approved, namely, the prior probability of approval, showing how often projects are approved in the dataset. <sup>[17, 21]</sup>

$$[EQ15] Prevalence = \frac{(FP + TP)}{(TN + FN + FP + TP)}$$

Prevalence's equation is given by EQ15.

### 6.3.16 F1 Score

F-measure is the weighted average of the true positive rate (recall) and the precision, measuring the model's accuracy; in other words, it is their harmonic mean, referencing the True Positives to the Arithmetic Mean of Predicted Positives and Real Positives. F-score, like recall and precision, only considers the positive predictions, equating the probabilities under the assumption that the positive labels and the positive predictions should have the same distribution and prevalence. A fundamental flaw in this is that it overlooks how well negative examples are handled, thus propagating the underlying

marginal prevalences and biases, failing to take the chance level performance into account. <sup>[17]</sup>

$$[EQ16] F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} = \frac{(2 \cdot TP)}{(2 \cdot TP) + FP + FN}$$

F1's equation is given by EQ16.

### 6.3.17 G-measure

G-measure is the Geometric Mean of Recall and Precision; a normalisation of True Positives to the geometric mean of Predicted Positives and Real Positives, indicating the central tendency of the predictions. A fundamental flaw in this is that it overlooks how well negative examples are handled, thus propagating the underlying marginal prevalences and biases, failing to take the chance level performance into account. <sup>[17]</sup>

$$[EQ17] G = \sqrt{Precision \cdot Recall} = \sqrt{\left(\frac{TP}{(TP + FP)}\right) \cdot \left(\frac{TP}{(TP + FN)}\right)}$$

G-measure's equation is given by EQ17.

### 6.3.18 Matthews correlation coefficient, $\phi$ (PhiMCC)

Matthews  $\phi$  is a correlation coefficient between the observed and predicted values, measuring a binary model's classification quality and is viewed as a balanced measure, used even if extended class imbalance is present.  $\phi$  ranges from -1 to +1 with "+1" illustrating a perfect prediction, "0" a random one and "-1" signifying total disagreement between predicted and actual values.

$$[EQ19] \phi = \frac{((TP * TN) - (FP * FN))}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}}$$

Matthews correlation coefficient's equation is given by EQ19.

### 6.3.19 Cohen's kappa coefficient, $\kappa$ (CohensK)

Cohen's  $\kappa$  is a measure of how well the classifier performed as compared to how well it would have performed simply by chance.  $\kappa$  is, as most measures are, biased. A high  $\kappa$  value implies a significant difference between the accuracy and the Null Error Rate. <sup>[17]</sup>

$$P_0 = \frac{(TP + TN)}{(TN + FN + FP + TP)}$$

$$M_a = \frac{((TP + FP) \cdot (TP + FN))}{(TN + FN + FP + TP)}$$

$$M_b = \frac{((FN + TN) \cdot (FP + TN))}{(TN + FN + FP + TP)}$$

$$P_e = \frac{(M_a + M_b)}{(TN + FN + FP + TP)}$$

$$[EQ20] \kappa = \frac{(P_0 - P_e)}{(1 - P_e)}$$

Cohen's kappa coefficient's equation is given by EQ20.

### 6.3.20 Youden's J statistic

With a range from -1 to 1, Youden's index is an estimation of the probability of an informed decision, meaning that it's 0 for a classifier that gives the same proportion of positive results for projects that were approved or disapproved. Its underlying logic begins with subtracting the proportion of incorrectly predicted approved projects from the correctly predicted ones as a measure of the model's success on the approved projects  $\left(\frac{TP-FN}{TP+FN}\right)$ . By extension,  $\left(\frac{TN-FP}{TN+FP}\right)$  applies for the cancelled projects. Youden's J takes the mean value of the two as its value, assuming parity on the cost of False Positives and False Negatives.

Youden's index is the maximum vertical distance between the Receiver Operating Characteristic (ROC) curve and the chance line and is often used alongside ROC analysis. Youden's J statistic is defined for all points of a ROC curve, and should a model yield a probability result rather than a binary, its maximum value can be used for selecting the optimum cut-off point. [16, 17, 19]

$$[EQ21] J = TPR + FNR - 1 = \left(\frac{TP}{TP + FN}\right) + \left(\frac{TN}{TN + FP}\right) - 1$$

Youden's J statistic's equation is given by EQ21.

### 6.3.21 Receiver Operating Characteristic (ROC) Curve

Receiver Operating Characteristic curves visualise and summarise the performance of a binary classifier over all possible thresholds by plotting the True Positive Rate (y-axis) against the False Positive Rate (x-axis) as the threshold for assigning observations to a given class varies. The benefit of it is vast, as other metrics typically only represent the

error rate for a single threshold. The default threshold is 0.5, yielding a certain amount of TPR and FPR, however, it can be overruled and the predictions can be fashioned to another design. The quicker the ROC curve reaches for the upper left corner, the better it does on classifying correctly, conversely, the more it stays in proximity to the diagonal line, the worst a job it does. This is because the upper left corner represents an idealised situation, whilst the diagonal line reflects a classification technique tantamount to random guessing. The ROC curve and by extension its AUC are insensitive to whether the predicted probabilities are properly calibrated to actually represent probabilities, meaning that they would remain identical for probabilities ranged from 0.9 to 1 as opposed to the norm of 0 to 1, so long as the ordering of observations by predicted probability remained the same. ROC curves are only sensitive to rank ordering for the crucial part, for the metric illustrates how good a job of class separation has been achieved by the classifier. It can be utilised for evaluation in highly balanced and highly unbalanced datasets alike for the underlying logic is the representation of the probability that a classifier will rank a randomly chosen positive observation higher than a randomly chosen negative observation. One has to decide for themselves whether they would rather minimise the False Positive Rate or maximise the True Positive Rate. The ROC curve aids in visualisation of that and comprehension of its impact (watching how the former rate raises with the latter).

### **6.3.22 Area Under the Curve (AUC)**

ROC curves have some great strengths and are consistently amongst the preferred evaluation methods, which means that for ease of use, a quantification method is needed for them. AUC is literally the area under the Receiver Operating Characteristic Curve. The probabilistic interpretation is that if a positive case and a negative case are randomly chosen, the probability that the positive case outranks the negative case according to the classifier is given by the AUC. This is evident from the figure, where the total area of the plot is normalised to one. The cells of the matrix enumerate all possible combinations of positive and negative cases, and the fraction under the curve comprises the cells where the positive case outranks the negative one.

# 7 Implementation

In order to achieve everything mentioned thus far in an autonomous manner, absent of the hurdles of trying to figure out the correct configurations for each computer and how to code them, only to be left with the task of laboriously iterating over the algorithms, the ‘HEDNO Oracle’ has been developed. It’s a fully Open Source and Free programme, licensed under the GNU GENERAL PUBLIC LICENSE GPLv2, and its complete source code is publicly available via the <https://github.com/N1h11sT/HEDNO-Oracle> github repository.

The programme offers full-scale customisation capabilities, both directly via the Settings form and localised settings options on each form, and indirectly via open access to all the SQL & R code which can be modified without a need of recompilation of HEDNO Oracle. Furthermore, it has a built-in automatic update feature so that if the need arises for a new version or build, the programme can update itself (after an explicit positive confirmation is issued by the user so that no update is forced under any circumstances). The update-check mechanism defaults to TRUE, but can safely be turned off from the settings form.

The programme also possesses a framework for multilingual support, such that the only thing needed for it being translated to another language is literally copying an existing language sub-folder (found inside the “Language” folder of the installation directory), e.g. the “en-GB” folder, renaming it to a designation of another language e.g, “el-GR”, and translating the texts of the files inside it to that language. The programme will automatically pick up the new language as an option on its Settings Form, and if selected, everything will be translated to that language thereafter.

Each time an input value is requested from the user, the input form validates results and only allows for correct values so that the programme can continue to run smoothly, minimising any unexpected behaviour whilst not assuming any expert knowledge on the user part. It is also worth mentioning that when numbers are requested as an input, basic maths functions are also supported (i.e. exponents, square roots, division, etc.)

HEDNO Oracle is a multi-threaded programme, using Microsoft R’s highly scalable R packages enabling solutions with concurrency through clusters and parallelised workloads. This is important, if not essential in big data, as the volume of the data has

such a detrimental effect as to hinder, impede, or even altogether prohibit execution. Complications span from the inevitability of tremendously extensive waiting times to terminal obstacles like memory insufficiency.

Being in possession of, or having been granted access to a cluster, is not, however, a prerequisite. The code is written in such a way that it can run from run-of-the-mill computers/laptops and clusters alike. To enable the latter, all which is needed is configuring a single line of code setting the ‘context’ in the beginning of the “[Initialisation].R” file under the “Functions” folder for the HEDNO Oracle, or “General.R” for the R-only version. The contexts available are:

- "RxLocalSeq" or "local" to execute the code from a local computer or laptop.
- "RxLocalParallel" or "localpar" to execute the code from a local computer in a parallel manner using the back-end for HPC computations, which is only used to distribute computations via the rxExec function.
- "RxSpark" or "spark" to connect to an Apache Spark general engine for large-scale data processing<sup>[131]</sup>.
- "RxHadoopMR" or "hadoopmr" to execute the code from a Hadoop cluster.
- "RxInSqlServer" or "sqlserver" to create a compute context for running RevoScaleR analyses inside Microsoft SQL Server.
- "RxInTeradata" or "teradata" to create a compute context for running RevoScaleR analyses inside a Teradata database.
- "RxForeachDoPar" or "dopar" to create a compute context object using the registered foreach parallel back end, which is only used to distribute computations via the rxExec function.

The line defaults to

```
rxSetComputeContext(computeContext = RxLocalSeq())
```

so as to render execution possible from any device to begin with.

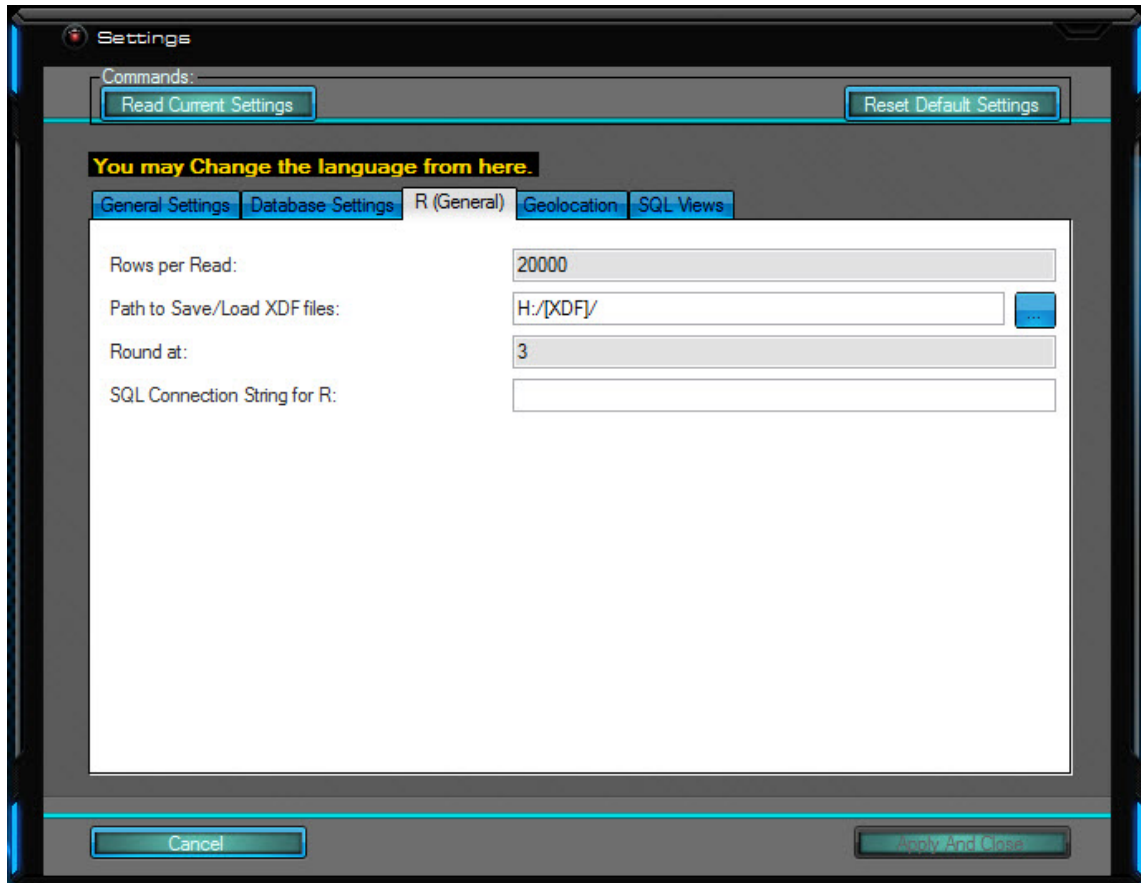
## 7.1 Settings

The first time HEDNO Oracle runs on a computer, it automatically identifies the most suitable settings for the given computer, like which language to be set to (Greek for Greek computers, English for English computers, and so on, provided the programme is translated to the computer’s language; otherwise it defaults to English), or the SQL Server (in this instance, the Microsoft SQL Server) and its connection string, as well as the R Server (or equivalent) and it defaults the settings so that everything can run without any



configuration whatsoever. That being said, configuration remains possible via the “Settings” form.

### 7.1.1 R Settings



**Figure 9:** R Settings

- **Rows per Read:** Determines how many blocks/chunks will be created. XDF files are optimised to be read in a certain way and the value of how many rows will be read at each time effectively determines the block size; For a dataset with 20,000 rows, there will only be 1 chunk, whilst for one with 133,098 like the current one, there will be  $Ceil(133,098/20,000) = 7$  chunks. The default, 20000, is an appropriate value for this dataset
- **Path to Save/Load XDF files:** This is the local directory in which all files created by the programme will be saved in, unless otherwise specified.
- **Round at:** When viewing statistics, all values will be rounded at that decimal point; this is only for viewing purposes and does not round numbers internally so as to not lose precision.
- **SQL Connection String for R:** This is the connection string that R is going to be ‘told’ to use; it is automatically figured out, but if need arises it can be overridden here.

## 7.1.2 Geolocation Settings

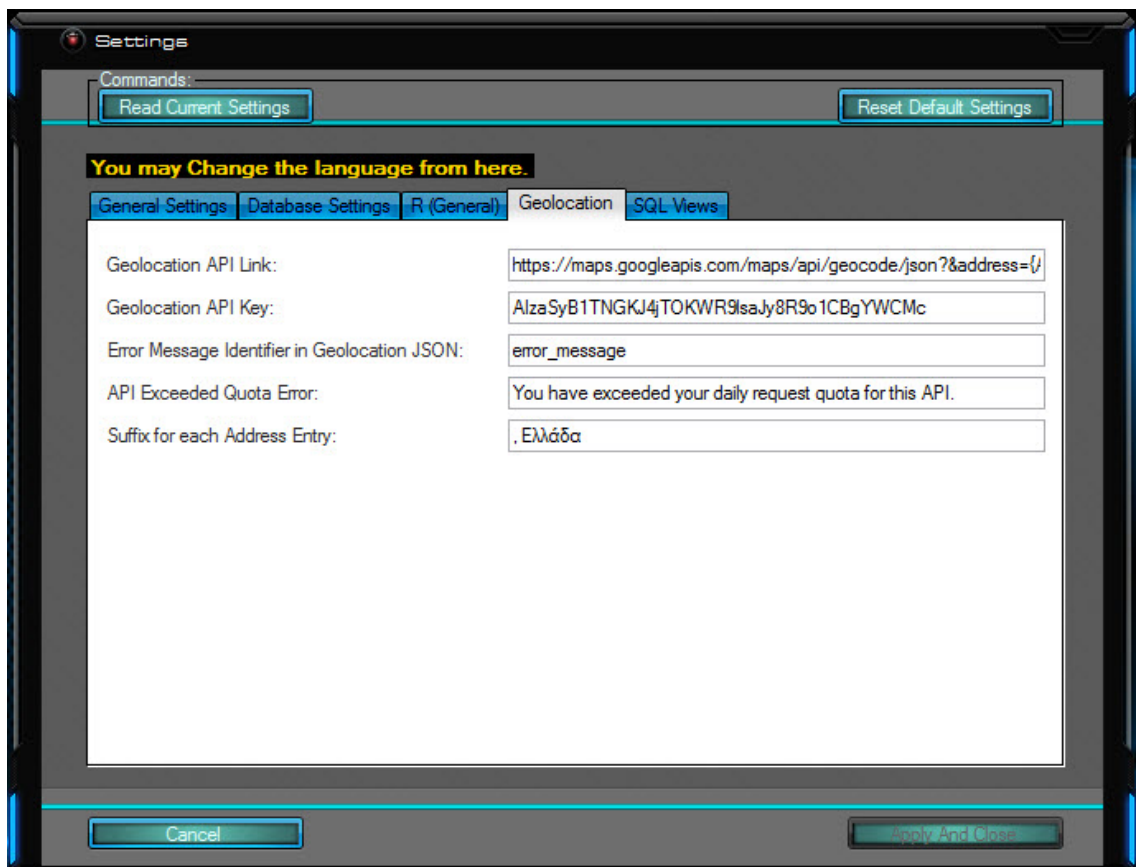


Figure 10: Geolocation Settings

- **Geolocation API Link:** This is the URL to the Geolocation API to be used by the programme to Geolocate the projects (needs to return a JSON). It defaults to Google's API.
- **Geolocation API Key:** In order to use the service, one has to acquire an API key. The API Key lifts the public use restrictions, however, it comes with restrictions of its own. A Google API Key can be acquired from google:  
<https://developers.google.com/maps/documentation/geolocation/get-api-key>.
- **Error Message Identifier in Geolocation JSON:** If an error occurs, such as that the address could not be identified or the service is down, which string identifier will appear in the JSON?
- **API Exceeded Quota Error:** The programme responds differently to a general API error and to *Quota Exceeded* one; this is the error message that the JSON will provide in case of the latter.
- **Suffix for each Entry:** When Geolocating, a string that should be suffixed to each address before Geolocation is attempted.

### 7.1.3 SQL Views Settings

Settings

Commands:

You may Change the language from here.

SQL View City Column Name:	<input type="text" value="Onoma_Polis"/>
SQL View ERGA Name:	<input type="text" value="v4Erga"/>
SQL View Column GeoLocationX Name:	<input type="text" value="GeoLocX"/>
SQL View Column GeoLocationY Name:	<input type="text" value="GeoLocY"/>
SQL View Column ID Name:	<input type="text" value="ID_Erga"/>
SQL View FinalDataset Name:	<input type="text" value="v9FinalDataset"/>
SQL Table EPFA Name:	<input type="text" value="EPFA"/>
SQL Table City Column Name:	<input type="text" value="ПОЛН"/>
SQL Table GeoLocationX Name:	<input type="text" value="GeoLocX"/>
SQL Table GeoLocationY Name:	<input type="text" value="GeoLocY"/>
Suffix for each Address Entry:	<input type="text" value="ID"/>

Figure 11: SQL Views Settings

- **SQL View City Column Name:** The name that the “city” field has on the SQL View.
- **SQL View ERGA Name:** The name of the SQL view containing the pre-processed data excluding pending projects.
- **SQL View Column GeoLocationX Name:** The Column Name that Longitude has on the SQL Views.
- **SQL View Column GeoLocationY Name:** The Column Name that Latitude has on the SQL Views.
- **SQL View Column ID Name:** The Column Name that Project ID has on the SQL Views.
- **SQL View FinalDataset Name:** The name of the SQL View containing the pre-processed data including pending projects to be used for predictions.
- **SQL Table EPFA Name:** The name of the SQL Table containing the original unprocessed data.
- **SQL Table City Column Name:** The name that the “city” field has on the original SQL Table.
- **SQL Table GeolocationX Name:** The Column Name that Longitude has on the original SQL Table.
- **SQL Table GeolocationY Name:** The Column Name that Latitude has on the original SQL Table.

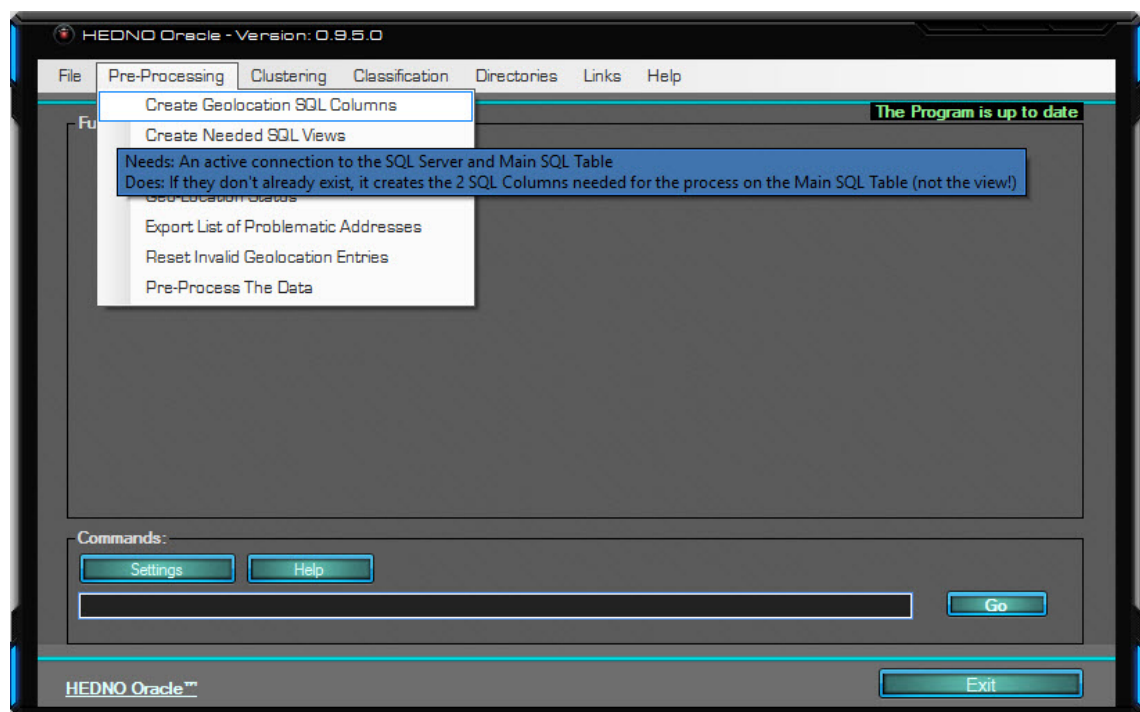
- **SQL Table Column ID Name:** The Column Name that Projects have on the original SQL Table.

## 7.2 Pre-Processing

Before any pre-processing takes place, several steps must first be completed. The “Pre-Processing” menu on HEDNO Oracle is comprised of 7 menu-items covering what needs to be done along with feeding information back to the end-user.

### 7.2.1 Create Geolocation SQL Columns:

Creates the GeolocationX and GeolocationY columns to be used to store the Longitude and Latitude of each project.



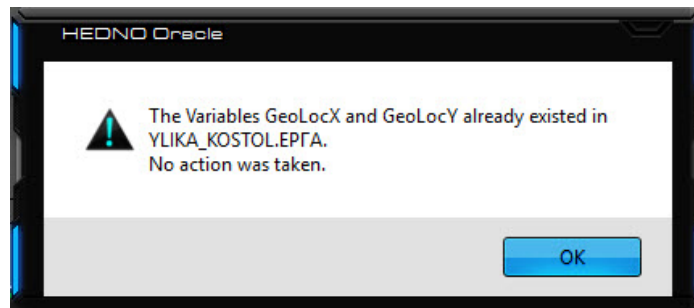
**Figure 12:** Create Geolocation SQL Columns Mouse Hover

Every control and item on the programme comes with a helpful tooltip appearing after the mouse cursor stays over the item for 500 milliseconds. Tooltips will typically inform the user of what an item needs and what it does.

For instance, the first option requires that the computer be connected to the SQL Server and the Main SQL Table (the Projects table, or as is called on the database itself, ‘EPFA’). What it does is, it checks whether each column exists, and if it does not, the programme creates it. This is something that happens to the original table itself, not to something created by the programme, and as such, it is explicitly written to inform the end-user that

they will be updating their *Original SQL Table*. Even so, the SQL Query only includes an “ADD” command to the “ALTER TABLE”.

This action is the only one taking place on the original table; everything else happens in a higher abstract level that does not cause any change to the original data or database.

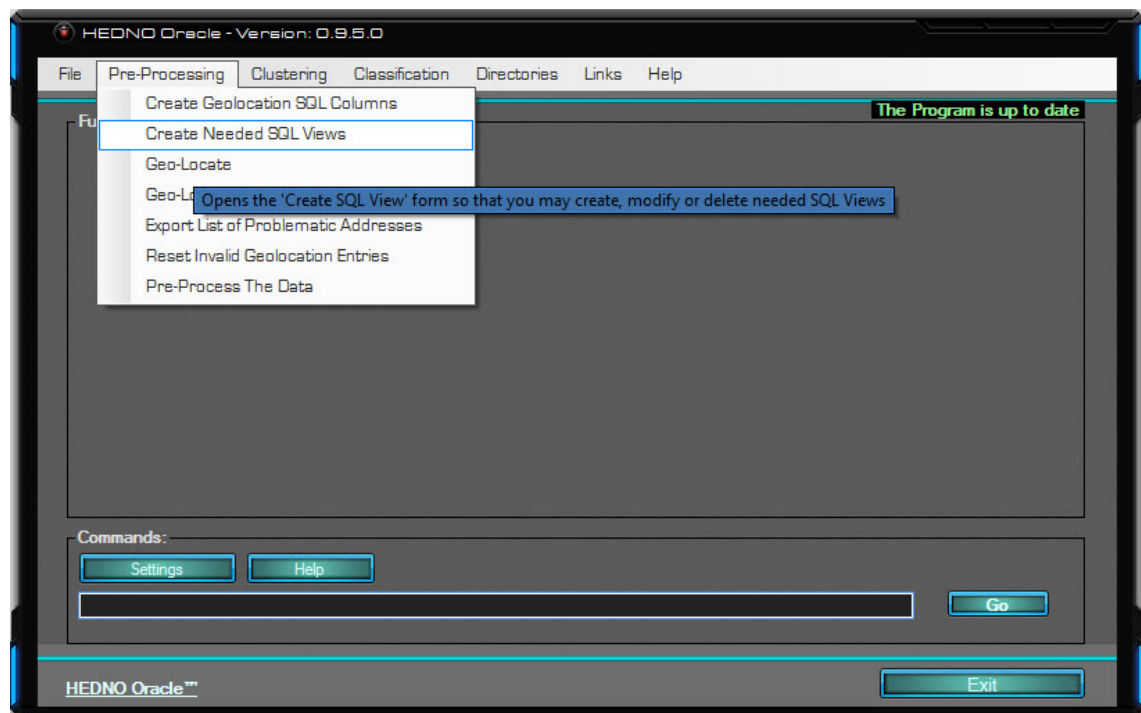


**Figure 13:** Create Geolocation SQL Columns Pushed

Since, on that instance, both GeoLocX and GeoLocY columns already existed on the ‘YLIKA\_KOSTOL’ server in the ‘EPTA’ SQL view, so no action was taken.

### 7.2.2 Create Needed SQL Views:

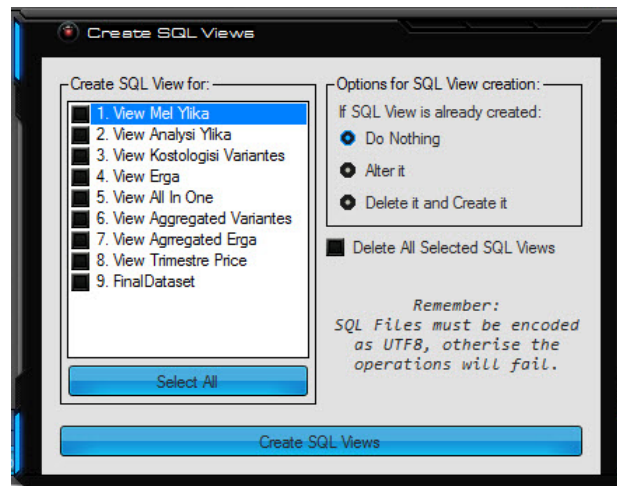
Opens a form in which the user can create, modify or delete SQL views that the programme needs.



**Figure 14:** Create Needed SQL Views Mouse Hover

The option can never delete any view other than those specified on the programme's "SQL" subfolder looking at it from the scope of each installation directory. This way it poses no danger to the Server unless either the SQL scripts are tampered with or a pre-existing view has the same name as the ones specified in the "SQL" subfolder, which is highly unlikely.

Once pushed, the "Create SQL Views" form appears



**Figure 15:** Create Needed SQL Views Pushed

One can select which SQL Views they wish to Create, Alter or Delete from the ListBox and the corresponding .SQL Files are in the Programme's Install Directory under the folder "SQL" and subfolder "Views".

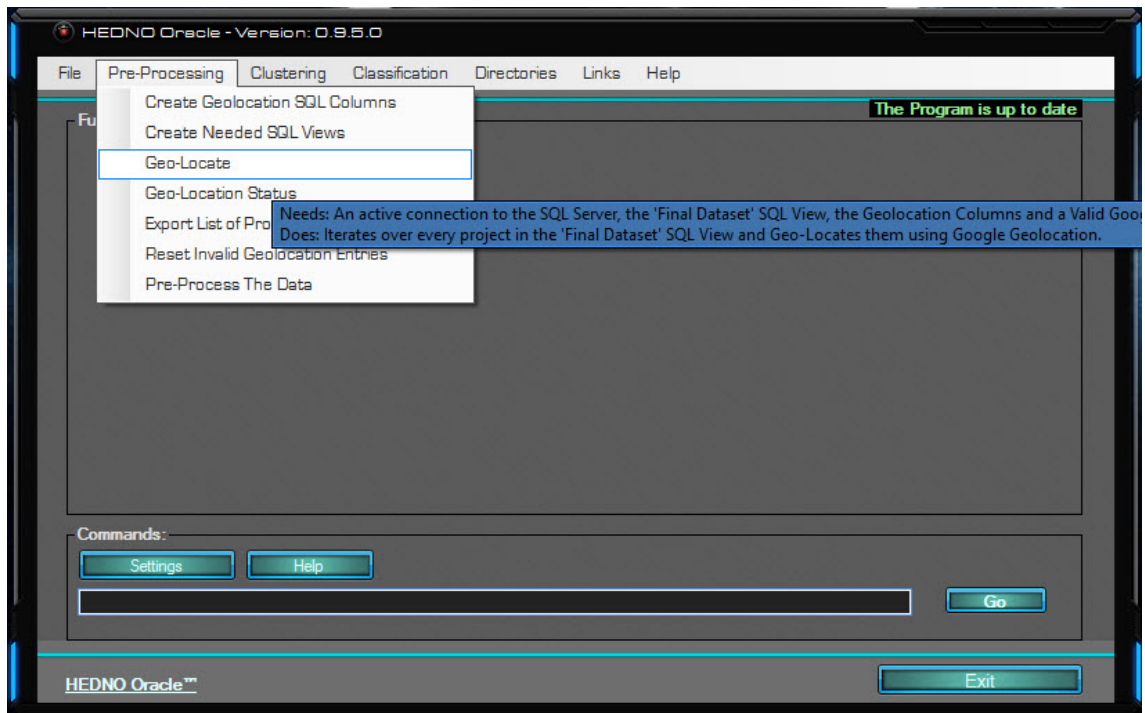
Before creating the view, the programme checks to see whether the view already exists. If it does, the course of action is specified from the "Options for SQL View creation" GroupBox. The programme can ignore that specific view and continue to the next on the list, it can use ALTER instead of CREATE, or it can DELETE it and use CREATE afterwards.

Checking the "Delete All Selected SQL Views" will permanently DELETE all SQL Views associated with the checked file name and will not create them afterwards.

Adding more SQL Views to the list is only a matter of copying the SQL file in the aforementioned subfolder in a UTF8 format, as non-latin characters are going to interfere with the readability of the file.

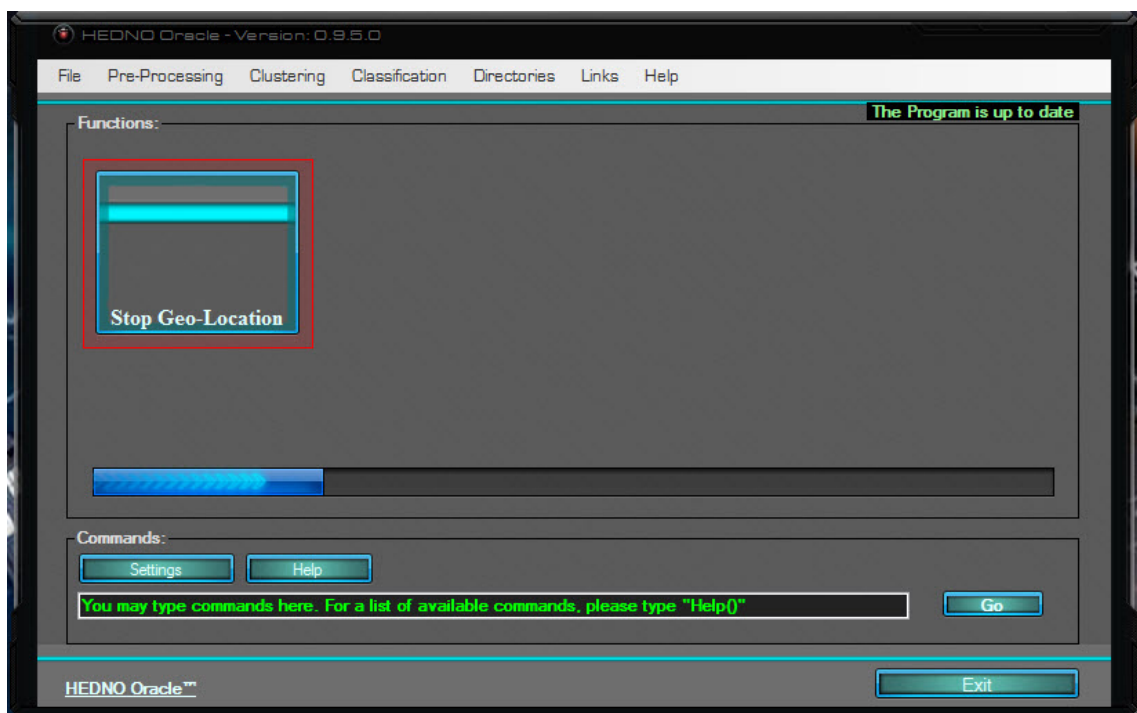
### 7.2.3 Geo-Locate:

Uses the specified API to geo-locate the projects.



**Figure 16:** Geo-Locate Mouse Hover

The amount of projects spans on hundreds of thousands, and this amount of projects can take from hours to days to Geo-locate on Google's geolocation API with a regular API key. Perhaps a paid key, if available, could go a long way in speeding up the process. The delay is also in no small part due to the incessant back and forth between the Server and the programme.





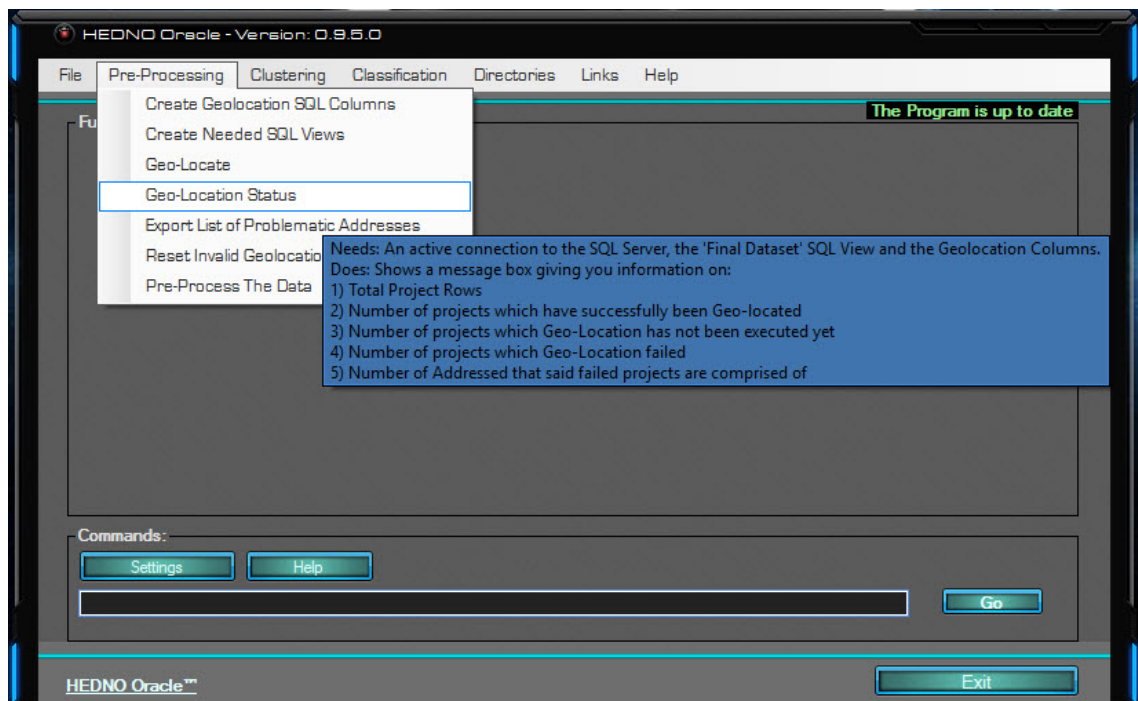
**Figure 17: Geo-Locate Pushed**

Once pushed, the programme fetches the data from the Server, isolates the distinct addresses, pushes them over to the geolocation API, retrieves their longitudes/latitudes and finally uploads them back to each qualified project on the Server.

The process can be terminated at any point before a successful completion by clicking the “Stop Geo-Location” button.

#### **7.2.4 Geo-Location Status:**

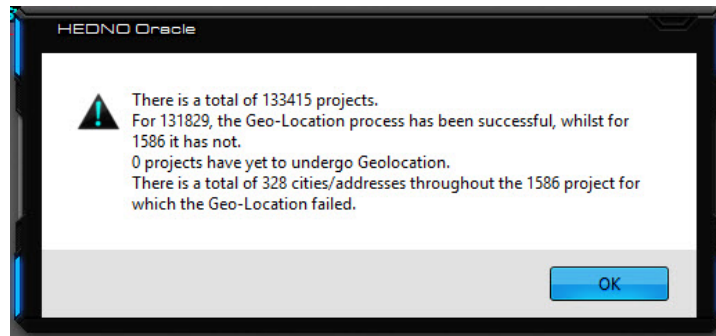
Retrieves Geolocation information portraying the current status of the projects on the server.



**Figure 18: Geo-Location Status Mouse Hover**

The information shown in the MessageBox is about projects that are going to be used for Unsupervised and Supervised learning, including pending projects which are going to be the ones to be predicted.





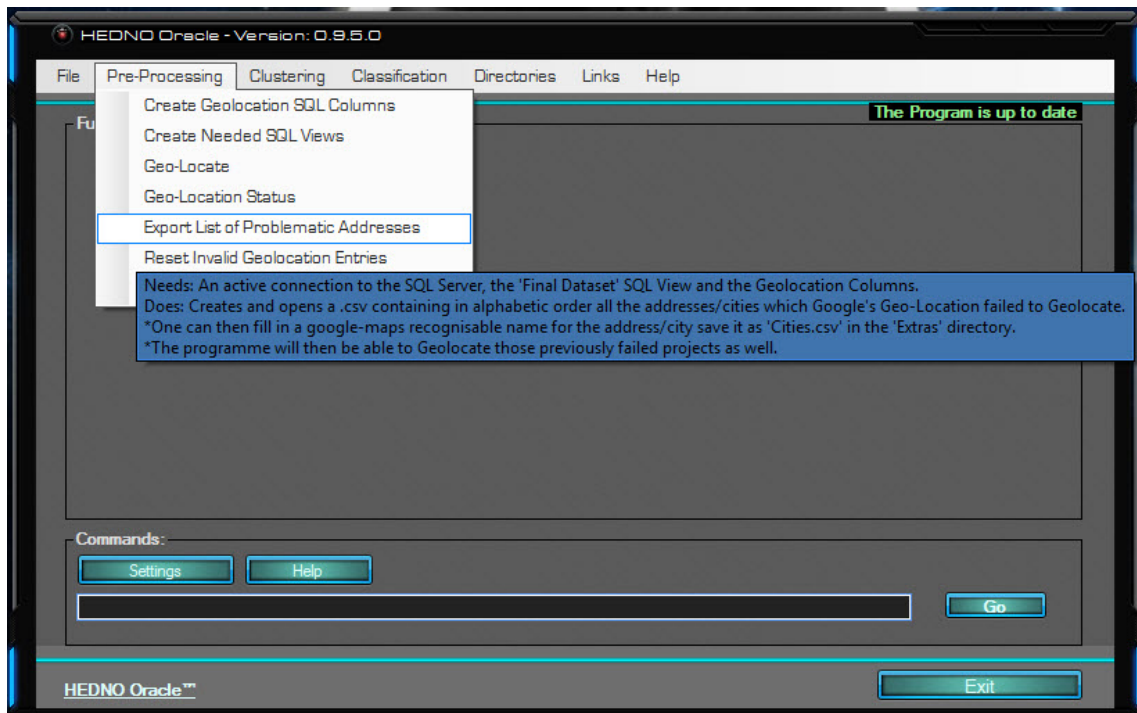
**Figure 19:** Geo-Location Status Pushed

In this instance, out of the 133415 projects that are of use to the cause, 131829 have been successfully Geo-Located, whilst all the rest failed. Should the geolocation process be stopped before it's finished, some of the potentially successful and some of the potentially unsuccessful will be reported on the 4<sup>th</sup> line where it states the number of projects that have yet to undergo Geolocation.

Because each city/village has numerous projects, the number of cities that failed geolocation is fairly lower than the number of the projects themselves. Amongst the next options is a way to overcome this problem and make the API recognise those cities too; which effectively means that all 1586 cities of this instance will be successfully geolocated by filling 328 fields, which is the number of failed cities.

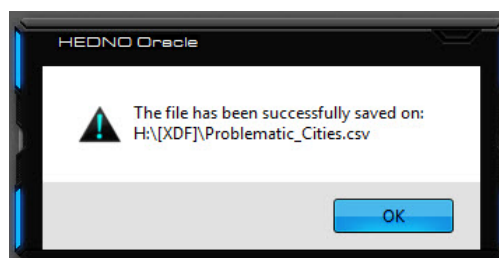
### **7.2.5 Export List of Problematic Addresses:**

Exports a list of all the addresses that cause the geolocation API to fail.



**Figure 20:** Export List of Problematic Addresses Mouse Hover

This option creates a .csv, or comma separated file, containing all the addresses causing a problem, one on each line, and can be used to fill in a google-maps recognisable alternative name in the cell next to the problematic one. Doing so and saving the file as “cities.csv” in the “Extras” folder under the programme’s installation directory enables the programme to geolocate successfully those problematic entries.

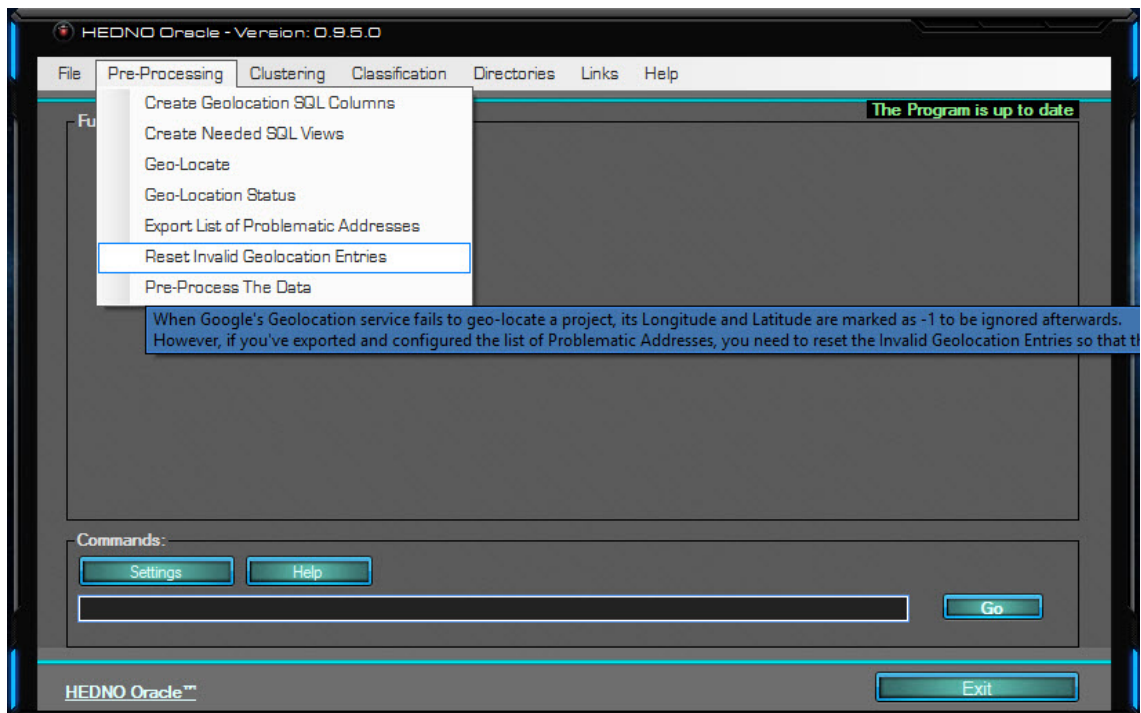


**Figure 21:** Export List of Problematic Addresses Pushed

Once pushed, the algorithm will instantaneously fetch the data from the SQL Server and transform it into a csv format which will be exported on the location configured in the Settings form. Regardless, the directory of the exported file will be reported in the MessageBox.

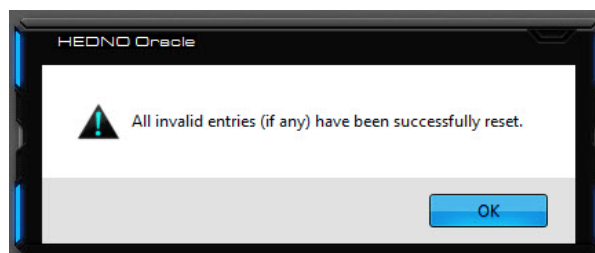
### 7.2.6 Reset Invalid Geolocation Entries:

Resets the values of failed geolocation rows.



**Figure 22:** Reset Invalid Geolocation Entries Mouse Hover

After every failure in the geolocation process, the programme assigns ‘-1’ as the corresponding project’s Longitude/Latitude values, and each time it attempts to geo-locate everything anew it ignores projects with ‘-1’ as a Longitude/Latitude value, hence ignoring previously failed projects which in all likelihood will fail again. Had you filled in valid addresses to the problematic ones, though, you may wish to Reset the invalid geolocation entries, allowing the programme to cease ignoring them.

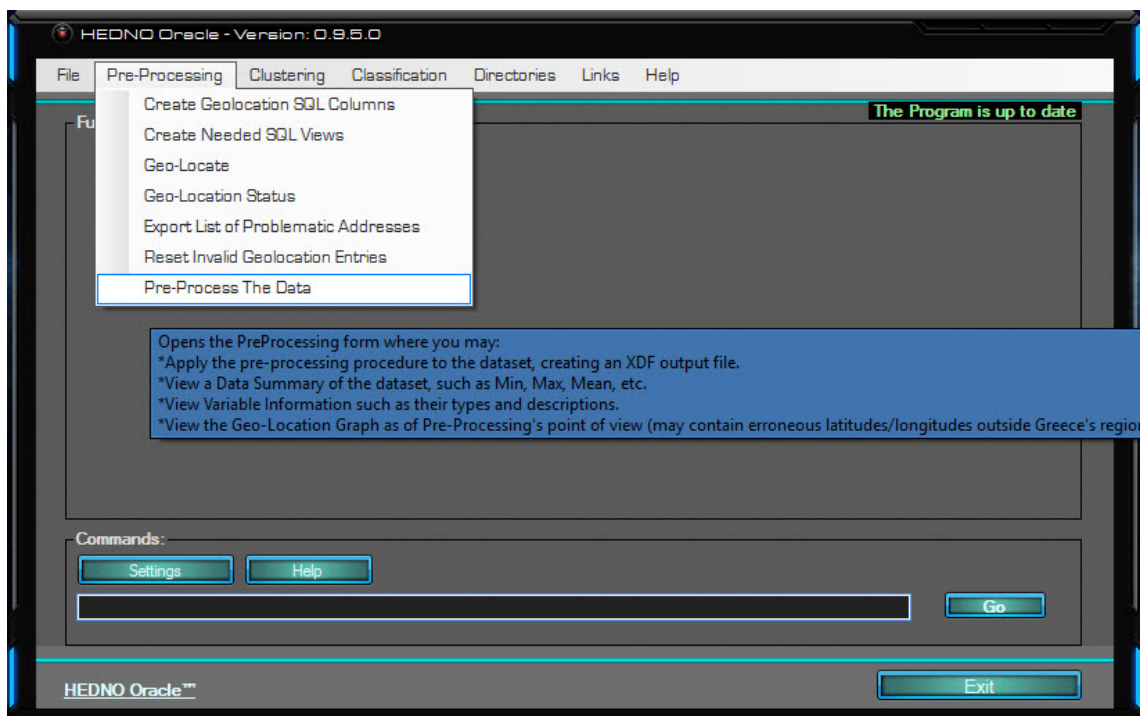


**Figure 23:** Reset Invalid Geolocation Entries Pushed

Once pushed, the programme will display the MessageBox above, informing the user that the action has been successfully completed. As in any other case, had the action failed, it would have displayed a message with insight as to how and why it failed.

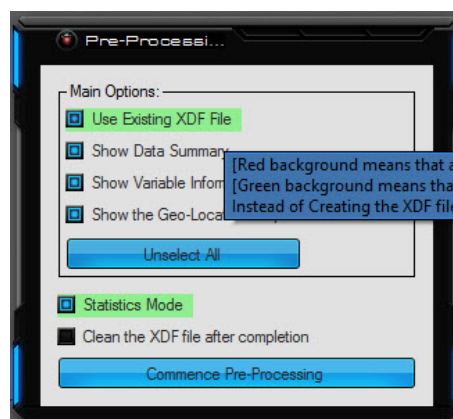
### 7.2.7 Pre-Process the Data:

Opens the 'PreProcessing' form where you may perform actions regarding pre-processing.



**Figure 24:** Pre-Process The Data Mouse Hover

Before clustering occurs, the data needs pre-processing and cleaning. The form below creates an XDF (External Data Frame) file with the data in such a condition, or reads one if it was previously created.



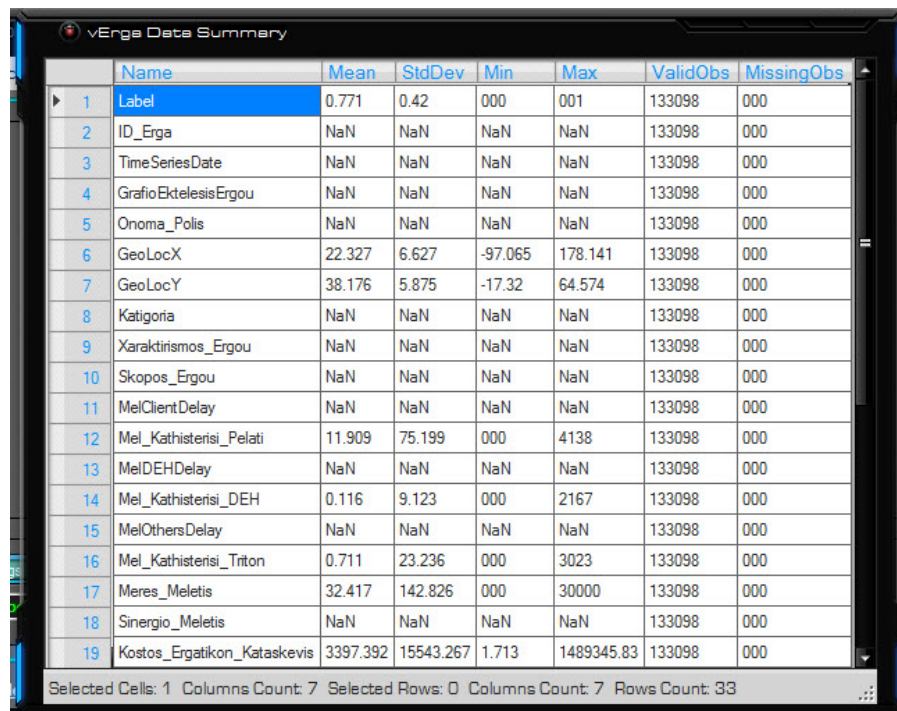
**Figure 25:** Pre-Process The Data Mouse Pushed

The "Use existing XDF file" option will read the dataset from a previously saved XDF file instead of fetching the data from the SQL Server.

If the background colour of the checkbox is green, it means that the XDF file representing the dataset at the pre-processing point is located and accessible in the directory specified in the settings form. Should the background colour be red, it means that the file is not found or is inaccessible and the option should not be checked, but the programme will not go as far as outright-forcing you to not check it.

Checking this red option is completely safe as the underlying R code will check again for the XDF file's status, and if it too finds it unreachable, then the file will be created normally as if the CheckBox had never been checked.

The background colour of the series of CheckBoxes of whether or not each file is reachable in following forms is not conditioned on any other control in the form or elsewhere, and the procedure always remains safe even if the background is red.



	Name	Mean	StdDev	Min	Max	ValidObs	MissingObs
1	Label	0.771	0.42	000	001	133098	000
2	ID_Erga	NaN	NaN	NaN	NaN	133098	000
3	TimeSeriesDate	NaN	NaN	NaN	NaN	133098	000
4	GrafioEktelesisErgou	NaN	NaN	NaN	NaN	133098	000
5	Onoma_Polis	NaN	NaN	NaN	NaN	133098	000
6	GeoLocX	22.327	6.627	-97.065	178.141	133098	000
7	GeoLocY	38.176	5.875	-17.32	64.574	133098	000
8	Katigoria	NaN	NaN	NaN	NaN	133098	000
9	Xaraktrismos_Ergou	NaN	NaN	NaN	NaN	133098	000
10	Skopos_Ergou	NaN	NaN	NaN	NaN	133098	000
11	MelClientDelay	NaN	NaN	NaN	NaN	133098	000
12	Mel_Kathisterisi_Pelati	11.909	75.199	000	4138	133098	000
13	MelDEHDelay	NaN	NaN	NaN	NaN	133098	000
14	Mel_Kathisterisi_DEH	0.116	9.123	000	2167	133098	000
15	MelOthersDelay	NaN	NaN	NaN	NaN	133098	000
16	Mel_Kathisterisi_Triton	0.711	23.236	000	3023	133098	000
17	Meres_Meletis	32.417	142.826	000	30000	133098	000
18	Sinergio_Meletis	NaN	NaN	NaN	NaN	133098	000
19	Kostos_Ergatikon_Kataskevis	3397.392	15543.267	1.713	1489345.83	133098	000

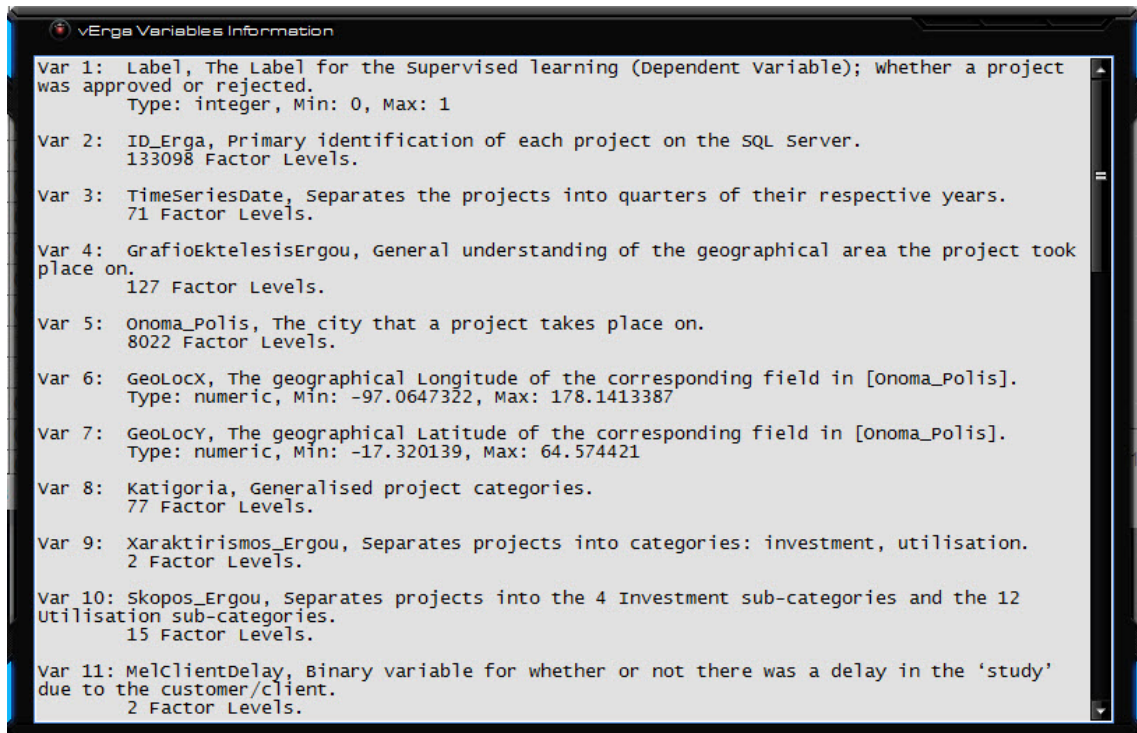
Selected Cells: 1 Columns Count: 7 Selected Rows: 0 Columns Count: 7 Rows Count: 33

**Figure 26:** Data Summary Form

Checking the “Show Data Summary” checkbox will display a summary of the data to the end user. The summary comprises of a variable's Mean value, Standard Deviation, Minimum Value, Maximum Value, Valid Observations, and Missing Values.

The decimal point at which numbers are rounded at is subject to the corresponding option set in the Settings form (defaults to 3). Zero values ending in an asterisk (i.e. 000\*) mean that the number was not originally zero, but became such as a direct result or

rounding. ‘NaN’ is code for “not a number”, which tends to means infinity, but in this case points to factor variables for which the notions of Mean, Standard Deviation, Minimum Value and Maximum Value do not apply.



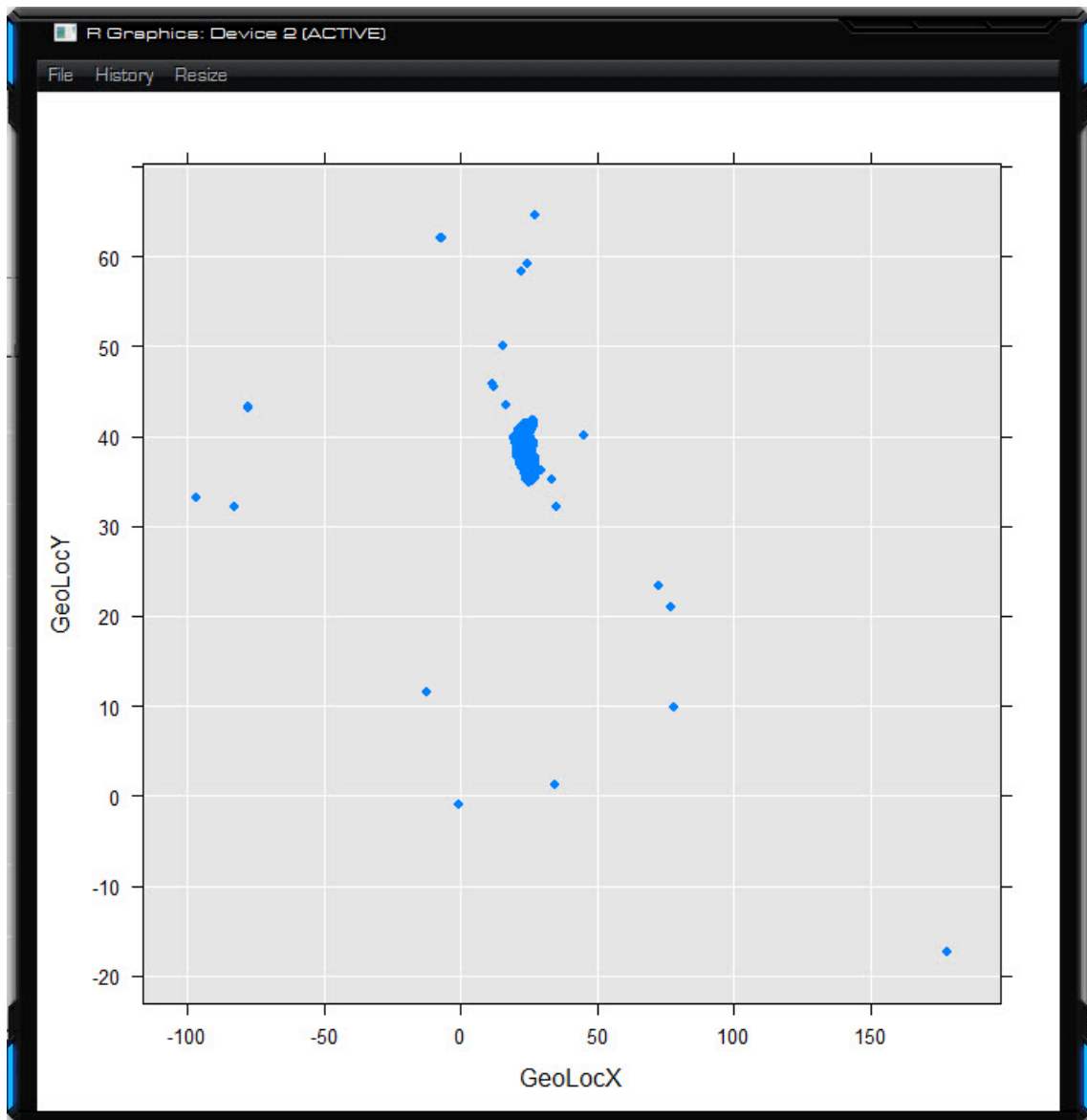
**Figure 27:** Variables Information Form

Checking the ‘Show Variables Information’ CheckBox yields this form at the end of the process. Here the user can see the names of the variables they’ll be working with, as well as a description of each one’s use. Numeric variables will report a type such as “integer”, a minimum value and a maximum value, whereas Factor variables report their factor levels, i.e. the number of different factors.

Checking the “Show the Geo-Location Graph” will display a plot of all projects’ coordinates using a graphical engine optimised for big data, meaning that the sheer number of entries on the plot does not render it disfigured and unreadable.

Although the data at this point are processed, as the geolocation runs, some projects might seemingly be successfully geolocated when in reality they’ve been miscategorised.





**Figure 28:** Pre-Processing Geolocation Visualisation

Those projects will show up as dots scattered around the canvas away from the group of valid projects. Knowing the database contains locations strictly under Greece's region, those points can safely be erased; something that is done in the very next step (Clustering Step 0).

The "Statistics Mode" means that the Dataset from which the Training and Testing sets are going to be formed is going to come from projects that their having been approved or cancelled has already been decided (data with a Label). Having labels allows for statistical analysis, meaning that information regarding how well the machine learning algorithms perform on the current dataset can be extrapolated by comparing the results given by the algorithms to the Label (what actually happened). The alternative is to use

all the clearly labelled data to train a model which in turn will be used to predict what will happen to the projects that are currently pending.

When creating a new XDF we can opt to either create it using the former method or the latter and either one is valid, hence the background colour of ‘Statistics Mode’ on such a case is transparent (it does not possess a background colour). If, however, the ‘Use Existing XDF File’ is checked, then the dataset represented by the file would have either been created using Statistics Mode, or not; in which 4 distinct possibilities exist:

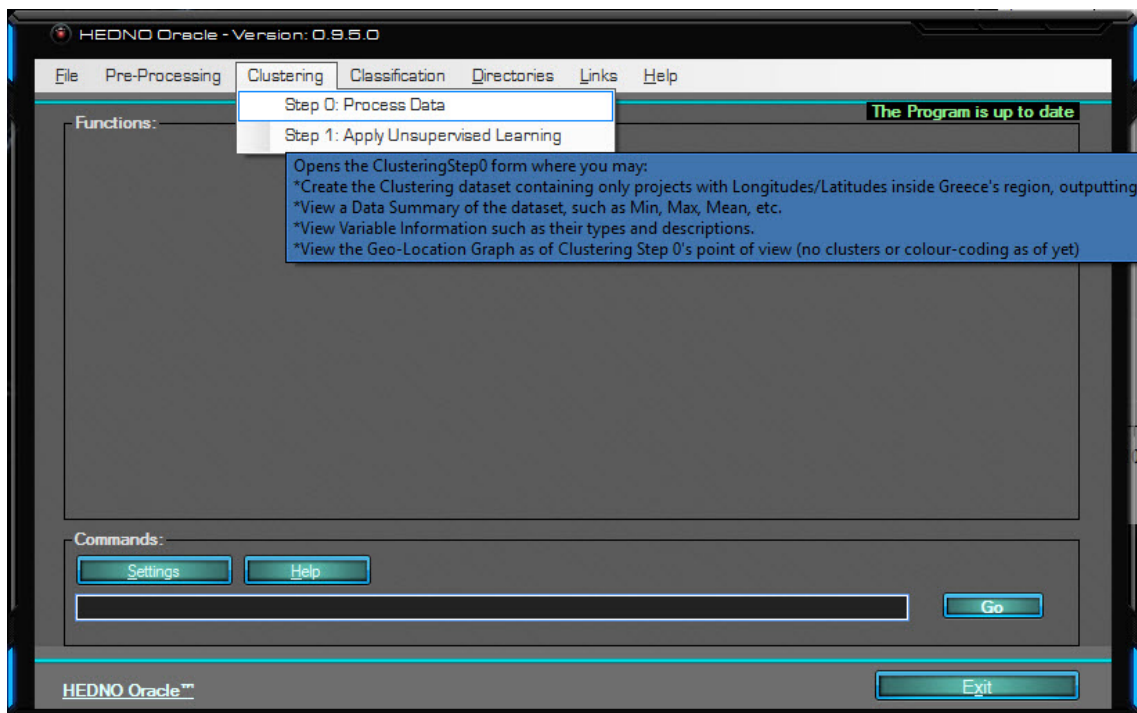
- If the ‘Statistics Mode’ *is* checked and the Existing XDF file *had been* created with Statistics Mode, the background will be Green for the file matches the settings.
- If the ‘Statistics Mode’ *is* checked but the Existing XDF files *had been* created with Statistics Mode Off, the background will be Red, signalling that using this dataset to extrapolate statistics will produce nonsensical data or fail to run altogether.
- If the ‘Statistics Mode’ *is not* checked but the Existing XDF file *had been* created with Statistics Mode, the background will be Red, signalling that Statistics are rendered unavailable even though it seems like they should be on.
- If the ‘Statistics Mode’ *is not* checked and the Existing XDF files *had been* created with Statistics Mode Off, the background will Green for the file matches the settings.

## 7.3 Clustering

Projects in close spatial proximity (in geological terms) may share more commonality than those farther away. Clustering the projects in geologically driven groups helps the algorithms tap into this information.

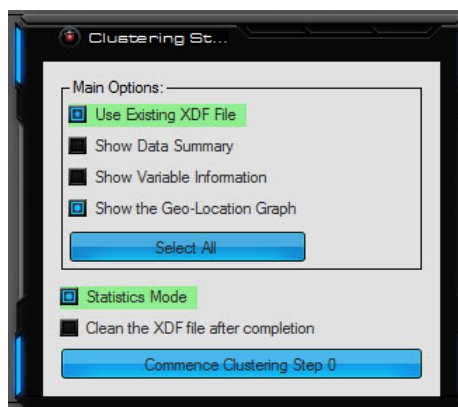


### 7.3.1 Step 0: Process Data



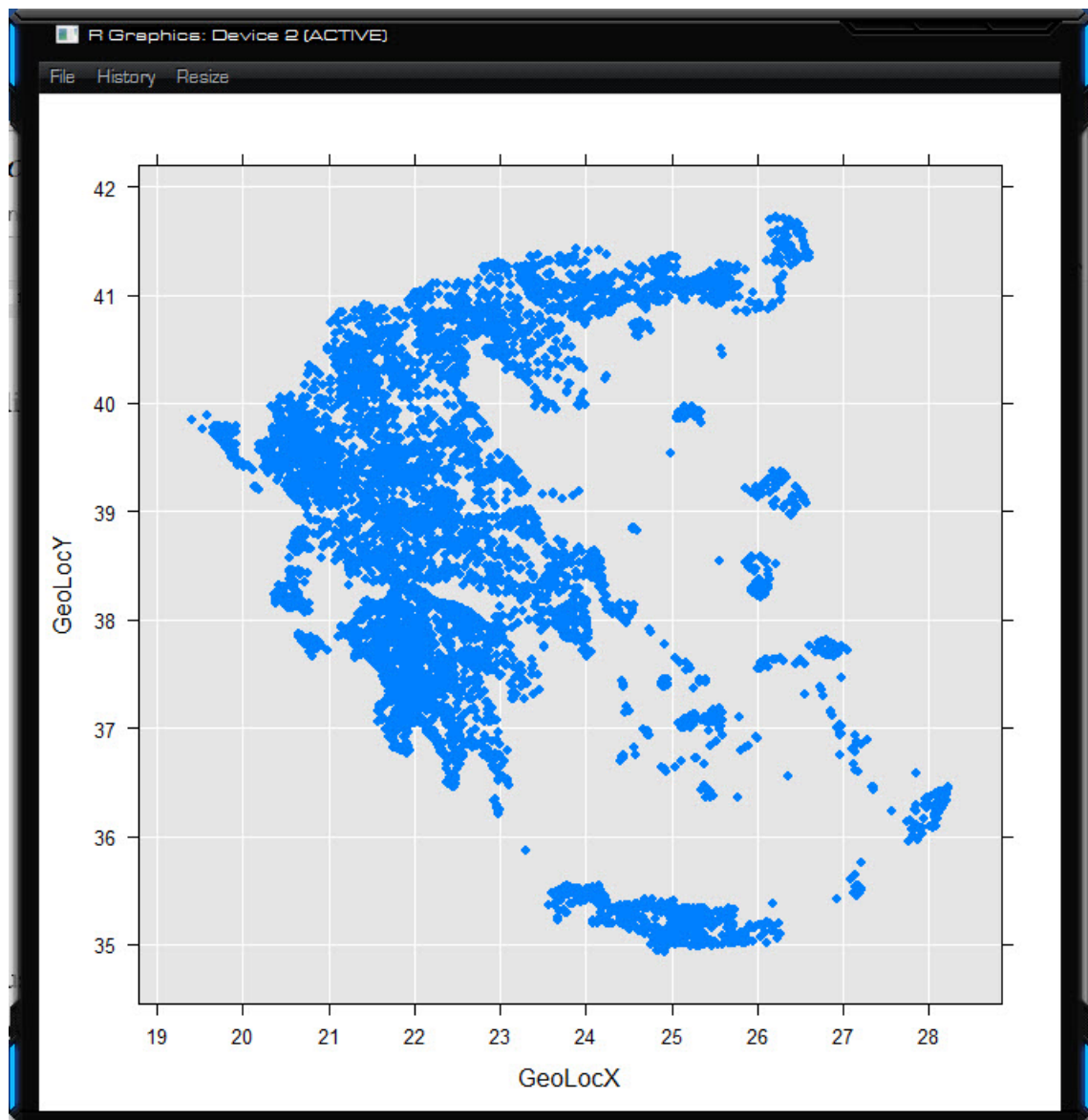
**Figure 29:** Step 0: Process Data Mouse Hover

This Clustering Step 0 form does the final process on the data and eliminates any projects whose geological location is outside Greece's rectangle.



**Figure 30:** Step 0: Process Data Mouse Pushed

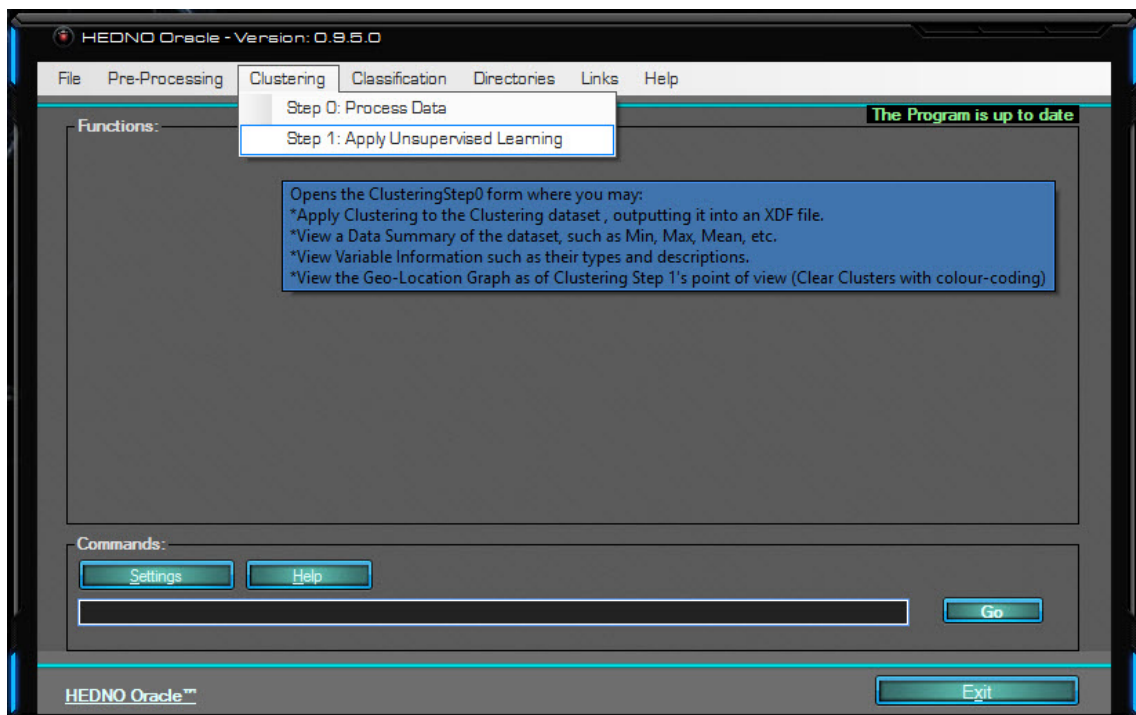
The CheckBox 'Clean the XDF file after completion' will delete the Clustering XDF file (or each form's corresponding XDF file). After all the computations are over with, statistics are drawn and graphs are plotted.



**Figure 31:** Step 0: Process Data Geolocation Graph

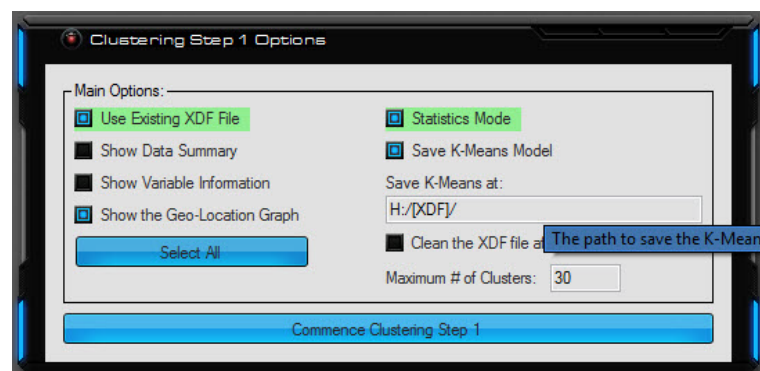
Since addresses were kept confidential for legal reasons, what each dot actually represents is a collection of projects on the same city, town or village. Even so, having kept only valid locations, the plot now clearly illustrates Greece's geology. When HEDNO itself uses the programme, with but a small adjustment in the SQL view, the full address can be used to increase the accuracy of the Geolocation Process.

### 7.3.2 Step 1: Apply Unsupervised Learning



**Figure 32:** Step 1: Apply Unsupervised Learning Mouse Hover

This step opens the Clustering form which performs scalable k-means clustering using the classical Lloyd algorithm to separate the data into  $k$  clusters.

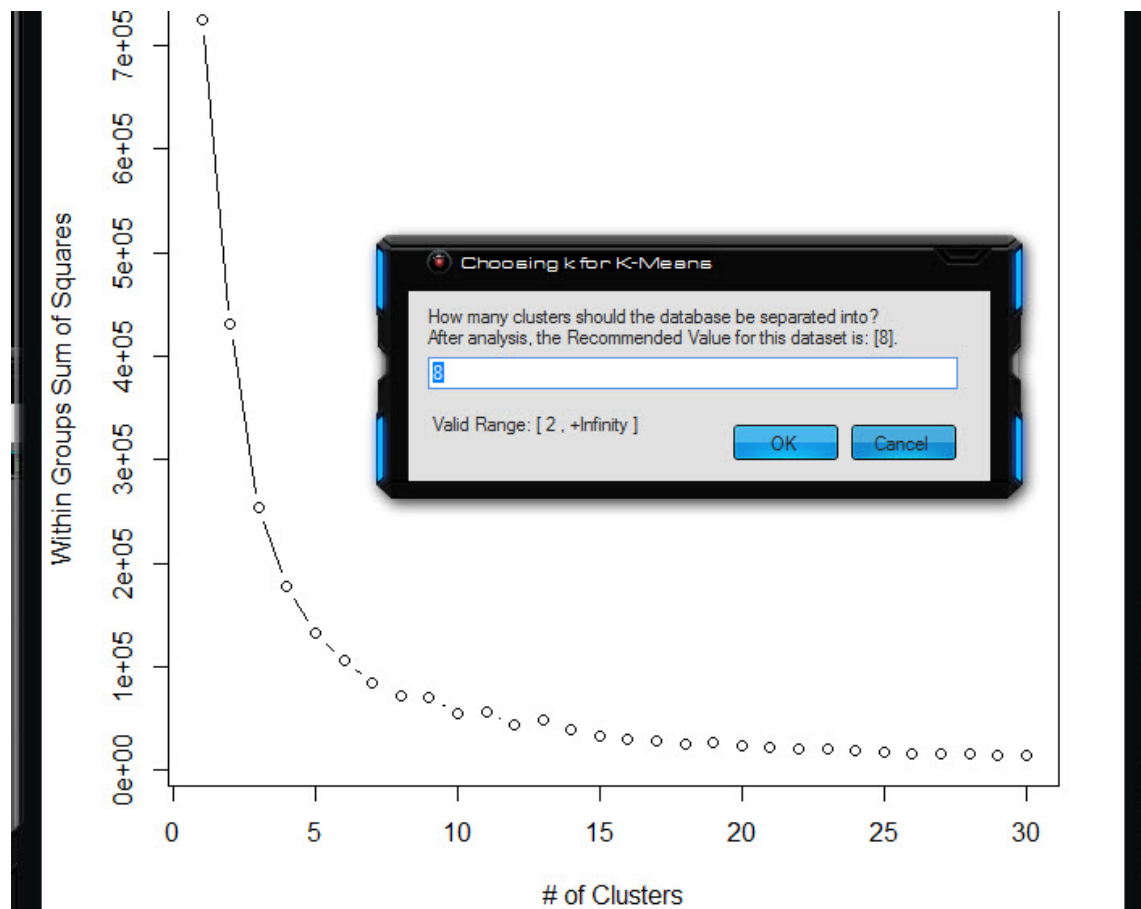


**Figure 33:** Step 1: Process Data Mouse Pushed

The k-means model created by this step can be saved using R's RDS format and it can be imported in an R environment with all its information intact. The path which will be saved defaults to the XDF path, but can be overridden by the "Save K-Means at:" TextBox.

During the process, as mentioned on the 'Applying Machine Learning' chapter, the algorithm attempts to extrapolate the optimal value for  $k$ ; to do so it tests for the best WG SSE for a number of clusters between 1 and 'Maximum # of Clusters', meaning that

setting the value to 30, the algorithm will attempt to find the optimal value between 1 and 30.

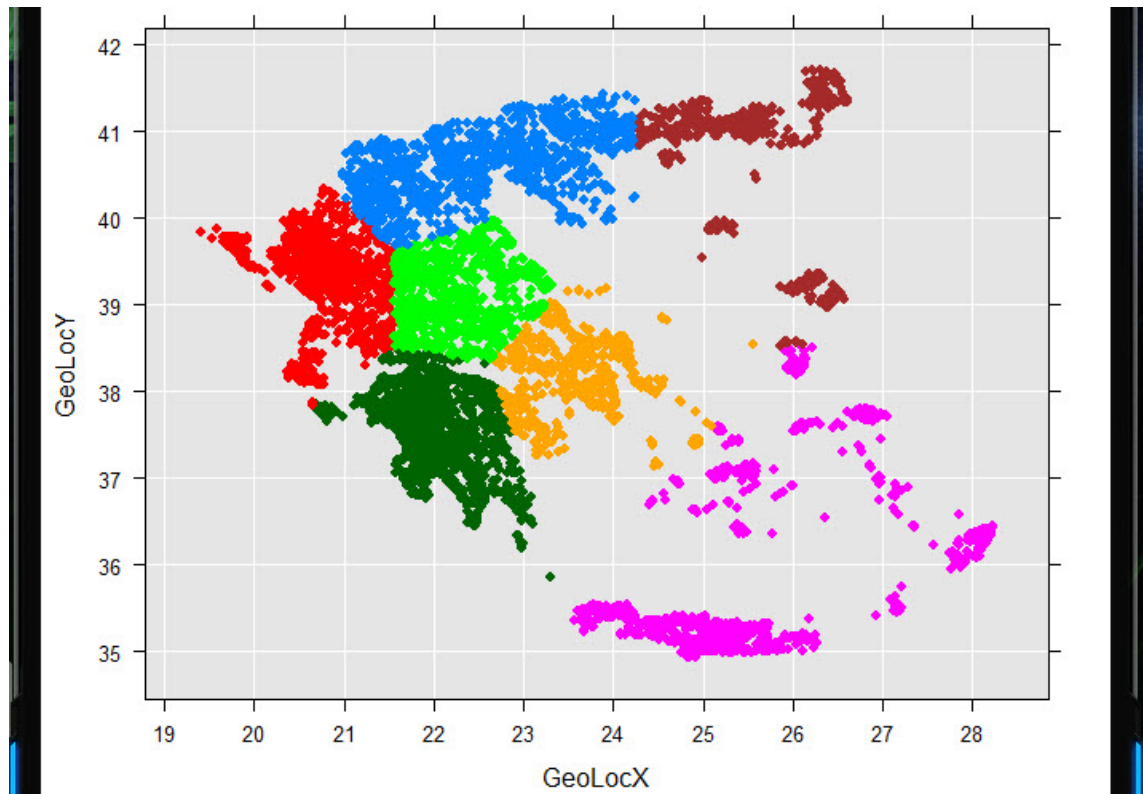


**Figure 34:** Step 1: Optimal k Value Selection

When the number is computed via comparisons on the Within Groups Sums of Squares vs the number of clusters, the programme will display an InputBox with the recommended value pre-loaded and the corresponding Scree-Plot so that the user may opt to choose an alternative number.

In most cases the default value is the best choice, so just pushing “OK” is highly recommended.

As clusters and cluster centres change with newly introduced projects (or even by re-running the process, since this is a non-deterministic algorithm), to permanently assign the cluster number to a project is ill-advised; unless each new project’s cluster is assigned via classification.

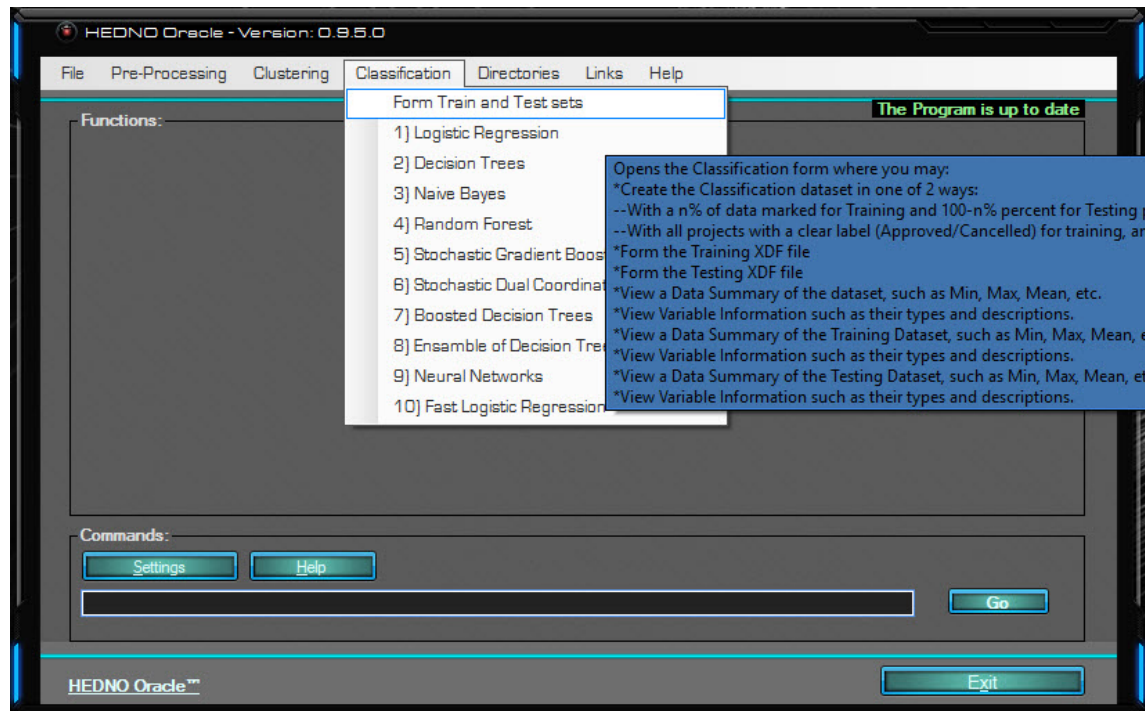


**Figure 35:** Step 1: Optimal k Value Selection

The end result for 8 clusters looks like this, and the information of which cluster each project belongs to is saved locally on the XDF file as a new column named ‘rxCluster’. In fact, with the exception of Geo-Location columns (GeoLocX and GeoLocY), everything else is always written locally, minimising interactions with the SQL Server and any possibility of introduced vulnerability.

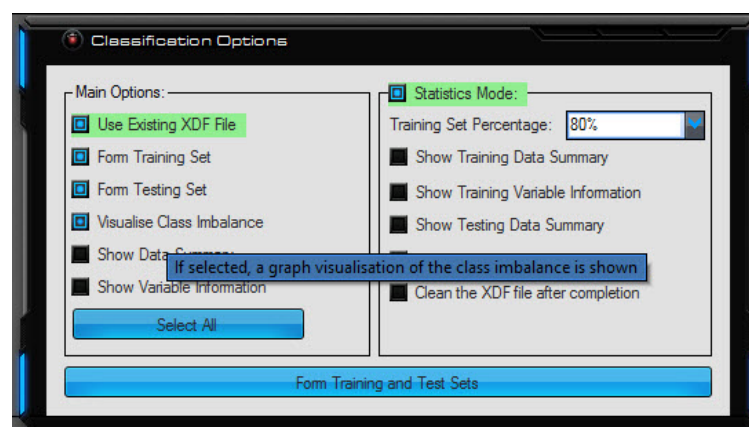
## 7.4 Classification

### 7.4.1 Form Train and Test sets



**Figure 36:** Form Train and Test Sets Mouse Hover

The clustering process demanded that all data, training and testing alike, be in the same dataset. Be that as it may, the classification process requires that the Training and Test datasets be separate. To this end, the ‘Form Train and Test sets’ menu-item opens a form which offers options to perform this very thing.



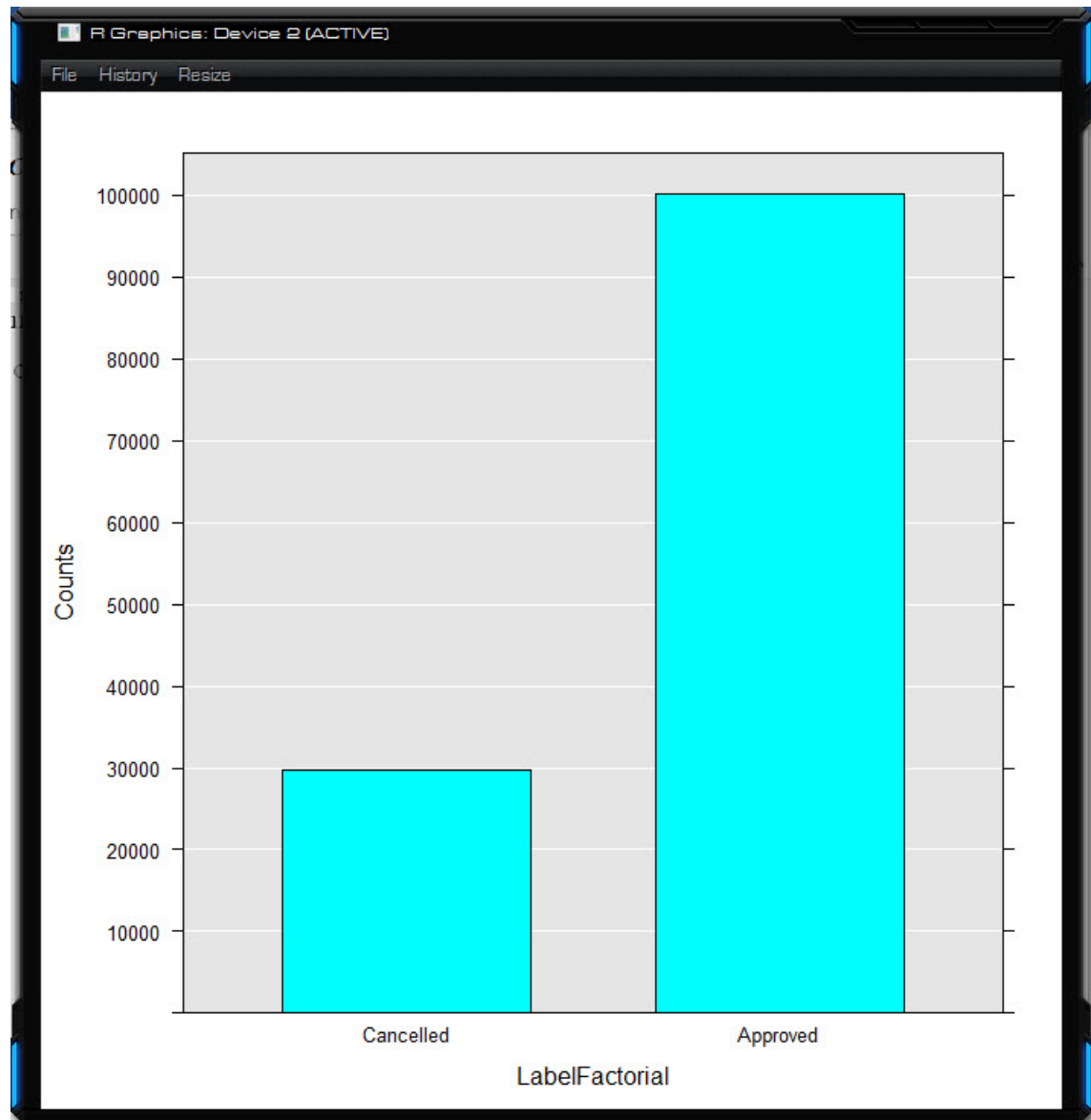
**Figure 37:** Form Train and Test Sets Pushed

Checking the “Use Existing XDF File” will use a previously generated Classification File (not available the first time it runs, or if “Clean the XDF file after completion” was

previously checked). Forming the Training and Testing sets is performed by checking their respective CheckBoxes, but one can use this form to just visualise the Class imbalance or to get information on the Data or Variables.

Should the sets be created, their Data & Variable information can also be viewed by checking their corresponding CheckBoxes in the left panel.

Checking the “Visualise Class Imbalance” CheckBox will produce the following plot:



**Figure 38:** Class Imbalance Plot

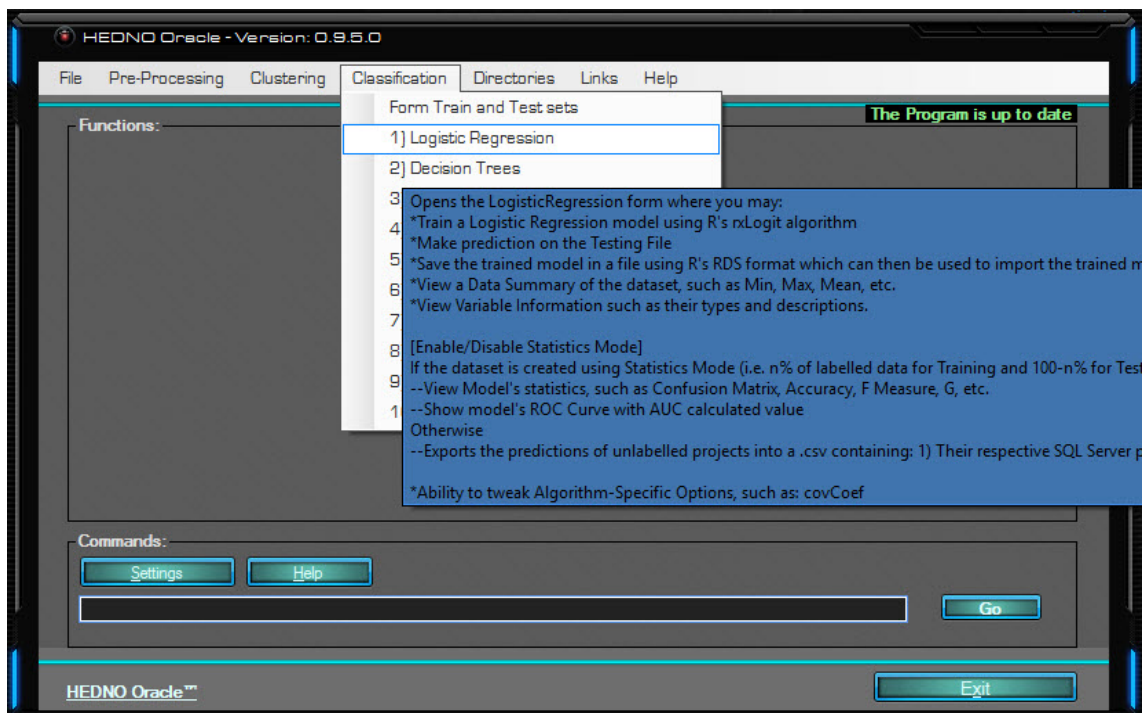
In the current dataset a substantial class imbalance is witnessed. A project's probability of being approved is more than 3 times higher than that of it being cancelled. This is an important factor that should be taken into account when viewing statistics such as Accuracy because it means that if a classifier which classifies everything as Approved



is built, then it will have an accuracy of about 70% even though it actually does nothing more than always prediction the positive class (approved). Nonetheless, other measures are impartial to class imbalance, like the AUC of the ROC Curve, which is amongst the main statistics considered when reviewing a classifier.

The “Training Set Percentage” ComboBox allows the user to select what percentage of the Dataset they want to be used for training whilst the rest will be used for prediction. This option only applies when Statistics Mode is one, as in non-statistics mode, the purpose is the most accurate prediction of the projects that are still pending, using the previous, labelled projects as input.

## 7.4.2 Logistic Regression

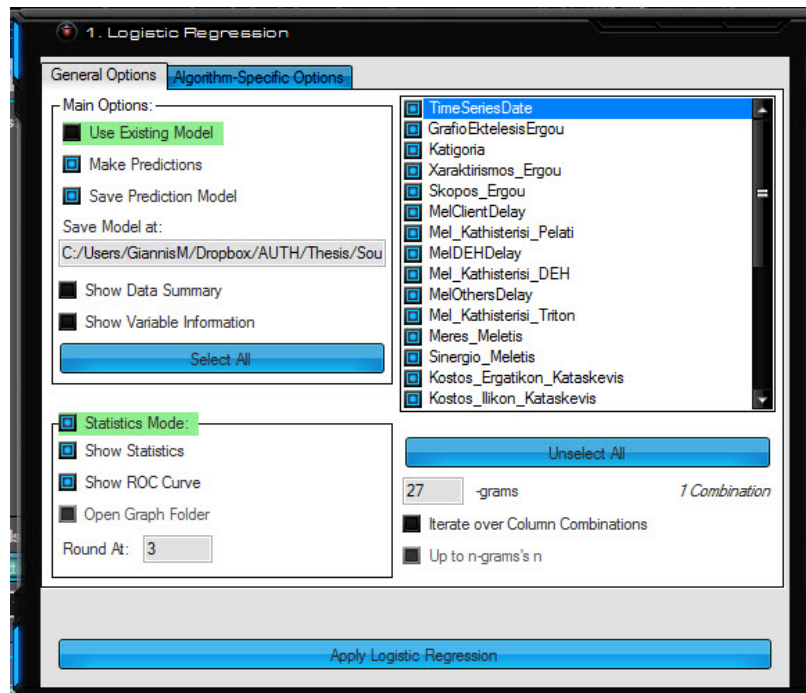


**Figure 39:** Logistic Regression Mouse Hover

This option opens the Logistic Regression form which can be used to build a logistic regression classifier, using R's rxLogit algorithm.

Each following Classification form will have, for the sake of uniformity, the same layout as this one, with the first of two tabs having a collection of options useable by any classifier, thus remaining the same amongst different classifiers presentation-wise, and the second comprising of classifier-specific options, meaning each following classifier will have their one unique one.





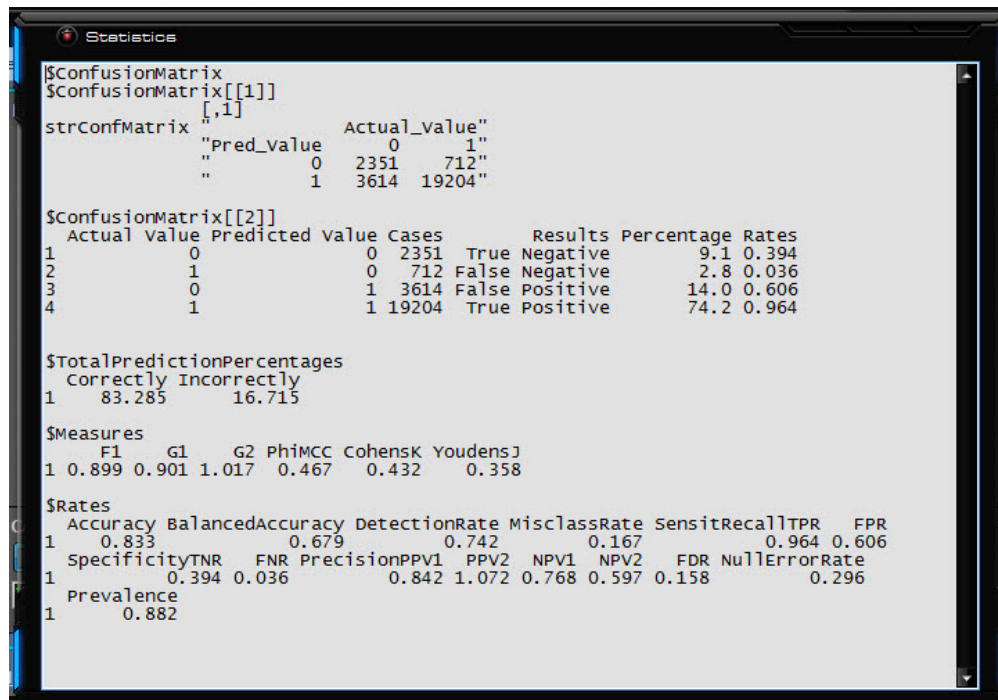
**Figure 40:** Logistic Regression Pushed

By checking ‘Use Existing Model’, instead of training the model, a previously saved model is loaded. Using an existing model is faster, but beware that should the data change significantly (in particular, if the Factor Level Order changes), then the whole procedure will fail. In that case, re-run it, unchecking the CheckBox to create the model anew.

Checking the ‘Make Predictions’ CheckBox uses the trained model to make prediction upon the Testing Dataset, and if Statistics Mode is off, then also outputting them in a .csv file.

Checking the ‘Save Prediction Model’ Checkbox, the programme saves the trained model to the location specified below it.

Checking the ‘Show Statistics’ CheckBox, provided that Statistics Mode is on and the XDF Files were created with the option on as well (as indicated by the Green background colour when the CheckBox is checked), produces the following statistics, measures and insight on the currently trained classification model:



**Figure 41:** Model Statistics

This form consists of the Confusion Matrix in its usual setting (\$ConfusionMatrix[[1]]), where we view:

- The True Negatives, the number of projects that were actually Cancelled and the classifier predicted as Cancelled (2351 in this instance),
- The False Negatives, the number of projects that were actually Approved but the classifier predicted as Cancelled (712 in this instance),
- The False Positives, the number of projects that were actually Cancelled but the classifier predicted as Approved (3614 in this instance),
- The True Positives, the number of projects that were actually Approved and the classifier predicted as Approved (19204 in this instance),

The confusion matrix in a more detailed binary-truth-table inspired setting (\$ConfusionMatrix[[2]]), which in addition to the above, it includes percentages and rates as well; as far as the rates are concerned, those of the True Negatives and False positives add up to 1 (all the negatives), and in the same logic, those of the False Negatives and True positives add up to 1 as well (all the positives),

The Correctly Predicted Percentages and Incorrectly Predicted Percentages (\$TotalPredictionPercentages),

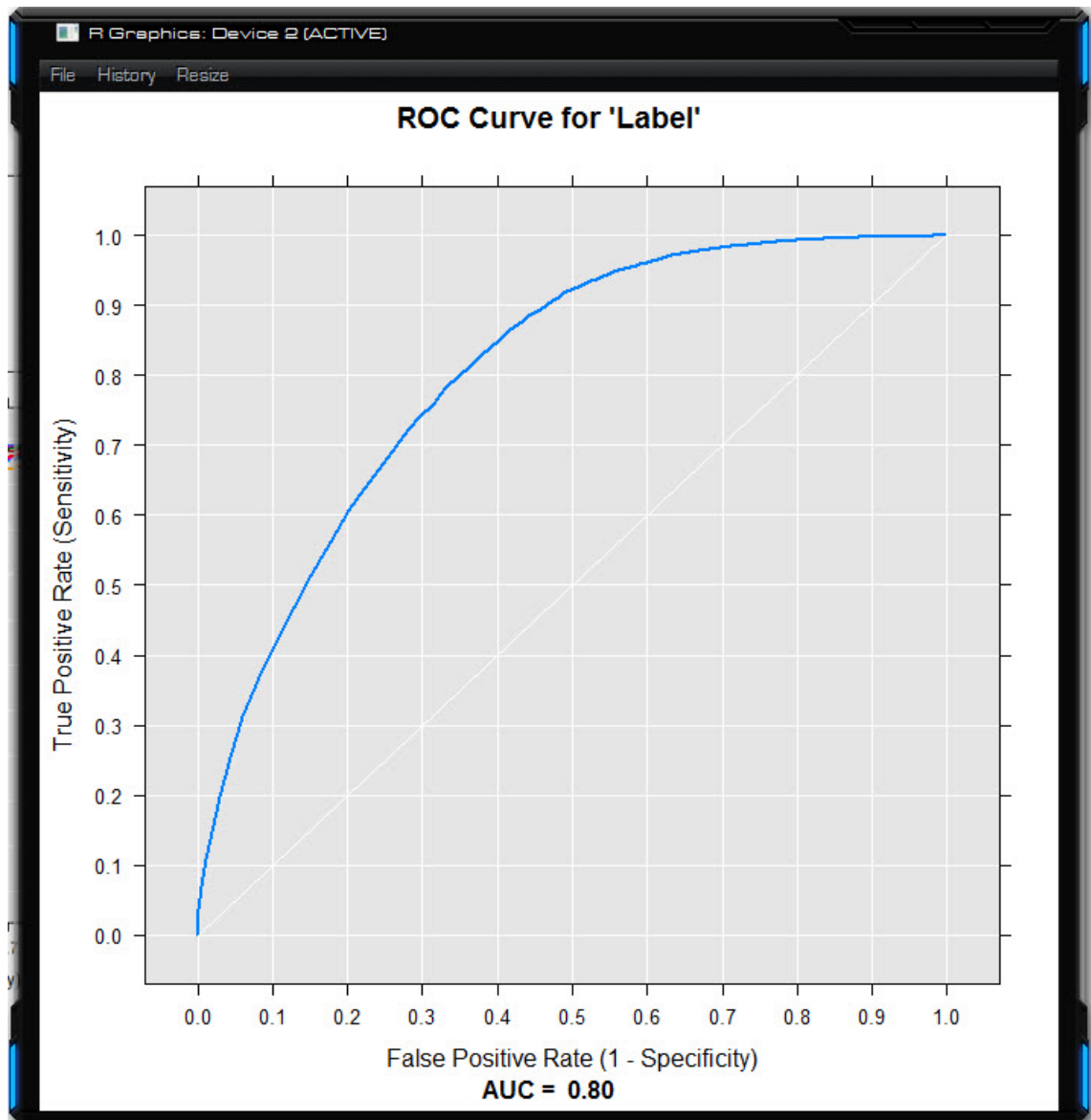
Statistical measures (\$Measures), which include:

- F1 Score, the weighted average of the true positive rate (recall) and precision; put another way, their harmonic mean,

- G-measure, an indication of the central tendency or typical value of the predictions,
- PhiMCC ( $\phi$ , Matthews correlation coefficient), a correlation coefficient between the observed and predicted values,
- CohensK ( $\kappa$ , Cohen's kappa coefficient), a measure of how well the classifier performed as compared to how well it would have performed simply by chance,
- YoudensJ (J, Youden's J statistic), an estimation of the probability of an informed decision,

Statistical Rates (\$Rates), which include:

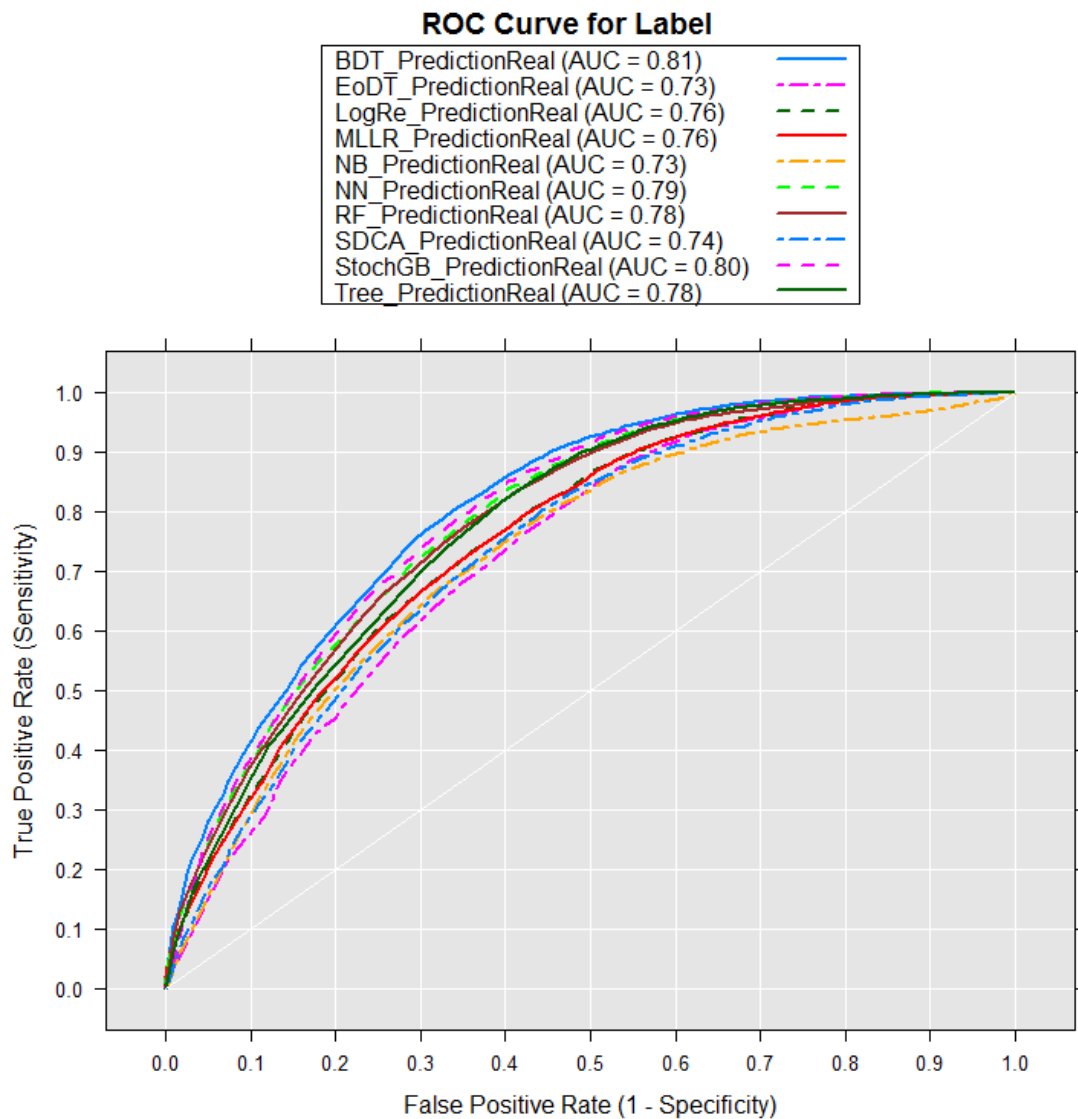
- Accuracy, which answers the question: overall, how often is the classifier correct?
- BalancedAccuracy, an alternative accuracy measure that does not lead to an optimistic estimate when a biased classifier is tested on an imbalanced dataset,
- DetectionRate, basically at which rate does the classifier identify the True Positive cases,
- MisclassRate (Misclassification Rate), which answers the question: overall, how often is it wrong?,
- SensitRecallTPR (Sensitivity or Recall or True Positive Rate), answering the question: when the project's actually approved, how often does the classifier predict approved?,
- FPR (False Positive Rate), answering the question: when the project's actually cancelled, how often does the classifier predict approved?,
- SpecificityTNR (Specificity or True Negative Rate), answering the question: when the project's actually cancelled, how often does the classifier predict cancelled?,
- FNR (False Negative Rate), answering the question: when the project's actually approved, how often does the classifier predict cancelled?,
- PrecisionPPV1 (Precision or Positive Predictive Value), answering the question: when the classifier predict approved, how often is it correct?,
- PPV2 (Positive Predictive Value), is similar to PrecisionPPV1, except that it takes prevalence into account,
- NPV1 (Negative Predictive Value), answering the question: when the classifier predicts cancelled, how often is it correct?
- NPV2 (Negative Predictive Value), is similar to NPV1, except that it takes prevalence into account,
- FDR (False Discovery Rate), is a way to measure the rate at which the classifier does Type I Error,
- NullErrorRate (Null Error Rate), answering the question: how often would the classifier be wrong if it only predicted the majority class; in this case, approved,
- Prevalence, shows how often projects are approved in the dataset.



**Figure 42:** Single-Model ROC Curve

If the 'Show ROC Curve' CheckBox is checked, a Receiver Operating Characteristic, that is, a visualisation of the classifier's performance summarising its performance over all possible thresholds, is shown.

'Open Graph Folder' has to do with the way R.NET behaves. Because multiple graphical devices are not supported, when the algorithm runs once, a form similar to the one above opens; but when there are multiple combinations running sequentially, then all single-model ROC Curves are saved on the local disk, and the juxtaposition of the curves opens in the form, as shown below.



**Figure 43:** ROC Curves of Multiple Models

The ‘Round At:’ TextBox defaults to the ‘Round At’ value of the settings form, but can be overridden in the form if one should want that single procedure to continue with statistical results rounded at a different decimal point.

On the right side of the form are the Columns/Variables which can be used to train the classifier. The ListBox is automatically populated with valid values as the programme reads the available variables from the Training dataset, which in turn are used to validate the testing dataset (both should have the same number of variables, or at least the Testing Dataset should contain all of the Training Dataset’s columns, and they should be coming from the same source, potentially containing a different number of rows).

The ‘n-grams’ TextBox takes values between 1 and the current count of variables, so that if there are 27 variables checked,  $n$  must be between 1 and 27. If “Iterate over Column

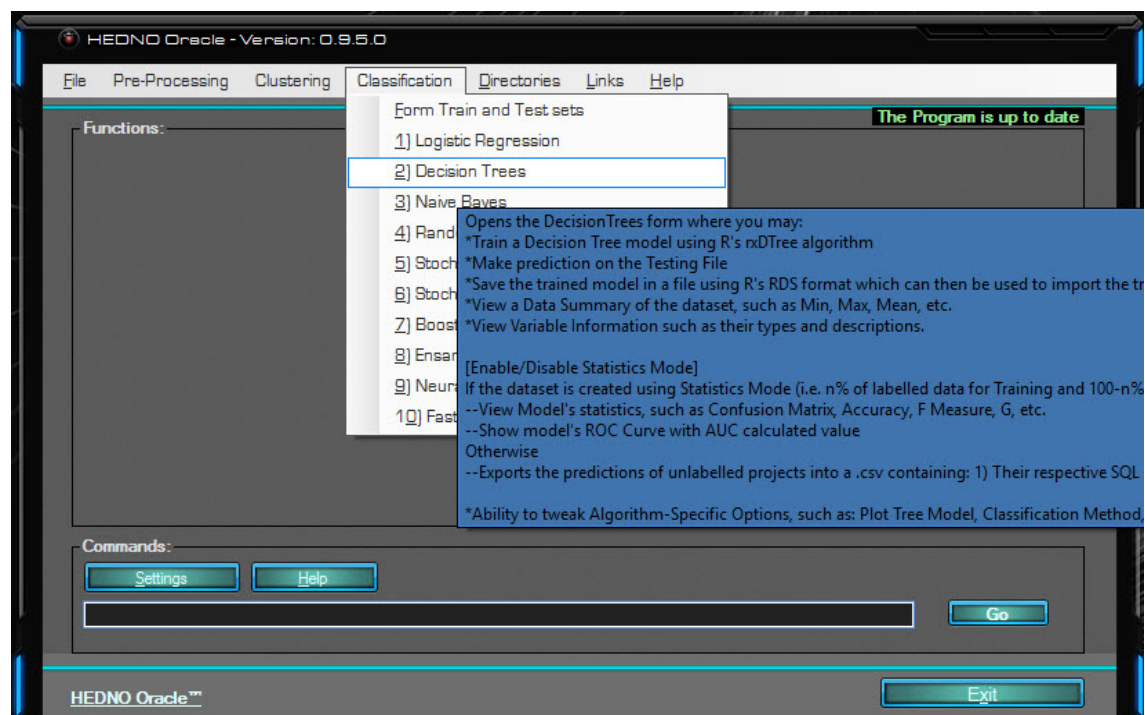
Combinations” and “Up to n-grams’s n” are unchecked, then the n is always the Checked Columns Count as there is only 1 combination – that with everything on.

If “Iterate over Column Combinations” is Checked, then the number of variables combinations depends on the  $n$  of n-grams, given by the Binomial coefficient formula  $c = \binom{v}{n} = \frac{v!}{n! \cdot (v-n)!}$ , where  $c$  is the number of combinations,  $v$  is the number of variables and  $n$  is the n-gram’s  $n$ . If, for instance, variables '1', '2', and '3' are selected, then

- a 1-gram will produce 3 results (1, 2, 3)
- a 2-gram will produce 3 results (1-2, 1-3, 2-3)
- whilst a 3-gram will produce 1 result (1-2-3)

Should “Up to n-grams's n” is checked as well, then the number of combinations changes per the formula  $c = \sum_{i=1}^n \binom{v}{n_i}$ , meaning that in the same case of 3 variables, an ‘n’ of 3 will yield these 7 results: (1, 2, 3, 1-2, 1-3, 2-3, 1-2-3)

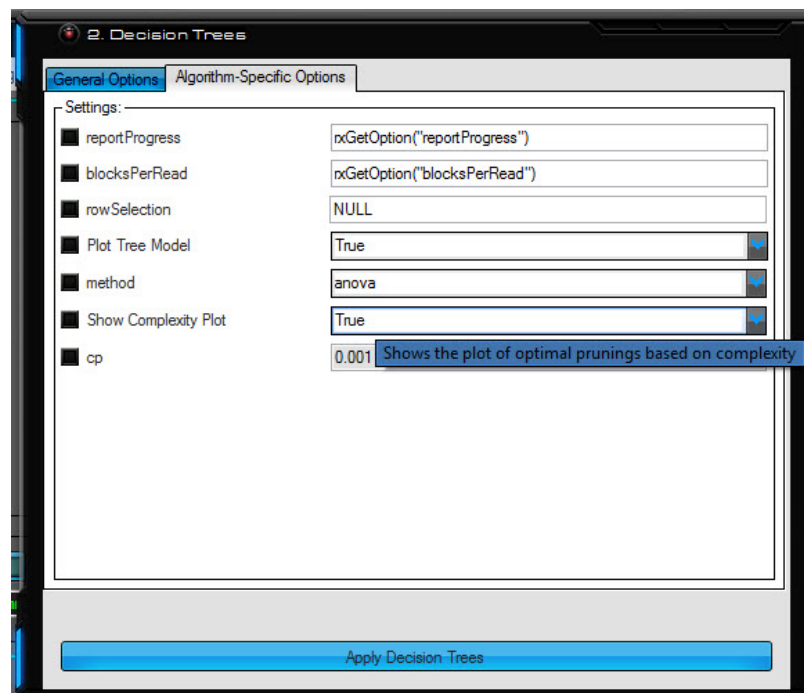
### 7.4.3 Decision Trees



**Figure 44:** Decision Trees Mouse Hover

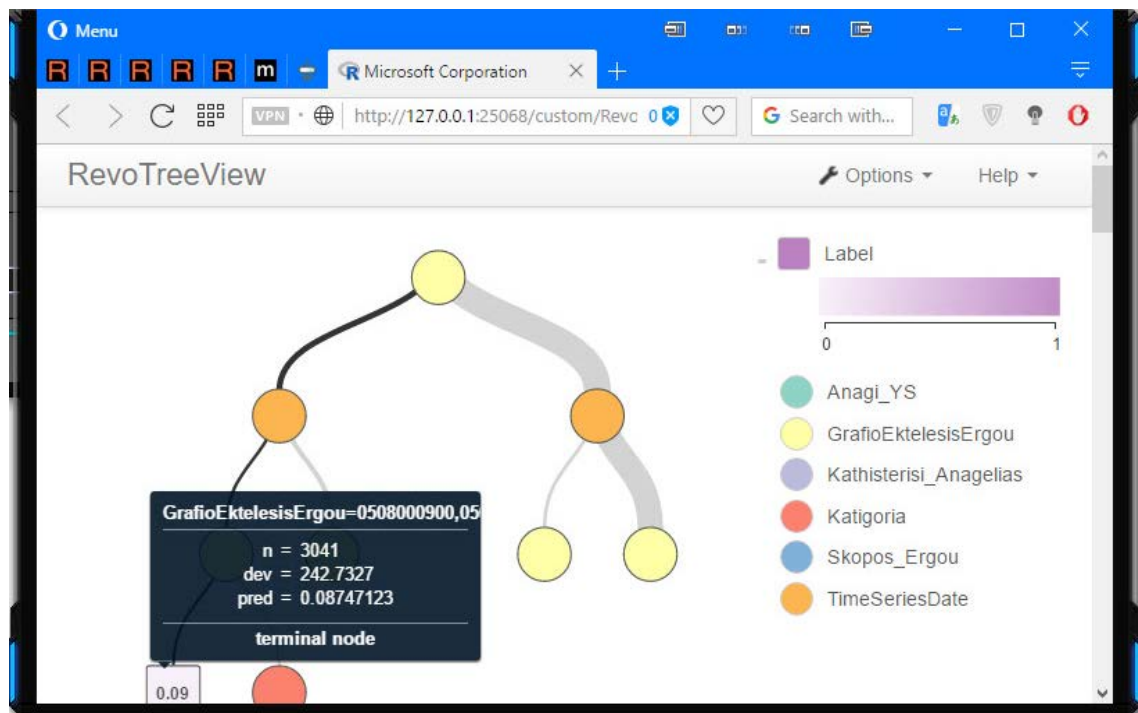
This menu item opens the Decision Trees form which can be used to build a Decision Trees classifier, using R’s rxDTree algorithm.

The first 3 items on the “Algorithm-Specific Options” are universal amongst the Classification forms, ergo they will be explained once here and not mentioned moving forwards.



**Figure 45:** Decision Trees Mouse Pushed

- **reportProgress:** governs how R reports back to the user and can vary from no progress reporting to full rows processed and timings reported. The default behaviour is to retrieve back the reporting value from the R Server, which is the recommended value.
- **blocksPerRead:** the number of blocks to read for each chunk of data read from the XDF file. The default behaviour is to retrieve back the value from the R Server, which is the recommended value.
- **rowSelection:** a formula that limits the number or rows retrieved from the Training set by the classifier to perform a targeted classification. The default value is a NULL formula so that all of the training set is used.
- **Plot Tree Model:** if True, the Default Web Browser will open, revealing an interactive Tree Structure for the trained model. <sup>[13]</sup>

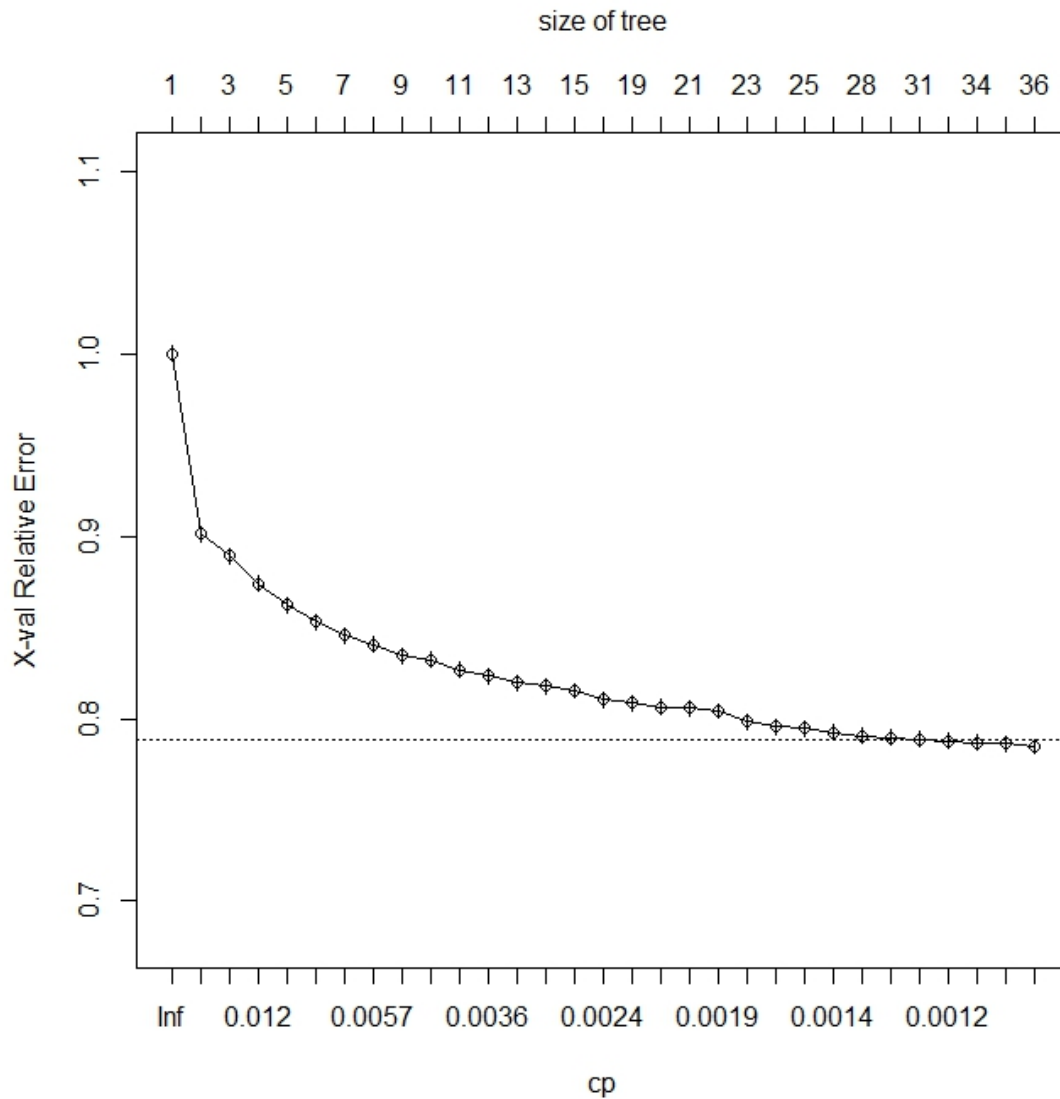


**Figure 46:** Decision Tree Model Plot

This visualisation begins with just the top node (circle), named root, and with each click on a node (or shortcut used), the split options for it appear. Squares denote a terminal node which doesn't split any further. The colour of a split node represents the variable split to create its child nodes. Hovering with the mouse on a node reveals the node name, the next split, and for classification trees, the n, loss, predicted values and a bar chart, whilst for regression trees such as the one in the picture, the n, deviance and predicted values. Lines have different thicknesses for the thickness of the line connecting two nodes is representative of the number of observations going into the child node.

- **method:** the type of tree to be built; "anova" for regression trees or "class" for classification ones.

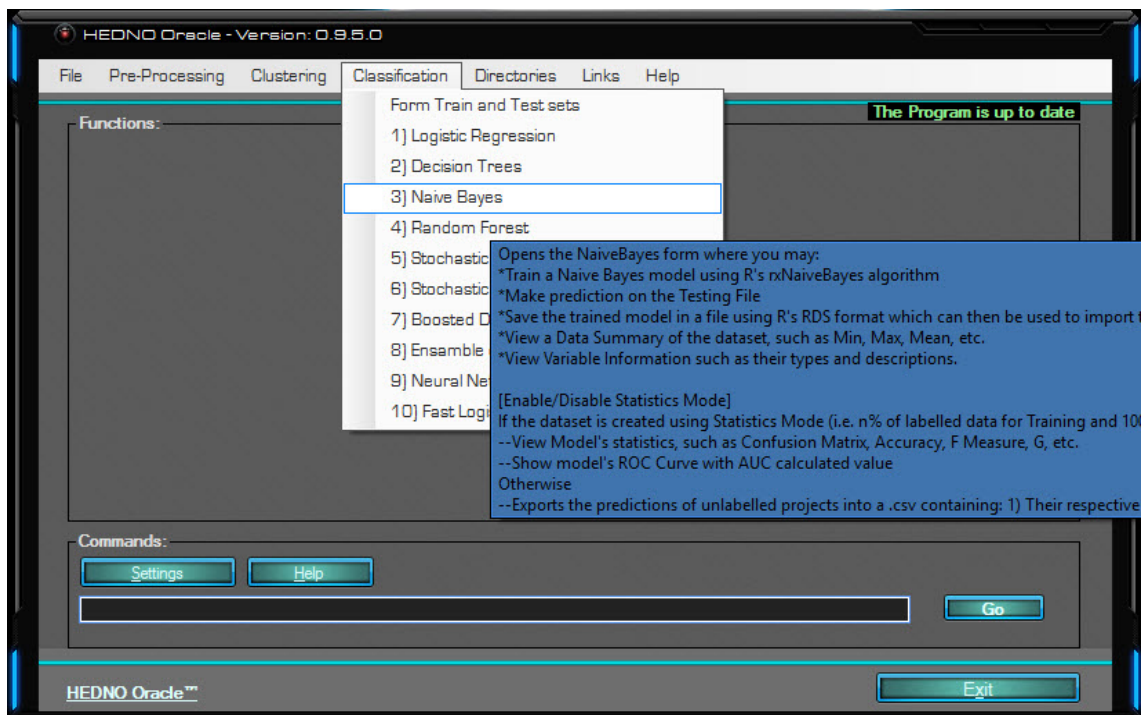




**Figure 47:** Plot of Optimal Pruning Based on Complexity

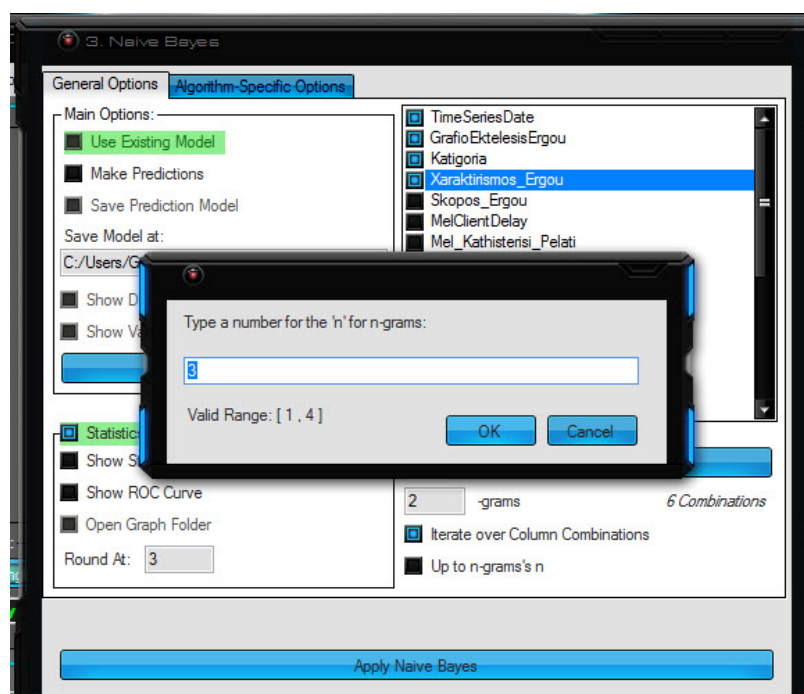
- **Show Complexity Plot:** if true, a plot appears showing how the complexity and X-val Relative Error drop as the number of trees gets higher. This can be used to determine the optimal complexity value for pruning, however the pruning process has been automated rendering manual pruning obsolete.
- **Cp:** A numeric scalar specifying the complexity parameter. Any split that does not decrease overall lack-of-fit by at least that number is not attempted. Its default value (0) produces a very large number of splits; specifying  $cp = 1e-5$  produces a more manageable set of splits. The default value leaves `maxDepth` and `minBucket` to control the tree sizes. `rpart`'s default `cp` value is 0.01 but for more splits, the `cp` can be decreased by powers of 10.

### 7.4.4 Naïve Bayes



**Figure 48:** Naïve Bayes Mouse Hover

This menu item opens the Naïve Bayes form which can be used to build a Naïve Bayes classifier, using R's rxNaiveBayes algorithm.



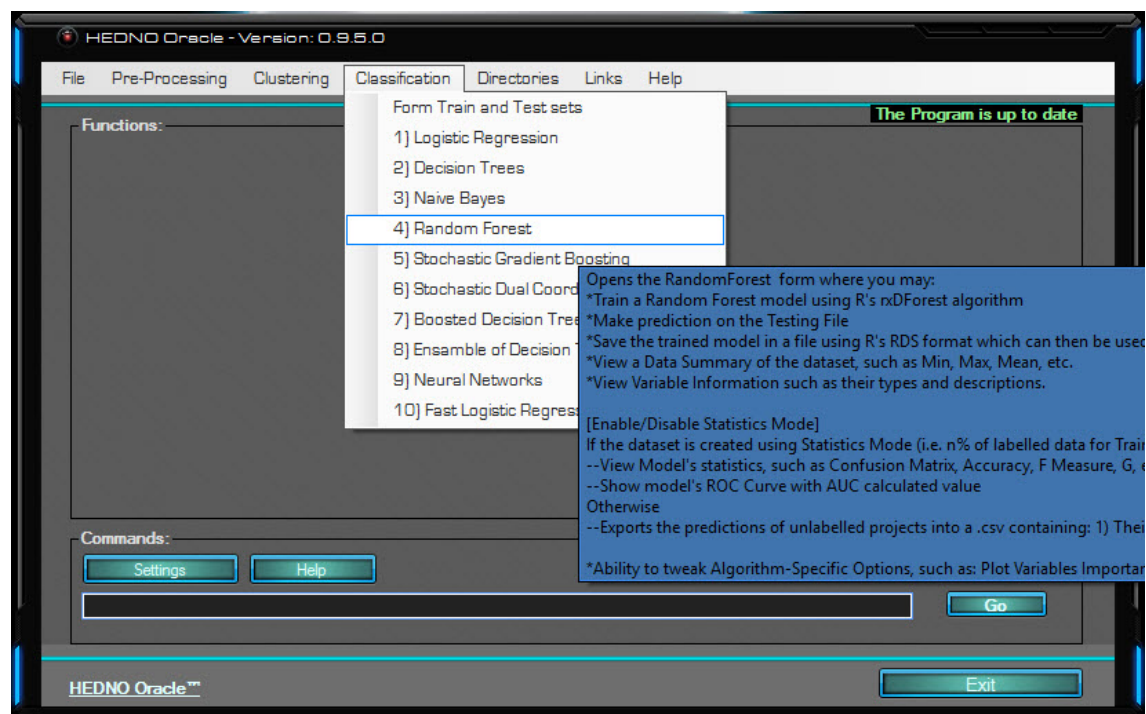
**Figure 49:** Naïve Bayes Pushed

When “Iterate over Column Combinations” is checked, the default value for n-grams is the (Checked Columns Count – 1). As four columns were checked, the default and pre-loaded value is 3, but it can change to anything spanning from 1 to the Checked Columns Count. The programme makes sure that invalid numbers cannot be returned, so cancelling the InputBox will serve only to leave the previously correctly loaded value intact. As with the previous algorithms, so with this and futures ones, upon clicking the “Apply Naïve Bayes” button, the algorithm will start running, and the button will be renamed to “Cancel” extending the ability of cancellation to the user.

Since there are several kinds of codes and environments running and interacting with one another, cancellation might not be instantaneous, and even if the form is forced to close, the underlying R Session might continue working. Once cancel is pushed, the best choice is to wait until everything can come to a complete halt.

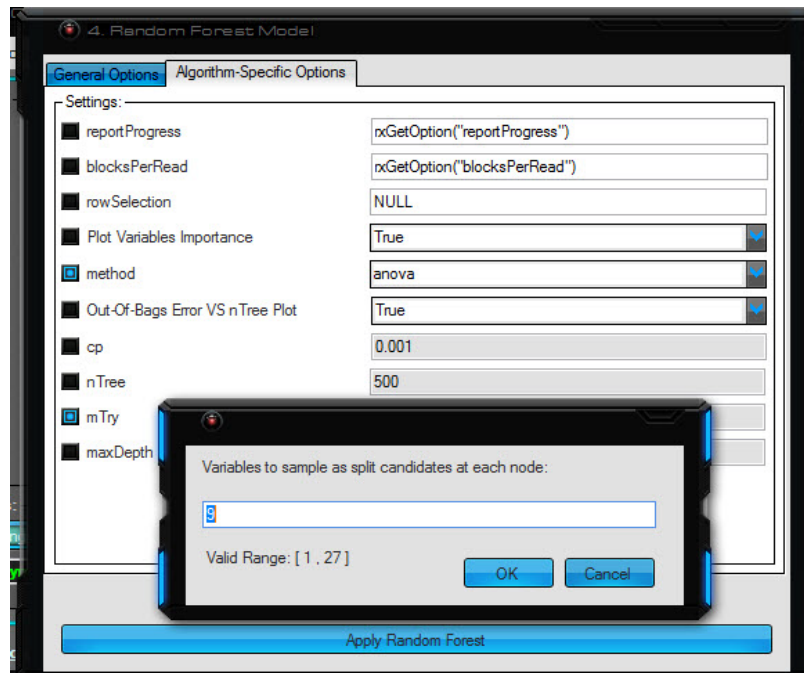
Under normal circumstances, provided everything is allowed to finish, all forms and plots checked will open sequentially as the conclude; this is, Data Summaries, Variable Information, Statistics, ROC Curves, and any algorithm-specific item.

### 7.4.5 Random Forest



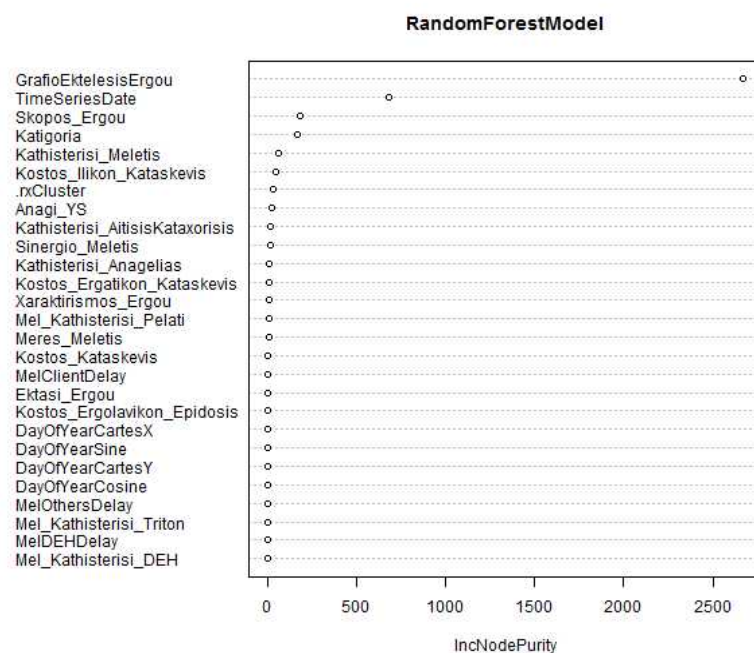
**Figure 50:** Random Forest Mouse Hover

This menu item opens the Decision Forest form which can be used to build a Random Forest classifier, using R’s rxDForest algorithm.



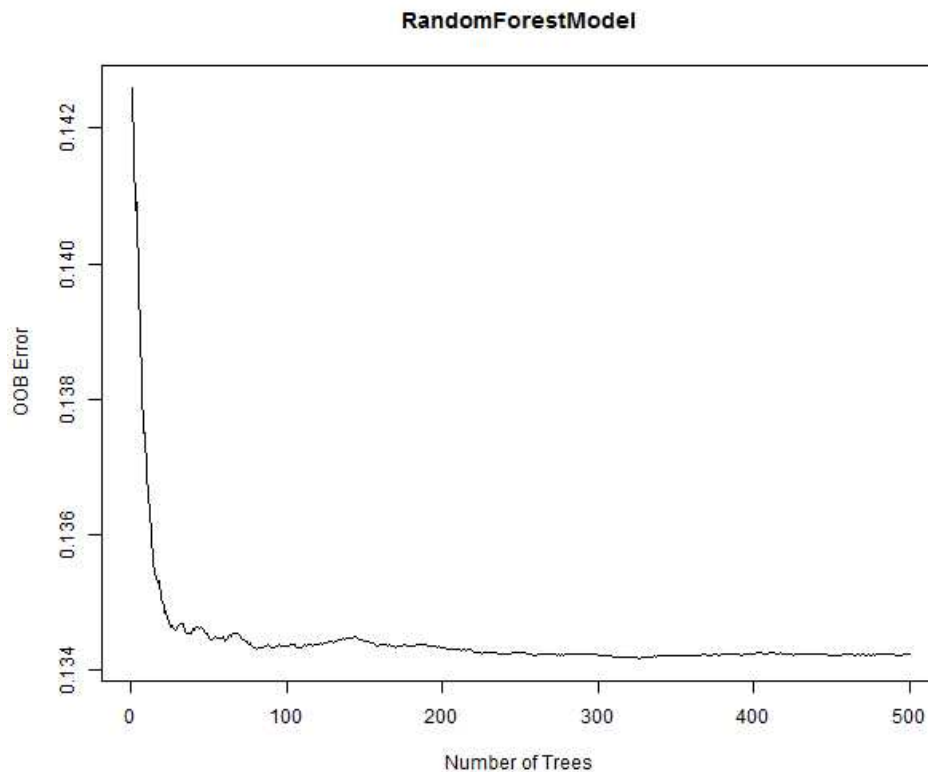
**Figure 51: Random Forest Pushed**

- **Plot Variables Importance:** if true, a plot delineating each variable's importance to the model's build is shown. In this example, the variable "GrafioEktelesisiErgou" is by far the most important factor by which the classifier decides what to predict. TimeSeriesDate comes as a not so close second, with SkoposErgou and Katigoria following. KathisterisiMeletis, KostosIlikonKataskevis, and .rxCluster are now of just a little importance, and all others following those are essentially inconsequential.



**Figure 52: Variables Importance Plot**

- **method:** the type of tree to be built; “anova” for regression trees or “class” for classification ones.
- **Out-Of-Bag Error VS nTree Plot:** A plot showing how the out-of-bag error (as described in “Applying Machine Learning” section) behaves as the number of trees gets higher.

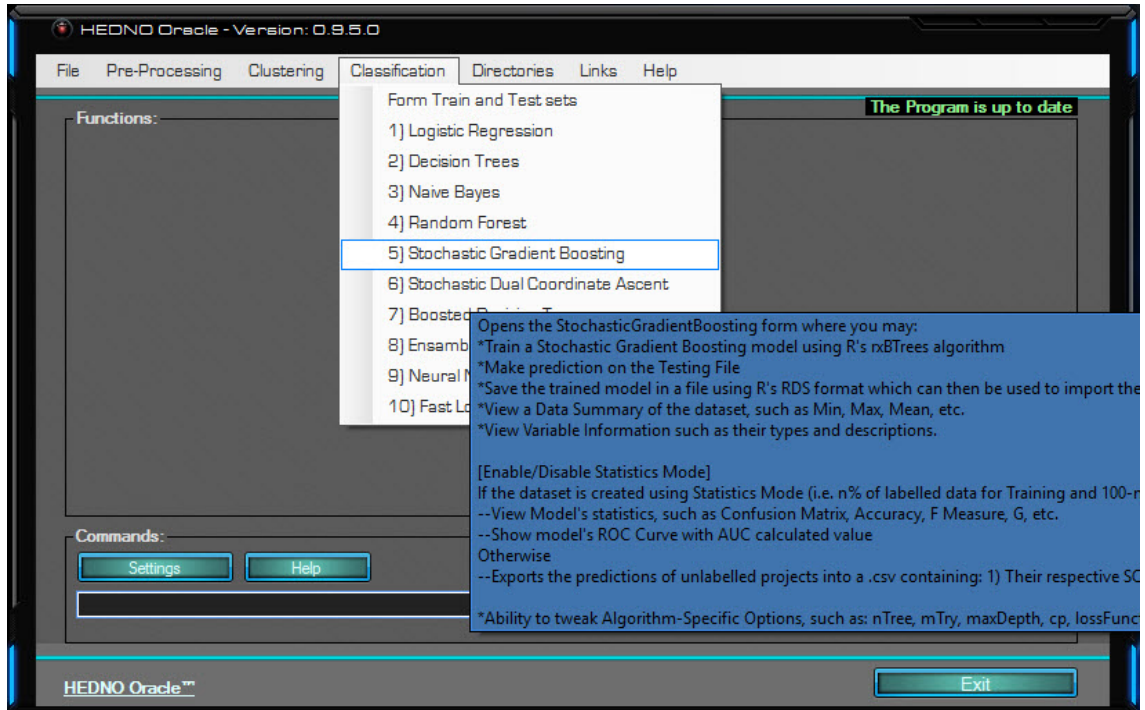


**Figure 53:** OOB Error vs nTree

- **cp:** A numeric scalar specifying the complexity parameter. Any split that does not decrease overall lack-of-fit by at least that number is not attempted. Its default value (0) produces a very large number of splits; specifying  $cp = 1e-5$  produces a more manageable set of splits. The default value leaves `maxDepth` and `minBucket` to control the tree sizes. `rpart`’s default `cp` value is 0.01 but for more splits, the `cp` can be decreased by powers of 10.
- **nTree:** The number of Trees to grow. Computations grow rapidly more expensive as the depth increases. The default value is 15, the maximum recommended default value according to R.
- **mTry:** A positive integer specifying the number of variables to sample as split candidates at each tree node. The default value is  $\sqrt{Variables\ Count}$  for classification and  $\frac{Variables\ Count}{3}$  for regression, which is the pre-loaded value when you click the textbox.

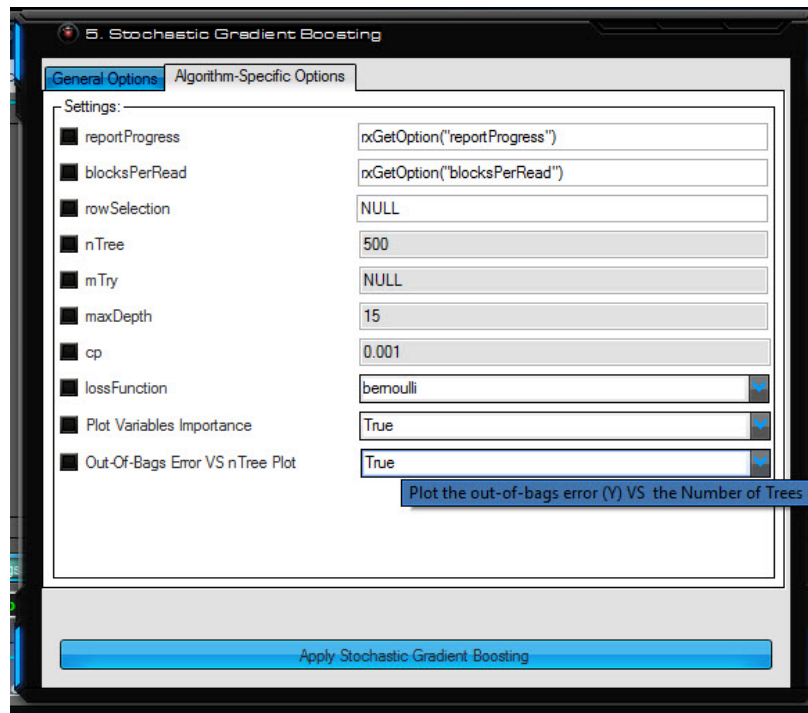
- **maxDepth:** The maximum depth of any tree node. The computations take much longer at greater depth, so lowering maxDepth can greatly speed up computation time. The default value is 15.

#### 7.4.6 Stochastic Gradient Boosting



**Figure 54:** Stochastic Gradient Boosting Mouse Hover

This menu item opens the Stochastic Gradient Boosting form which can be used to build a Stochastic Gradient Boosting classifier, using R's rxBTrees algorithm. This and every following algorithm are part of MicrosoftML package for R. <sup>[12]</sup>

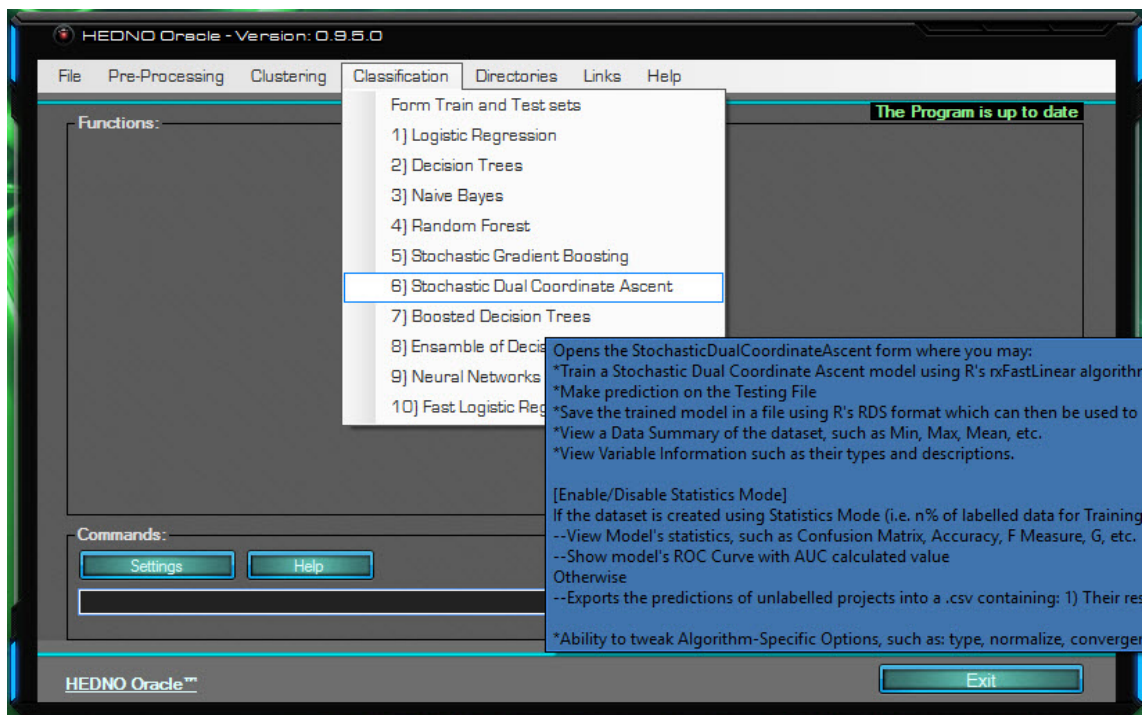


**Figure 55:** Stochastic Gradient Boosting Pushed

- **nTree:** The number of Trees to grow.
- **mTry:** A positive integer specifying the number of variables to sample as split candidates at each tree node. The default value is  $\sqrt{\text{Variables Count}}$  for classification and  $\frac{\text{Variables Count}}{3}$  for regression, which is the pre-loaded value when you click the textbox.
- **maxDepth:** The maximum depth of any tree node. The computations take much longer at greater depth, so lowering maxDepth can greatly speed up computation time. The default value is 15.
- **cp:** A numeric scalar specifying the complexity parameter. Any split that does not decrease overall lack-of-fit by at least that number is not attempted. Its default value (0) produces a very large number of splits; specifying  $cp = 1e-5$  produces a more manageable set of splits. The default value leaves maxDepth and minBucket to control the tree sizes. rpart's default cp value is 0.01 but for more splits, the cp can be decreased by powers of 10.
- **lossFunction:** A string specifying the name of the loss function to use; gaussian for regression, bernoulli for binary.
- **Plot Variables Importance:** if true, a plot delineating each variable's importance to the model's build is shown, just like in the example of Random Forest.
- **Out-of-bag Error VS nTree Plot:** A plot showing how the out-of-bag error (as described on section "Applying Machine Learning") behaves as the number of trees gets higher.

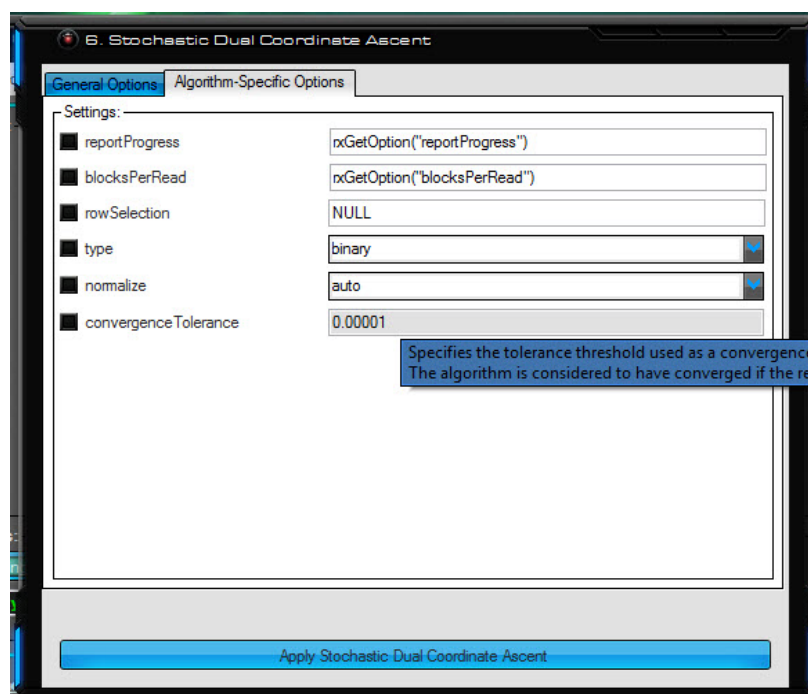


## 7.4.7 Stochastic Dual Coordinate Ascent



**Figure 56:** Stochastic Dual Coordinate Ascent Mouse Hover

This menu item opens the Stochastic Dual Coordinate Ascent form which can be used to build a Stochastic Dual Coordinate Ascent classifier, using R's rxFastLinear algorithm.

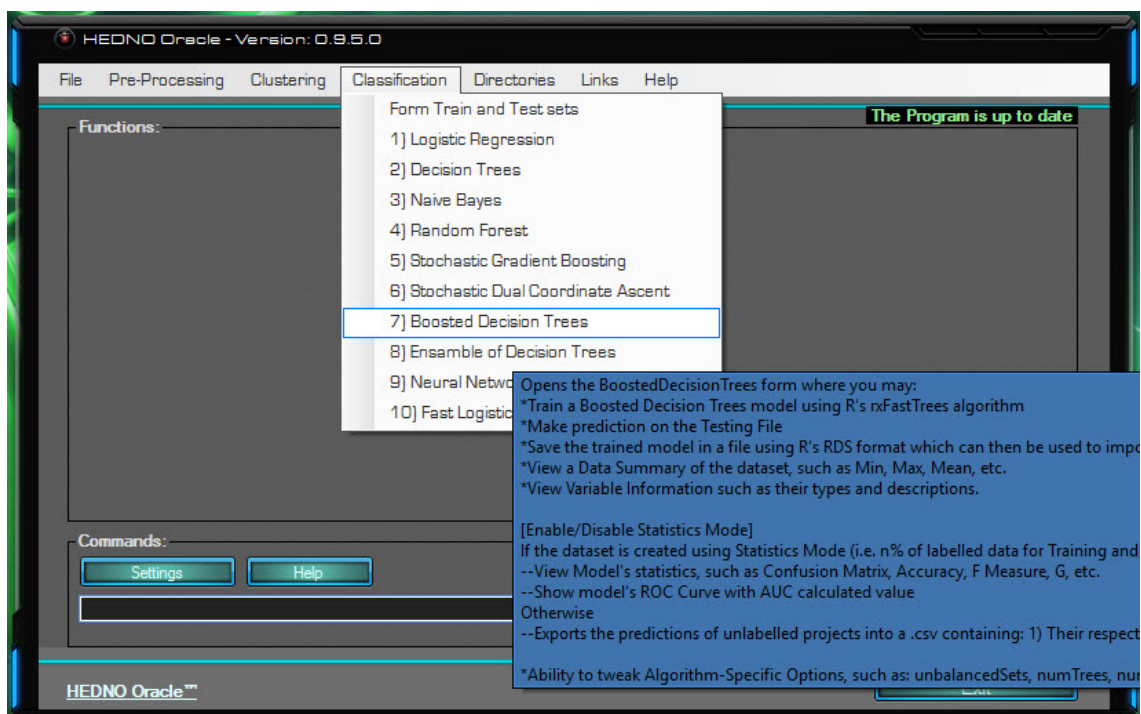


**Figure 57:** Stochastic Dual Coordinate Ascent Pushed



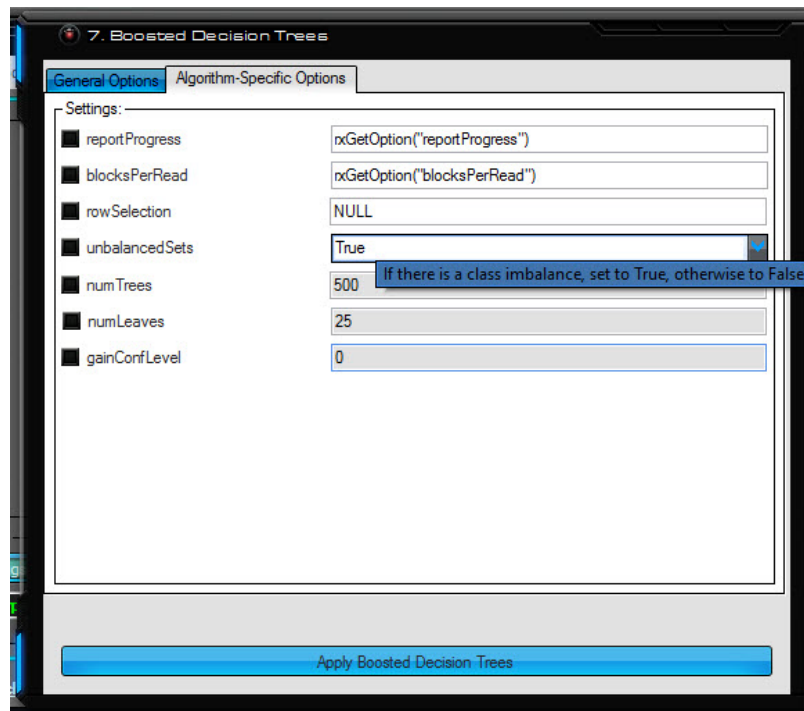
- **type:** Specifies the model type with a string: "binary" for the default binary classification or "regression" for linear regression
- **normalize:** Normalisation standardises disparate data ranges, keeping sparsity intact. Feature scaling ensures proportionality of the distances between data points and employs various optimisation methods like "gradient descent" for faster convergence. Should normalisation be performed, a MaxMin normaliser is used to normalise values in an interval  $[a, b]$  where  $-1 \leq a \leq 0$ ,  $0 \leq b \leq 1$ , and  $b - a = 1$ . The available types of automatic normalisation are:
  - "auto": if normalisation is needed, it is automatically performed. This is the default value.
  - "no": no normalisation is performed.
  - "yes": normalisation is performed.
  - "warn": if normalisation is needed, a warning message is displayed, but normalisation is not performed.
- **convergenceTolerance:** Specifies the tolerance threshold used as a convergence criterion. It must be between 0 and 1 and the default value is  $1e-5$ . The algorithm converges when the ratio between the duality gap and the primal loss (relative duality gap) gets lower than the value of convergence tolerance specified.

#### 7.4.8 Boosted Decision Trees



**Figure 58:** Boosted Decision Trees Mouse Hover

This menu item opens the Boosted Decision Trees form which can be used to build a Boosted Decision Trees classifier, using R's rxFastTrees algorithm.



**Figure 59:** Boosted Decision Trees Pushed

- **unbalancedSets:** If there is a class imbalance, set to True, otherwise to False; in the current dataset, there most definitely is class imbalance, so its value defaults to True.
- **numTrees:** The number of Trees to grow. More decision trees potentially mean better coverage, but the at the cost of an increased training time. The default value is 500.
- **numLeaves:** The maximum number of leaves (terminal nodes) that can be created in any tree. The more leaves the bigger (potentially) the size of the tree and the better the precision. However this comes with the risk of overfitting and it takes longer training times. The default value is 25.
- **gainConfLevel:** A value for the tree fitting gain confidence requirement which has to be in the range [0,1). The default value is 0.

### 7.4.9 Ensemble of Decision Trees

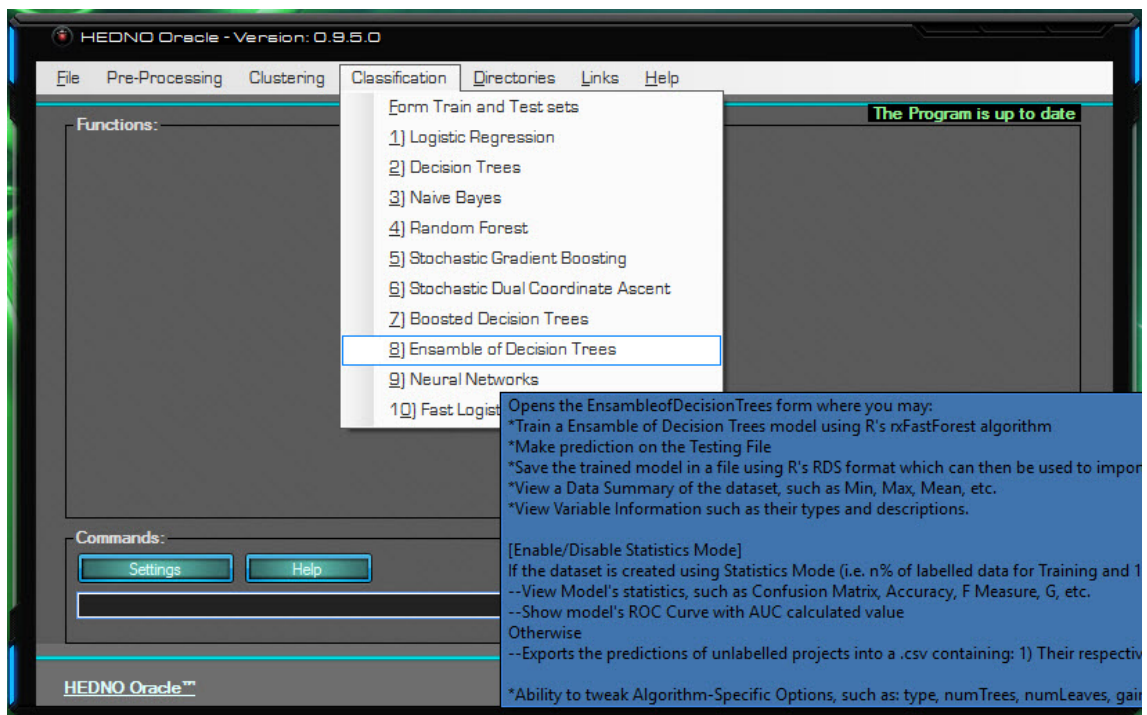


Figure 60: Ensemble of Decision Trees Mouse Hover

This menu item opens the Ensemble of Decision Trees form which can be used to build an Ensemble of Decision Trees classifier, using R's rxFastForest algorithm.

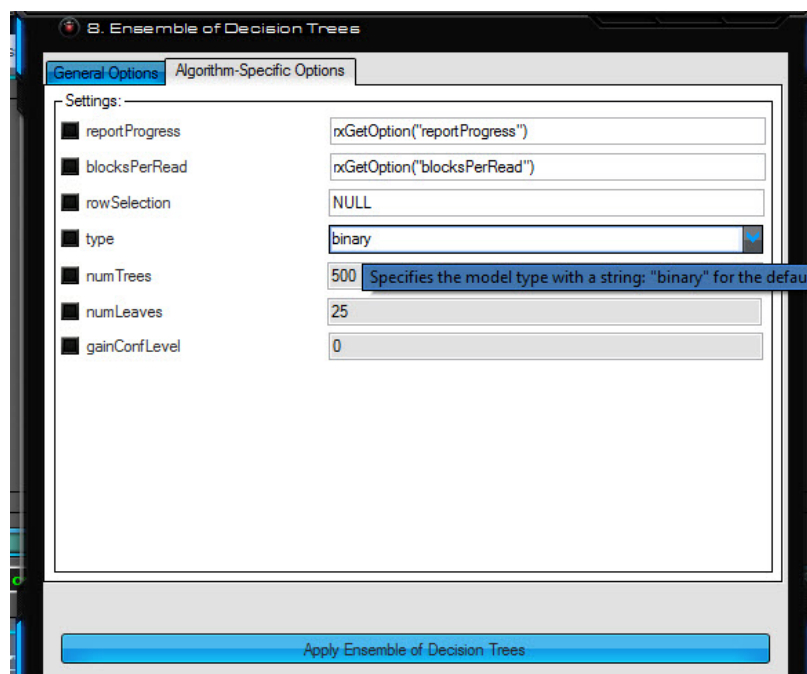
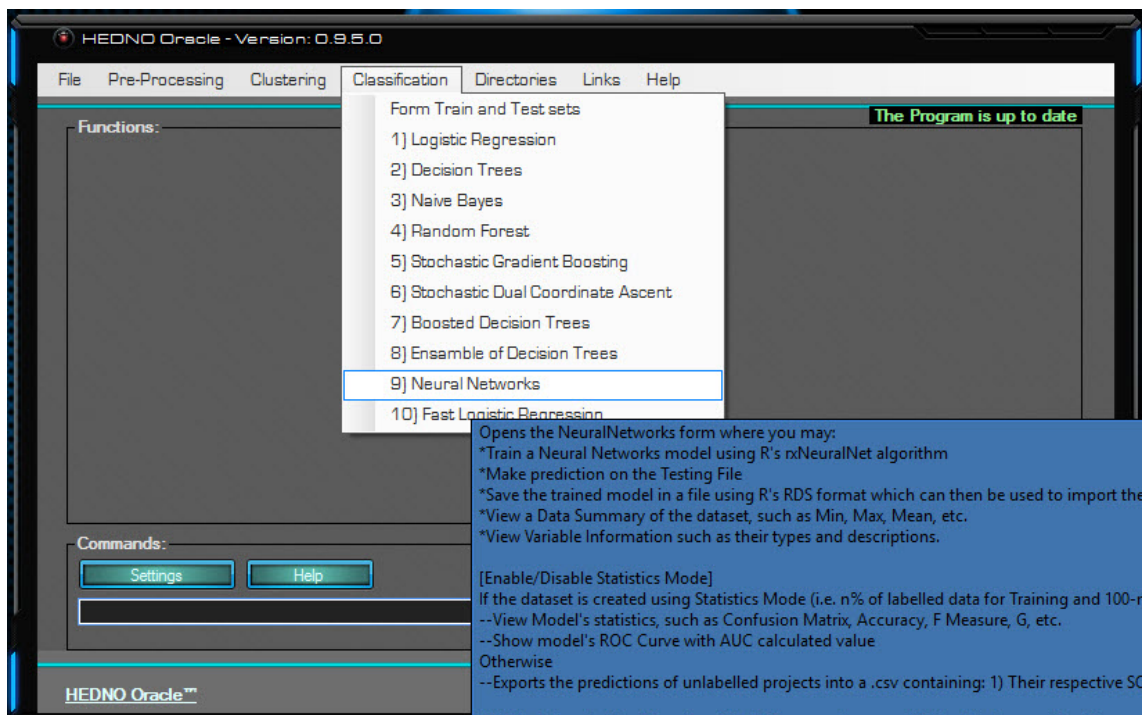


Figure 61: Ensemble of Decision Trees Mouse Hover

- **type**: Specifies the model type with a string: "binary" for the default binary classification or "regression" for linear regression

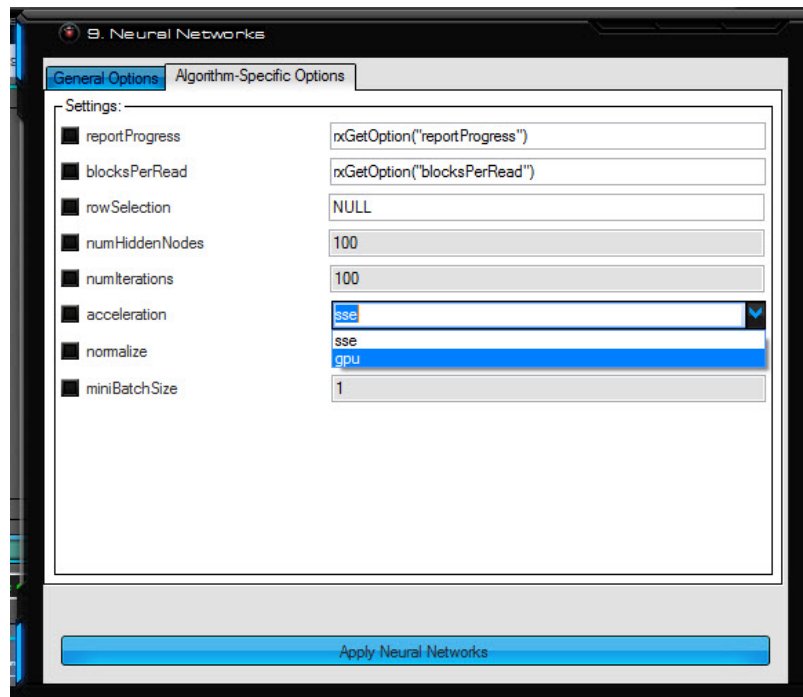
- **numTrees:** The number of Trees to grow. More decision trees potentially mean better coverage, but at the cost of an increased training time. The default value is 500.
- **numLeaves:** The maximum number of leaves (terminal nodes) that can be created in any tree. The more leaves the bigger (potentially) the size of the tree and the better the precision. However, this comes with the risk of overfitting and it takes longer training times. The default value is 25.
- **gainConfLevel:** A value for the tree fitting gain confidence requirement which has to be in the range [0,1) and the default value is 0.

#### 7.4.10 Neural Networks



**Figure 62:** Neural Networks Mouse Hover

This menu item opens the Neural Networks form which can be used to build a Neural Network classifier, using R's rxNeuralNet algorithm.

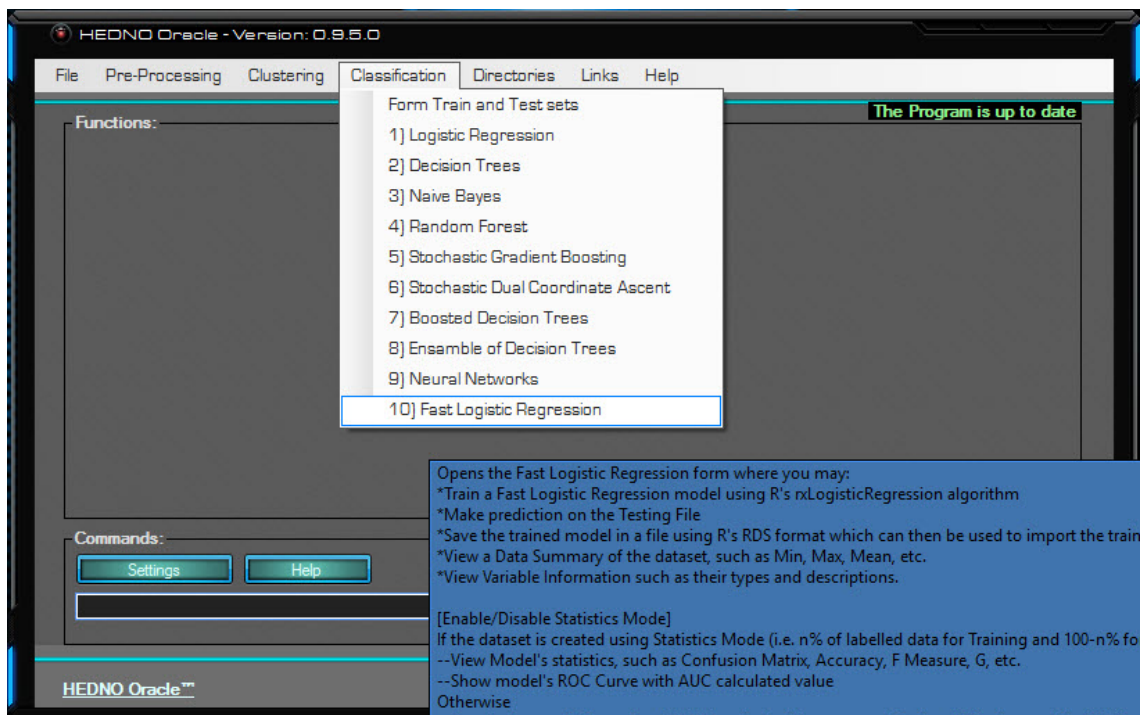


**Figure 63:** Neural Networks Pushed

- **numHiddenNodes:** The number of hidden nodes in the neural net. The default value is 100.
- **numIterations:** The number of iterations on the full training set. The default value is 100.
- **acceleration:** Specifies the type of hardware acceleration to use; either "sse" for CPU or "gpu" for GPU acceleration. For GPU acceleration, a miniBatchSize value higher than one is recommended. To use the GPU acceleration, there are additional manual setup steps are required:
  - Download and install NVidia CUDA Toolkit 6.5 (<https://developer.nvidia.com/cuda-toolkit-65>).
  - Download and install NVidia cuDNN v2 Library (<https://developer.nvidia.com/rdp/cudnn-archive>).
  - Find the libs directory of the MicrosoftRML package by calling `system.file("mxLibs/x64", package = "MicrosoftML")`.
  - Copy cublas64\_65.dll, cudart64\_65.dll and cusparse64\_65.dll from the CUDA Toolkit 6.5 into the libs directory of the MicrosoftML package.
  - Copy cudnn64\_65.dll from the cuDNN v2 Library into the libs directory of the MicrosoftML package.
- **normalize:** Normalisation standardises disparate data ranges, keeping sparsity intact. Feature scaling ensures proportionality of the distances between data points and employs various optimisation methods like "gradient descent" for faster convergence. Should normalisation be performed, a MaxMin normaliser is used to normalise values in an interval [a, b] where  $-1 \leq a \leq 0$ ,  $0 \leq b \leq 1$ , and  $b - a = 1$ . The available types of automatic normalisation are:

- "auto": if normalisation is needed, it is automatically performed. This is the default value.
- "no": no normalisation is performed.
- "yes": normalisation is performed.
- "warn": if normalisation is needed, a warning message is displayed, but normalisation is not performed.
- **miniBatchSize**: Sets the mini-batch size which is only used when the acceleration is GPU. Higher values improve the speed of training, but there's potential for negatively affecting the accuracy. It is recommended that its value be in the range [1, 256].

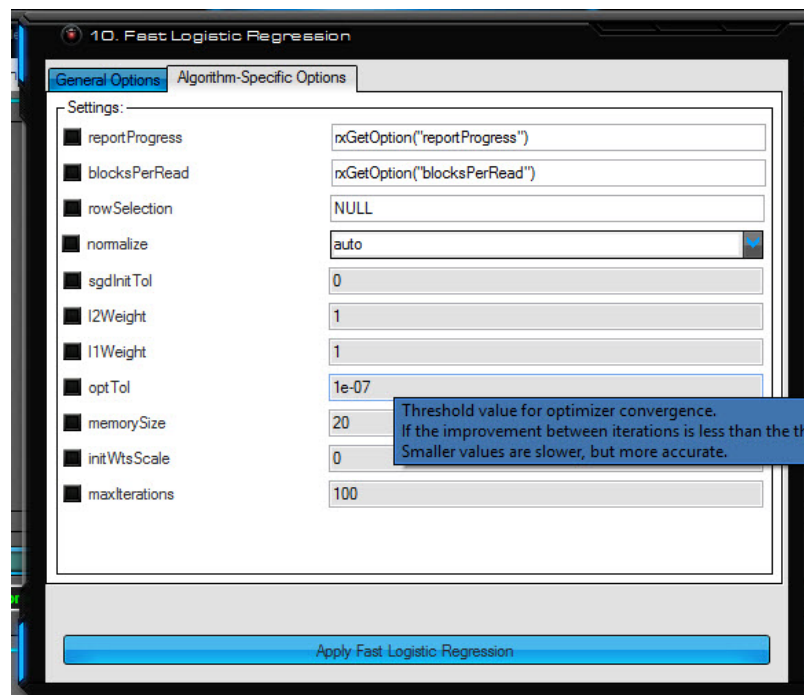
### 7.4.11 Fast Logistic Regression



**Figure 64:** Fast Logistic Regression Mouse Hover

This menu item opens the Fast Logistic Regression form which can be used to build a Fast Logistic Regression classifier, using R's rxLogisticRegression algorithm.





**Figure 65:** Fast Logistic Regression Pushed

- normalize:** Normalisation standardises disparate data ranges, keeping sparsity intact. Feature scaling ensures proportionality of the distances between data points and employs various optimisation methods like "gradient descent" for faster convergence. Should normalisation be performed, a MaxMin normaliser is used to normalise values in an interval  $[a, b]$  where  $-1 \leq a \leq 0$ ,  $0 \leq b \leq 1$ , and  $b - a = 1$ . The available types of automatic normalisation are:
  - "auto": if normalisation is needed, it is automatically performed. This is the default value.
  - "no": no normalisation is performed.
  - "yes": normalisation is performed.
  - "warn": if normalisation is needed, a warning message is displayed, but normalisation is not performed.
- sgdInitTol:** A number greater than 0 uses Stochastic Gradient Descent (SGD) to find the initial parameters. A non-zero value specifies the tolerance SGD uses to determine convergence whilst 0 means SGF will not be used.
- l2Weight:** The L2 regularisation (ridge) weight to pull large weights towards zero. Its value must be greater than or equal to 0 and 1 is preferable for data that is not sparse.
- l1Weight:** The L1 regularisation weight (lasso) to pulls small weights associated features that are relatively unimportant towards 0. Its value must be greater than or equal to 0 and 1 is preferable to sparse models when working with high-dimensional data.
- optTol:** Threshold value for optimiser convergence. If the improvement between iterations is less than the threshold, the algorithm stops and returns the current model. Smaller values are slower, but more accurate and it defaults to 1e-07.

- **memorySize:** Memory size for L-BFGS, specifying the number of past positions and gradients to store for the computation of the next step. It cannot be less than 1 and it defaults to 20. This optimisation parameter limits the amount of memory that is used to compute the magnitude and direction of the next step. When less memory is specified, training is faster but less accurate.
- **initWtsScale:** Sets the initial weights diameter that specifies the range from which values are drawn for the initial weights. These weights are initialised randomly from within this range.
- **maxIterations:** Sets the maximum number of iterations after which the algorithm stops even if it has not satisfied convergence criteria.



## 7.4.12 Model Evaluation

**Table 10:** Model Evaluation

Model Name	Logistic Regression	Decision Trees	Naive Bayes	Random Forest	Stochastic Gradient Boosting	Stochastic Dual Coordinate Ascent	Boosted Decision Trees	Ensemble of Decision Trees	Neural Networks	Logistic Regression
Algorithm Name	rxLogit	rxDTree	rxNaiveBayes	rxDForest	rxBTrees	rxFastLinear	rxFastTrees	rxFastForest	rxNeuralNet	rxLogisticRegression
Correctly Classified	80.878%	82.635%	77.648%	81.098%	82.542%	78.072%	79.639%	80.305%	82.565%	80.932%
Incorrectly	19.122%	17.365%	22.352%	18.902%	17.458%	21.928%	20.361%	19.695%	17.435%	19.068%
AUC	0.756	0.778	0.730	0.784	0.796	0.738	0.807	0.731	0.791	0.756
F1	0.885	0.895	0.868	0.889	0.891	0.860	0.866	0.885	0.896	0.886
G	0.888	0.897	0.872	0.893	0.892	0.860	0.866	0.890	0.899	0.889
PhiMCC	0.369	0.444	0.213	0.368	0.463	0.353	0.445	0.329	0.435	0.370
CohensK	0.329	0.413	0.175	0.286	0.453	0.352	0.444	0.241	0.383	0.327
YoudensJ	0.265	0.345	0.134	0.214	0.408	0.336	0.458	0.176	0.305	0.261
Accuracy	0.809	0.826	0.776	0.811	0.825	0.781	0.796	0.803	0.826	0.809
BalancedAccuracy	0.632	0.673	0.567	0.607	0.704	0.668	0.729	0.588	0.652	0.630
DetectionRate	0.738	0.737	0.735	0.758	0.715	0.675	0.657	0.759	0.749	0.740
MisclassRate	0.191	0.174	0.224	0.189	0.175	0.219	0.204	0.197	0.174	0.191
SensitRecallTPR	0.960	0.958	0.956	0.985	0.929	0.877	0.854	0.987	0.974	0.962
FPR	0.695	0.613	0.822	0.771	0.521	0.541	0.395	0.811	0.669	0.701
SpecificityTNR	0.305	0.387	0.178	0.229	0.479	0.459	0.605	0.189	0.331	0.299
FNR	0.040	0.042	0.044	0.015	0.071	0.123	0.146	0.013	0.026	0.038
PrecisionPPV1	0.822	0.839	0.795	0.810	0.856	0.844	0.878	0.803	0.829	0.821
PPV2	1.070	1.075	1.062	1.049	1.086	1.108	1.100	1.044	1.065	1.069
NPV1	0.693	0.733	0.545	0.824	0.670	0.528	0.553	0.812	0.791	0.703

NPV2	0.460	0.560	0.246	0.516	0.572	0.483	0.582	0.450	0.574	0.462
FDR	0.178	0.161	0.205	0.190	0.144	0.156	0.122	0.197	0.171	0.179

There are two types of errors that can be made in a binary problem, Type I Error (False Positive) and type II error (False Negative). The cost of making a Type I Error is negligible in comparison to the cost of making a type II error for the HEDNO company. This is because making certain that needed items for a project that is ultimately not going to be approved, are readily available, is not that detrimental as the items will be utilised somewhere down the road. Being reassured, however, that a set of items is not going to be needed in the near future, when in fact it is because the project is ultimately going to be approved, is unfavourable as the whole project could be delayed for a considerable amount of time. Therefore, a positive result is achieved by minimising type II error, even at the expense of making more Type I Errors.

The algorithms: Logistic Regression, Naive Bayes, Stochastic Dual Coordinate Ascent, Fast Logistic Regression, Random Forest, have no substantial results relative to the rest of the algorithms.

The Decision Trees classifier has relatively exceptional results on Correctly Classified, Incorrectly Classified, Accuracy, and Misclassification Rate. Its predictions are the most accurate in that it has the most True Positives and True negatives out of all the algorithms.

Stochastic Gradient Boosting classifier's PhiMCC and CohensK are relatively the best, meaning that its result's correlation coefficient between the observed and predicted values surpass all others', as does their performance compared to the expected performance if the classifier were predicting according to pure chance.

The Boosted Decision Trees classifier ranked best on AUC, YoudensJ, BalancedAccuracy, FPR, SpecificityTNR, PrecisionPPV1, NPV2, and FDR. Its performance over all possible thresholds is superior to that of all other algorithms as given by AUC. It has the highest maximum vertical distance between the ROC curve and the diagonal. Its balanced accuracy is the highest, meaning that its estimate remains true irrespective of its being used on a dataset with or without class imbalance. The proportion of its predicting Approved when the project is Cancelled is the lowest, its 'cancelled' predictions are the most accurate, which also stands true for its 'Approved' ones, and the rate at which false positives occur is the lowest. Despite its being the overall best fit algorithm for the dataset, what matters the most is type II error minimisation, which is not its main strength.

The Ensemble of Decision Trees classifier enjoys highest scores on DetectionRate, SensitRecallTPR, and FNR. The rate at which the classifier identifies the True Positive

cases is the highest amongst all the other algorithms. When the project is actually approved, the classifier identifies it better than the rest whilst also keeping the rate at which false negatives occur to the lowest. The Ensemble of Decision Trees behave best at the main variable, which is the best prediction of positives among all positives (correctly predicted positively and falsely predicted negatively).

The Neural Networks Classifier ranked best on F1, G, and MisclassRate. The algorithm does comparably great on the weighted average of its ability to predict “Approved” when in fact it is approved and its ability to be correct when it predicts “Approved”. It also does overall the least mistakes, but the fact remains that the Ensemble of Decision Trees minimises type II error, which is the main objective.

## 8 Conclusions

Running with high efficiency is a matter of utmost importance to a company, for therein lies the recipe for reaching the end goal as quickly and effortlessly as possible, maximising both financial outcome and work potential. For the task at hand, it means getting a foresight as to which projects are going to be approved and which are going to be cancelled. This allows for items to be readily available, preventing a need to put projects on hold. Having worked with real data meant dealing with a high degree of noise in the data and having to invest a lot to pre-processing in addition to the main work. A user-friendly and customisable programme has been developed, automating every task leading to the use of 10 highly scalable machine learning algorithms for prediction purposes. Having trained the models on the dataset in question (HEDNO S.A.'s), statistical rates and measures reveal that the optimal results with regards to minimising type II error are given by the Ensemble of Decision Trees classifier.



# References

1. 1988, David E. Goldberg, J. H. Genetic Algorithms and Machine Learning. Springer.
2. 1986, Kodratoff, Y. Introduction to machine learning. Long Acre, London: Pitman Publishing. Retrieved from [https://books.google.gr/books?hl=en&lr=&id=AQyJBQAAQBAJ&oi=fnd&pg=PP1&dq=introduction+to+machine+learning&ots=XnsT6zF3KJ&sig=W3cIq85LIpp31r2tGVSR07tnCi4&redir\\_esc=y#v=onepage&q=introduction%20to%20machine%20learning&f=false](https://books.google.gr/books?hl=en&lr=&id=AQyJBQAAQBAJ&oi=fnd&pg=PP1&dq=introduction+to+machine+learning&ots=XnsT6zF3KJ&sig=W3cIq85LIpp31r2tGVSR07tnCi4&redir_esc=y#v=onepage&q=introduction%20to%20machine%20learning&f=false)
3. 2012, Lison, P. An introduction to machine learning. Language Technology Group (LTG), Dept. of Informatics, University of Oslo.
4. 2006, Pattern Recognition and Machine Learning. C. Bishop. Springer.
5. 1984, R.S. Michalski, J. C. Machine Learning: An Artificial Intelligence Approach. Springer-Verlag.
6. 2007, Ryszard Michalski; Shaped How Machines Learn, <http://www.washingtonpost.com/wp-dyn/content/article/2007/09/30/AR2007093001569.html>
7. 2009, Physics of the Impossible, Anchor Publications, ISBN-13: 978-0307278821
8. R Core Team (2015). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
9. APACHE Spark, <http://spark.apache.org/>.
10. Microsoft SQL Server, <https://www.microsoft.com/en-us/sql-server/sql-server-2016>
11. Microsoft Corporation (2016). RevoScaleR: Scalable, distributable, fast, and extensible Data Analysis in R. R package version 9.0.1.
12. Microsoft Corporation (2016). MicrosoftML: Microsoft Machine Learning for R. R package version 1.0.0.
13. Microsoft Corporation (2016). RevoTreeView: Decision Tree Visualization from Microsoft Corporation. R package version 10.0.0.
14. Microsoft R. <https://msdn.microsoft.com/en-us/microsoft-r/index>
15. 2010, The balanced accuracy and its posterior distribution, <http://dl.acm.org/citation.cfm?id=1905533>
16. 2005, Optimal Cut-point and Its Corresponding Youden Index to Discriminate Individuals Using Pooled Blood Samples, <https://www.ncbi.nlm.nih.gov/pubmed/15613948>
17. 2011, AILab, Evaluation: From Precision, Recall And F-Measure to ROC, Informedness, Markedness & Correlation
18. 1994, Diagnostic tests 2: predictive values, BMJ Volume 309, <https://www.ncbi.nlm.nih.gov/pubmed/8038641>
19. 1950, Youden, W.J., Index for rating diagnostic tests, doi:10.1002/1097-0142(1950)3:1<32::aid-cnrcr2820030106>3.0.co;2-3
20. 2005, Tom Fawcett, An introduction to ROC analysis, doi>10.1016/j.patrec.2005.10.010

21. 2013, Max Kuhn, Kjell Johnson, Applied Predictive Modeling, doi> 10.1007/978-1-4614-6849-3
22. 2008, Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, Geoffrey J. McLachlan, Angus Ng, Bing Liu, Philip S. Yu, Zhi-Hua Zhou, Michael Steinbach, David J. Hand and Dan Steinberg, Top 10 Algorithms in Data Mining, Knowledge and Information Systems
23. Clustering Algorithms and Evaluations PhD Thesis, University of Stuttgart, <http://www.ims.uni-stuttgart.de/institut/mitarbeiter/schulte/theses/phd/algorithm.pdf>
24. HEDNO S.A., <http://www.deddie.gr/en>