# Ανοικτό Πανεπιστήμιο Κύπρου

**Σχολή Θετικών και Εφαρμοσμένων Επιστημών**

**Μεταπτυχιακό Πρόγραμμα Σπουδών**
***Πληροφοριακά και Επικοινωνιακά Συστήματα***


## Μεταπτυχιακή Διατριβή

# Ανάπτυξη Web-based Περιβάλλοντος για Ημι-παραμετρικό Μοντέλο Πρόβλεψης Κόστους Λογισμικού


**Χαριστή Φραγκιαδουλάκη**

**Επιβλέπων Καθηγητής**
**Δρ. Νικόλαος Μήττας**



**Δεκέμβριος 2017**

# Ανοικτό Πανεπιστήμιο Κύπρου

**Σχολή Θετικών και Εφαρμοσμένων Επιστημών**

**Μεταπτυχιακό Πρόγραμμα Σπουδών**

***Πληροφοριακά και Επικοινωνιακά Συστήματα***


# Μεταπτυχιακή Διατριβή

## Ανάπτυξη Web-based Περιβάλλοντος για Ημι-παραμετρικό Μοντέλο Πρόβλεψης Κόστους Λογισμικού


**Χαριστή Φραγκιαδουλάκη**

**Επιβλέπων Καθηγητής**
**Δρ. Νικόλαος Μήττας**



Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων για απόκτηση μεταπτυχιακού τίτλου σπουδών στα Πληροφοριακά και Επικοινωνιακά Συστήματα από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών του Ανοικτού Πανεπιστημίου Κύπρου.

**Δεκέμβριος 2017**

# Περίληψη

Η Εκτίμηση Κόστους Λογισμικού είναι ένα σημαντικό ερευνητικό πρόβλημα, το οποίο έχει προσελκύσει το ενδιαφέρον της επιστημονικής κοινότητας από την δεκαετία του 80 και έπειτα. Ο βασικός σκοπός είναι η ρεαλιστικότερη και ακριβέστερη πρόβλεψη της απαιτούμενης προσπάθειας για την ολοκλήρωση ενός έργου λογισμικού. Το αποτέλεσμα αυτής της διαδικασίας μπορεί να χρησιμοποιηθεί από διοικητές έργων για την βελτιστοποίηση του σχεδιασμού ενός έργου, την καλύτερη διοίκηση ανθρωπίνων πόρων αλλά και την διασφάλιση της βέλτιστης διαχείρισης κινδύνων.

Μέχρι σήμερα οι ερευνητές επικεντρώνονται σε δύο γνωστές κλάσεις μοντέλων πρόβλεψης, τις παραμετρικές και τις μη-παραμετρικές μεθοδολογίες. Παρά τις μελέτες σύγκρισης που έχουν πραγματοποιηθεί αυτά τα χρόνια φαίνεται ότι υπάρχει δυσκολία στην επιλογή της καλύτερης μεθόδου πρόβλεψης. Για τον λόγο αυτόν έχουν προταθεί οι ημι-παραμετρικές μεθοδολογίες που συνδυάζουν τα καλύτερα χαρακτηριστικά των δύο προηγούμενων κλάσεων.

Στην παρούσα διατριβή, γίνεται μία συγκριτική μελέτη και αποτιμάται η προβλεπτική ικανότητα των μη-παραμετρικών και ημι-παραμετρικών μεθόδων. Πιο συγκεκριμένα, εξετάζονται τέσσερα γνωστά μη-παραμετρικά μοντέλα και εκτιμάται κατά πόσο η εισαγωγή ενός παραμετρικού όρου σε αυτά και το ημι-παραμετρικό μοντέλο που προκύπτει αποτελεί μία ρεαλιστικότερη προσέγγιση για τη μοντελοποίηση της προσπάθειας που απαιτείται για την ανάπτυξη έργων λογισμικού. Η αξιολόγηση των μοντέλων γίνεται βάσει γνωστών μέτρων ακρίβειας και ενός αλγορίθμου συσταδοποίησης (Scott-Knott) που βασίζεται σε στατιστικούς ελέγχους πολλαπλών υποθέσεων.

Η παραπάνω προσέγγιση αποτελεί ένα ολοκληρωμένο πλαίσιο εργασίας, το οποίο υλοποιήθηκε σε μία ολοκληρωμένη διαδικτυακή εφαρμογή στην γλώσσα προγραμματισμού R και το πακέτο Shiny. Τούτο μας επέτρεψε να αυτοματοποιήσουμε ολόκληρη τη διαδικασία συγκριτικής μελέτης δίνοντας την δυνατότητα στους διοικητές έργου και σε άλλους επαγγελματίες και ερευνητές να αντιληφθούν καλύτερα ομάδες από τα καταλληλότερα μοντέλα ΕΚΛ και να διαλέξουν ενδεχομένως και το

καταλληλότερο κατά το δοκούν μέσα από μία ποικιλία ανάλογα με τις τρέχουσες ανάγκες τους.

# Summary

The Software Cost Estimation is an important scientific research problem where it has attracted the interest of the scientific community from the early 80s. It's main goal is to predict as realistic as possible the precise prediction of the effort required for completing a Software project. The result of this process can be used by Project Managers for improving the Project Planning, the resource allocation and assuring the optimum Risk Management.

Until today the researches interest has been focused on two well known classes of models, namely the parametric Regression Analysis and the non-parametric Estimation by Analogy. Despite the several comparison studies that have taken place over the years, there seems to be a discrepancy in choosing the best prediction technique between them. To this end the semi-parametric methods have been introduced, that have the ability to combine the advantages of both categories.

In this thesis, a comparison is realized where we study and evaluate the predictive ability of eight Prediction Models. More specifically, four known non-parametric models are studied and it is assessed whether the introduction of a parametric component to them and the resulting semi-parametric model is a more realistic approach to modelling the effort needed to develop software projects. The evaluation of the models is based on known precision measures and a clustering algorithm (Scott-Knott) based on multi-assay statistical tests.

The above approach is an integrated framework, which was implemented in a comprehensive web application in the R programming language and the Shiny package. This has allowed us to automate the entire comparative study process by enabling project managers and other professionals and researchers to better understand the groups of the most suitable SCE models and select the most appropriate for a diverse variety according to their current needs.

# Ευχαριστίες

Η παρούσα διπλωματική εργασία με τίτλο “Ανάπτυξη Web-based Περιβάλλοντος για Ημι-παραμετρικό Μοντέλο Πρόβλεψης Κόστους Λογισμικού” αναπτύχθηκε στα πλαίσια του Μεταπτυχιακού μου στα Πληροφοριακά και Επικοινωνιακά Συστήματα στην Σχολή Θετικών και Εφαρμοσμένων Επιστημών του Ανοικτού Πανεπιστημίου Κύπρου.

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή της Διπλωματικής μου εργασίας Δρ.Νικόλαο Μήττα, για την εμπιστοσύνη που μου έδειξε κατά την επιλογή του θέματος, τις πολύτιμες συμβουλές του και τις ουσιαστικές παρατηρήσεις για την ολοκλήρωση της εργασίας. Ακόμα, τον ευχαριστώ για τη γνώση που μου μετέφερε στο ειδικό και τόσο ενδιαφέρον θέμα που πραγματοποίησα με την βοήθεια του.

Ευχαριστίες όμως, οφείλω σε όλους τους καθηγητές του Τμήματος Πληροφορικής που με τις γνώσεις, τις συμβουλές και τις παρατηρήσεις των οποίων συνέβαλαν στη λήψη του Μεταπτυχιακού πτυχίου μου.

Τέλος θα ήθελα να ευχαριστήσω τον σύζυγο μου και την οικογένεια μου για την αμέριστη υποστήριξη και την συνεχή ενθάρρυνση που μου επέδειξαν κατά την διάρκεια του Μεταπτυχιακού μου η οποία αποτέλεσε ισχυρό κίνητρο για την επιτυχή ολοκλήρωση του.

Χαριστή Φραγκιαδουλάκη

Θεσσαλονίκη, Δεκέμβριος 2017

# Περιεχόμενα

# Πίνακας με ακρωνύμια

| Ακρωνύμιο | Περιγραφή |
|---|---|
| AE | Absolute Error |
| ANOVA | Analysis of Variance |
| Bagging | Bootstrap aggregation |
| BRACE | Bootstrap based Analogy Cost Estimation |
| BRE | Balance Relative Error |
| CART | Classification and Regression Tree |
| CBR | Case Based Reasoning |
| COCOMO | Constructive Cost Model |
| EbA | Estimation by Analogy |
| FP | Function Points |
| IBRE | Inverted Balance Relative Error |
| IDE | Integrated Development Environment |
| ISBSG | International Software Benchmarking Standards Group |
| LSEbA | Least Square Estimation by Analogy |
| LOOCV | Leave one out cross validation |
| LM | Linear Model |
| LOES | Locally Weighted Regression |
| LOC | Lines of Code |
| LS | Least squares |
| MER | Magnitude of Relative Error to the Estimate |
| MRE | Magnitude of Relative Error |
| MSE | Mean Squared Error |
| MVC | Model-View-Controller |
| OSR | Optimized Set Reduction |
| REC | Regression Error Characteristic Curves |
| SCE | Software Cost Estimation |
| SLIM | Software Life-Cycle Model |
| SLOC | Source Line of Code |
| SPQR | Software Productivity, Quality and Reliability |
| SPR | Software Productivity Research |
| SQE | Square Error |
| UI | User Interface |
| ΕΚΛ | Εκτίμηση Κόστους Λογισμικού |
| ΚΝΝ | Αριθμός κ-πλησιέστερων γειτόνων |
| Test Set | Σύνολο Ελέγχου |
| Training Set | Σύνολο Εκπαίδευσης |

# Κεφάλαιο 1
# Εισαγωγή

Στο πρώτο κεφάλαιο της παρούσας Διατριβής πραγματοποιείται μια ανάλυση που αφορά το πρόβλημα της Εκτίμησης Κόστους Λογισμικού-ΕΚΛ (Software Cost Estimation -SCE), εστιάζοντας στο να προσδιοριστεί επαρκώς το κόστος για την υλοποίηση ενός έργου Λογισμικού και στα προβλήματα που μπορεί να προκύψουν οποιαδήποτε στιγμή κατά τον υπολογισμό αυτού. Στην συνέχεια, ακολουθεί μια σύντομη ιστορική αναδρομή στις μεθόδους ΕΚΛ που έχουν χρησιμοποιηθεί τις τελευταίες δεκαετίες από την επιστημονική κοινότητα. Τέλος γίνεται αναφορά στην δομή της παρούσας διατριβής και στην συνεισφορά αυτής στο θέμα της ΕΚΛ.

## 1.1 Η Εκτίμηση Κόστους Λογισμικού

Το λογισμικό τα τελευταία χρόνια αποτελεί αναπόσπαστο κομμάτι στην καθημερινότητα εκατομμυρίων χρηστών, υποστηρίζει κρίσιμες δομές μέσων μεταφοράς, υποδομών υγείας και εκτελείται σε πληθώρα συσκευών από μικρο-ηλεκτρονικά έως διατάξεις ηλεκτρονικών ελέγχων πυρηνικών εργοστασίων. Οι εταιρείες του κλάδου ανάπτυξης λογισμικού αυξάνονται όλο και περισσότερο, όπως αντίστοιχα αυξητική τάση ακολουθούν οι απαιτήσεις των χρηστών από τα έργα λογισμικού, τόσο σε ευχρηστία όσο και σε πολυπλοκότητα αλλά και αξιοπιστία.

Το λογισμικό ως προϊόν διέπεται από άυλα ή αφηρημένα χαρακτηριστικά. Έτσι είναι δύσκολο να προσδιοριστεί επαρκώς από τους φυσικούς νόμους, γεγονός που οδηγεί κάποιες φορές στην εφαρμογή πολύπλοκων διαδικασιών ανάπτυξης που έχουν ως αποτέλεσμα την υλοποίηση αναξιόπιστου λογισμικού, με κακή απόδοση και το υπέρογκο κόστος (Kittlaus & Clough, 2009).

Επομένως, η διεργασία εκτίμησης κόστους λογισμικού, καθίστανται απαραίτητη ώστε να προβλεφθεί με ακρίβεια η πρόβλεψη είτε της προσπάθειας που καταβάλλεται, είτε

του χρηματικού ποσού που απαιτείται για την ολοκλήρωση ενός έργου. Ωστόσο πολλές φορές φαίνεται πως η εκτίμηση λογισμικού γίνεται με εμπειρικές μεθόδους ή βασιζόμενη σε κάποια παράκαιρα ιστορικά στοιχεία, με αποτέλεσμα οι διοικητές έργου να μην μπορούν να εκτιμήσουν επαρκώς το μέγεθος αλλά και το τελικό κόστος ενός έργου (Boehm et al., 2000b). Αυτό οδηγεί ενίοτε σε αποκλίσεις από τις αρχικές εκτιμήσεις οι οποίες δεν είναι θεμιτές από τα ενδιαφερόμενα μέρη (stakeholders). Για τον λόγο αυτό τις τελευταίες δεκαετίες έχουν διερευνηθεί διάφορες μέθοδοι εκτίμησης του κόστους όπου φαίνεται να έχουν έναν ικανοποιητικό ρυθμό ακρίβειας.

Ως ΕΚΛ εννοείται η διαδικασία πρόβλεψης της προσπάθεια που απαιτείται για την ανάπτυξη και την συντήρηση του λογισμικού όταν τα στοιχεία μας είναι ελλιπή, χαρακτηρίζονται από μεγάλη ασάφεια και εμπεριέχουν θόρυβο. Ως μέθοδος χρησιμοποιείται στο αρχικό στάδιο σχεδιασμού έργων (Project Planning), ώστε να υποβοηθηθούν οι προβλέψεις που αφορούν τον προϋπολογισμό έργου.

Ωστόσο, η ΕΚΛ δεν ήταν ποτέ μία εύκολη διεργασία και συνεπώς υπάρχουν πολλές προτάσεις και μέθοδοι για το πώς μπορεί να κοστολογηθεί επαρκώς ένα έργο υπό ανάπτυξη. Μεγάλο μέρος της βιβλιογραφίας καταλαμβάνεται από ορθολογικές μεθόδους όπου γίνεται προσπάθεια προσέγγισης του κόστους ενός έργου λογισμικού. Έτσι μέσω μιας εκτενούς διερευνητικής μελέτης αφενός των παραμετρικών και αφετέρου των μη-παραμετρικών μοντέλων πρόβλεψης, επιδιώκουμε να αναδείξουμε τα καλύτερα χαρακτηριστικά τους. Αυτά στην συνέχεια θα συγκεραστούν με απώτερο σκοπό τη δημιουργία μίας ομάδας ημι-παραμετρικών μοντέλων, τα οποία θα μας βοηθήσουν στο να πετύχουμε την πρόβλεψη του κόστους διαφόρων έργων λογισμικού με όσο το δυνατόν μεγαλύτερη ακρίβεια.

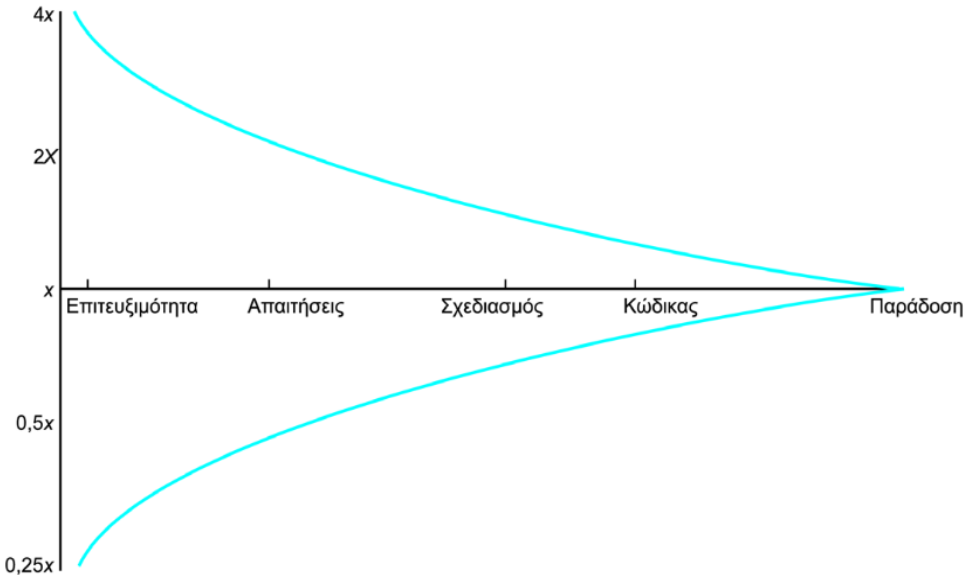
## 1.2 Το Πρόβλημα Της Εκτίμησης Κόστους Λογισμικού

Γενικότερα, η διαδικασία της πρόβλεψης, με ακρίβεια του κόστους που απαιτείται για την ολοκλήρωση ενός νέου έργου λογισμικού είναι ένα δύσκολο ζήτημα στην περιοχή της ΕΚΛ. Αυτό συμβαίνει καθώς αυτή η διαδικασία είναι στενά συνδεδεμένη με τις δραστηριότητες της διαχείρισης του ίδιου του έργου όπως επίσης και την μεθοδολογία ανάπτυξης λογισμικού που ακολουθεί ο εκάστοτε οργανισμός. Έτσι, η ακριβής πρόβλεψη του κόστους δεν επιτυγχάνεται και τις περισσότερες φορές υφίσταται

μεγάλη αβεβαιότητα. Παρόλα αυτά ως διαδικασία είναι αδιαμφισβήτητης αξίας και σημασίας όχι μόνο σε έργα μικρού εύρους αλλά και σε αυτά μεγαλύτερης κλίμακας.

Μελέτες ερευνητών έχουν καταδείξει μέχρι σήμερα ότι όταν τα έργα βρίσκονται στην φάση της επιτευξιμότητας (όπου ορίζονται οι λειτουργικές τους απαιτήσεις), έχουν αβεβαιότητα στο άνω και στο κάτω άκρο έναν παράγοντα x4. Όσο η ανάπτυξη τους προχωράει στο χρόνο η αβεβαιότητα τους υποβαθμίζεται τείνοντας προς την εξάλειψη.

Το φαινόμενο αυτό προσδιορίστηκε από τον Boehm ως ο κώνος της αβεβαιότητας και ενδεικτικά παρατίθεται στο Διάγραμμα 1 (Boehm, 1981).




**Διάγραμμα 1.1:** Αβεβαιότητα της Εκτίμησης.

Στις προηγούμενες δεκαετίες υπάρχει πληθώρα βιβλιογραφικών αναφορών, σχετικά με έργα λογισμικού τα οποία καθυστέρησαν ή ξεπέρασαν τον αρχικό προϋπολογισμό κατά πολύ με αποτέλεσμα ακόμα και να ακυρωθούν. Ενδεικτικά θα μπορούσαμε να αναφέρουμε κάποια όπως το έργο Digital Media Initiative το οποίο διήρκησε από το 2008 έως το 2013 και αφορούσε ένα λογισμικό διαχείρισης ψηφιακών παραγωγών, το οποίο καταστάθηκε παρωχημένο καθώς μέχρι να ολοκληρωθεί, ανακαλύφθηκαν εναλλακτικά αλλά και καλύτερα εμπορικά προϊόντα (The BBC Trust's Finance and Compliance, 2011). Ένα ακόμα έργο είναι το TAURUS (Transfer and Automated Registration of Uncertified Stock) το οποίο διήρκησε από το 1980 έως το 1993 και

αφορούσε μια ηλεκτρονική πλατφόρμα συναλλαγών, το συγκεκριμένο έργο δεν ολοκληρώθηκε ποτέ λόγω του υπέρογκου προϋπολογισμού (Flowers, 1996) και (Currie, 1995). Ένα τρίτο εξίσου σημαντικό έργο, το οποίο οδηγήθηκε σε ακύρωση λόγω εκτεταμένης χρονικής περιόδου ανάπτυξης, είναι το Expeditionary Combat Support System που αναπτύχθηκε λόγο ενός προγράμματος της Πολεμικής Αεροπορίας των ΗΠΑ (Aronin et al., 2011).

Από τα πρώιμα χρόνια της ανάπτυξης έργων λογισμικού μέχρι σήμερα, δεδομένου ότι το πρόβλημα της ΕΚΛ είναι ιδιαίτερα έκδηλο η ερευνητική κοινότητα επικεντρώθηκε στην μελέτη και ανάπτυξη μεθόδων όπου μέσα από την αξιοποίηση και την σύγκριση των αποτελεσμάτων τους αναζητείται η αποτελεσματικότερη μέθοδος που θα είναι προσαρμοσμένη στην εκάστοτε περίπτωση.

## 1.3 Ιστορική Αναδρομή

Η διαδικασία της ΕΚΛ, άρχισε να απασχολεί τον ευρύτερο τομέα της Τεχνολογίας και Ανάπτυξης Λογισμικού από την δεκαετία του 1950, όμως πιο εμπεριστατωμένα τις τελευταίες τρεις δεκαετίες, ομάδες έρευνας και πειραματισμού έχουν συσταθεί με σκοπό την μελέτη αυτού μέσω υπολογιστικών μεθόδων που επιδιώκουν εκτιμητική ακρίβεια. Η ευρύτερη λογική των ερευνών άπτεται στην σύγκριση των διαφόρων μοντέλων δια των αποτελεσμάτων τους, ώστε να αξιολογηθεί η ορθότητα και κατά πόσο απέχουν από το τελικό κόστος.

Τα δύο πρώτα δημοφιλέστερα μοντέλα, τα οποία βασίζονται στην μέθοδο παραμετρικής παλινδρόμησης είναι η COCOMO (Constructive Cost Model) και COCOMO II, τα οποία προτάθηκαν το 1981 και 1995 αντίστοιχα από τον Δρ. Barry W. Boehm (Boehm et al., 2000a). Στην συνέχεια προτάθηκαν άλλα μοντέλα όπως το ANGEL, το οποίο στηρίζεται στην εκτίμηση με αναλογίες (Shepperd & Schofield, 1997). Έπειτα προτάθηκαν και αναπτύχθηκαν μοντέλα όπως το SLIM (Software Life-cycle Model) (Putnam & Myers, 1992) και το SPR (Software Productivity Research) (Jones, 1991) αλλά και το παραμετρικό μοντέλο PRICE-S, το οποίο χρησιμοποιήθηκε στο διαστημικό πρόγραμμα Apollo (Boehm et al., 2000b). Την ίδια εποχή αναπτύχθηκαν και εργαλεία όπως το BRACE (Bootstrap based Analogy Cost Estimation) (Stamelos et al., 2001), που

βασίζεται ως λογισμικό σε μια μεθοδολογία αναδειγματοληψίας, την μη παραμετρική τεχνική bootstrap (Efron & Tibshirani, 1993).

Άλλα πολύ σημαντικά μοντέλα τα οποία είναι πολύ δημοφιλή στην βιβλιογραφία είναι τα μοντέλα παλινδρόμησης. Ένα από τα βασικότερα αυτής της κατηγορίας είναι το λογισμικό Early Checker, το οποίο αναπτύχθηκε από την εταιρία International Software Benchmarking Standards Group (ISBSG) (Mittas et al., 2015a).

Αργότερα αναπτύχθηκαν μέθοδοι, που στηρίζονται στα επιστημονικά πεδία της Τεχνικής νοημοσύνης, της αναγνώρισης προτύπων και της εξόρυξης δεδομένων. Μια από αυτές είναι η μέθοδος Χαρακτηριστικών Καμπυλών Σφάλματος Παλινδρόμησης (Regression Error Characteristic Curves-REC) (Mittas & Angelis, 2008a).

Παράλληλα άλλοι ερευνητές επικεντρώθηκαν στην καθιέρωση κανόνων, οι οποίοι προσανατολίζονται στην αξιοποίηση κατάλληλων ελέγχων στατιστικής υπόθεσης, για την σύγκριση διαφορετικών μοντέλων (Kitchenham & Mendes, 2009).

Η εκτενής ανάλυση των παραπάνω μοντέλων θα πραγματοποιηθεί στο Κεφάλαιο 2.

## 1.4 Συνεισφορά Της Παρούσας Διατριβής

Στην παρούσα μεταπτυχιακή διατριβή επιδιώκεται η ανάλυση του περισπούδαστου θέματος της πρόβλεψης του κόστους λογισμικού. Η μεθοδολογία που θα εφαρμοστεί είναι αυτή της ανάλυσης πηγών και ερευνητικών έργων από προηγούμενους ερευνητές, η οποία θα συνδυαστεί με την πειραματική έρευνα. Αυτό προσεγγίζεται μέσω μιας εκτενούς μελέτης αφενός των παραμετρικών και αφετέρου των μη-παραμετρικών μοντέλων πρόβλεψης, με απώτερο σκοπό τον συγκερασμό των καλών χαρακτηριστικών τους, για την δημιουργία ενός ημι-παραμετρικού μοντέλου με μεγάλη ακρίβεια στην πρόβλεψη του κόστους.

Στα πλαίσια αυτού του συγκερασμού, θα πραγματοποιηθεί πειραματική έρευνα σε έναν κατάλληλα σχεδιασμένο αλγόριθμο, ο οποίος θα συνδυάζει τις προαναφερόμενες οικογένειες μεθόδων σε ένα υβριδικό ημι-παραμετρικό μοντέλο πρόβλεψης. Από αυτό με κατάλληλους μετασχηματισμούς των μη-παραμετρικών μοντέλων πρόβλεψης που θα μελετηθούν θα προκύψουν και τα ανάλογα ημι-παραμετρικά. Παράλληλα, θα

χρησιμοποιηθούν κατάλληλα μέτρα ακριβείας για την αξιολόγηση των προβλέψεων που αποδίδονται από τα υβριδικά αυτό μοντέλα σε σχέση με αυτές που προβλέπονται από τα μη-παραμετρικά μοντέλα.

Εν συνεχεία, θα προβούμε στην ανάπτυξη ενός αυτοματοποιημένο εργαλείου κατασκευής που θα ενσωματώνει τα μοντέλα πρόβλεψης στην περιοχή της ΕΚΛ, το οποίο θα εξυπηρετεί στην ερμηνεία των αποτελεσμάτων των διαφόρων μεθόδων και στην συνέχεια στην λήψη αποφάσεων για την επιλογή της κατάλληλης στρατηγικής που θα πρέπει να ακολουθηθεί. Στην ουσία, στόχος είναι η υιοθέτηση ελεύθερου λογισμικού και αυτοματοποιημένων εργαλείων στην περιοχή του Software Cost Estimation.

## 1.5 Δομή της Μεταπτυχιακής Διατριβής

Η διπλωματική εργασία αποτελείται από 8 κεφάλαια τα οποία περιγράφονται στην συνέχεια.

Το Πρώτο Κεφάλαιο αποτελεί μια μορφή παρουσίασης του προβλήματος της ΕΚΛ. Πραγματοποιείται αναφορά στην κρισιμότητα που φέρει η εκτίμηση στο κόστους λογισμικού για να υπάρξει η συνέχεια στην πορεία ενός έργου και γίνεται αναφορά στα προβλήματα που μπορούν να συμβάλουν στην διαδικασία της εκτίμησης. Εν συνεχεία ακολουθεί μια ιστορική αναδρομή, ο σκοπός της οποίας είναι να αναφερθούν ποια μοντέλα έχουν χρησιμοποιηθεί από την επιστημονική κοινότητα για την προσπάθεια της ΕΚΛ. Τέλος, το πρώτο κεφάλαιο κλείνει με την συνεισφορά και την δομή της διπλωματικής εργασίας.

Στο Δεύτερο Κεφάλαιο παρατίθεται μια εκτενής παρουσίαση που αφορά το πρόβλημα της ΕΚΛ και στο ποια και πόσο μεγάλα μπορεί να είναι τα αποτελέσματα αυτού στην ανάπτυξη ενός έργου. Αμέσως μετά ακολουθεί η περιγραφή των μεθόδων ΕΚΛ που χρησιμοποιούνται μέχρι και σήμερα.

Στο Τρίτο Κεφάλαιο αναλύονται τα ημι-παραμετρικά μοντέλα, τα οποία αποτελούν την σύγκλισης των παραμετρικών και μη-παραμετρικών μοντέλων, λαμβάνοντας υπόψη τα καλύτερα χαρακτηριστικά αυτών. Τέλος ακολουθεί η περιγραφή της ημι-παραμετρικής στατιστικής μεθοδολογίας LSEbA (Least Squares Estimation by Analogy). Η οποία

χρησιμοποιήθηκε για την μοντελοποίηση του κόστους των έργων και συνδυάζει τα πλεονεκτήματα της παραμετρικής Ανάλυσης Παλινδρόμησης και της μη-παραμετρικής με Αναλογίες.

Στο Τέταρτο Κεφάλαιο αναλύεται με λεπτομέρεια τα μέτρα ακριβείας τα οποία χρησιμοποιήθηκαν για την επικύρωση των μοντέλων ΕΚΛ. Ακολουθεί η περιγραφή της διαδικασίας βαθμολόγησης και συσταδοποίησης των μοντέλων πρόβλεψης ΕΚΛ.

Στο Πέμπτο Κεφάλαιο γίνεται μια περιγραφή της γλώσσας R και των βασικών στοιχείων που την χαρακτηρίζουν. Εν συνεχεία παρατίθεται μια παρουσίαση του περιβάλλοντος και των δυνατοτήτων που αυτό μας παρέχει. Τέλος, αναλύονται οι απαιτήσεις που έχει η εφαρμογή και πως αυτές καλύφθηκαν.

Στο Έκτο Κεφάλαιο πραγματοποιείται η παρουσίαση της εφαρμογής που υλοποιήθηκε στα πλαίσια της εργασίας της παρούσας Διατριβής. Πιο συγκεκριμένα παρατίθενται μια ανάλυση των κυριότερων σημείων αυτής και των διαδικασιών μέσα από κάποια στιγμιότυπα οθόνης.

Στο Έβδομο Κεφάλαιο γίνεται η παρουσίαση της πειραματικής διαδικασίας που ακολουθήθηκε. Πραγματοποιείται μια αναφορά στα σύνολα δεδομένων, στα μοντέλα καθώς και τα μέτρα ακριβείας που χρησιμοποιήθηκαν για την πραγματοποίηση των συγκρίσεων καταλληλότητας και ακρίβειας των υπό μελέτη μοντέλων. Στην συνέχεια παρατίθενται τα αποτελέσματα της συσταδοποίησης όπως διαμορφώθηκαν από τον αλγόριθμο Scott-Knott και ακολουθεί ένας σχολιασμός αυτών.

Στο τελευταίο Όγδοο Κεφάλαιο παρουσιάζονται τα συμπεράσματα που προέκυψαν από την συγκεκριμένη διπλωματική εργασία τόσο από την θεωρητική, όσο και από τη πειραματική διαδικασία που ακολουθήθηκε. Παράλληλα προτείνονται κάποιες ιδέες που αφορούν την περαιτέρω μελέτη των ημι-παραμετρικών μοντέλων, όπως και την επέκταση της εφαρμογής.

# Κεφάλαιο 2
# Μέθοδοι Εκτίμησης Κόστους Λογισμικού

Στο δεύτερο κεφάλαιο της παρούσας Διατριβής γίνεται αναφορά στις μεθόδους ΕΚΛ με μια εκτενή παρουσίαση που αφορά το πρόβλημα της, ρίχνοντας περισσότερο βάρος στο να προσδιοριστεί επαρκώς το κόστος για την υλοποίηση ενός έργου Λογισμικού και στα προβλήματα που μπορεί να προκύψουν οποιαδήποτε στιγμή κατά τον υπολογισμό αυτού. Στην συνέχεια ακολουθεί μια περιγραφή των μεθόδων ΕΚΛ που έχουν χρησιμοποιηθεί τις τελευταίες δεκαετίες από την επιστημονική κοινότητα. Το κεφάλαιο ολοκληρώνεται με την ανάλυση των βασικότερων χαρακτηριστικών ορισμένων συνόλων που χρησιμοποιήθηκαν την πειραματική διαδικασία.

## 2.1 Μέθοδοι ΕΚΛ

Η ΕΚΛ εμπεριέχει την διαδικασία της δημιουργίας και της εφαρμογής ενός κατάλληλου μοντέλου ώστε να εκτιμηθούν οι πόροι που απαιτούνται για να αναπτυχθεί ένα πλήρως λειτουργικό σύστημα λογισμικού. Ένας από τους βασικούς παράγοντες είναι η ανθρώπινη προσπάθεια. Μια κατηγορία διαδεδομένων μοντέλων εκτίμησης κόστους λογισμικού χρησιμοποιεί κατά κύριο λόγω φόρμουλες για να προβλέψει τους ανθρωπομήνες που απαιτούνται για την ανάπτυξη ενός συστήματος, εν τούτοις η άυλη και πολυεπίπεδη φύση του λογισμικού όπως και το γεγονός ότι κάθε ένα έργο είναι παραμετροποιημένο για τις ανάγκες ενός συγκεκριμένου πελάτη ή οργανισμού καθιστούν την προσπάθεια της (εκτίμησης) δύσκολη. Ο συνηθισμένος τρόπος πρόβλεψης είναι κατά κύριο λόγω προσανατολισμένος σε εμπειρικές μεθόδους, όπου χρησιμοποιούνται μετρικές όπως τα Function Points (FP), που περιγράφουν τις λειτουργίες του λογισμικού σε όρους εισόδων, εξόδων, πληροφοριών, αρχείων και εξωτερικών διεπαφών. Σε αυτά τα εμπειρικά μοντέλα η συσχέτιση μεταξύ της

προσπάθειας και της λειτουργικότητας ενός συστήματος περιγράφεται συνήθως με τρόπο μη-γραμμικό.

Ακόμη λοιπόν, παρόλο που αυτά τα μοντέλα έχουν χρησιμοποιηθεί κατά κόρων από την ερευνητική κοινότητα, αλλά και από τον εταιρικό κόσμο θα λέγαμε ότι τα αποτελέσματα τους δεν είναι πάντα τα βέλτιστα (Mittas et al., 2015b).

Τα τελευταία χρόνια έχουν πραγματοποιηθεί πληθώρα ερευνών για την αξιολόγηση διάφορων τεχνικών που αποσκοπούν στην βελτίωση του σχεδιασμού των έργων λογισμικού, την κατανομή των ανθρωπίνων πόρων, τον σχεδιασμό και την ποιότητα του τελικού προϊόντος. Το βασικό ερώτημα το οποίο επιδιώκεται να απαντηθεί άπτεται στην επιλογή της καλύτερης μεθόδου πρόβλεψης κάτω από συγκεκριμένες συνθήκες.

Δύο είναι οι κύριες κατηγορίες μοντέλων στις οποίες έχει επικεντρωθεί η επιστημονική ερευνητική κοινότητα αυτή των παραμετρικών μοντέλων και αυτή των μη-παραμετρικών μοντέλων.

## 2.2 Παραμετρικά Μοντέλα

Τα παραμετρικά μοντέλα ΕΚΛ είναι χρήσιμα εργαλεία για τον πρώιμο υπολογισμό εκτιμήσεων σε έργα όπου υφίστανται αρχικά ελάχιστες τεχνικές λεπτομέρειες. Αυτό συμβαίνει διότι μπορεί να μην υπάρχουν επαρκή στοιχεία ώστε να υποστηριχθούν άλλες πιο λεπτομερείς μέθοδοι ΕΚΛ (Dysert, 2008).

Ένα παραμετρικό μοντέλο είναι μια μαθηματική αναπαράσταση των σχέσεων που διέπουν το εκτιμώμενο κόστος ενός υπό ανάπτυξη έργου, βάσει των φυσικών ή των λειτουργικών χαρακτηριστικών του. Πιο συγκεκριμένα αποτελείται από συναρτήσεις που παρέχουν λογικές και επαναλαμβανόμενες συσχετίσεις μεταξύ των ανεξάρτητων μεταβλητών, που μπορεί να είναι κάποιες παράμετροι σχεδίασης ενός έργου λογισμικού και την εξαρτημένη μεταβλητή, το κόστος. Η είσοδος που δίδεται συνήθως στις συναρτήσεις των μοντέλων αυτής της κατηγορίας μπορεί να είναι είτε Γραμμές Πηγαίου Κώδικα (SLOC), είτε Function Points ή κάποιο άλλο μέγεθος.

Στην συνέχεια παρατίθενται τα κυριότερα μοντέλα αυτής της κατηγορίας.

### 2.2.1 Μέθοδος COCOMO

Το Δομικό Μοντέλο Κόστους ή αλλιώς γνωστό ως μέθοδος COCOMO (Constructive Cost Model) είναι ένα παραμετρικό μοντέλο εκτίμησης κόστους λογισμικού που αναπτύχθηκε από τον Δρ. Barry W. Boehm και δημοσιεύτηκε για πρώτη φορά το 1981 στο βιβλίο Software Engineering Economics (Boehm, 1981). Θα λέγαμε πως είναι ευρέως γνωστό και αποτελεί τον κυριότερο εκφραστή των αλγοριθμικών μοντέλων. Στηρίζεται στην μελέτη πολλών έργων πληροφορικής και χρησιμοποιείται από πολλούς διοικητές έργων λογισμικού σε όλο τον κόσμο (Boehm et al., 2000a).

Από πλευράς υλοποίησης αξιοποιεί έναν βασικό τύπο παλινδρόμησης με παραμέτρους που προκύπτουν από ιστορικά στοιχεία του έργου αλλά και τα τρέχοντα, όπως και επίσης και τα εν δυνάμει μελλοντικά χαρακτηριστικά του έργου. Επί της ουσίας το εν λόγω εμπειρικό μοντέλο υπολογίζει μία εκτίμηση της διάρκειας και του κόστους ενός έργου λογισμικού, βασιζόμενο στο μέγεθος του προϊόντος και την ποιότητα της ομάδος ανάπτυξης. Σημαντικό στοιχείο αποτελεί το γεγονός ότι έχει μεγάλη ανεξαρτησία και δε δεσμεύεται από κάποιο συγκεκριμένο προμηθευτή λογισμικού.

Οι πρώτες αναφορές σε αυτό το μοντέλο εμφανίστηκαν με τον όρο COCOMO 81. Εν τούτοις το 1995 αναπτύχθηκε το COCOMO II και τελικά δημοσιεύθηκε το 2000 στο βιβλίο Software Cost Estimation (Heemstra, 1992). Αυτό αποτέλεσε και τον διάδοχο του COCOMO 81 και θεωρείται καταλληλότερο για την εκτίμηση σύγχρονων έργων ανάπτυξης λογισμικού. Γενικότερα θα λέγαμε πως παρέχει μεγαλύτερη στήριξη για τις σύγχρονες μεθόδους ανάπτυξης λογισμικού και μια ενημερωμένη βάση δεδομένων του έργου.

Αξίζει να σημειωθεί πως η ακρίβεια της τεχνικής COCOMO81 εξαρτάται από το είδος των έργων που χρησιμοποιήθηκαν για τη βαθμονόμηση (calibration) των παραμέτρων του μοντέλου. Ωστόσο ένα βασικό μειονέκτημά της, σχετίζεται με το γεγονός ότι για τον υπολογισμό του κόστους λογισμικού, απαιτείται η γνώση του πλήθους των γραμμών κώδικα (Lines of Code–LOC), στοιχείο που δεν είναι γνωστό εκ των προτέρων στις αρχικές φάσεις ανάπτυξης έργων λογισμικού. Την αδυναμία αυτή κάλυψε το μοντέλο COCOMO II, το οποίο αναπτύχθηκε μεταγενέστερα στα τέλη της δεκαετίας του '90.

Τρεις είναι οι εκδόσεις του μοντέλου, το βασικό, το ενδιάμεσο και το λεπτομερειακό με κάθε να έχει τις δικές τις παραμέτρους.

Το αποτέλεσμα του μοντέλου είναι να εκτιμηθεί η προσπάθεια και η διάρκεια ενός έργου, στηριζόμενο σε εισόδους σχετικές με το μέγεθος των συστημάτων που θα δημιουργηθούν αλλά και σε ορισμένους συντελεστές κόστους που επηρεάζουν την παραγωγικότητα. Ο βασικότερος υπολογισμός του Δομικού Μοντέλου είναι η χρήση της Συνάρτησης προσπάθειας (Effort Equation) για την εκτίμηση του αριθμού των Ανθρωπομηνών που απαιτούνται για να αναπτυχθεί ένα έργο.

### 2.2.2 Μέθοδος COCOMO II

Αποτελεί την νεότερη έκδοση του πρώτου μοντέλου COCOMO, το κύριο χαρακτηριστικό του COCOMO II είναι ότι εφαρμόζει τύπους σε τρία στάδια για να υπολογίσει την προσπάθεια, το χρονοδιάγραμμα και το κόστος που απαιτείται για την ανάπτυξη ενός προϊόντος λογισμικού. Το πρώτο στάδιο υποστηρίζει την κοστολόγηση των προσπαθειών διαμόρφωσης πρωτοτύπου ή σύνθεσης εφαρμογών. Το δεύτερο στάδιο υποστηρίζει την κοστολόγηση στο αρχικό στάδιο του σχεδιασμού ενός έργου όταν ακόμη δεν είναι γνωστά πολλά για τους οδηγούς κόστους του. Το τρίτο στάδιο υποστηρίζει την κοστολόγηση στην μετα-αρχιτεκτονική φάση ενός έργου.

Ο υπεύθυνος ανάπτυξης ενός έργου, μπορεί να αναπτύσσει μοντέλα τόσο πριν όσο και κατά τη διάρκεια που αναπτύσσεται το έργο προκειμένου να μπορεί, να εντοπίσει πιθανά προβλήματα στους πόρους του έργου, το προσωπικό ανάπτυξης, τον προϋπολογισμό και το χρονοδιάγραμμα. Το COCOMO II παρέχει ανάλυση της προσπάθειας και του χρονοδιαγράμματος σε λογισμικές φάσεις του κύκλου ζωής και ενέργειες από το αρχικό εγχειρίδιο του COCOMO (Boehm et al., 2000a).

### 2.2.3 Μέθοδος SLIM

Στα τέλη της δεκαετίας του 1970, αναπτύχθηκε η παραμετρική μέθοδος εκτίμησης κόστους λογισμικού SLIM (Software Life-cycle Model) (Putnam & Myers, 1992).

Χρησιμοποιώντας την κατανομή Rayleigh για να αναλύει τον κύκλο ζωής λογισμικού, με απώτερο σκοπό την εκτίμηση της προσπάθειας και τον ρυθμό εμφάνισης ελαττωμάτων (error defect rate).

Η εκτίμηση του κόστους λογισμικού, εξαρτάται και σε αυτήν την μέθοδο αντιστοίχως από τον όγκο των γραμμών του κώδικα και αμέσως μετά γίνεται χρήση του μοντέλου Rayleigh, με βάση το οποίο πραγματοποιούνται τροποποιήσεις, ώστε να μπορέσει να εκτιμηθεί η παραγόμενη προσπάθεια (Effort) (Putnam & Myers, 1992).

### 2.2.4 Μέθοδος Ανάλυσης Λειτουργικών Σημείων (FPA)

Η μέθοδος ανάλυσης λειτουργικών σημείων, αναπτύχθηκε από την IBM ως μια μέθοδος μέτρησης της παραγωγικότητας στα έργα ανάπτυξης λογισμικού. Θεωρήθηκε μια εναλλακτική μετρική, σε σχέση με τις μεθόδους που καταμετρούν τις γραμμές κώδικα και είναι ανεξάρτητη από την γλώσσα προγραμματισμού που χρησιμοποιείται για την ανάπτυξη ενός έργου (Albrecht & Gaffney, 1983). Η αρχή της είναι απλή και βασίζεται στην καταγραφή του αριθμού των λειτουργιών που θα πρέπει να πραγματοποιεί το λογισμικό. Οι λειτουργίες αυτές βασίζονται στους τύπους των δεδομένων που χρησιμοποιεί και παράγει.

Τα πέντε βασικά συστατικά (components) του υπό αξιολόγηση λογισμικού είναι τα παρακάτω:

α. Είσοδοι (External Inputs)
β. Έξοδοι (External Outputs)
γ. Επερωτήσεις (External Inquiries)
δ. Εσωτερικά Λογικά Αρχεία (Internal Logical File)
ε. Εξωτερικά Αρχεία Διεπαφής (External Interface Files)

### 2.2.5 Μέθοδος SPQR-20

Το ακρωνύμιο της μεθόδου SPQR (Software Productivity, Quality and Reliability), προκύπτει από τους όρους Παραγωγικότητα, Ποιότητα και Αξιοπιστία (Jones, 1986). Ο βασικός ισχυρισμός κατά την δημιουργία της, είναι πως είναι εφαρμόσιμη για όλα τα είδη έργων ανάπτυξης λογισμικού. Αυτό καθώς παρέχει την δυνατότητα εκτιμήσεων

και ως προς την διάρκεια και το κόστος ανάπτυξης αλλά και δίνει δυνατότητες εκτίμησης του κόστους συντήρησης.

Η SPQR αξιοποιεί το FPA (ανάλυση λειτουργικών σημείων) για να προσδιορίσει την τάξη μεγέθους ενός έργου. Η μέθοδος βασίζεται στην διατήρηση μιας εκτεταμένης βάσης δεδομένων με προηγούμενα έργα λογισμικού. Υπάρχουν τέσσερις εκδόσεις αυτής της μεθόδου η SPQR 10, 20, 50 και 100. Η SPQR-20 είναι η μόνη εμπορική διαθέσιμη εκδοχή (Wittig & Finnie, 1997).

### 2.2.6 Παραμετρικά Μοντέλα Μηχανικής Μάθησης

Τα τελευταία χρόνια οι ερευνητές που ασχολούνται με τον τομέα ΕΚΛ, έχουν επικεντρωθεί στην αξιοποίηση και αξιολόγηση παραμετρικών μοντέλων που βασίζονται στην μηχανική μάθηση (Machine learning). Η μηχανική μάθηση αποτελεί ένα ερευνητικό πεδίο της επιστήμης των υπολογιστών, που αναπτύχθηκε από την μελέτη της αναγνώρισης προτύπων και στην υπολογιστική θεωρία μάθησης στην τεχνητή νοημοσύνη.

Θα μπορούσαμε να συνοψίσουμε την μηχανική εκμάθηση ως μια συνάρτηση εκμάθησης F, η οποία συσχετίζει ένα σύνολο μεταβλητών εισόδου X με την μεταβλητή εξόδου Y (Brownlee, 2016).

$$Y = f(x) \quad \text{(Εξ. 2.1)}$$

Η μορφή της συνάρτησης αυτής είναι άγνωστη, επομένως οι ερευνητές του εν λόγω πεδίου αξιολογούν ένας πλήθος από διαφορετικούς αλγορίθμους διαφορετικής μάθησης, ώστε να ανακαλύψουν ποιος είναι ο βέλτιστος και που προσεγγίζει καλύτερα την συνάρτηση με γνώμονα τα προσδοκώμενα αποτελέσματα.

Τα οφέλη των παραμετρικών μοντέλων μηχανικής εκμάθησης:

Απλά: Αυτές οι μέθοδοι είναι πιο εύκολες στην κατανόηση και στην ερμηνεία των αποτελεσμάτων.
Ταχύτατα: Μαθαίνουν πολύ γρήγορα από τα δεδομένα τα οποία τροφοδοτούνται.

Λιγότερα δεδομένα: Δεν απαιτούν περισσότερα δεδομένα εκπαίδευσης και μπορούν να λειτουργήσουν σωστά, ακόμη και αν η προσαρμογή στα δεδομένα δεν είναι τέλεια.

Οι περιορισμοί των παραμετρικών μοντέλων μηχανικής εκμάθησης:

- Προβλήματα: Η επιλογή μιας συγκεκριμένης μορφής συνάρτησης μας περιορίζει να εργαστούμε μόνο πάνω σε αυτήν.

-Περιορισμένη Πολυπλοκότητα: Οι μέθοδοι είναι καταλληλότεροι για απλούστερα προβλήματα και όχι για σύνθετα.

### 2.2.7 Παλινδρόμηση Ελαχίστων τετραγώνων (Ordinary Least Squares)

Η παλινδρόμηση Ελαχίστων τετραγώνων είναι μία μέθοδος που στηρίζεται στην γενική γραμμική παλινδρόμηση της κλασσικής στατιστικής. Είναι μία σχετικά εννοιολογικά και υπολογιστικά απλή και εύχρηστη μέθοδος που συναντά κανείς σε δημοφιλή πακέτα λογισμικού συμπεριλαμβανομένης της R. Μέχρι σήμερα έχει μελετηθεί εκτενώς από πληθώρα ερευνητών (Mittas, 2011) και (Hayes & Cai, 2008).

Η εν λόγω μέθοδος εξηγεί την συσχέτιση μεταξύ μερικών ανεξάρτητων μεταβλητών που είναι οι παράγοντες του κόστους και μίας εξαρτημένης μεταβλητής που είναι η προσπάθεια (effort) με την μορφή μίας παραμετρικής γραμμικής σχέσης. Καθώς οι μεταβλητές συνήθως δεν ακολουθούν την κανονική κατανομή, απαιτείται κάποιος λογαριθμικός μετασχηματισμός ώστε να καταλήξουμε σε ένα έγκυρο γραμμικό μοντέλο. Επιπρόσθετα, για να καταφέρουμε να διαχειριστούμε ανάμικτα δεδομένα με κατηγορικές και συνεχείς μεταβλητές, αντικαθιστούμε τις κατηγορικές μεταβλητές με μεταβλητές δυαδικής φύσεως.

Ένα μαθηματικό μοντέλο που χρησιμοποιεί την μέθοδο OLS μπορεί να διατυπωθεί ως εξής:

$$Y_1 = B_1 + B_2 x'_{12} + \ldots + B_K x'_{ik} + e_t \quad \text{(Εξ. 2.2)}$$

όπου $x_{12}$...$x_{ik}$ είναι οι ανεξάρτητες μεταβλητές (ή regressors) που συμβάλλουν στην εκτίμηση της τιμής της παρατήρησης, οι $\beta_2$,...$\beta_κ$ είναι οι συντελεστές απόκρισης, η $\beta_1$ είναι ο σταθερός όρος ή αλλιώς intercept και η $y_1$ είναι η εξαρτημένη μεταβλητή. Επίσης υπάρχει και ένας όρος σφάλματος που καθορίζεται απλό τον συντελεστή $e_t$, όπου είναι

μία τυχαία μεταβλητή με κατανομή πιθανότητας που συνήθως ακολουθεί την κανονική κατανομή.

Η μέθοδος OLS λειτουργεί υπολογίζοντας τους συντελεστές απόκρισης και τον όρο intercept, ελαχιστοποιώντας τα ελάχιστα τετράγωνα ri^2. Η τιμή ri προκύπτει από την διαφορά της παρατηρηθείσας με την εκτιμώμενη τιμή για την ν-οστή παρατήρηση. Ως εκ τούτου όλες οι παρατηρήσεις έχουν ισοδύναμη επιρροή στην εξίσωση της παλινδρόμησης και έτσι οι έκτοπες τιμές (outliers) ασκούν αρνητική επίδραση στο μοντέλο.

Τέλος αξίζει να σημειωθεί πως μία παραλλαγή της OLS χρησιμοποιήθηκε για την αξιολόγηση του μοντέλου COCOMO II.

### 2.2.8 Η Εύρωστη Παλινδρόμηση (Robust OLS)

Η εύρωστη παλινδρόμηση αποτελεί μία βελτιωμένη εκδοχή της παλινδρόμησης ελαχίστων τετραγώνων. Το βασικό της χαρακτηριστικό είναι πως εξαλείφει ένα βασικό πρόβλημα που συναντάται στην τελευταία, τις έκτοπες τιμές (outliers). Είναι σύνηθες κατά την εκτίμηση έργων λογισμικού να εμφανίζονται τέτοιες τιμές καθώς μπορεί να υπάρχει έλλειψη σε ποσοτικά στοιχεία από ανεπαρκείς μετρικές και ελάχιστα ιστορικά δεδομένα. Αξίζει αν σημειωθεί πως η συγκεκριμένη μέθοδος μπορεί να χρησιμοποιηθεί σε μοντέλα εκτίμησης λογισμικού που υφίσταται μικρό πλήθος ανεξάρτητων μεταβλητών (Miyazaki, 1994).

Μία προσέγγιση που ανήκει στην κατηγορία της εύρωστης παλινδρόμησης είναι μία τεχνική που χρησιμοποιεί τα σημεία που βρίσκονται ανάμεσα σε δύο ή τρείς τυπικές αποκλίσεις της μέσης τιμής της εξαρτημένης μεταβλητής. Η εν λόγω μέθοδος ξεπερνά το πρόβλημα με τις έκτοπες τιμές. Ωστόσο μπορεί να αξιοποιηθεί μόνο αν υπάρχει ένα ικανοποιητικό μέγεθος του δείγματος, ώστε να μην υπάρξει σημαντικός αντίκτυπος στους βαθμούς ελευθερίας του προτύπου.

Αξίζει να σημειωθεί, πως τα περισσότερα μοντέλα που προαναφέρθηκαν στις προηγούμενες υπο-ενότητες χρησιμοποιούν κάποια μορφή μεθόδων παλινδρόμησης.

## 2.3 Μη-Παραμετρικά Μοντέλα

Τα μη–παραμετρικά μοντέλα περιλαμβάνουν τις τεχνικές ΕΚΛ, που βασίζονται σε ένα σύνολο από μεθόδους που εφαρμόζουν αρχές και μεθόδους που συναντώνται στο επιστημονικό πεδίο της τεχνητής νοημοσύνης. Τέτοιες είναι τα τεχνητά νευρωνικά δίκτυα (neural networks), η συλλογιστική βασισμένη στην αναλογική σκέψη (analogy based reasoning), τα δέντρα παλινδρόμησης (regression trees), οι γενετικοί αλγόριθμοι (γενετικοί αλγόριθμοι) και τα συστήματα κανόνων (rule based induction)(Poonam, 2013).

Το κύριο χαρακτηριστικό των μη-παραμετρικών μοντέλων είναι ότι αποφεύγουν να προβούν σε περιοριστικές υποθέσεις για την δομή που θα έχει η συνάρτηση, η οποία θα χρησιμοποιηθεί. Ως εκ τούτου δεν έχουν απόλυτες εξαρτήσεις σε συγκεκριμένες κατανομές δεδομένων και δεν προκαθορίζουν ρητά μια σταθερή δομή για το μοντέλο. Αυτό αποτελεί και ένα βασικό πλεονέκτημα σε σχέση με τα παραμετρικά μοντέλα, καθώς σε αυτή την κατηγορία μπορεί να δίδονται ως είσοδος δεδομένα με τιμές που έχουν μεικτή φύση, όπως επίσης και μεγαλύτερη ανοχή σε περίπτωση που έχουμε μεγάλες ποσότητες ελλειπόντων δεδομένων (Mittas et al., 2015b).

Στην συνέχεια παρατίθενται τα κυριότερα μοντέλα αυτής της κατηγορίας.

### 2.3.1 Μέθοδος Estimation by Analogy (EbA)

Μια από τις περισσότερο διαδεδομένες μεθόδους αυτής της κατηγορίας μοντέλων είναι η Εκτίμηση με Αναλογίες (Estimation by Analogy-EbA). Θεωρείται μια μέθοδος εκτίμησης κόστους που λαμβάνει υπόψη αφενός στοιχεία από την κρίση των ειδικών και αφετέρου αξιοποιεί αλγοριθμικά μοντέλα. Αποτελεί μια ολοκληρωμένη μεθοδολογία και σήμερα υπάρχουν πολλά διαθέσιμα εργαλεία για την αυτοματοποίηση της.

Η εν λόγω μέθοδος αποτελεί ομάδα τεχνικών, η οποία ανήκει στην ευρύτερη κατηγορία της Συλλογιστικής Βασισμένη σε Περιπτώσεις (Case Based Reasoning- CBR). Το βασικό προαπαιτούμενο για την εφαρμογή της μεθόδου EbA, είναι η ύπαρξη ιστορικών στοιχείων από παλαιότερα έργα λογισμικού. Επιπλέον η αρχή στην οποία βασίζεται είναι πως έργα με όμοια χαρακτηριστικά αναμένεται να έχουν και όμοιο κόστος.

Στην τεχνική EbA ακολουθούνται τα παρακάτω βήματα:

- Χρήση των χαρακτηριστικών όμοιων-παλαιότερων λογισμικών σε ένα σύνολο δεδομένων.
- Συλλογή των χαρακτηριστικών του υπό εκτίμηση λογισμικού που αντιστοιχούν με τα χαρακτηριστικά των έργων στο ιστορικό σύνολο δεδομένων.
- Εκτίμηση της ομοιότητας μεταξύ του νέου λογισμικού με τα όμοια-παλαιότερα λογισμικά.
- Πρόβλεψη του κόστους του νέου λογισμικού κάνοντας χρήση των κοντινότερων αναλογιών.

Μαθηματικά μοντελοποιείται ως μια τεχνική μη-παραμετρικής παλινδρόμησης που έχει την ακόλουθη μορφή: $Y_i = f(X_i) + \varepsilon_i$ $E(e_i X_i) = 0$ $(i = 1,\ldots n)$. Το βασικό πρόβλημα της μεθόδου είναι ότι πως εκδηλώνονται κάποιες δυσκολίες ως προς την θεωρητική μελέτη του σφάλματος πρόβλεψης (prediction error). Επομένως στο πρόσφατο παρελθόν προτάθηκε η χρήση τεχνικών bootstrap, ώστε να αποδίδονται ακριβέστερες προβλέψεις. Αυτό εξυπηρετεί την βαθμονόμηση (calibration) της μεθόδου και για την εύρεση διαστημάτων εμπιστοσύνης των προβλέψεων (Angelis & Stamelos, 2000).

### 2.3.2 Μέθοδος Τοπικά βεβαρημένη Παλινδρόμηση (LOES)

Η μέθοδος LOESS (Locally weighted regression) (Cleveland, 1979), είναι μια μη-παραμετρική τεχνική παλινδρόμησης, η οποία συνδυάζει πολλαπλά μοντέλα παλινδρόμησης σε ένα βασιζόμενο μετα-μοντέλο στους k-NN κοντινότερους γειτόνους. Σχετίζεται στενά με την EbA, κατασκευάζοντας μια συνάρτηση πρόβλεψης μέσω της γειτνίασης δεδομένων. Η κατασκευή της συνάρτησης διαφοροποιείται από αυτήν της EbA, καθώς η EbA αξιοποιεί ένα μέσο σε μια διαφορετική γειτονιά, που ορίζεται από τα έργα τον k-NN πλησιέστερων γειτόνων του νέου έργου (Mittas et al., 2015).

### 2.3.3 Μέθοδος Εμπειρικής πρόβλεψης (Empiric non-parametric estimation model)

Το βασικό χαρακτηριστικό των εμπειρικών μη-παραμετρικών μοντέλων εκτίμησης είναι ότι χρησιμοποιούν δεδομένα από έργα που δημιουργήθηκαν προγενέστερα. Ωστόσο η εκτίμηση δεν πραγματοποιείται εφαρμόζοντας μαθηματικές φόρμουλες, κατά

τους τρόπους που ακολουθούν άλλες προσεγγίσεις. Στα πλαίσια αυτά κάποιοι ερευνητές (Zivadinovic & Medic, 2011), ανέπτυξαν την τεχνική της βελτιστοποιημένης μείωσης ενός συνόλου (optimized set reduction - OSR), η οποία επιλέγει ένα υποσύνολο από έργα και βάση αυτών εκτιμά την παραγωγικότητα του νέου έργου. Η παραγωγικότητα ορίζεται ως η προσπάθεια σε ανθρωπομήνες διαιρούμενη με τον αριθμό γραμμών κώδικα. Τα έργα ομαδοποιούνται σε ένα βέλτιστο υποσύνολο, έχοντας όμοιους συντελεστές κόστους , όπως το νέο υπό ανάπτυξη έργο.

Η OSR λαμβάνει υπόψη τις τιμές των παραγόντων κόστους, κατά τέτοιο τρόπο ώστε η κατανομή της παραγωγικότητας, στα πλαίσια του υποσυνόλου των επιλεγμένων έργων, να είναι καλή βάση των στατιστικών κριτηρίων. Η κατανομή της πιθανότητας σε σχέση με την παραγωγικότητα απορρέει από την κατανομή της συχνότητας των επιλεγμένων έργων, πάνω από τον όγκο του διαστήματος της παραγωγικότητας. Η παραγωγικότητα σε σχέση με το νέο έργο εκτιμάται υπολογίζοντας την αναμενόμενη τιμή βάση των πιθανοκρατικών κατανομών.

Οι ερευνητές συνέκριναν την ακρίβεια της τεχνικής OSR με το παραμετρικό μοντέλο COCOMO και κατέληξαν σε αποτελέσματα, που καταδεικνύουν ότι το πρώτο έχει μεγαλύτερη ακρίβεια.

### 2.3.4 Δέντρα ταξινόμησης και παλινδρόμησης (Classification and Regression Trees - CART)

Το CART είναι μέρος ενός συνόλου μιας κλάσης μη-παραμετρικών και μη-γραμμικών μοντέλων πρόβλεψης, τα οποία μπορούν να διαχωριστούν σε δύο βασικές κατηγορίες: τα δέντρα ταξινόμησης (Classification trees) και στα δέντρα παλινδρόμησης (Regression trees). Τα CART μοιράζουν τα δεδομένα σε δύο κατηγορίες-υποσύνολα, έτσι ώστε οι καταγραφές εντός των υποσυνόλων να είναι περισσότερο ομοιογενείς μεταξύ τους από ότι βρίσκονταν στο αρχικό σύνολο. Αποτελεί μια επαναληπτική (recursive) διαδικασία, καθώς καθένα από τα δύο υποσύνολα διασπάται ξανά και ξανά, μέχρι να βρεθεί κάποιο κριτήριο τερματισμού της διαδικασίας.

Η μεθοδολογία CART βασίζεται σε τρία βήματα. Στο πρώτο βήμα δημιουργείται το πλήρες δέντρο, με την χρήση διαδοχικών ερωτήσεων σχετικά με τις τιμές κάθε μεταβλητής χωριστά. Σε αυτό το βήμα το βασικό πρόβλημα είναι το «overfitting». Στο

δεύτερο βήμα αντιμετωπίζεται το overfitting, με την διαδικασία του κλαδέματος (pruning) του πλήρους δέντρου σε έναν ικανοποιητικό βαθμό. Αποτέλεσμα αυτού του κλαδέματος είναι η παραγωγή πολλών λιγότερο πολύπλοκων δέντρων. Στο τρίτο βήμα με την χρήση της διαδικασίας cross-validation, γίνεται επιλογή του πιο βέλτιστου δέντρου (Shalizi, 1999).

### 2.3.5 Μέθοδος Τυχαίου Δάσους (Random Forest)

Τα τυχαία Δάση (Random Forests) είναι μία μέθοδος αλληλένδετη με την μέθοδο των δέντρων απόφασης/ταξινόμησης. Πιο συγκεκριμένα πρόκειται για μια συλλογή από Δέντρα Απόφασης (Decision Trees) (Quilan, 1986) και (Breiman, 2001).

Στην κατασκευή των δέντρων απόφασης γίνεται πρώτα η ανάθεση του συνόλου των δειγμάτων εκπαίδευσης στις ρίζες. Κάθε επιπλέον κόμβος περιλαμβάνει ένα υποσύνολο των δειγμάτων που μετά από κάποιο έλεγχο, διαχωρίζονται σε μικρότερα υποσύνολα (παιδιά). Τα μέτρα βάση των οποίων πραγματοποιείται ο διαχωρισμός μπορεί να είναι π.χ. η εντροπία, το Gini index κτλ. Τα Τυχαία Δέντρα αναπτύσσονται στο μέγιστο δυνατό βαθμό, χωρίς να υφίστανται κάποιου είδους κλαδέματος (no pruning).

Όσον αφορά το βαθμό λάθους που έχει η μέθοδος στην ταξινόμηση των δεδομένων και την πρόβλεψη των μελλοντικών αποτελεσμάτων (classification and prediction error rate), έχει αποδειχθεί (Quilan, 1986) ότι εξαρτάται από δύο συγκεκριμένες μεταβλητές την συσχέτιση μεταξύ δύο δέντρων (correlation) και την δύναμη κάθε δέντρου (strength). Για παρόμοια δέντρα η συσχέτιση είναι υψηλή και κατά συνέπεια τόσο μεγαλύτερο είναι και το error rate του δάσους. Η δύναμη κάθε δέντρου έχει να κάνει με το κατά πόσο καλός είναι ο ταξινομητής του κάθε δέντρου απόφασης. Εάν ένα δέντρο έχει μικρό error rate τότε αποτελεί έναν πολύ δυνατό ταξινομητή. Δηλαδή αν αυξηθεί η δύναμη των δέντρων μειώνεται το error rate του δάσους.

### 2.3.6 Μέθοδος Bagging

Η μέθοδος Bagging (Bootstrap aggregation) προτάθηκε από τον Leo Breiman το 1996 για την βελτίωση της ταξινόμησης συνδυάζοντας τις ταξινομήσεις τυχαίας παραγόμενων εκπαιδευτικών συνόλων (Breiman, 1996).

Η μέθοδος συγκέντρωσης Bootstrap είναι ένας μετα-αλγόριθμος μηχανικής μάθησης, ο οποίος έχει σχεδιαστεί για να βελτιώσει την σταθερότητα και την ακρίβεια των αλγορίθμων μηχανικής μάθησης που χρησιμοποιούνται στην στατιστική ταξινόμηση και παλινδρόμηση. Επίσης χρησιμοποιείται για την μείωση της διακύμανσης και βοηθάει στην αποφυγή της υπερφόρτωσης. Εφαρμόζεται συνήθως στις μεθόδους με δέντρα απόφασης, αλλά μπορεί να χρησιμοποιηθεί και με οποιοδήποτε μέθοδο. Η Bagging αποτελεί μια ειδική περίπτωση της προσέγγισης του μέσου όρου του μοντέλου.

Όταν το σύνολο εκπαίδευσης είναι μικρό, δεδομένου ότι το συνολικό πλήθος των παραδειγμάτων εκπαίδευσης είναι n, κάθε νέο σύνολο εκπαίδευσης παράγεται επιλέγοντας n φορές ένα παράδειγμα από το αρχικό σύνολο. Η επιλογή πραγματοποιείται με την επανατοποθέτηση σύμφωνα με την ομοιόμορφη κατανομή. Έτσι σε κάποια παραδείγματα εκπαίδευσης μπορεί κάποιες παρατηρήσεις μπορεί να εμφανίζονται πολλές φορές ή και καθόλου (Opitz & Maclin, 1999) και (Breiman, 1996). Κάθε ένα από τα νέα σύνολα μπορούν να χρησιμοποιηθούν ως είσοδο στον αλγόριθμο μάθησης. Έτσι δημιουργούνται ένα σύνολο, μια σειρά από διαφορετικά μοντέλα που χρησιμοποιούνται για να γίνει η πρόβλεψη της εξαρτημένης μεταβλητής κάνοντας χρήση και των επιμέρους προβλέψεων των μοντέλων.

Η Bagging οδηγεί σε "βελτιώσεις για ασταθείς διαδικασίες-ταξινομητές" (Breiman, 1996), οι οποίοι μεταβάλλονται αρκετά σε μικρές διαταραχές του συνόλου εκπαίδευσης (που περιλαμβάνουν τα τεχνητά νευρωνικά δίκτυα, τα δέντρα ταξινόμησης και παλινδρόμησης και την επιλογή υποομάδων σε γραμμική παλινδρόμηση). Από την άλλη πλευρά, μπορεί να υποβαθμίσει ήπια την απόδοση σταθερών μεθόδων όπως οι Κ-πλησιέστεροι γείτονες (Breiman, 1996).

## 2.4 Ανασκόπηση μεθόδων ΕΚΛ

Τις τελευταίες δεκαετίες όπως μελετήσαμε προτάθηκε και αναπτύχθηκε ένας μεγάλος αριθμός διαφορετικών μοντέλων και μεθόδων ΕΚΛ. Το γεγονός αυτό από μόνο του αναδεικνύει πως η επιστημονική κοινότητα έχει επικεντρωθεί στο να καλύψει την ανάγκη βελτίωσης της εκτίμησης κόστους στα σύγχρονα έργα λογισμικού. Ωστόσο ο κλάδος της ΕΚΛ, φαίνεται να είναι πλούσιος από παραδείγματα ημιτελούς συμπερασματολογίας και από προτάσεις για βέλτιστα μοντέλα, χωρίς να υπάρχουν

επαρκή στοιχεία για κάποια σε βάθος συγκριτική αποτίμηση. Έτσι πολλές φορές οι ερευνητές οδηγούνται σε μελέτες με αντιφατικά αποτελέσματα. Η έρευνα λοιπόν δε θα πρέπει να εστιάζεται μόνο στην προσπάθεια εύρεσης του «βέλτιστου» μοντέλου πρόβλεψης αλλά εξίσου καίριας σημασίας είναι η εξασφάλιση της αξιοπιστίας και της εγκυρότητας των ερευνών.

# Κεφάλαιο 3
# Ημι-Παραμετρικά Μοντέλα

Στο παρόν κεφάλαιο της παρούσας Διατριβής περιγράφεται μία προσέγγιση για την δημιουργία υβριδικών μεθόδων ΕΚΛ που συνδυάζουν τα πλεονεκτήματα που φέρουν τα μη-παραμετρικά και τα παραμετρικά μοντέλα που μελετήθηκαν ανωτέρω.

## 3.1 Ημι-Παραμετρικά Μοντέλα

Μέχρι πρότινος χρησιμοποιούνταν ξεχωριστά τα παραμετρικά και τα μη-παραμετρικά μοντέλα πρόβλεψης κόστους λογισμικού. Εν τούτοις σήμερα που οι ανάγκες και οι απαιτήσεις του πελάτη μεταβάλλονται με ραγδαίους ρυθμούς, καθίσταται επιτακτική η ανάγκη της σύγκλισης των δυο μοντέλων λαμβάνοντας υπόψη τα καλύτερα χαρακτηριστικά αυτών. Ο σκοπός της παρούσας διατριβής δεν είναι να αντικαταστήσει την χρήση των παραμετρικών ή την χρήση των μη-παραμετρικών προσεγγίσεων αλλά να χρησιμοποιήσει και να συγκεράσει τα καλύτερα χαρακτηριστικά των δύο κατηγοριών στα ημι-παραμετρικά μοντέλα (Mittas & Angelis, 2010). Αυτό μπορεί να πραγματοποιηθεί μέσω της εξέτασης των μέτρων ακρίβειας στα μοντέλων ΕΚΛ. Τα εν λόγω μέτρα αξιοποιούν την πραγματική ($Y_{Ai}$) καθώς και την εκτιμώμενη ($Y_{Ei}$) τιμή του κόστους ενός έργου για να προβούν στις απαραίτητες συγκρίσεις και να προσδιορίσουν το σχετικό σφάλμα (Miyazaki et al., 1994).

Σύμφωνα με τους Mittas & Aggelis (2015b) στις μεθόδους ημι-παραμετρικής παλινδρόμησης εμπεριέχονται τα μοντέλα παλινδρόμησης, που συνδυάζουν τα πλεονεκτήματα των παραμετρικών και μη-παραμετρικών μοντέλων. Αυτά βρίσκουν χρησιμότητα σε περιπτώσεις όπου τα μη-παραμετρικά μοντέλα δεν αποδίδουν τα μέγιστα ή σε περιπτώσεις όπου ο ερευνητής επιθυμεί την χρήση παραμετρικών μοντέλων. Αυτές οι προσεγγίσεις επεκτείνονται πέραν από την στατιστική και στο επιστημονικό πεδίο που πραγματεύεται την ΕΚΛ.

Τα μοντέλα αυτής της κατηγορίας, φαίνεται να είναι τα καταλληλότερα, ιδιαίτερα σε περιπτώσεις όπου υπάρχουν αποσπασματικές και ελλιπείς πληροφορίες σχετικά με τις υποκείμενες σχέσεις μεταξύ της εξαρτημένης και της ανεξάρτητης μεταβλητής. Επιπρόσθετα τα ημι-παραμετρικά μοντέλα μπορούν να αποτελέσουν την λύση στο πρόβλημα του εντοπισμού πολύπλοκων σχέσεων μεταξύ των μεταβλητών, όπου για τις οποίες, ένα προκαθορισμένο μοντέλο δεν είναι προφανές ή για περιπτώσεις όπου η χρήση μη-παραμετρικών μοντέλων αποτυγχάνει. Αξιοσημείωτο επίσης και εξίσου σημαντικό είναι το γεγονός ότι τα μη-παραμετρικά μοντέλα δεν αυξάνουν το επίπεδο της πολυπλοκότητας, αλλά και δεν φαίνεται να υπονομεύουν την δυνατότητα κατανόησης των παραμετρικών μοντέλων. Αυτά τα χαρακτηριστικά έχουν από μόνα τους ερευνητικό ενδιαφέρον και για τον λόγω αυτό σκοπεύουμε στην θεωρητική ανάλυση των κυριότερων μοντέλων αυτής της κατηγορίας που θα συνοδεύεται από πειραματική μελέτη με την χρήση της γλώσσας R και του μοντέλου Shiny. Ο συνδυασμός των παραμετρικών και των μη-παραμετρικών τεχνικών μπορεί να μειώσει τις αδυναμίες που αντιμετωπίζουν οι "κλασσικές μεθοδολογίες".

### 3.1.1 Μέθοδος Least Square – Περίπτωση της LSEbA

Όπως έχουμε ήδη αναφέρει, η μέθοδος των Ελαχίστων Τετραγώνων (Least Square) αποτελεί μία από τις πιο γνωστές μεθόδους ΕΚΛ για την κατασκευή ενός γραμμικού παραμετρικού μοντέλου με μία εξαρτημένη μεταβλητή που αναπαριστά το κόστος ενός έργου λογισμικού. Εναλλακτικά στο παρελθόν χρησιμοποιήθηκαν εκτενώς εμπειρικά μοντέλα όπως η εκτίμηση με αναλογίες (Eba-Estimation by Analogy). Αυτά χρησιμοποιώντας γνωρίσματα και ιστορικά στοιχεία από σύνολα δεδομένων (datasets) για όμοια έργα λογισμικού, μέσα από κάποια κριτήρια ομοιότητας καταλήγουν σε προβλέψεις για το κόστος μελλοντικών έργων που έχουν συνάφεια με αυτά που έχουν αναλυθεί.

Εντούτοις σε διάφορες εργασίες (Mittas & Aggelis, 2009; Mittas et al., 2015b) κατέδειξαν πως είναι δυνατόν να συνδυαστούν οι τεχνικές παραμετρικών και μη-παραμετρικών μοντέλων και να εκφραστούν με μαθηματικές σχέσεις. Αυτή είναι η υβριδική προσέγγιση που επιτρέπει την αξιοποίηση παραμετρικών και μη-παραμετρικών χαρακτηριστικών συνδυάζοντας κατάλληλα τα οφέλη που έχει η κάθε μία μέθοδος. Στην συνέχεια παραθέτουμε την μαθηματική περιγραφή της ημι-παραμετρικής κλάσης μοντέλων πρόβλεψης και ακολούθως θα προχωρήσουμε στην

περιγραφή του αλγορίθμου που αφορά τον συγκερασμό της μεθόδου Ελαχίστων Τετραγώνων με το μη-παραμετρικό μοντέλο εκτίμηση με αναλογίες (EbA). Αξίζει να σημειωθεί πως αναλόγως η ίδια μέθοδος μπορεί να γενικευτεί και να εφαρμοστεί σε οποιοδήποτε μη-παραμετρικό μοντέλο αξιοποιώντας τα ανάλογα χαρακτηρίστηκα που αυτό προσφέρει και δεν περιορίζεται αποκλειστικά σε αυτό που προαναφέραμε.

Σύμφωνα με τους Mittas & Aggelis (2009), έστω ότι έχουμε ως $Y_i$ το διάνυσμα που αποτελεί το κόστος ενός έργου λογισμικού i. Το $X_i = (X_{i1},...X_{ip})'$ είναι ένα διάνυσμα ανεξάρτητων μεταβλητών που συσχετίζονται γραμμικά με το $Y_i$ και το $T_i = (T_{i1},...T_{id})'$ ένα άλλο διάνυσμα μεταβλητών που δεν συσχετίζονται γραμμικά με την εξαρτημένη μεταβλητή $Y_i$. Με αυτά τα στοιχεία θα μπορούσαμε να κατασκευάσουμε ένα μοντέλο που θα διέπετε από την ακόλουθη εξίσωση

$$Y_i = X_i^{'}\beta + g(T_i) + e_i \quad (1), \qquad (\text{Εξ. } 3.1)$$

όπου β είναι ένα άγνωστο διάνυσμα από παραμέτρους (συντελεστές παλινδρόμησης) και g(X) είναι μια άγνωστη μη-γραμμική συνάρτηση του διανύσματος $T_i$. Τέλος, τα σφάλματα $\varepsilon_i$ θεωρούνται ασυσχέτιστα με την μηδενική μέση τιμή και την διακύμανση $\sigma^2$. Ο Robinson κατέδειξε ότι το μοντέλο αυτό θα μπορούσε να ξαναγραφτεί ως

$$Y_i - E(Y_i \mid T_i) = X_i^{'}\beta + g(T_i) + e_i \,(2), \quad (\text{Εξ. } 3.2)$$

ως μια απόπειρα εύρεσης του διανύσματος β σε δύο μόλις βήματα (Robinson, 1998) και (Anglin & Gencay, 1996):

1. Υπολογισμός των αγνώστων υπό συνθήκη μέσων τιμών $E(Y_i|T_i)$ και $E(X_i|T_i)$ από μία μη-παραμετρική μέθοδο.

2. Αντικατάσταση των αποτελεσμάτων του βήματος 1 στους αγνώστους τις εξίσωσης (2) και εφαρμογή της παλινδρόμησης συνήθων ελαχίστων τετραγώνων για τον υπολογισμό του β.

Λαμβάνοντας υπόψη μας τα παραπάνω θα μπορούσε να προσδιοριστεί ένας αλγόριθμος για τον υπολογισμό εκτίμησης κόστους, αξιοποιώντας αφενός την EbA και

αφετέρου υπολογίζοντας τα $E(Y_i|T_i)$ και $E(X_i|T_i)$ και εφαρμόζοντας την παραμετρική μέθοδο LS για την εκτίμηση του β.

Βήμα 1: Καθορισμός των ανεξάρτητων μεταβλητών στο σύνολο δεδομένων που είναι γραμμικά συσχετισμένες με την ανεξάρτητη μεταβλητή $Y_i$. Από αυτές θα σχηματιστεί το σύνολο LS-set ενώ οι υπόλοιπες θα σχηματίσουν το EbA-Set.

Βήμα 2: Καθορισμός του διανύσματος $X_i = (X_{i1},...X_{ip})'$, το οποίο θα αντιπροσωπεύει το σύνολο LS-set και του διανύσματος $T_i=(T_{i1},...,T_{id})'$ που θα αντιπροσωπεύει το EbA-Set. Ο σκοπός μας είναι για ένα νέο έργο λογισμικού να υπολογίσουμε το εκτιμώμενο κόστος $Y_{new}$ το οποίο θα απορρέει από τα $(X_{new},T_{new})$.

Βήμα 3: Για το νέο έργο εφαρμόζουμε την EbA, όπου αναζητούμε τους k κοντινότερους γείτονες χρησιμοποιώντας τις μεταβλητές $T_i$ και $T_{new}$. Έτσι θα παραχθεί το σύνολο κοντινότερων γειτόνων $J_{new}$.

Βήμα 4: Για όλα τα έργα λογισμικού i=1,...n (από 1 μέχρι n) υπολόγισε:

1. μέση τιμή $\tilde{Y}_i = Y_i - \frac{1}{k}\sum_{j\in Ji} Y_j$ (Εξ. 3.3)
2. μέση τιμή $\tilde{X}_i = X_i - \frac{1}{k}\sum_{j\in Ji} X_j$ (Εξ. 3.4)

Εδώ πραγματοποιείται η εκτίμηση αφενός της εξαρτημένης μεταβλητής βάση της EbA, αφετέρου των ανεξάρτητων μεταβλητών βάσει της μεθόδους LS και αφαιρούνται οι εκτιμήσεις από τις αρχικές τιμές.

Βήμα 5: Προσαρμογή του μοντέλου παλινδρόμησης (fitting) $Y_i$ = μέση τιμή $X_i'\beta + \varepsilon_i$. Αυτό πραγματοποιείται με την εκτίμηση του συντελεστή παλινδρόμησης βLS

Βήμα 6: Η τελική εκτίμηση της εξαρτημένης μεταβλητής για το νέο έργο προκύπτει από την σχέση:

$$\tilde{Y}_{new} = X_{new}\beta_{LS} + \frac{1}{k}\sum_{j\in J_{new}}\left(Y_j - X_j\beta_{LS}\right) \qquad (Εξ. 3.5)$$

Για όλα αυτά τα παραπάνω προβλήματα που προαναφέραμε προτάθηκε η μέθοδος LSEbA (Least Square Estimation by Analogy), για να αποδείξει μέσω της εφαρμογής της ότι σε αρκετά πολύπλοκες και δύσκολες περιπτώσεις, αν χρησιμοποιηθούν παραμετρικά και μη-παραμετρικά κομμάτια μπορεί να δώσει αρκετά βελτιωμένο το σφάλμα πρόβλεψης.

Στην παρούσα εργασία, γίνεται χρήση τριών νέων διαφορετικών μη-παραμετρικών μεθοδολογιών (CART, Bagging, Random Forest) εκτός από την ήδη χρησιμοποιούμενη μη-παραμετρική τεχνική EbA, για τον υπολογισμό του μη-παραμετρικού μέρους της εξίσωσης 2. Στόχος είναι να εξετάσουμε την ακρίβεια των αντίστοιχων ημι-παραμετρικών μοντέλων που προκύπτουν (LSCart, LSBagging, LSRandomForest) κάνοντας χρήση και υπολογισμό του μη-παραμετρικού τους μέρους μέσω των παραπάνω τεσσάρων τεχνικών.

# Κεφάλαιο 4
# Σύγκριση Μοντέλων Εκτίμησης Κόστους Λογισμικού

Σε αυτή την ενότητα παρουσιάζονται τα μέτρα ακριβείας, τα οποία χρησιμοποιούνται για την επικύρωση των μοντέλων εκτίμησης.

## 4.1 Μέτρα ακριβείας

Στις έρευνες που έχουν πραγματοποιηθεί έχει καταδειχθεί ότι η διαδικασία λήψης απόφασης για την καταλληλότητα ενός μοντέλου, δεν θα πρέπει να στηρίζεται σε καθιερωμένες στατιστικές μεθόδους επεξεργασίας δεδομένων. Επίσης οι διοικητές έργων θα πρέπει να επιλέγουν το καταλληλότερο μοντέλο ανάλογα με το έργο που έχουν και μετά από την σύγκριση συγκεκριμένων δεικτών ακριβείας (Kitchenham & Mendes, 2009). Εκτός αυτού, η μεμονωμένη μέτρηση ενός σφάλματος χρησιμοποιώντας απλά στατιστικά στοιχεία όπως η μέση τιμή ή η διάμεσος, μπορεί να αποδώσει διαφοροποιημένα αποτελέσματα, οπότε και να οδηγήσει σε εσφαλμένα συμπεράσματα (Mittas & Angelis, 2008b).

Έτσι έχουν προταθεί ορισμένα μέτρα ακριβείας, τα οποία χρησιμοποιούνται για την επικύρωση των μοντέλων ΕΚΛ. Η διαδικασία της επικύρωσης βασίζεται σε δύο τιμές, την πραγματική ($ΥΑ_i$) και την εκτιμώμενη τιμή ($ΥΕ_i$) κόστους ενός έργου τα οποία αξιολογούν την προβλεπτική ικανότητα ενός μονέλου ΕΚΛ.

Η επιλογή του εκάστοτε χρησιμοποιούμενου μέτρου ακρίβειας αποτελεί έναν εξίσου σημαντικό παράγοντα. Μάλιστα διάφοροι ερευνητές διατείνονται ότι τα συμπεράσματα για την επιλογή του καλύτερου μοντέλου πρόβλεψης, εξαρτώνται σε ένα μεγάλο βαθμό από τον δείκτη ακρίβειας που θα χρησιμοποιηθεί (Myrtveit et al., 2005). Επομένως αρκετά συχνά παρατηρείται το φαινόμενο, ένα μοντέλο πρόβλεψης να θεωρείται πως έχει μεγαλύτερη ακρίβεια από κάποιο άλλο, όταν αξιολογείται με ένα συγκεκριμένο

μέτρο ακριβείας, ενώ όταν χρησιμοποιείται κάποιο άλλο μέτρο, το αποτέλεσμα να είναι τελείως διαφορετικό. Αυτό συμβαίνει καθώς υφίστανται πληθώρα παραγόντων που πρέπει να ληφθούν υπόψη κατά την αξιολόγηση ενός μοντέλου. Τέτοιοι είναι α) η εξάρτηση των μεθοδολογιών πρόβλεψης από τα διαθέσιμα δεδομένα, β) η καταλληλότητα των τοπικών μεγεθών σφάλματος που χρησιμοποιούνται, γ) η στρατηγική προώθησης ενός συγκεκριμένου μοντέλου πρόβλεψης έναντι ενός ανταγωνιστικού με χρήση ανεδαφικών στατιστικών διαδικασιών και δ) οι πειραματικές διαδικασίες, όπως ο διαχωρισμός των δειγμάτων σε σύνολα εκπαίδευσης (training sets) και ελέγχου (test sets), καθώς και η μέθοδος επικύρωσης που ακολουθεί για την αξιολόγηση της προβλεπτικής ικανότητας του προτεινόμενου μοντέλου (Sommerville, 2004). Τα μέτρα ακριβείας (accuracy measures) που χρησιμοποιήσαμε στην παρούσα διατριβή ορίζονται ακολούθως:

Το πρώτο είναι το απόλυτο σφάλμα (Absolute Error-AE) που ορίζεται ως η απόλυτη τιμή της διαφοράς της εκτιμώμενης από την πραγματική τιμή

$$AE_i = \left| X_{Ai} - Y_{\mathrm{Ei}} \right| \qquad \text{(Εξ. 4.1)}$$

Σε περιπτώσεις όμως που η πραγματική τιμή της εξαρτημένης μεταβλητής ($Y_{Ai}$) αλλάζει συχνά, όπως συμβαίνει με το μέγεθος του λογισμικού σε γραμμές κώδικα, τότε το σχετικό σφάλμα (Relative Error-RES) χρησιμοποιείται για την εκτίμηση της σωστής ή προσαρμογής ενός μοντέλου στα πραγματικά δεδομένα (Miyazaki et al., 1994). Το σχετικό σφάλμα ορίζεται ως ο λόγος του απόλυτου σφάλματος προς την πραγματική τιμή

$$RE_i = \frac{AE_i}{Y_{\mathrm{i}}} \left| X_{Ai} - Y_{\mathrm{Ei}} \right| \qquad \text{(Εξ. 4.2)}$$

Αυτά τα δύο σφάλματα το απόλυτο και το σχετικό ανήκουν στην κατηγορία των μέτρων τοπικού (local) σφάλματος, καθώς αναφέρονται στη μέτρηση του σφάλματος που γίνεται «τοπικά» στην εκτίμηση ενός μόνον έργου.

Ένα άλλο μέτρο ακριβείας είναι αυτό που αφορά το μέγεθος σχετικού σφάλματος (Magnitude Relative Error-MRE) (Conte et al., 1986) και ορίζεται ως ο λόγος της

απόλυτης διαφοράς της εκτιμώμενης από την πραγματική τιμή προς την πραγματική τιμή

$$MRE_i = \frac{|Y_{\mathrm{Ai}} - Y_{\mathrm{Ei}}|}{Y_{\mathrm{Ai}}} = \frac{AE_i}{Y_{\mathrm{Ai}}} \qquad \text{(Εξ. 4.3)}$$

Επιπρόσθετα ένα άλλο σημαντικό μέτρο ακριβείας είναι το μέγεθος σχετικού σφάλματος ως προς την εκτίμηση (Magnitude of Relative Error to the Estimate-MER). Ως μέγεθος σχετικού σφάλματος ορίζεται ο λόγος της απόλυτης διαφοράς της εκτιμώμενης τιμής από την πραγματική τιμή προς την εκτιμώμενη τιμή, δηλαδή

$$MRE_i = \frac{|Y_{\mathrm{Ai}} - Y_{\mathrm{Ei}}|}{Y_{\mathrm{Ai}}} = \frac{AE_i}{Y_{\mathrm{Ei}}} \qquad \text{(Εξ. 4.4)}$$

Τέλος το ισορροπημένο σχετικό σφάλμα (Balance Relative Error-BRE) και το αντίστροφο ισορροπημένο σχετικό σφάλμα (Inverted Balance Relative Error-IBRE) (Miyazaki et al., 1991) για τα οποία ισχύει αν

$$y_{Ei} - y_{Ai} >= 0 \text{ τότε } \frac{|Y_{\mathrm{Ai}} - Y_{\mathrm{Ei}}|}{Y_{\mathrm{Ai}}} = \frac{AE_i}{Y_{\mathrm{Ei}}} \qquad \text{(Εξ. 4.5)}$$

Μετά τη χρήση των μέτρων τοπικού σφάλματος είναι δυνατόν να προκύψουν τα καθολικά (global) μέτρα ακρίβειας. Τα μέτρα αυτά στην πλειοψηφία των περιπτώσεων υπολογίζονται με τη χρήση κάποιου στατιστικού μέτρου κεντρικής τάσης (μέση τιμή ή διάμεσος) ή ενός ποσοστού σχετικών λαθών που είναι μικρότερο από μία τιμή p (συνήθως p=0.20 ή p=0.25). Ο υπολογισμός αυτός πραγματοποιείται στα τοπικά μέτρα ακριβείας και μία μέθοδος εκτίμησης μπορεί να θεωρηθεί ακριβής όταν τα μέτρα ακριβείας MMRE, MdMRE, MMER, MdMER, MBER και MIBRE έχουν μικρές τιμές.

## 4.2 Βαθμολόγηση και Συσταδοποίηση Μοντέλων Πρόβλεψης

Η ανάγκη για ακρίβεια στις προβλέψεις είναι επιτακτική καθώς λανθασμένες εκτιμήσεις μπορεί να αποβούν καταστροφικές για τους προγραμματιστές και τους πελάτες καθώς μπορεί να οδηγήσουν στην ακύρωση ενός σημαντικού συμβολαίου. Έτσι το ενδιαφέρον επικεντρώθηκε στο ανοιχτό και υπό διερεύνηση ζήτημα της αναζήτησης της καλύτερης τεχνικής πρόβλεψης. Ωστόσο, η γενίκευση ορισμένων μέχρι τώρα ευρημάτων που δεν αξιοποιούν κατάλληλο στατιστικό έλεγχο υποθέσεων, μπορεί πολλές φορές να είναι παραπλανητική και να οδηγήσει σε εσφαλμένα συμπεράσματα.

Επίσης, αναδεικνύονται διάφορα ζητήματα που αφορούν την στατιστική διαδικασία σύγκρισης των τεχνικών πρόβλεψης. Στην περίπτωση που πραγματοποιείται σύγκριση μεταξύ δύο ανταγωνιστικών μοντέλων πρόβλεψης η μηδενική υπόθεση εξετάζεται μέσα από κλασσικά τεστ (t-test, Wilcoxon rank test κτλ.). Εντούτοις στην περίπτωση που συγκρίνονται περισσότερα από 2 μοντέλα πρόβλεψης, η έννοια της στατιστικής σημαντικότητας γίνεται περισσότερο πολύπλοκη και τα θέματα που σχετίζονται με αυτή τη προσέγγιση είναι γνωστά στην στατιστική ως “σύγκριση πολλαπλών προβλημάτων”. Επομένως οι διοικητές έργου θα πρέπει να επιλέγουν το ανάλογο μοντέλο πρόβλεψης μέσα από μία καλά εδραιωμένη και αξιόπιστη διαδικασία που δίνει την δυνατότητα επιλογής ενός συνόλου μοντέλων αντί ενός και μόνο επικρατέστερου. Αυτό επαυξάνει την αντικειμενικότητα.

Για την δημιουργία ομάδων με μοντέλα τα οποία είναι καταλληλότερα από άλλα οι ερευνητές έχουν καταφύγει σε αλγορίθμους κατηγοριοποίησης και συσταδοποίησης οι οποίοι πραγματοποιούν την διαφοροποίηση των μοντέλων έναντι κάποιων άλλων βάση των μέτρων ακριβείας. Στα πλαίσια της παρούσας διατριβής μελετήσαμε και υλοποιήσαμε στην εφαρμογή τον αλγόριθμο Scott-Knott διότι φέρει ως κύριο χαρακτηριστικό την δημιουργία μη-επικαλυπτόμενων συστάδων (clusters) (Mittas et al., 2015b).

Η προτεινόμενη στατιστική μεθοδολογία που ακολουθείται βασίζεται σε μία αλγοριθμική διαδικασία που δίνει την δυνατότητα να παράγουμε μη επικαλυπτόμενες συστάδες από μοντέλα πρόβλεψης. Αυτές οι συστάδες χαρακτηρίζονται από ομοιογένεια ως προς την απόδοση των μοντέλων πρόβλεψης. Για την υλοποίηση αυτής

της προσέγγισης αξιοποιείται ο επονομαζόμενος έλεγχος Scott-Knott, όπου βαθμολογεί τα μοντέλα και τα κατατάσσει σε διάφορες επιμέρους συστάδες (Mittas & Agelis, 2013).

Ο αλγόριθμος ομαδοποίησης Scott-Knott είναι ένας ευρέως διαδεδομένος αλγόριθμος ομαδοποίησης που χρησιμοποιείται ως μια στατιστική μέθοδο πολλαπλών συγκρίσεων στην ανάλυση της διακύμανσης (Scott and Knott, 1974). Θα λέγαμε πως αποτελεί μια διαδικασία ελέγχου πολλαπλών συγκρίσεων που βασίζεται στις αρχές της συσταδοποίησης (cluster analysis). Η συσταδοποίηση αναφέρεται στον τρόπο που με τον οποίο, αποτελέσματα μίας διαδικασίας, στην περίπτωση μας οι προβλέψεις των μοντέλων, συγκρίνονται ώστε να δημιουργηθεί μία ομαδοποίηση βάση της στατιστικής σημαντικότητας. Η στατιστική σημαντικότητα καθορίζεται από τυχόν διαφοροποιήσεις που μπορεί να έχουν μεταξύ τους οι μέσες τιμές των μεγεθών ενός δείγματος.

Η επιλογή του συγκεκριμένου τρόπου αξιολόγησης έγινε με βάση το γνώρισμα που φέρει να αποτρέπει αλληλοεπικαλύψεις μεταξύ των συστάδων. Όλες οι μέθοδοι που χρησιμοποιούνται σήμερα όπως t-test, Tukey, Duncan, διαδικασίες Newman-Keuls έχουν προβλήματα επικάλυψης. Με τον όρο επικάλυψη εννοείται το γεγονός μια περίπτωση να κατατάσσεται σε παραπάνω από μια ομάδα. Η μέθοδος Scott-Knott δεν αντιμετωπίζει αυτό το πρόβλημα, το οποίο είναι στα θετικά χαρακτηριστικά αυτής.

Η διαδικασία ξεκινά με τον διαχωρισμό των ομάδων έτσι ώστε να μεγιστοποιηθεί το άθροισμα των τετραγώνων (Sum of Squared Errors) μεταξύ των ομάδων. Στην συνέχεια οι ομάδες ταξινομούνται βάση των μέσων τιμών ώστε να μειωθεί ο μέγιστος αριθμός των δυνατών διαχωρισμών.

Για τις ανάγκες της παρούσας διατριβής και για την επιλογή ενός κατάλληλου υποχώρου μοντέλων, ο εν λόγω έλεγχος εφαρμόστηκε σε ένα σύνολο 8 μοντέλων χρησιμοποιώντας 6 σύνολα δεδομένων (data sets), τα οποία είναι κοινά διαθέσιμα και εμπεριέχουν έργα λογισμικού.

Οι τιμές που τροφοδοτούμε στον έλεγχο απορρέουν από τα λάθη πρόβλεψης που αποδίδει το εκάστοτε μοντέλο σε σχέση με τις πραγματικές. Η διαδικασία Scott-Knott χρησιμοποιεί την ανάλυση διακύμανσης κατά ένα κριτήριο (one way ANOVA). Η μεθοδολογία αυτή αποσκοπεί στο να ανιχνεύσει διαφορές μεταξύ των μέσων τιμών ορισμένων πληθυσμών. Στην προκειμένη περίπτωση χρησιμοποιήθηκε ώστε να

επαληθεύσουμε την μηδενική μας υπόθεση, η οποία ορίζει πως δεν υφίσταται κάποια διαφορά μεταξύ των μέσων τιμών των μέτρων ακριβείας που ανακτήσαμε από τα συγκρινόμενα μοντέλα πρόβλεψης (Montgomery, 1991).

Συμπληρωματικά, η εναλλακτική υπόθεση που θεωρούμε είναι πως τα μοντέλα μας μπορεί να κατατμηθούν σε δύο αμοιβαία αποκλειόμενα και διαμερισμένα (collectively exhaustive) υποσύνολα. Έτσι όταν η μέθοδος ANOVA καταδεικνύει ότι δεν υπάρχει κάποια στατιστική διαφορά για την απόρριψη της μηδενικής ερευνητικής υπόθεσης, η σύγκριση των τιμών από τα επιμέρους μοντέλα πρόβλεψης μας οδηγεί στην συγκρότηση ομοιογενών ομάδων που δεν μπορούν να κατατμηθούν περαιτέρω.

Για τον διαχωρισμό των δειγμάτων κάθε ομάδας σε υπο-ομάδες γίνεται χρήση μιας στατιστικής κατανομής που προσεγγίζει την κατανομή Chi-square. Εν τούτοις η εμπειρία καταδεικνύει ότι οι κατανομές των σφαλμάτων δεν ακολουθούν την κανονική κατανομή. Κατά συνέπεια είναι απαραίτητο στον ανωτέρω αλγόριθμο (Scott-Knott) να εφαρμόσουμε ένα μετασχηματισμό για να κανονικοποιήσουμε τις τιμές του λάθους πριν εφαρμόσουμε τον αλγόριθμο. Για τον λόγο αυτό ο κατάλληλος μετασχηματισμός που αξιοποιείται είναι ο μετασχηματισμός Blom ο οποίος παράγει αριθμητικές τιμές κανονικά κατανεμημένες που τροφοδοτούνται στον αλγόριθμο αντί των πρωτότυπων τιμών σφάλματος.

Τα παραπάνω έχουν ως αποτέλεσμα τα μοντέλα να βαθμολογούνται βάση των μετασχηματισμένων τιμών λάθους των μέτρων ακριβείας και εκτός τούτου να δημιουργείται ένα σχήμα συσταδοποίησης, όπου κάθε συστάδα αποτελείται ταξινομημένα μοντέλα που δεν έχουν κάποια σημαντική διαφορά στα σφάλματα μετρήσεων (Error Measures) για περισσότερες λεπτομέρειες μπορεί να ανατρέξει κανείς στην ερευνητική μελέτη των Mitta et al., (2015b).

# Κεφάλαιο 5
# Γλώσσα R

Στο κεφάλαιο αυτό θα πραγματοποιηθεί μια παρουσίαση των δυνατοτήτων της γλώσσας R, η οποία χρησιμοποιήθηκε για την ανάπτυξη της διαδικτυακής εφαρμογής.

## 5.1 Γλώσσα R

Η R είναι ένα υπολογιστικό πακέτο που προσφέρει στον χρήστη δυνατότητες διαχείρισης και στατιστικής ανάλυσης δεδομένων καθώς και δυνατότητες κατασκευής γραφημάτων (Kurt, 2015 ; Richard et al., 1988). Η εν λόγω γλώσσα αποτελεί μια εφαρμογή της γλώσσας S, η οποία αναπτύχθηκε στα Bell Laboratories από τους Rick Becker, John Champers και Allan Wilks (Gentleman, 2006). Οι διαφορές μεταξύ της R και S είναι ελάχιστες και έτσι κώδικας που γράφεται για το R τρέχει σχεδόν αμετάβλητος και στις δύο S μηχανές (Chambers, 1998).

Διατίθεται ελεύθερα στο διαδίκτυο από τους όρους του CNU. Για να εγκαταστήσει κάποιος την R, μπορεί να επισκεφθεί την ιστοσελίδα http://www.r-project.org/, η οποία περιέχει περαιτέρω πληροφορίες που αφορούν την εγκατάσταση και αποθήκευση του προγράμματος σε λειτουργικά συστήματα όπως Linux, MAC OS και Windows.

Η γλώσσα προγραμματισμού R, χρησιμεύει κυρίως για την επεξηγηματική ανάλυση δεδομένων μέσω της παραγωγής κάποιον διαγραμμάτων, ιστογραμμάτων και στην εφαρμογή διάφορων στατιστικών μοντέλων (Φωκιανός & Χαραλάμπους, 2010).

Ένα από τα σημαντικά πλεονεκτήματα της, είναι ότι με την χρήση μία διαλέκτου της γλώσσας S (η οποία είναι μια διερμηνέας γλώσσα) οι εντολές διαβάζονται και μετά εκτελούνται αμέσως, δηλαδή μια συνάρτηση μπορεί να δημιουργηθεί, να εκτελεσθεί και μετά να δημιουργήσει μια καινούργια συνάρτηση η οποία να καλεί την προηγούμενη. Σε

αντίθεση με άλλες γλώσσες προγραμματισμού στις οποίες ολόκληρα προγράμματα μεταγλωττίζονται στην κατάλληλη γλώσσα μηχανής. Επιπρόσθετα δίδονται διάφορες δυνατότητες στους χρήστες, με την χρήση διαφόρων στατιστικών μοντέλων, όπως ανάλυση γραμμικών και μη γραμμικών μοντέλων, στατιστικοί έλεγχοι, μέθοδοι ανάλυσης χρονοσειρών, αλγόριθμοι ομαδοποίησης και κατηγοριοποίησης με την δυνατότητα αναπαράστασης αυτών μέσω γραφικού περιβάλλοντος.

Η R είναι λειτουργικά επεκτάσιμη γλώσσα, καθώς οποιαδήποτε στιγμή ο χρήστης μέσω εγκατάστασης βιβλιοθηκών (packages) μπορεί να επεξεργαστεί τα δεδομένα του. Οι βιβλιοθήκες αυτές αναπτύσσονται για να καλύψουν ένα ευρύ φάσμα των σημερινών στατιστικών αναγκών των χρηστών και είναι διαθέσιμες από τους διαδικτυακούς τόπους CRAN.

Εκτός από τις βιβλιοθήκες , η R εμπεριέχει πληθώρα εργαλείων τα οποία μπορούν να χρησιμοποιηθούν για την αποτελεσματική επεξεργασία και απεικόνιση των δεδομένων, όπως εκτέλεση πράξεων με πίνακες, εργαλεία για την απεικόνιση γραφικών αναπαραστάσεων και δυνατότητα ανάπτυξης συναρτήσεων από τους χρήστες και εργαλεία για εισαγωγή και έξοδο δεδομένων. Επίσης κώδικας γραμμένος σε διάφορες γλώσσες όπως C,C++ μπορούν να φορτωθούν και να εκτελεσθούν.

Τέλος η γλώσσα προγραμματισμού R συμβαδίζει με τις εξελίξεις στην τεχνολογία και ταυτόχρονα στις τεχνολογικές απαιτήσεις της εργασίας, ενώ ταυτόχρονα το περιβάλλον χρήσης της είναι λειτουργικό και φιλικό προς τον χρήστη.

## 5.2 Το RStudio

Το RStudio αποτελεί ένα ελεύθερο και ενεργό ολοκληρωμένο περιβάλλον (Integrated development environment-IDE) ανοιχτού κώδικα της γλώσσας R (Verzani, 2011). Περιλαμβάνει ένα σύνολο από ολοκληρωμένα εργαλεία όπως κονσόλα, έναν συντάκτη – επεξεργαστή για την άμεση εκτέλεση κώδικα, διάθεση λίστας ιστορικού εντολών, ποικιλία εργαλείων απεικόνισης και δυνατότητα εντοπισμού σφαλμάτων στον κώδικα.

## 5.3 Το πακέτο shiny

Στα πλαίσια της εκπόνησης της παρούσας διατριβής θα γίνει χρήση ενός πακέτου της R, το Shiny. Αυτό είναι ένα νέο πακέτο, μέσω του οποίου μπορούν να δημιουργηθούν διαδικτυακές εφαρμογές με χρήση κώδικα R με εξαιρετικά εύκολο τρόπο στις οποίες η απεικόνιση είναι δυναμική.

Κάποια από τα βασικά πλεονεκτήματα του πακέτου Shiny παρουσιάζονται παρακάτω:

- Συνήθως ο όγκος γραμμών κώδικα άλλων γλωσσών, οι οποίες απαιτούνται για την ανάπτυξη μιας ολοκληρωμένης εφαρμογής συγκριτικά με τον κώδικά που απαιτείται για μια R εφαρμογή είναι πολύ μεγαλύτερος.
- Κώδικας γραμμένος σε άλλες γλώσσες (C,C++) μπορεί να εκτελεσθεί σε αυτό.
- Οι εφαρμογές αναπτύσσονται με την χρήση του Bootstrap, το οποίο είναι μια συλλογή εργαλείων ανοιχτού κώδικα για τη δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών,
- Η εύκολη μετατροπή διαφόρων αναλύσεων σε διαδικτυακές διαδραστικές εφαρμογές χωρίς την ανάγκη χρήσης HTML, CSS και JavaScript.
- Συγκερασμός των ικανοτήτων της R με τις ικανότητες παρουσίασης μιας διαδικτυακής εφαρμογής.
- Η οπτική παρουσίαση των αποτελεσμάτων γίνεται με δυναμικό τρόπο (δυναμική μεταβολή δεδομένων ενός ιστογράμματος).
- Δυνατότητα ταξινόμησης ή και εμφάνισης των δεδομένων υπό συνθήκες δυναμικά (sorting functions).
- Χρήση μπαρών πλοήγησης.
- Επιτρέπεται η χρήση κομματιών γλώσσας HTML/CSS μέσα στον κώδικα της R και με επέκταση με χρήση JavaScript.
- Επιτρέπεται η προσαρμογή των δεδομένων με χρήση λειτουργιών drag και drop και οι όποιες αλλαγές προσαρμόζονται αυτόματα.
- Δημιουργία εφαρμογών με χειριζόμενα συστατικά (components).
- Άμεση επικοινωνία μεταξύ του φυλλομετρητή και της R αξιοποιώντας πακέτα “websockets”.
- Υποστήριξη ανάπτυξης widgets από χρήστες και αποθήκευση αυτών για μελλοντική επαναχρησιμοποίηση τους από άλλους προγραμματιστές-χρήστες.

Αξιοποιώντας το Shiny αναπτύχθηκε μία web-based εφαρμογή, η οποία ενσωματώνει 8 από τα πιο γνωστά μοντέλα πρόβλεψης στην περιοχή της ΕΚΛ και έχοντας ένα πλήθος μέτρων ακριβείας, επιστρέφει αποτελέσματα που βοηθούν στην συγκριτική αποτίμηση των μοντέλων. Αυτό έχει ως σκοπό να διευκολύνει την λήψη αποφάσεων για την επιλογή της κατάλληλης στρατηγικής που θα πρέπει να ακολουθηθεί σε κάθε έργο κατά την φάση της εκτίμησής του.

# Κεφάλαιο 6
# Υλοποίηση της Web-Based Εφαρμογής

Στο κεφάλαιο αυτό θα πραγματοποιηθεί η παρουσίαση της διαδικτυακής εφαρμογής (Web application), η οποία αναπτύχθηκε στα πλαίσια της Διατριβής κάνοντας χρήση της γλώσσας R και του πακέτου Shiny στο περιβάλλον του RStudio.

## 6.1 Η Εφαρμογή

Στην παρούσα ενότητα πραγματοποιείται μια παρουσίαση των κυριοτέρων σημείων της εφαρμογής που αναπτύχθηκε με την χρήση της γλώσσας προγραμματισμού R και την χρήση του πακέτου Shiny που υποστηρίζει πολλαπλές λειτουργίες για την υλοποίηση τμημάτων που εκτελούνται στην πλευρά του χρήστη αλλά και στην πλευρά ενός εξυπηρετητή.

Ετσι δημιουργήθηκαν τέσσερις υβριδικές ημι-παραμετρικές μέθοδοι που συνδυάζουν τα πλεονεκτήματα παραμετρικών και μη-παραμετρικών μοντέλων και ενσωματώθηκαν στην εφαρμογή. Για την αξιολόγηση των μεθόδων η εφαρμογή προσφέρει τα παρακάτω:

1. Την επιλογή και την εκτέλεση προβλέψεων για ΕΚΛ, χρησιμοποιώντας τα μοντέλα που υλοποιήθηκαν. Αυτά είναι τα μη-παραμετρικά EbA, Random Forest, CART και Bagging, όπως επίσης και τα αντίστοιχα ημι-παραμετρικά που προέκυψαν από τα πρώτα LSEbA, LSRandomForest, LSCART και LSBagging.
2. Τον υπολογισμό γνωστών μέτρων ακρίβειας ανά μέθοδο για ένα συγκεκριμένο σύνολο δεδομένων.
3. Στατιστικούς ελέγχους υποθέσεων για την αξιολόγηση της ακρίβειας πρόβλεψης των μοντέλων και τη δημιουργία συστάδων από μεθόδους με παρόμοιες

δυνατότητες πρόβλεψης μέσω του αλγόριθμου πολλαπλών συγκρίσεων Scott-Knott.

Ειδικότερα αξιοποιώντας το περιβάλλον shiny μπορούμε να επιτύχουμε αρχιτεκτονική Model-view-controller (MVC) (Burbeck, 1987). Το μοντέλο αρχιτεκτονικής λογισμικού MVC, χρησιμοποιείται σε περιπτώσεις που επιθυμούμε να δημιουργήσουμε διεπαφές χρήστη που επικοινωνούν απρόσκοπτα και δυναμικά με τις λειτουργίες που εκτελούνται στο παρασκήνιο (backend) αξιοποιώντας υπολογιστικούς πόρους ενός συστήματος. Οποιαδήποτε εφαρμογή ακολουθεί αυτό το μοντέλο, διαιρείται σε τρία διασυνδεμένα μέρη ώστε να διαχωριστεί η παρουσίαση της πληροφορίας στον χρήση από την μορφή που έχει αποθηκευτεί στο σύστημα. Πιο συγκεκριμένα, το κύριο μέρος του, είναι το μοντέλο (Model) που αποτελείται από τα δεδομένα της εφαρμογής, τους κανόνες και τις συναρτήσεις. Το αντικείμενο-όψη (View) που μπορεί να είναι οποιαδήποτε παράσταση εξόδου των πληροφοριών, όπως ένα γράφημα ή διάγραμμα. Το τρίτο μέρος είναι ο ελεγκτής (Controller), που δέχεται μια είσοδο και τη μετατρέπει σε εντολές για το μοντέλο ή την όψη.

Στην διαδικτυακή εφαρμογή που αναπτύχθηκε τον ρόλο του αντικειμένου όψης κατέχει το αρχείο “ui.R” ενώ τον ρόλο του μοντέλου και ταυτόχρονα του ελεγκτή κατέχει το αρχείο “server.R”, όπου μεταξύ άλλων περιλαμβάνει υλοποιήσεις όλων των υπομελέτη μεθόδων των μέτρων ακριβείας και του αλγορίθμου Scott-Knott.

## 6.2 Διαδικασίες για εκτέλεση της Εφαρμογής

Για να μπορέσει κάποιος να χρησιμοποιήσει την εφαρμογή θα πρέπει να προβεί στην εγκατάσταση του πακέτου. Επίσης από την γραμμή εντολών του περιβάλλοντος που παρέχεται θα πρέπει να εγκατασταθεί το πακέτο Shinny και να φορτώσει την εν λόγω βιβλιοθήκη. Τότε μπορεί να χρησιμοποιήσει την εφαρμογή που υλοποιήθηκε.

## 6.3 Δομή της Εφαρμογής

Όπως προαναφέρθηκε τα δύο βασικά αρχεία του πακέτου Shiny, της γλώσσας R, στα οποία αναπτύχθηκε το μεγαλύτερο μέρος της εφαρμογής είναι τα ui.R και το server.R. Στο αρχείο “ui.R” (User Interface) αναπτύχθηκε κώδικας στην γλώσσα HTML, για την

υλοποίηση των φορμών και πεδίων εισόδου-εξόδου. Με αυτά μπορεί ο χρήστης να τροφοδοτήσει δεδομένα στην εφαρμογή αλλά και να παρατηρήσει τα αποτελέσματα. Από την άλλη πλευρά στο αρχείο “serve.R” βρίσκεται ο βασικός κώδικας της διατριβής, που αφορά την διαχείριση των δεδομένων που εισάγει ο χρήστης, αλλά και κλήσεις των συναρτήσεων που υλοποιούν τα μοντέλα πρόβλεψης. Εκτός τούτων στο ίδιο αρχείο περιλαμβάνονται οι υλοποιήσεις που αφορούν τους μηχανισμούς των μέτρων ακριβείας για την πραγματοποίηση στατιστικών υπολογισμών και η υλοποίηση του αλγορίθμου Scott-Knott μαζί με τον μετασχηματισμό Blom για την κανονικοποίηση των δειγμάτων.

## 6.4 Ανάπτυξη της εφαρμογής - Δομή της Εφαρμογής

Όπως θα είναι εύκολα κατανοητό οι απαιτήσεις αλλά και ο χρόνος που χρειάζεται κάποιος για να προχωρήσει στην σύγκριση των μοντέλων ΕΚΛ είναι πολύπλοκες και δυσνόητες. Πολλές φορές η συγκεκριμένη προσπάθεια μπορεί να μην επιφέρει τα επιθυμητά αποτελέσματα, όταν οι διοικητές έργου δεν κατέχουν εξειδικευμένες γνώσεις που αφορούν την πρόβλεψη έργων λογισμικού αλλά και δεν υποβοηθούνται από ανάλογα εργαλεία.

Κρίνεται λοιπόν απαραίτητη η ανάγκη δημιουργίας μιας εφαρμογής που θα δίνει την δυνατότητα στον χρήστη – ερευνητή να διεξάγει άμεση σύγκριση, μιας κρίσιμης μάζας μοντέλων ΕΚΛ σε ένα περιβάλλον φιλικό ως προς τον χρήση. Αυτή η απλότητα μεταφράζεται ως ευχρηστία της εφαρμογής όπου ο χρήστης μπορεί με την φόρτωση μιας βάσης δεδομένων με παλαιότερα έργα λογισμικού υπό την μορφή ενός συνόλου δεδομένων να πραγματοποιήσει υπολογισμούς και προβλέψεις.

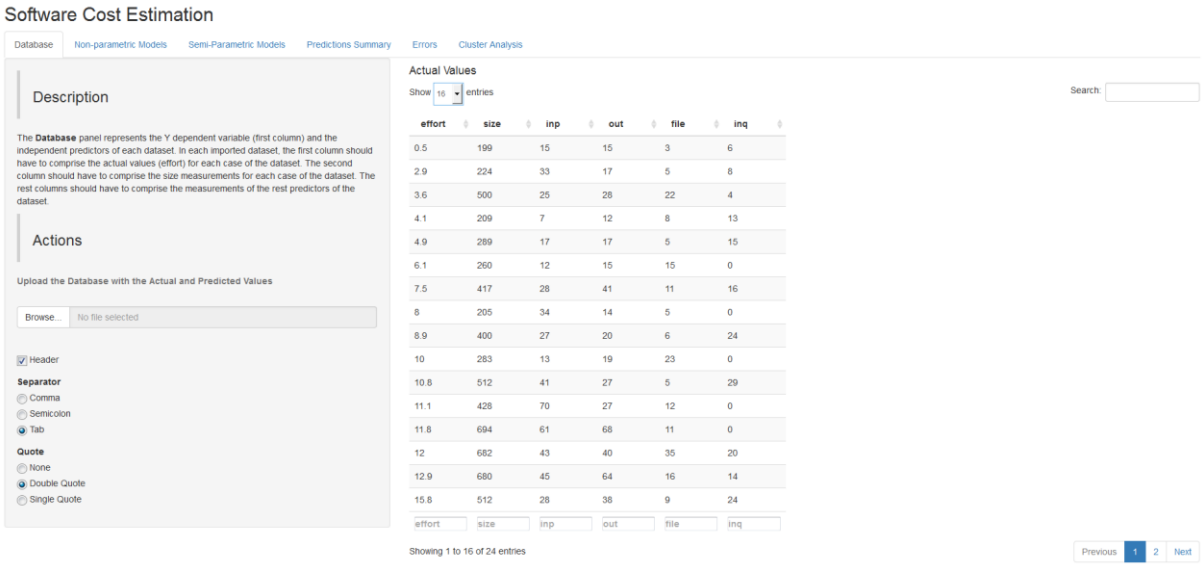
Η ανάπτυξη της εφαρμογής έγινε αποκλειστικά με την χρήση του RStudio στην γλώσσας προγραμματισμού R και οι δύο θεμελιώδης ομάδες αρχείων είναι αυτή που αφορά την διεπαφή χρήστη “ui.R” (User Interface) και αυτούς που αφορά τους μηχανισμούς του εξυπηρετητή “server.R”.

Το βασικό χαρακτηριστικό της εφαρμογής είναι η δυνατότητα που αφορά την συγκριτική πρόβλεψη όλων των μεθόδων για ένα συγκεκριμένο σύνολο δεδομένων. Αυτό επιτρέπει σε ένα χρήστη – διαχειριστή έργου να κατανοήσει και να ερμηνεύσει τα αποτελέσματα των διαφόρων μεθόδων. Στην συνέχεια αφού συγκεντρώσει

επιστημονικά στοιχεία μπορεί να προβεί εύκολα στην λήψη αποφάσεων για την επιλογή της καταλληλότερης στρατηγικής που μπορεί να ακολουθηθεί. Αξίζει να σημειωθεί ότι δεν υπάρχει περιορισμός στο πλήθος των συνόλων δεδομένων με τα οποία μπορεί να πραγματοποιηθεί συγκριτική μελέτη.

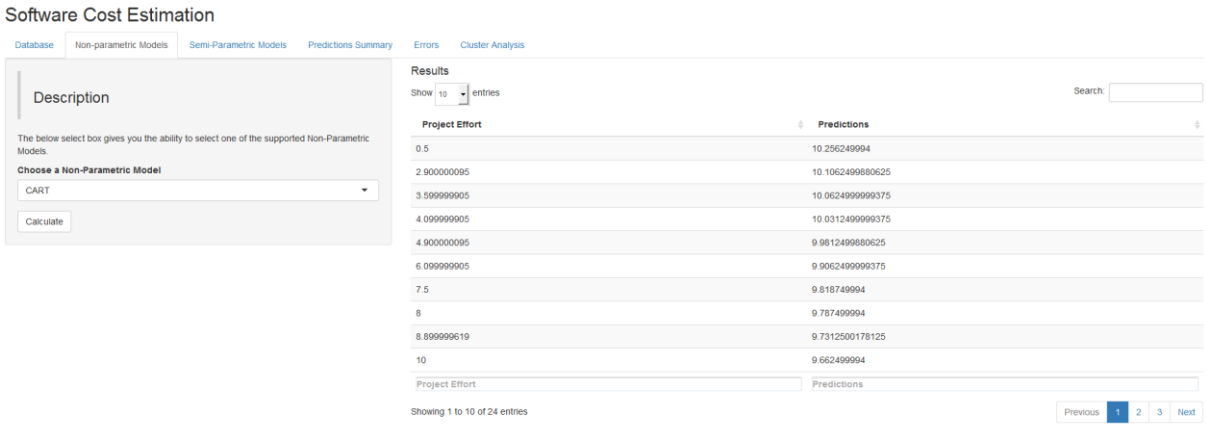
Τα υποστηριζόμενα μοντέλα της εφαρμογής είναι στο σύνολο τους οκτώ, εκ των οποίων τα τέσσερα είναι μη-παραμετρικά και τα υπόλοιπα τέσσερα είναι ημι-παραμετρικά. Τα ημι-παραμετρικά σχηματιστήκαν από την πρώτη τετράδα από την μέθοδο των ελαχίστων τετραγώνων (Least Square) (Mittas & Angelis, 2009). Επίσης η εφαρμογή αξιοποίησε πέντε στατιστικές συναρτήσεις σφάλματος και έναν αλγόριθμο συσταδοποίησης για την ομαδοποίηση των καλύτερων μεθόδων σε ένα σύνολο δεδομένων.

Κατά την εκκίνηση στην αρχική οθόνη ο χρήστης έρχεται αντιμέτωπος με μια φόρμα εισόδου όπου μπορεί να τροφοδοτήσει μέσω αυτής ένα αρχείο συνόλου δεδομένων. Ο κώδικας της εφαρμογής έχει αναπτυχθεί με τέτοιο τρόπο ώστε να χρησιμοποιείται αυτόματα το σύνολο δεδομένων Albrecht σε περίπτωση που δεν έχει γίνει επιλογή κάποιου άλλου συνόλου δεδομένων. Εδώ υπάρχει και μια φόρμα εμφάνισης του επιλεγμένου συνόλου δεδομένων υπό την μορφή πίνακα. Επίσης δίνεται η επιλογή στον χρήστη να επιλέξει τον τρόπο με τον οποίο διαχωρίζονται τα δεδομένα μέσα στο αρχείο ώστε να διαχειριστούμε όλες τις πιθανές περιπτώσεις (διαχωρισμός τιμών με “,” ή “tab” ή “κενό διάστημα”).

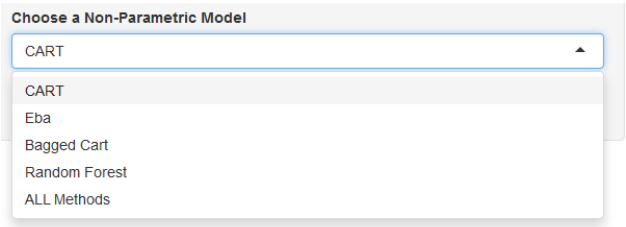



**Εικόνα 6.1:** Απεικόνιση της υλοποίησης της Web-Based εφαρμογής για τον σκοπό του "uploading" συνόλου δεδομένων.

Συνεχίζοντας στην επόμενη καρτέλα που ονομάζεται "Non-parametric Models", ο χρήστης έχει την δυνατότητα να επιλέξει ένα από τα διαθέσιμα μη-παραμετρικά μοντέλα. Τα μοντέλα τα οποία μπορούν να επιλεχθούν είναι το CART, η EbA, το Bagged Cart και η Random Forest. Με το πάτημα του πλήκτρου "Calculate" εμφανίζεται το μέγεθος της πραγματικής προσπάθειας (Project Effort) που απαιτήθηκε από το έργο αλλά και της προβλεπόμενης προσπάθειας (Predictions) σε αντιδιαστολή. Τα μοντέλα τα οποία μπορούν να επιλεχθούν είναι το CART, η EbA, το Bagged Cart και η Random Forest.

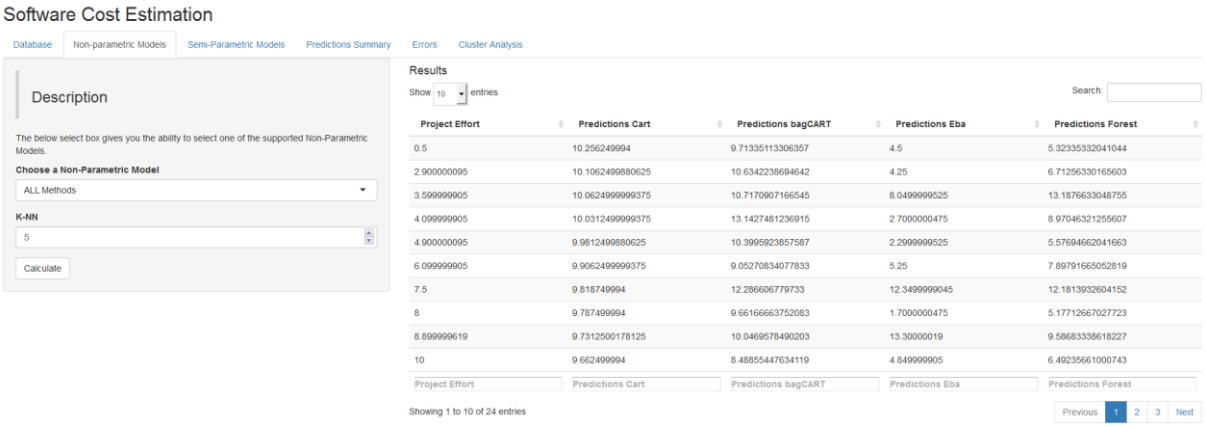



**Εικόνα 6.2:** Επιλογή μη-παραμετρικού μοντέλου για υπολογισμό των προβλέψεων.



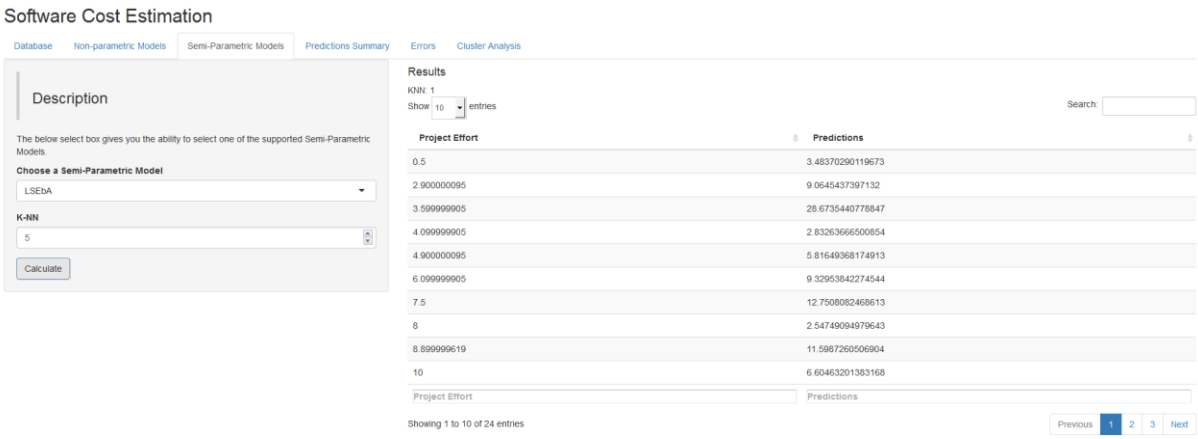

**Εικόνα 6.3:** Επιλογή μη-παραμετρικού μοντέλου.

Εκτός τούτων όπως διαφαίνεται στην Εικόνα 6.3, η εφαρμογή δίνει την δυνατότητα επιλογής όλων των μεθόδων, ώστε να υπολογιστούν συγκεντρωτικά τα αποτελέσματα. Αξίζει να σημειωθεί πως η μέθοδος EbA πραγματοποιεί μια βαθμονόμηση ώστε να αυτορυθμιστεί και να καταλήξει σε ένα βέλτιστο αριθμό κοντινότερων γειτόνων. Για τον λόγω αυτό η εφαρμογή δίνει την δυνατότητα στο χρήστη να εισάγει έναν αριθμό.

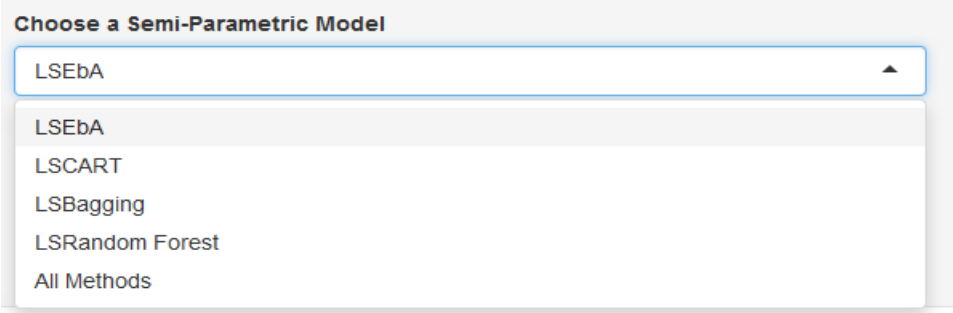



**Εικόνα 6.4:** Απεικόνιση αποτελεσμάτων όλων των μη-παραμετρικών μοντέλων .

Έπειτα ακολουθεί η καρτέλα που αφορά τα ημι-παραμετρικά μοντέλα και φέρει το όνομα “Semi-parametric Models”. Εδώ κατά αντιστοιχία ο χρήστης έχει την δυνατότητα να επιλέξει ένα ημι-παραμετρικό μοντέλο από τα διαθέσιμα και να υπολογίσει την προβλεπόμενη προσπάθεια (Predictions). Τα μοντέλα τα οποία μπορούν να επιλεχθούν είναι το LSCART, η LSEbA, το LSBagging και η LSRandom Forest.

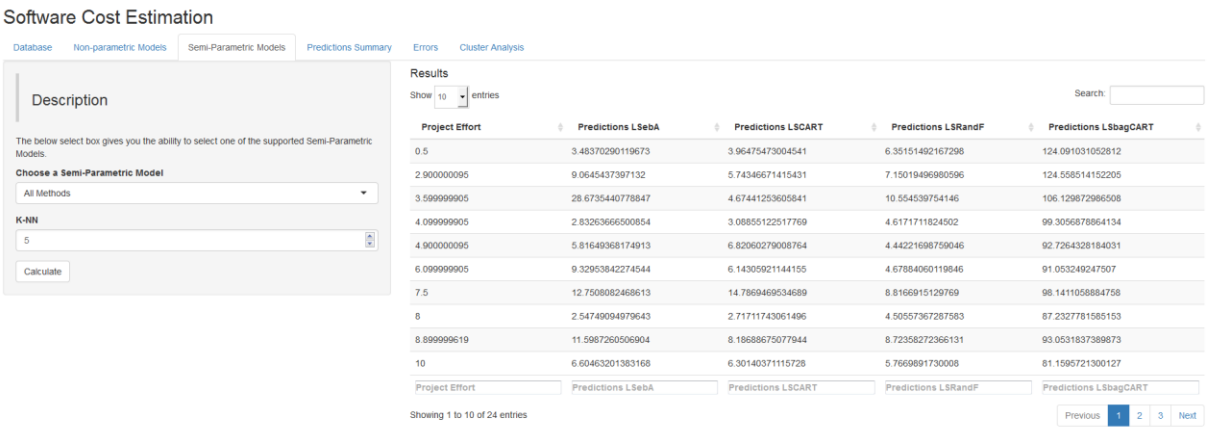



**Εικόνα 6.5:** Επιλογή ημι-παραμετρικού μοντέλου για υπολογισμό των προβλέψεων.



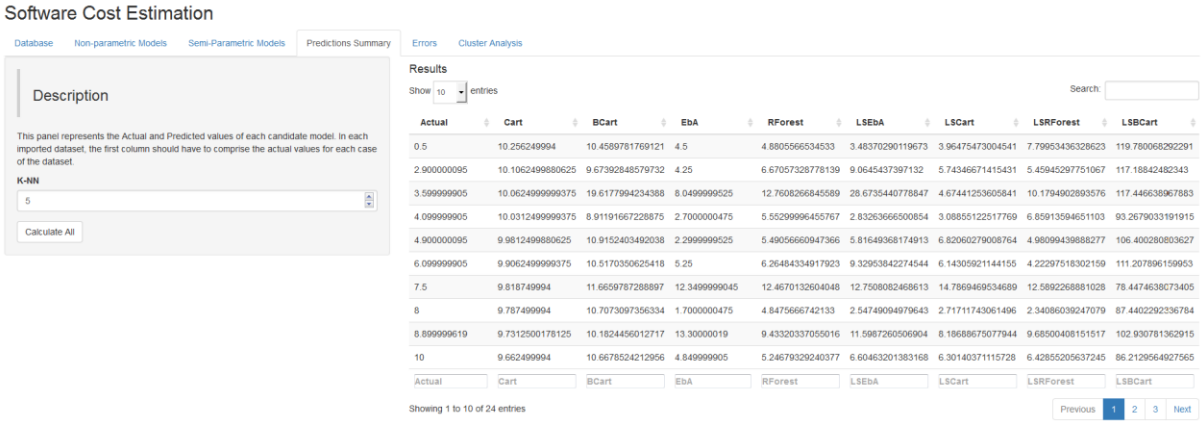

**Εικόνα 6.6:** Επιλογή ημι-παραμετρικού μοντέλου.

Επίσης και εδώ παρουσιάζονται αντίστοιχοι πίνακες όπου παραθέτουν τα στοιχεία της πραγματικής προσπάθειας (Project Effort) που καταβλήθηκε έναντι της προβλεπόμενης (Predictions). Αντιστοίχως και σε αυτή την περίπτωση δίνεται η δυνατότητα επιλογής "ALL Methods" που με την οποία θα πραγματοποιηθεί μια συγκριτική διαδικασία όλων των ημι-παραμετρικών μοντέλων σε σχέση με την προσπάθεια που απαιτείται. Ομοίως για την περίπτωση της μεθόδου LSEbA δίδεται η δυνατότητα στον χρήστη να ορίσει έναν αριθμό για τους κ-πλησιέστερους γείτονες, με απώτερο σκοπό να αυτορυθμιστεί η ανωτέρω μέθοδος.




**Εικόνα 6.7:** Απεικόνιση αποτελεσμάτων όλων των ημι-παραμετρικών μοντέλων .

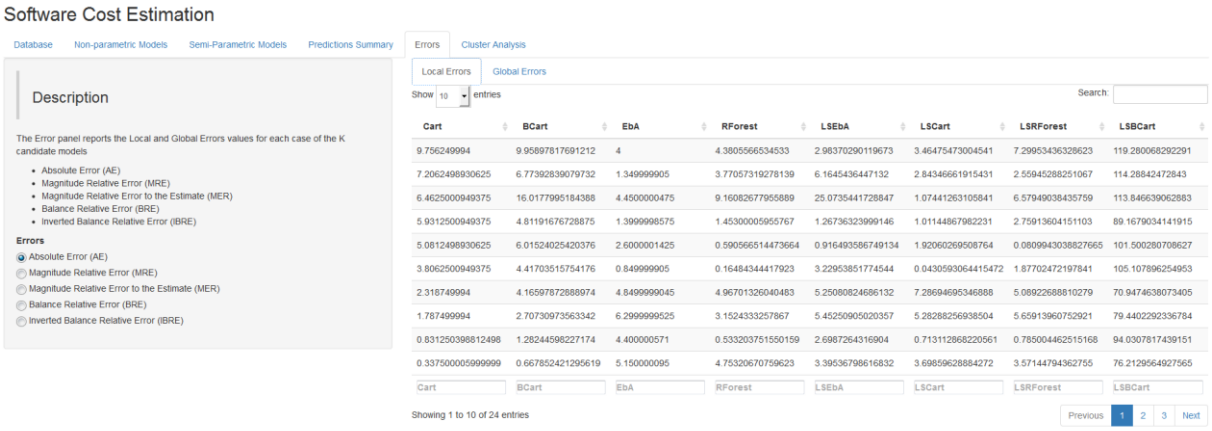
Όλα τα παραπάνω αποτελέσματα μπορούν να εν δύναμη να απεικονιστούν στην καρτέλα Predictions Summary. Εδώ εμφανίζονται για ένα σύνολο έργων τόσο οι πραγματικές τιμές κόστους όσο και οι εκτιμώμενες τιμές πρόβλεψης για όλο το πλήθος μη-παραμετρικών και ημι-παραμετρικών μοντέλων που αναπτύχθηκαν.



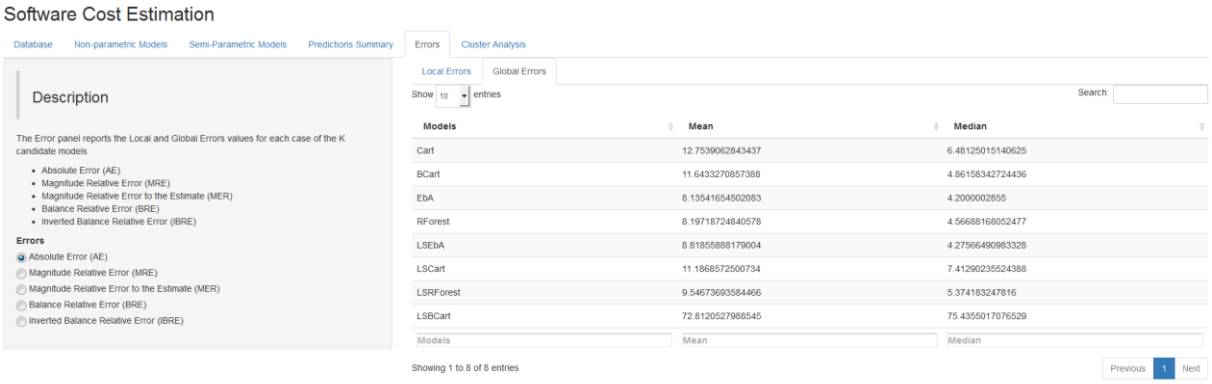

**Εικόνα 6.8:** Απεικόνιση αποτελεσμάτων όλων των μη-παραμετρικών και ημι-παραμετρικών μοντέλων .

Ακολουθώντας γραμμικά τις καρτέλες καταλήγουμε στην καρτέλα Errors που φέρει δύο υπο-καρτέλες μια για τα τοπικά λάθη (Local Errors) και μια για τα καθολικά λάθη (Global Errors). Εδώ δίνεται η δυνατότητα στον χρήστη να επιλέξει ένα από τα διαθέσιμα μέτρα ακριβείας τα οποία είναι τα Absolute Error (AE), Magnitude Relative Error (MRE), Magnitude Relative Error to the Estimate (MER), Balance Relative Error (BRE) και Inverted Balance Relative Error (IBRE).

Πρέπει να σημειωθεί ότι τα αποτελέσματα που θα ληφθούν από την σύγκριση μπορεί να μην συμφωνούν μεταξύ, έτσι θα είναι πιο δόκιμο να χρησιμοποιηθούν περισσότερα του ενός μέτρου ακριβείας ώστε ο χρήστης να μπορέσει να έχει μια πλήρη εικόνα του σφάλματος πρόβλεψης.

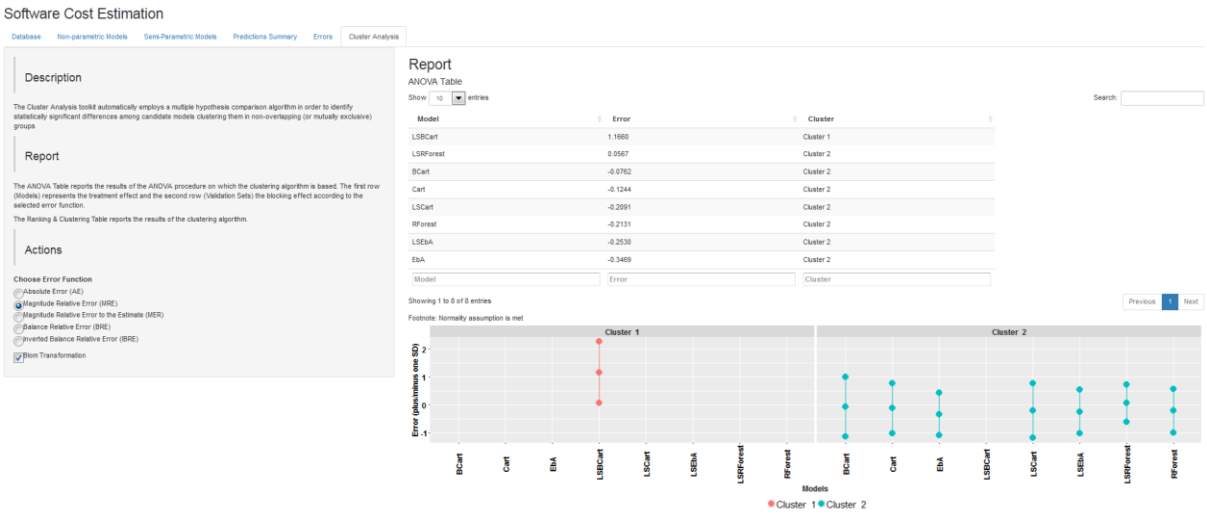



**Εικόνα 6.9:** Επιλογή τοπικής συναρτήσεως σφάλματος.




**Εικόνα 6.10:** Επιλογή καθολικού μέτρου ακριβείας.

Στην τελική καρτέλα Cluster Analysis πραγματοποιείται η διαδικασία της συσταδοποίησης για το επιλεγμένο σύνολο δεδομένων και για κάθε ένα μέτρο ακριβείας. Αυτή η καρτέλα αξιοποιεί στο παρασκήνιο τον αλγόριθμο Scott-Knott και με κάποια συστατικά στοιχεία όπως ο καμβάς σχεδίασης, εμφανίζονται γραφήματα με συστάδες. Εκτός αυτών, εμφανίζεται και ένας πίνακας με τα αποτελέσματα της ανάλυσης ANOVA όπου φέρει τρεις στήλες. Αυτή που αναφέρει το όνομα του μοντέλου που κατηγοριοποιήθηκε, την δεύτερη που αφορά το λάθος πρόβλεψης σε σχέση με πραγματική προσπάθεια (effort) και την στήλη Cluster όπου προσδιορίζει την πραγματική ομάδα στην οποία έχει κατηγοριοποιηθεί μοναδικά το εκάστοτε μοντέλο.




**Εικόνα 6.11:** Παρουσίαση αποτελεσμάτων συσταδοποίησης.

# Κεφάλαιο 7
# Πειραματική Διαδικασία

Στην παρούσα ενότητα περιγράφονται οι λεπτομέρειες που αφορούν τον πειραματικό σχεδιασμό της παρούσας Διατριβής. Τα στοιχεία που λήφθηκαν υπόψη για την δημιουργία του πειραματικού περιβάλλοντος ήταν τα εξής:

- Χρησιμοποιήθηκαν τέσσερις μη-παραμετρικές και ημι-παραμετρικές μέθοδοι.
- Γίνεται χρήση έξι διαφορετικών συνόλων δεδομένων, ώστε να μελετηθεί η συμπεριφορά των μοντέλων σε διάφορα σύνολα δεδομένων που έχουν χρησιμοποιηθεί εκτενώς στην βιβλιογραφία.
- Αξιοποιούνται πέντε μέτρα ακριβείας για την μέτρηση του σφάλματος πρόβλεψης.

Εκτός αυτών, τα πειράματα λαμβάνουν υπόψη την επίδραση που έχει σε κάθε περίπτωση ο διαχωρισμός του συνόλου δεδομένων σε σύνολα εκπαίδευσης και σύνολα δοκιμών.

## 7.1 Σύνολα Δεδομένων

Στα πλαίσια της πειραματικής διαδικασίας που ακολουθήθηκε και για την σύγκριση των μεθόδων ΕΚΛ, μέσω του υπολογισμού σφαλμάτων εκτίμησης, έγινε χρήση όπως προαναφέραμε, έξι συνόλων, τα οποία περιλαμβάνουν δεδομένα από πραγματικά έργα. Κάθε σύνολο δεδομένων που μελετήθηκε περιλαμβάνει ένα αριθμό ανεξάρτητων μεταβλητών (cost-drivers) και μια εξαρτημένη μεταβλητή (effort).

### 7.1.1 Σύνολο Δεδομένων COCOMO81

Το σύνολο δεδομένων COCOMO81 ((Boehm, 1981) είναι μια διαδεδομένη βάση για την εκτίμηση κόστους, η οποία χρησιμοποιήθηκε για τη βαθμονόμηση του γνωστού μοντέλου COCOMO και περιλαμβάνει διάφορα έργα λογισμικού της NASA. Αποτελείται

από 63 ολοκληρωμένα έργα λογισμικού, τα οποία περιλαμβάνουν 22 ανεξάρτητες μεταβλητές κόστους, από αυτές οι δύο είναι συνεχείς, οι δεκαεπτά διάταξης και οι τρεις ονομαστικής κλίμακας.

| Μεταβλητή | Περιγραφή | Επίπεδα |
|---|---|---|
| **Productivity** | Παραγωγικότητα ομάδας ανάπτυξης | Συνεχής |
| **Duration** | Διάρκεια σε μήνες | Συνεχής |
| **Year** | Χρονολογία ολοκλήρωσης του έργου | Συνεχής |
| **Type** | Τύπος έργου | 1=BUS<br>2=CTL<br>3=HMI<br>4=SCI<br>5=SUP<br>6=SYS |
| **Type_c** | Τύπος υπολογιστή | 1=MAX<br>2=MID<br>3=MIN<br>4=MIC |
| **Rmode** | Τρόπος ανάπτυξης λογισμικού | 1=Embedded |
| **Rely** | Απαιτούμενη αξιοπιστία λογισμικού | |
| **Data** | Μέγεθος βάσης δεδομένων | |
| **Cplx** | Πολυπλοκότητα | |
| **Time** | Χρονικοί περιορισμοί εκτέλεσης | |
| **Stor** | Περιορισμοί στην κύρια μνήμη | |
| **Virt** | Αλλαγές στο σύστημα HW/SW | |
| **Turn** | Χρόνος απόκρισης υπολογιστή | 1=Very low<br>2=Low<br>3=Nominal<br>4=High<br>5=Very high<br>6=Extra high |
| **Acap** | Ικανότητα αναλυτών | |
| **Aexp** | Εμπειρία σε εφαρμογές | |
| **Pcap** | Ικανότητα προγραμματιστών | |
| **Vexp** | Εμπειρία με το σύστημα HW/SW | |
| **Lexp** | Εμπειρία στις γλώσσες προγραμματισμού | |
| **Cont** | Συνέχεια προσωπικού | |
| **Modp** | Χρήση μοντέρνων πρακτικών προγραμματισμού | |
| **Tool** | Χρήση εργαλείων προγραμματισμού | |
| **Sced** | Πίεση από το χρονοδιάγραμμα ανάπτυξης | |
| **Rvol** | Αλλαγές στις απαιτήσεις | |

**Πίνακας 7.1:** Περιγραφή του συνόλου δεδομένων COCOMO81

### 7.1.2 Σύνολο Δεδομένων NASA93

Το σύνολο δεδομένων NASA93 (NASA93, 2007) είναι ένα μικρό σύνολο δεδομένων, που αφορά έργα λογισμικού. Αποτελείται από δεδομένα για 93 έργα λογισμικού και εμπεριέχει 23 ανεξάρτητες μεταβλητές και 1 εξαρτημένη μεταβλητή, την πραγματική

προσπάθεια (actual effort), η οποία καταβλήθηκε για την ανάπτυξη των έργων μετρημένη σε εργατομήνες (Idri & Abran, 2000).

| Μεταβλητή | Κλίμακα | Περιγραφή | Επίπεδα |
|---|---|---|---|
| **Effort** | Συνεχής | Προσπάθεια σε εργατομήνες | |
| **SLOC** | Συνεχής | Γραμμές κώδικα (χιλιάδες) | |
| **Forg** | Ονομαστική | Σύστημα αέρους ή εδάφους | Flight, Ground |
| **Mode** | Ονομαστική | Τρόπος ανάπτυξης λογισμικού | Embedded, Organic,<br><br>Semidetached |
| **Acap** | Διατάξιμη | Ικανότητα αναλυτών | |
| **Pcap** | Διατάξιμη | Ικανότητα προγραμματιστών | |
| **Aexp** | Διατάξιμη | Εμπειρία σε εφαρμογές | |
| **Modp** | Διατάξιμη | Χρήση μοντέρνων πρακτικών προγραμματισμού | |
| **Tool** | Διατάξιμη | Χρήση εργαλείων προγραμματισμού | |
| **Vexp** | Διατάξιμη | Εμπειρία σε συστήματα HW/SW | |
| **Lexp** | Διατάξιμη | Εμπειρία στις γλώσσες προγραμματισμού | Very low<br>Low<br>Nominal<br>High<br>Very high<br>Extra high |
| **Sced** | Διατάξιμη | Πίεση από χρονοδιάγραμμα ανάπτυξης | |
| **Stor** | Διατάξιμη | Περιορισμοί στην κύρια μνήμη | |
| **Data** | Διατάξιμη | Μέγεθος βάσης δεδομένων | |
| **Time** | Διατάξιμη | Χρονικοί περιορισμοί εκτέλεσης | |
| **Turn** | Διατάξιμη | Χρόνος απόκρισης υπολογιστή | |
| **Virt** | Διατάξιμη | Αλλαγές στο σύστημα HW/SW | |
| **Cplx** | Διατάξιμη | Πολυπλοκότητα | |
| **Rely** | Διατάξιμη | Απαιτούμενη αξιοπιστία λογισμικού | |

**Πίνακας 7.2:** Περιγραφή του συνόλου δεδομένων NASA93

### 7.1.3 Σύνολο Δεδομένων ALBRECHT

Το σύνολο δεδομένων ALBRECHT (Albrecht & Gaffney, 1983) περιέχει 24 έργα και αποτελείται από 7 μεταβλητές, 6 από αυτές είναι ανεξάρτητες και μία είναι η εξαρτημένη (κόστος). Η εξαρτημένη μεταβλητή έχει να κάνει με την προσπάθεια που καταβλήθηκε για την ολοκλήρωση της εφαρμογής σε ανθρωπομήνες (Albrecht & Gaffney, 1983).

| Μεταβλητή | Πλήρες Όνομα | Περιγραφή |
|---|---|---|
| **ProjNo** | Project Number | Μοναδικός αριθμός που χαρακτηρίζει κάθε έργο. |
| **Effort** | Effort | Πραγματική προσπάθεια σε εργατομήνες. |
| **In** | In | Αριθμός εξωτερικών δεδομένων εισόδου. |
| **Out** | Out | Αριθμός εξωτερικών δεδομένων εξόδου. |
| **File** | File | Αριθμός εξωτερικών/εσωτερικών αρχείων. |
| **Inq** | Inquires | Αριθμός των ερωτοαποκρίσεων των χρηστών. |
| **F.P** | Function Points | Βαθμοί Λειτουργίας. |
| **SLOC** | Source Lines of Code | Γραμμές Κώδικα. |

**Πίνακας 7.3:** Περιγραφή του συνόλου δεδομένων ALBRECHT

### 7.1.4 Σύνολο Δεδομένων DESHARNAIS

Το σύνολο Desharnais (Desharnais, 1988) αποτελείται από 77 έργα, τα οποία έχουν ληφθεί από έναν Καναδικό Οίκο Ανάπτυξης Λογισμικού στα τέλη της δεκαετίας του 1980 και που συγκεντρώθηκαν μέσα σε τέσσερα χρόνια. Τα δεδομένα αυτά χρησιμοποιήθηκαν σε πολλές μελέτες για την πρόβλεψη της προσπάθειας ή του κόστους ανάπτυξης. Ετσι το σύνολο περιλαμβάνει μετρήσεις της προσπάθειας (effort) που χρειάζεται για την υλοποίηση ενός έργου (project effort), τον χρόνο που απαιτήθηκε για να ολοκληρωθεί το έργο (project duration), το επίπεδο της εμπειρίας του προσωπικού ανάπτυξης και την διοίκηση του έργου, τον αριθμό των βασικών διεργασιών και των οντοτήτων (basic transactions and data entities) και τις ακατέργαστες μετρήσεις των Function Points (FP) (MacDonell & Gray, 1997).

| Μεταβλητή | Πλήρες Όνομα | Περιγραφή |
|---|---|---|
| **Project id** | Numeric identifier | Μοναδικός αριθμός που χαρακτηρίζει κάθε έργο. |
| **Effort** | Measured in hours | Πραγματική προσπάθεια σε ώρες. |
| **ExpEquip** | Team experience in years | Εμπειρία της ομάδας σε χρόνια. |
| **ExpProjMan** | Project manager's experience in years | Εμπειρία του Διαχειριστή Έργου σε χρόνια. |
| **Trans** | Number of transactions processed | Αριθμός ολοκληρωμένων συναλλαγών. |
| **Entities** | Number of entities | Αριθμός οντοτήτων. |
| **RawFPs** | Unadjusted function points | Μη διορθωμένα σημεία λειτουργίας. |
| **AdjFPs** | Adjusted function points | Προσαρμοσμένα σημεία λειτουργίας. |
| **DevEnv** | Development environment | Περιβάλλον ανάπτυξης. |
| **YearFin** | Year of completion | Έτος ολοκλήρωσης. |

**Πίνακας 7.4:** Περιγραφή του συνόλου δεδομένων DESHRNAIS

### 7.1.5 Σύνολο Δεδομένων MAXWELL

Το σύνολο Maxwell χαρακτηρίζεται ως ένα από τα πιο χρησιμοποιούμενα σύνολα δεδομένων στις έρευνες για την εκτίμηση κόστους λογισμικού παρά την πρόσφατη δημιουργία του. Αναπτύχθηκε για μια από τις μεγαλύτερες εμπορικές τράπεζες της Φιλανδίας και αποτελείται από 62 έργα (Maxwell et al., 1996).

| Μεταβλητή | Πλήρες Όνομα | Περιγραφή |
|---|---|---|
| **Effort** | Effort | Πραγματική προσπάθεια σε εργατομήνες. |
| **SizeFP** | Size | Μέγεθος λειτουργικών σημείων. |
| **Nlan** | Number of different development languages Used | Πόσες διαφορετικές γλώσσες προγραμματισμού χρησιμοποιούνται. |
| **T01** | customer participation | Συμμετοχή του πελάτη. |
| **T02** | development environment adequacy | Επάρκεια του περιβάλλοντος ανάπτυξης. |
| **T03** | staff availability | Διαθεσιμότητα του προσωπικού. |
| **T04** | Standards used | Χρήση των προτύπων. |
| **T05** | methods used | Χρήση των μεθόδων. |
| **T06** | tools used | Χρήση των εργαλείων. |
| **T07** | software's logical complexity | Λογική πολυπλοκότητα του Λογισμικού. |
| **T08** | requirements volatility | Μεταβλητότητα των απαιτήσεων. |
| **T09** | quality requirements | Απαιτήσεις ποιότητας. |
| **T10** | effciency requirements | Απαιτήσεις αποτελεσματικότητας. |
| **T11** | Installation requirements | Απαιτήσεις εγκαταστάσεις. |
| **T12** | staff analysis skills | Ανάλυση των ικανοτήτων του προσωπικού. |
| **T13** | staff application knowledge | Γνώση του προσωπικού στις εφαρμογές. |
| **T14** | staff tool skills | Ικανότητες προσωπικού στα εργαλεία. |
| **T15** | staff team skills | Ικανότητες ομάδας προσωπικού. |

**Πίνακας 7.5:** Περιγραφή του συνόλου δεδομένων MAXWELL

### 7.1.6 Σύνολο Δεδομένων MIYAZAKI

Το σύνολο Miyazaki αποτελείται από 48 έργα, τα οποία αναπτύχθηκαν από 20 διαφορετικές εταιρείες λογισμικού του Fujitsu Large και από την ομάδα χρηστών συστημάτων αυτής (Miyazaki, 1994).

| Μεταβλητή | Πλήρες Όνομα | Περιγραφή |
|---|---|---|
| **Effort** | Effort | |
| **Size** | Μέγεθος λειτουργικών σημείων. | Size |
| **SCRN** | The number of different input or output screens | Ο αριθμός των διαφορετικών οθονών εισόδου ή εξόδου. |
| **FORM** | the number of different report forms | Ο αριθμός των διάφορων φορμών αναφοράς. |
| **FILE** | the number of different record format | Ο αριθμός των εγγραφών που έχουν διαφορετική διαμόρφωση. |
| **ESCRN** | EScreen | Ηλεκτρονική Οθόνη |
| **EFORM** | Electronic Form | Ηλεκτρονική Φόρμα |
| **EFILE** | Electronic File | Ηλεκτρονική Φόρμα |

**Πίνακας 7.6:** Περιγραφή του συνόλου δεδομένων MIYAZAKI

# 7.2 Χαρακτηριστικά της Πειραματικής διαδικασίας

Στην παρούσα ενότητα αναλύεται η πειραματική διαδικασία που ακολουθήθηκε για την πραγματοποίηση των συγκρίσεων καταλληλότητας και ακρίβειας των υπό μελέτη μοντέλων. Τα αποτελέσματα των πειραμάτων που διεξήχθησαν διερευνούν την ικανότητα πρόβλεψης των ημι-παραμετρικών μοντέλων που δημιουργήθηκαν έναντι των αντίστοιχων μη-παραμετρικών.

Οι τέσσερις από τις οκτώ μεθόδους που μελετήθηκαν, ανήκουν στην κατηγορία των μη-παραμετρικών μοντέλων. Τρεις εξ'αυτών ανήκουν στους αλγορίθμους μηχανικής μάθησης (Machine Learning), ενώ η μία μέθοδος πραγματοποιεί εκτίμηση με αναλογίες (EbA). Οι υπόλοιπες τέσσερις είναι υβριδικές μέθοδοι που συνδυάζουν γραμμικά και μη-γραμμικά χαρακτηριστικά, αξιοποιώντας την μέθοδο των ελαχίστων τετραγώνων που μελέτηθησαν οι Μήττας και Αγγελής (2010).

Για την αποτίμηση των μοντέλων που παράγει η κάθε μέθοδος, χρησιμοποιείται στην βιβλιογραφία κατά κόρων η Κ-πλη Διασταυρωμένη επικύρωση (k-fold cross validation), η οποία αποτελείται από τα ακόλουθα βήματα:

1. Χωρίζουμε τυχαία τα δεδομένα σε Κ ίσα υποσύνολα.
2. Κάνουμε εκπαίδευση με τα Κ-1 υποσύνολα και το τελευταίο χρησιμοποιείται για έλεγχο (test set).

3. Η διαδικασία επαναλαμβάνεται Κ φορές, κάθε φορά και με διαφορετικό σύνολο επαλήθευσης.
4. Τα Κ εκπαιδευμένα δίκτυα χρησιμοποιούνται για την τελική πρόβλεψη (π.χ πλειοψηφία ή μέσος όρος).

Για τις ανάγκες της Διατριβής αξιοποιήσαμε μια ειδική περίπτωση της μεθόδου k-fold όπου πραγματοποιείται η διαμέριση του συνόλου δεδομένων σε δύο υποσύνολα, το πρώτο υποσύνολο θα χρησιμοποιηθεί σαν σύνολο εκπαίδευσης και το δεύτερο ως σύνολο ελέγχου. Η όλη διαδικασία επαναλαμβάνεται k φορές, Στην συνέχεια δημιουργούνται τα k ζεύγη συνόλων εκπαίδευσης και ελέγχου. Οι διαμερίσεις αυτών των συνόλων πραγματοποιούνται με την διαδικασία leave-one-out cross validation (LOOCV) (Kohavi, 1995). Με αυτή την προσέγγιση στην ουσία κάθε περίπτωση (έργο λογισμικού) του αρχικού συνόλου δεδομένων τοποθετείται στο σύνολο ελέγχου ακριβώς μια φορά και στην συνέχεια πραγματοποιείται η εκπαίδευση του μοντέλου δημιουργώντας ένα σύνολο εκπαίδευσης με όλες τις εναπομείνασες περιπτώσεις (έργα λογισμικού). Αυτό επαναλαμβάνεται για όλες τις περιπτώσεις ενώ στο τέλος λαμβάνουμε υπόψη το μέσο όρο των σφαλμάτων πρόβλεψης για κάθε σύνολο ελέγχου.

Η πειραματική μας διαδικασία βασίστηκε στη χρήση της εφαρμογής που αναπτύχθηκε, όπου και τροφοδοτήθηκαν σε αυτήν 6 διαφορετικά σύνολα δεδομένων με απώτερο σκοπό να παραχθούν αποτελέσματα προβλέψεων από το εκάστοτε μοντέλο και να πραγματοποιηθεί η συγκριτική αποτίμηση ως προς την ακρίβεια των προβλέψεων. Κάθε ένα από αυτά τα σύνολα περιλαμβάνει έναν αριθμό από έργα λογισμικού, έναν διαφορετικό αριθμό ανεξάρτητων μεταβλητών και μία εξαρτημένη μεταβλητή η οποία αντιπροσωπεύει την προσπάθεια που καταβλήθηκε στο έργο λογισμικού (effort).

Η σύγκριση των αποτελεσμάτων πραγματοποιήθηκε με τα μέτρα ακριβείας που περιγράφηκαν στο Κεφάλαιο 4. Αυτά ενσωματώθηκαν στην εφαρμογή και για κάθε μοντέλο και σύνολο δεδομένων υπολογίστηκαν οι τιμές του τοπικού σφάλματος. Από την χρήση των μέτρων τοπικού σφάλματος προέκυψαν τα καθολικά μέτρα ακριβείας. Αυτά υπολογίστηκαν με την χρήση στατιστικών μέτρων κεντρικής τάσης όπως είναι η μέση τιμή και η διάμεσος.

Τα ανωτέρω αποτελέσματα τροφοδοτήθηκαν στον αλγόριθμο Scott-Knott. Στην ουσία αποτελούν τις μέσες τιμές των τοπικών μέτρων ακριβείας ανά μοντέλο πρόβλεψης για

ένα συγκεκριμένο υπό μελέτη σύνολο δεδομένων. Ο αλγόριθμος Scott-Knott ταξινομώντας αυτές τις μέσες τιμές και πραγματοποιώντας στατιστικούς ελέγχους υποθέσεων (statistical hypothesis testing) καθορίζει την συστάδα στην οποία ανήκει το εκάστοτε μοντέλο πρόβλεψης. Επίσης, έχει την δυνατότητα της γραφικής αναπαράστασης των αποτελεσμάτων στα διαγράμματα που παράγει. Στον άξονα Χ παραθέτει τα μοντέλα ενώ στον άξονα Υ τις μετασχηματισμένες μέσες τιμές του λάθους.

Αξίζει να σημειωθεί πως οι μέθοδοι ταξινομούνται μοναδικά σε μια από τις συστάδες που δημιουργούνται, βάση της βαθμολογίας που τους αποδίδεται. Οι ομάδες θα λέγαμε πως χαρακτηρίζονται από ομοιογένεια καθώς τα μοντέλα στην ίδια ομάδα έχουν παρόμοια απόδοση. Το βασικό κριτήριο της συσταδοποίησης είναι η στατιστική σημαντικότητα των διαφορών μεταξύ των μέσων τιμών τους.

Για τον υπολογισμό των προβλέψεων, τα έξι από τα οκτώ μοντέλα που εμπεριέχει η εφαρμογή βαθμονομούνται μέσα από μια διαδικασία προσαρμογής (fitting) που χρησιμοποιεί την μέθοδο LOOCV. Αυτό γίνεται κατά την διάρκεια της εκτέλεσης του εκάστοτε μοντέλου σε πραγματικό χρόνο μέσω της εφαρμογής, αξιοποιώντας την μετρική που ονομάζεται Ρίζα του Μέσου Τετραγώνου του Σφάλματος (Root Mean Squared Error–RMSE). Ωστόσο υπάρχουν δύο μοντέλα το EbA και LSEbA, όπου η προσαρμογή τους πραγματοποιείται με διαφορετικό τρόπο. Πιο συγκεκριμένα, μέσω της εύρεση των κ-πλησιέστερων γειτόνων.

Επομένως η πειραματική διαδικασία χωρίστηκε σε δύο στάδια αποκλειστικά για την εύρεση των βέλτιστων κ-πλησιέστερων γειτόνων στα μοντέλα EbA και LSEbA. Έτσι προτού πραγματοποιηθεί ο αλγόριθμος συσταδοποίησης, δοκιμάστηκαν διάφορες τιμές από 0 έως 25 με βήμα 1, χρησιμοποιώντας όλα τα σύνολα δεδομένων έως ότου να βρεθεί από κοινού το πλήθος βέλτιστων κ-πλησιέστερων γειτόνων.

Στον παρακάτω πίνακα παρατίθενται οι τιμές που υπολογίστηκαν από την εφαρμογή. Ορισμένες από αυτές που διαφαίνονται με έντονο χρώμα είναι οι μέγιστες που αποδόθηκαν από τους αλγορίθμους ανά σύνολο δεδομένων. Αυτές αξιοποιήθηκαν κατά επέκταση στην επόμενη φάση της διαδικασίας.

| | Parametric / Non-Parametric Models | | | | | Semi-Parametric Models | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **DataSets** | | KNN=5 | KNN=10 | KNN=15 | KNN=20 | | KNN=5 | KNN=10 | KNN=15 | KNN=20 |
| Albrecht | **EbA** | 2 | 2 | **2** | 2 | **LSEbA** | 1 | 10 | **14** | 14 |
| Cocomo81 | | **2** | 2 | 2 | 2 | | **3** | 3 | 3 | 3 |
| Nasa93 | | 5 | **8** | 8 | 8 | | 4 | **4** | 4 | 4 |
| Maxwell | | 6 | **6** | 6 | 6 | | 4 | **4** | 4 | 4 |
| Desharnais | | 4 | 7 | **7** | 7 | | 3 | 9 | **9** | 9 |
| Miyazaki | | 9 | **9** | 9 | 9 | | 1 | **9** | 9 | 9 |

**Πίνακας 7.7:** Αριθμός κ-πλησιέστερων γειτόνων για κάθε Σύνολο Δεδομένων

Στο δεύτερο στάδιο αφενός αξιοποιώντας τις ανωτέρω τιμές και αφετέρου την δυνατότητα που δίνει η εφαρμογή για τον υπολογισμό εκτιμήσεων όλων των μοντέλων συγκεντρωτικά για όλα τα σύνολα δεδομένων υπολογίσαμε τις συστάδες.

# 7.3 Εφαρμογή των Μοντέλων στα Σύνολα Δεδομένων

Στις ενότητες που ακολουθούν παραθέτουμε τα αποτελέσματα της πειραματικής διαδικασίας που εφαρμόστηκε όπου ήταν όμοια για κάθε σύνολο δεδομένων. Έτσι σε κάθε σύνολο δεδομένων αφότου υπολογίστηκαν οι εκτιμήσεις για το πλήρες πλήθος των μοντέλων ΕΚΛ που υποστηρίζονται (μη-παραμετρικά και ημι-παραμετρικά), υπολογίστηκαν τα τοπικά και τα καθολικά σφάλματα για όλα τα μέτρα ακριβείας και τα αποτελέσματα αυτών τροφοδοτούνταν κάθε φορά στον αλγόριθμο Scott-Knott. Εντούτοις στα μοντέλα EbA και το LSEbA όπου απαιτείται ο καθορισμός ενός αριθμού κ-πλησιέστερων γειτόνων χρησιμοποιήθηκαν οι τιμές του πίνακα 7.7.

Χρησιμοποιώντας την εφαρμογή, τροφοδοτήθηκαν διαδοχικά τα έξι σύνολα δεδομένων που μελετήθηκαν στην παρούσα Διατριβή. Έτσι μέσω της καρτέλας “Cluster Analysis” υπολογίστηκαν τα τοπικά και καθολικά σφάλματα, επιλέγοντας τα πέντε μέτρα ακριβείας ένα προς ένα χωρίς την χρήση του μετασχηματισμού Blom. Στον ακόλουθο Πίνακα 7.8, παρατίθενται συγκεντρωτικά τα αποτελέσματα που προέκυψαν από τα μέτρα ακριβείας για όλα τα σύνολα δεδομένων και για όλα τα μοντέλα. Σε κάθε στήλη ανά μέτρο ακριβείας σημειώνεται με έντονη γραμματοσειρά η μέθοδος με την καλύτερη απόδοση ανά σύνολο δεδομένων.

| Dataset | Models | MAE | MMRE | MMER | MBRE | MIBRE |
|---|---|---|---|---|---|---|
| **Albrecht** | EbA | 8.1354 | 0.7241 | 0.6402 | 1.0042 | 0.3601 |
| | Cart | 12.7539 | 1.4948 | 0.6945 | 1.7620 | 0.4273 |
| | Bagging | 12.6743 | 1.5712 | 0.5732 | 1.7288 | 0.4157 |
| | Random Forest | 8.1217 | 0.8715 | 0.6231 | 0.9541 | 0.3383 |
| | LSEbA | **6.0007** | **0.6186** | **0.4823** | **0.8036** | **0.2973** |
| | LSCart | 11.1869 | 0.8317 | 0.7337 | 1.1563 | 0.4091 |
| | LSBagging | 70.7162 | 18.5629 | 0.7681 | 18.5637 | 0.7673 |
| | LSRandomForest | 11.6035 | 1.1116 | 0.6231 | 1.3219 | 0.4128 |
| **Cocomo81** | EbA | **464.7540** | 2.9695 | 1.9736 | 4.3646 | 0.5786 |
| | Cart | 584.7137 | 2.6879 | 1.1682 | 3.2391 | 0.6171 |
| | Bagging | 587.9496 | 2.0880 | 0.9018 | 2.4075 | 0.5823 |
| | Random Forest | 579.3401 | 3.4699 | **0.7133** | 3.5730 | 0.6101 |
| | LSEbA | 476.1016 | 1.6368 | 2.3330 | 3.3982 | 0.5716 |
| | LSCart | 545.5483 | 1.3631 | 1.5132 | 2.2997 | 0.5765 |
| | LSBagging | 611.6016 | 1.7284 | 2.7408 | 3.8456 | 0.6237 |
| | LSRandomForest | 487.7447 | **1.0940** | 1.5698 | **2.1614** | **0.5024** |
| **Desharnais** | EbA | 2675.091 | 0.6626 | 0.5695 | 0.8638 | 0.3683 |
| | Cart | 2181.775 | 0.7639 | 0.4274 | 0.8539 | 0.3373 |
| | Bagging | 2362.375 | 0.8388 | 0.4722 | 0.9495 | 0.3615 |
| | Random Forest | 2206.035 | 0.6503 | 0.4302 | 0.7357 | 0.3448 |
| | LSEbA | **1731.083** | 0.4927 | **0.3715** | 0.5678 | 0.2964 |
| | LSCart | 2011.131 | 0.4558 | 0.4866 | 0.6194 | 0.3229 |
| | LSBagging | 9950.301 | 4.2764 | 0.6950 | 4.2890 | 0.6823 |
| | LSRandomForest | 1904.925 | **0.4252** | 0.4200 | **0.5485** | **0.2967** |
| **Maxwell** | EbA | 4502.185 | 0.9962 | 0.7419 | 1.3347 | 0.4033 |
| | Cart | 4591.513 | 0.8482 | 0.5982 | 1.0382 | 0.4081 |
| | Bagging | 4481.213 | 1.0424 | 0.6023 | 1.2194 | 0.4253 |
| | Random Forest | 4272.647 | 0.9692 | **0.4999** | 1.0730 | 0.3960 |
| | LSEbA | **3469.511** | **0.5034** | 0.5408 | **0.7104** | **0.3339** |
| | LSCart | 4439.883 | 0.7158 | 0.7541 | 1.0800 | 0.3900 |
| | LSBagging | 23230.411 | 8.6475 | 0.7747 | 8.6612 | 0.7611 |
| | LSRandomForest | 4112.239 | 0.6022 | 0.6416 | 0.8615 | 0.3824 |
| **Miyazaki94** | EbA | 57.9906 | 0.6527 | 0.8677 | 1.1230 | 0.3974 |
| | Cart | 94.2107 | 1.5892 | 0.7895 | 1.9276 | 0.4511 |
| | Bagging | 71.6092 | 1.4365 | 0.6799 | 1.6781 | 0.4383 |
| | Random Forest | 49.5398 | 0.5464 | 0.4804 | 0.7206 | 0.3062 |
| | LSEbA | **46.6827** | **0.4099** | **0.4270** | **0.5533** | **0.2836** |
| | LSCart | 62.0334 | 0.5053 | 0.7472 | 0.9083 | 0.3442 |
| | LSBagging | 68.9043 | 1.0091 | 2.4404 | 2.9656 | 0.4840 |
| | LSRandomForest | 55.0916 | 0.4349 | 0.5133 | 0.6453 | 0.3029 |
| **Nasa93** | EbA | 526.6522 | 1.5996 | 1.4813 | 2.6211 | 0.4598 |
| | Cart | 436.9306 | 1.6229 | 0.8950 | 2.0793 | 0.4386 |
| | Bagging | 351.1967 | 1.2117 | 0.4711 | 1.3134 | 0.3695 |
| | Random Forest | 323.0350 | 1.2383 | **0.4364** | 1.3435 | 0.3312 |
| | LSEbA | 245.8784 | **0.4930** | 0.4784 | **0.6848** | 0.2866 |
| | LSCart | 446.0624 | 1.4201 | 1.8973 | 2.8965 | 0.4209 |
| | LSBagging | 603.1700 | 0.8725 | 26.6298 | 26.6544 | 0.8480 |
| | LSRandomForest | **245.8697** | 0.5321 | 0.5009 | 0.7614 | **0.2716** |

**Πίνακας 7.8:** Συγκεντρωτικός πίνακας αποτελεσμάτων χωρίς τον μετασχηματισμό Blom

Από τα ανωτέρω διακρίνεται ένα σύνολο μοντέλων ΕΚΛ, το οποίο παρουσιάζει το μικρότερο σφάλμα και πολύ μικρή διακύμανση. Αυτά είναι με φθίνουσα σειρά

κατάταξης τα ημι-παραμετρικά μοντέλα LSEbA και LSRandomForest, όπως επίσης και τα μη-παραμετρικά μοντέλα RandomForest και EbA.

Ακολούθως επαναλάβαμε την παραπάνω διαδικασία αξιοποιώντας τον μετασχηματισμό Blom. Αυτό έχει σαν αποτέλεσμα να παραχθεί ο πίνακας 7.9. Σε κάθε στήλη ανά μέτρο ακριβείας σημειώνεται με έντονη γραμματοσειρά η μέθοδος με την καλύτερη απόδοση ανά σύνολο δεδομένων.

| Dataset | Models | MAE | MMRE | MMER | MBRE | MIBRE |
|---|---|---|---|---|---|---|
| **Albrecht** | EbA | -0.3007 | -0.2595 | 0.0036 | -0.2262 | -0.2262 |
| | CART | -0.0719 | -0.0469 | 0.1128 | -0.0071 | -0.0071 |
| | Bagging | -0.0930 | -0.0536 | -0.1122 | -0.0889 | -0.0889 |
| | Random Forest | -0.1836 | -0.1780 | 0.1760 | -0.2246 | -0.2246 |
| | LSEbA | **-0.5233** | **-0.5515** | **-0.4094** | **-0.5010** | **-0.5010** |
| | LSCART | -0.2042 | -0.1374 | 0.0120 | -0.1506 | -0.1506 |
| | LSBagging | 1.4221 | 1.2280 | 0.4678 | 0.0089 | 0.0089 |
| | LSRandom Forest | -0.0454 | -0.0010 | -0.2506 | 1.1894 | 1.1894 |
| **Cocomo81** | EbA | 0.0060 | 0.0172 | 0.0270 | 0.0728 | 0.0728 |
| | CART | 0.1850 | 0.2081 | 0.0595 | 0.1339 | 0.1339 |
| | Bagging | 0.0957 | 0.0853 | -0.1610 | -0.0969 | -0.0969 |
| | Random Forest | 0.3157 | 0.4032 | -0.1687 | 0.1195 | 0.1195 |
| | LSEbA | -0.1804 | -0.1983 | 0.1190 | -0.0162 | -0.0162 |
| | LSCART | -0.0968 | -0.1170 | 0.1011 | -0.0542 | -0.0542 |
| | LSBagging | 0.0050 | -0.0138 | 0.2502 | 0.1518 | 0.1518 |
| | LSRandom Forest | **-0.3302** | **-0.3847** | **-0.2276** | **-0.3107** | **-0.3107** |
| **Desharnais** | EbA | -0.1225 | -0.1010 | 0.6706 | -0.0801 | -0.0801 |
| | CART | -0.1892 | -0.1792 | -0.1474 | -0.1725 | -0.1725 |
| | Bagging | -0.0567 | -0.0392 | 0.0245 | -0.0394 | -0.0394 |
| | Random Forest | -0.1307 | -0.1051 | -0.1005 | -0.1210 | -0.1210 |
| | LSEbA | **-0.3066** | -0.2811 | **-0.2498** | **-0.2977** | **-0.2977** |
| | LSCART | -0.2473 | -0.2037 | -0.0357 | -0.1865 | -0.1865 |
| | LSBagging | 1.3520 | 1.1950 | -0.0245 | 1.1628 | 1.1628 |
| | LSRandom Forest | -0.2990 | **-0.2856** | -0.1468 | -0.2657 | -0.2657 |
| **Maxwell** | EbA | -0,1844 | -0.2009 | -0,0098 | -0.1341 | -0.1341 |
| | CART | -0,1327 | -0.1046 | -0.1178 | -0.1325 | -0.1325 |
| | Bagging | -0,0089 | 0.0141 | -0.0081 | -0.0218 | -0.0218 |
| | Random Forest | -0,1037 | -0.1037 | -0.1603 | -0.1471 | -0.1471 |
| | LSEbA | **-0,3734** | **-0.3632** | **-0.2569** | **-0.3627** | **-0.3627** |
| | LSCART | -0,3020 | -0.3002 | -0,0376 | -0.2279 | -0.2279 |
| | LSBagging | 1,3842 | 1.3251 | 0.6884 | 1.2678 | 1.2678 |
| | LSRandom Forest | -0,2790 | -0.2666 | -0,0979 | -0.2417 | -0.2417 |
| **Miyazaki94** | EbA | 0.0209 | 0.0266 | 0.2018 | 0.0801 | 0.0801 |
| | CART | 0.2390 | 0.3322 | 0.1521 | 0.2745 | 0.2745 |
| | Bagging | 0.2962 | 0.3545 | -0.0460 | 0.2233 | 0.2233 |
| | Random Forest | -0.1227 | -0.1008 | -0.2474 | -0.1620 | -0.1620 |
| | LSEbA | **-0.3072** | **-0.4021** | **-0.3205** | **-0.4108** | **-0.4108** |
| | LSCART | -0.1272 | -0.1688 | -0.0549 | -0.1447 | -0.1447 |
| | LSBagging | 0.2071 | 0.2370 | 0.5148 | 0.4164 | 0.4164 |
| | LSRandom Forest | -0.2062 | -0.2786 | -0.1999 | -0.2769 | -0.2769 |
| **Nasa93** | EbA | 0.1014 | 0.1173 | 0.0577 | 0.0558 | 0.0558 |
| | CART | 0.1150 | 0.1915 | -0.0187 | 0.0188 | 0.0188 |
| | Bagging | 0.0330 | 0.0401 | -0.2299 | -0.1568 | -0.1568 |
| | Random Forest | -0.0834 | -0.1373 | -0.3264 | -0.2842 | -0.2842 |
| | LSEbA | -0.2660 | -0.4049 | -0.4194 | -0.4452 | -0.4452 |
| | LSCART | 0.0236 | 0.0115 | -0.0407 | -0.0322 | -0.0322 |
| | LSBagging | 0.3570 | 0.6289 | 1.4358 | 1.3265 | 1.3265 |
| | LSRandom Forest | **-0.2805** | **-0.4472** | **-0.4584** | **-0.4825** | **-0.4825** |

**Πίνακας 7.9:** Συγκεντρωτικός πίνακας αποτελεσμάτων με τον μετασχηματισμό Blom

Από τα ανωτέρω διακρίνεται ένα σύνολο μοντέλων ΕΚΛ, το οποίο παρουσιάζει το μικρότερο σφάλμα και πολύ μικρή διακύμανση. Αυτά είναι με φθίνουσα σειρά κατάταξης τα ημι-παραμετρικά μοντέλα LSEbA και LSRandomForest.

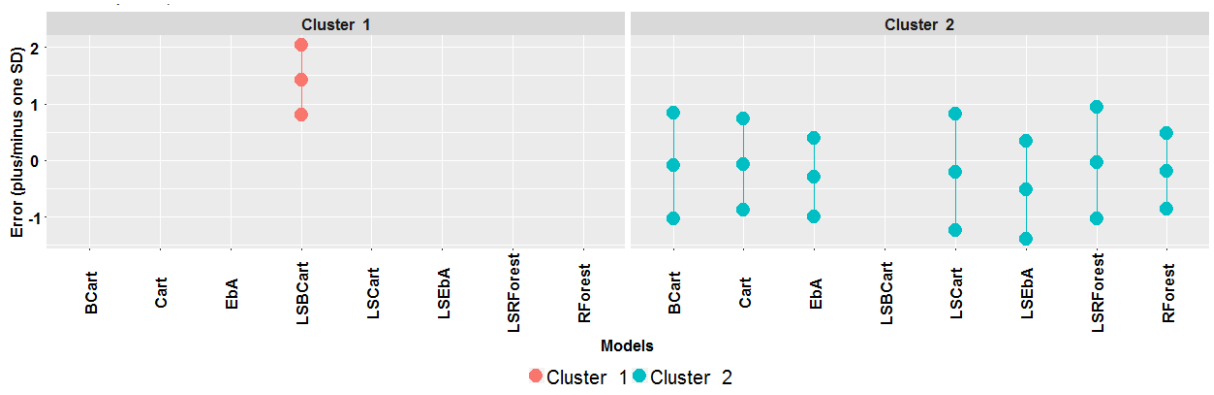
Στις ενότητες που ακολουθούν παραθέτουμε τα αποτελέσματα της πειραματικής διαδικασίας που εφαρμόστηκαν κάνοντας χρήση του μετασχηματισμού Blom ανά σύνολο δεδομένων.

### 7.3.1 Σύνολο Δεδομένων ALBRECHT

Το πρώτο σύνολο δεδομένων που τροφοδοτήθηκε στην εφαρμογή της παρούσας διατριβής ήταν το "Albrecht". Αφότου υπολογίστηκαν οι εκτιμήσεις για όλα τα μοντέλα που είναι διαθέσιμα στην εφαρμογή και ορίζοντας και τον αριθμό 14 για κ-πλησιέστερους γείτονες στα μοντέλα EbA και LSEbA, λάβαμε τα αποτελέσματα στην καρτέλα που εμπεριέχει την σύνοψη των προβλέψεων ("Predictions Summary") μαζί με την αρχική τιμή προσπάθειας (effort) του έργου.

Στην καρτέλα "Cluster Analysis", που αφορά την ανάλυση συστάδων, αφού ενεργοποιήσαμε την επιλογή που αφορά τον μετασχηματισμό Blom, υπολογίσαμε τα τοπικά και τα καθολικά σφάλματα επιλέγοντας τα υποστηριζόμενα μέτρα ακριβείας. Στο σημείο αυτό η εφαρμογή μας απέδωσε τις σχετικές τιμές αυτών, μαζί με το διάγραμμα που παράχθηκε από τον αλγόριθμο Scott-Knott.

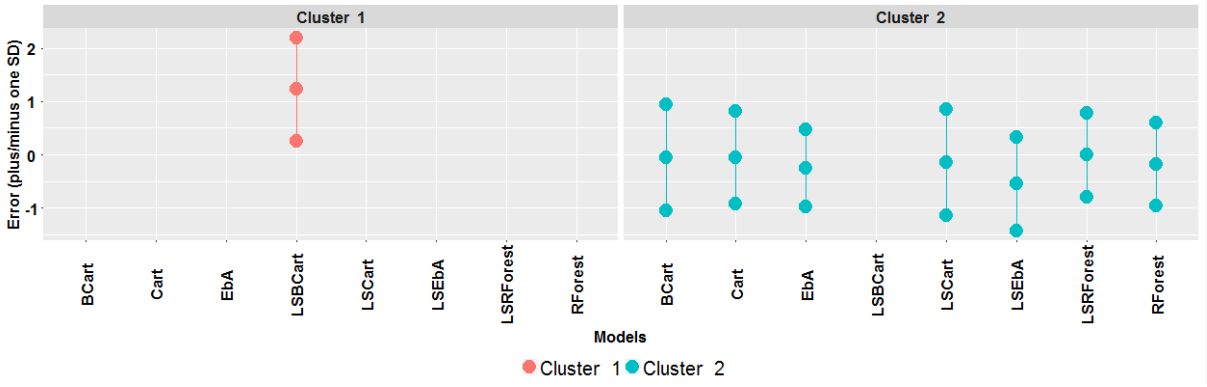
Η γραφική παράσταση για το μέτρο ακριβείας Absolute Error (AE) που αφορά την συσταδοποίηση παρατίθεται στην συνέχεια.




**Διάγραμμα 7.1:** Γραφική απεικόνιση του μέτρου AE.

Όπως μπορούμε να παρατηρήσουμε για το σύνολο δεδομένων Albrecht με το μέτρο ακριβείας Absolute Error (AE), τα 8 μοντέλα συσταδοποιούνται σε δύο συστάδες. Στην ίδια συστάδα εντάσσονται τα μοντέλα τα οποία δεν παρουσιάζουν στατιστικά σημαντική διαφορά μεταξύ τους. Από τα ανωτέρω διαφαίνεται πως στην συστάδα νούμερο 2 την καλύτερη απόδοση φαίνεται να έχει το ημι-παραμετρικό μοντέλο LSEbA. Αυτό καθώς το μοντέλο που κατατάσσεται σε αυτήν έχει το μικρότερο σφάλμα καθώς και πολύ μικρή διακύμανση.

Στην συνέχεια παρατίθεται η γραφική παράσταση που λάβαμε για το μέτρο ακριβείας Magnitude Relative Error (MRE).




**Διάγραμμα7.2:** Γραφική απεικόνιση του μέτρου MRE.

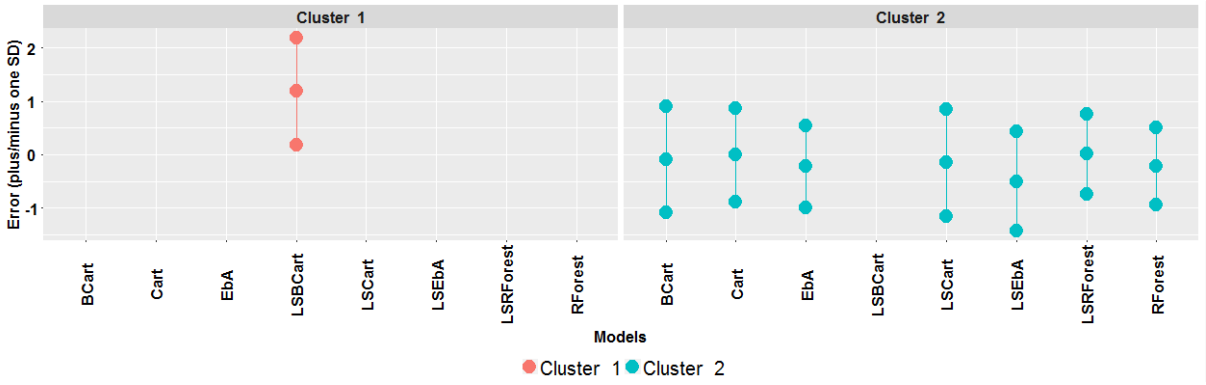
Όπως μπορούμε να παρατηρήσουμε και σε αυτό το μέτρο ακριβείας η συστάδα νούμερο 2, η οποία εμπεριέχει την μέθοδο LSEbA, παρουσιάζει καλύτερη απόδοση συγκριτικά με τα υπόλοιπα μοντέλα. Αυτό καθώς το μοντέλο που κατατάσσεται σε αυτήν έχει το μικρότερο σφάλμα και πολύ μικρή διακύμανση, όπως προηγουμένως.

Στην συνέχεια επιλέγοντας το μέτρο ακριβείας Magnitude Relative Error to the Estimate (MER) λάβαμε την ακόλουθη γραφική παράσταση:


**Διάγραμμα 7.3:** Γραφική απεικόνιση του μέτρου MER.

Μετά την συσταδοποίηση που πραγματοποιήθηκε για το Magnitude Relative Error to the Estimate (MER), δημιουργήθηκε μόλις μια συστάδα. Αξίζει να σημειωθεί ότι και σε αυτή την περίπτωση, ότι το μοντέλο LSEbA φαίνεται να παρουσιάζει την καλύτερη απόδοση.

Τα δύο τελευταία μέτρα ακριβείας που εφαρμόστηκαν για το συγκεκριμένο σύνολο δεδομένων ήταν το Balance Relative Error (BRE) και το Inverted Balance Relative Error (IBRE). Δεδομένου ότι χρησιμοποιήθηκε ο μετασχηματισμός Blom, τα αποτελέσματα ήταν πανομοιότυπα. Ακολούθως παρατίθεται η γραφική παράσταση των συστάδων.




**Διάγραμμα 7.4:** Γραφική απεικόνιση των μέτρων BRE/IBRE.

Από τα ανωτέρω, παρατηρήθηκε πως η συστάδα νούμερο 2, παρουσιάζει καλύτερη απόδοση με την συστάδα νούμερο 1. Επιπρόσθετα αξίζει να σημειωθεί ότι το μοντέλο LSEbA παρουσιάζει την καλύτερη απόδοση καθότι το σφάλμα του είναι μικρότερο από αυτό που παρουσιάζουν τα υπόλοιπα μοντέλα.

Η αποτίμηση πραγματοποιήθηκε ταξινομώντας τα μοντέλα βάση των τιμών καθολικού σφάλματος από το καλύτερο προς το χειρότερο. Αυτό έγινε ανά μέτρο ακριβείας και υπολογίστηκε η μέση κατάταξη ανά μοντέλο συνολικά. Τα αποτελέσματα συνοψίζονται ακολούθως.

| Albrecht | Models | MAE | MMRE | MMER | MBRE | MIBRE | Μέση Κατάταξη |
|---|---|---|---|---|---|---|---|
| | **EbA** | 2 | 2 | 4 | 2 | 2 | **2,4** |
| | CART | 6 | 6 | 6 | 6 | 6 | 6 |
| | Bagging | 5 | 5 | 3 | 5 | 5 | 4,6 |
| | **Random Forest** | 4 | 3 | 2 | 3 | 3 | **3** |
| | **LSEbA** | 1 | 1 | 1 | 1 | 1 | **1** |
| | LSCART | 3 | 4 | 5 | 4 | 4 | 4 |
| | LSBagging | 8 | 8 | 8 | 8 | 8 | 8 |
| | LSRandom Forest | 7 | 7 | 7 | 7 | 7 | 7 |

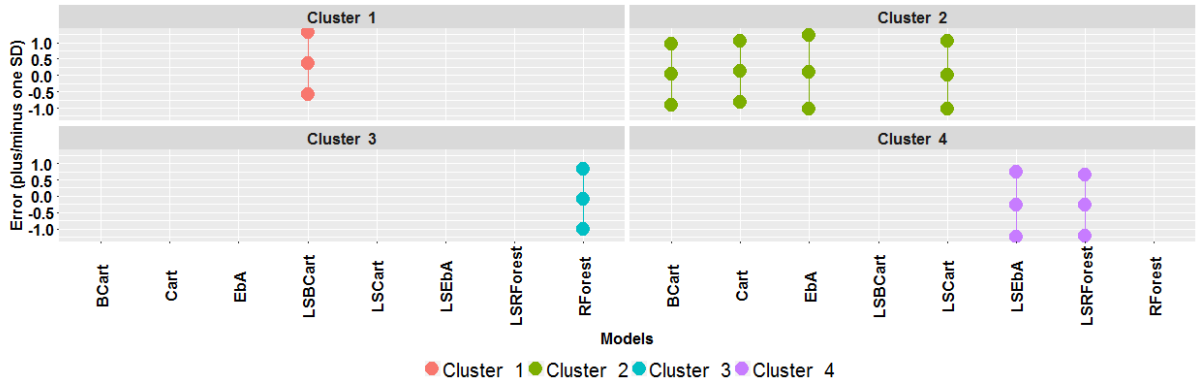
**Πίνακας 7.10:** Συγκεντρωτικός πίνακας κατάταξης συνόλου δεδομένων Albrecht

Σύμφωνα με τον πίνακα 7.10, η σειρά κατάταξης των τριών επικρατέστερων μεθόδων για το σύνολο δεδομένων Albrecht είναι η εξής: LSEbA, EbA και Random Forest.

### 7.3.2 Σύνολο Δεδομένων NASA93

Το δεύτερο σύνολο δεδομένων που τροφοδοτήθηκε στην εφαρμογή της παρούσας διατριβής ήταν το “Nasa93”. Αφότου υπολογίστηκαν οι εκτιμήσεις για όλα τα μοντέλα που είναι διαθέσιμα στην εφαρμογή και ορίζοντας τον αριθμό 8 για κ-πλησιέστερους γείτονες στα μοντέλα EbA και LSEbA.

Τα αποτελέσματα που λάβαμε για το μέτρο ακριβείας Absolute Error (AE) παρουσιάζονται στην παρακάτω γραφική παράσταση της συσταδοποίησης.




**Διάγραμμα 7.5:** Γραφική απεικόνιση του μέτρου AE.

Στην περίπτωση του συγκεκριμένου συνόλου δεδομένων δημιουργούνται 4 συστάδες. Η συστάδα που εμπεριέχει τα μοντέλα με τις καλύτερες αποδόσεις είναι η τέταρτη και σε αυτήν ανήκουν τα μοντέλα LSEbA και LSRForest.

Επιλέγοντας το μέτρο ακριβείας Magnitude Relative Error (MRE) η γραφική απεικόνιση είναι η ακόλουθη.

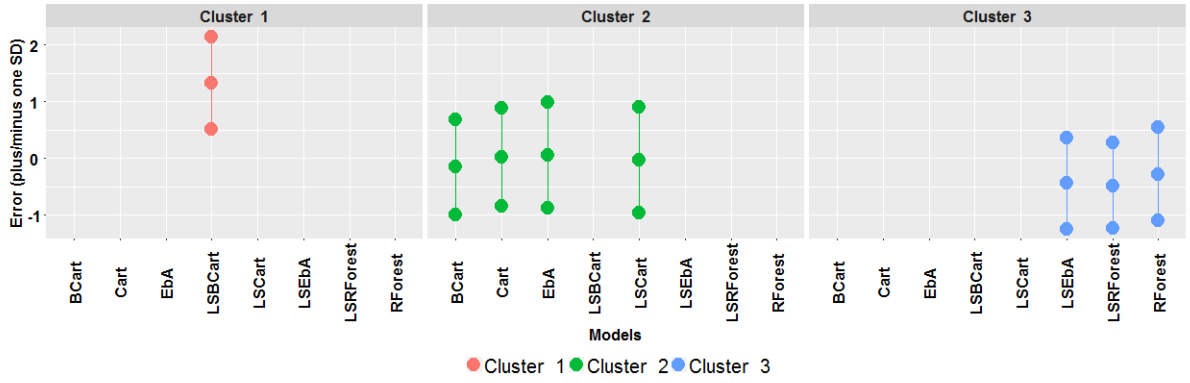

**Διάγραμμα 7.6:** Γραφική απεικόνιση του μέτρου MRE.

Σε αυτό το μέτρο ακριβείας παρόλο που τα αριθμητικά αποτελέσματα που λάβαμε είναι διαφορετικά, παρατηρείτε πως ο αριθμός των συστάδων παρέμεινε ο ίδιος εμπεριέχοντας τα ίδια μοντέλα.

Στην συνέχεια επιλέξαμε από την εφαρμογή το μέτρο ακριβείας Magnitude Relative Error to the Estimate (MER) και λάβαμε το ακόλουθο διάγραμμα.


**Διάγραμμα 7.7:** Γραφική απεικόνιση του μέτρου MER.

Τέλος για τα μέτρα ακριβείας το Balance Relative Error (BRE) και το Inverted Balance Relative Error (IBRE) η γραφική απεικόνιση είναι η παρακάτω.




**Διάγραμμα 7.8:** Γραφική απεικόνιση των μέτρων BRE/IBRE.

Σε αυτές τις περιπτώσεις δημιουργήθηκαν 3 συστάδες. Επίσης παρατηρήθηκε πως η συστάδα νούμερο 3 περιλαμβάνει με την καλύτερη απόδοση και δύο από τα καλύτερα είναι τα ημι-παραμετρικά LSEbA και LSRForest.

Όπως και στο πρώτο σύνολο δεδομένων πραγματοποιήθηκε η αποτίμηση των μοντέλων, ταξινομώντας τα βάση των τιμών καθολικού σφάλματος από το καλύτερο προς το χειρότερο. Αυτό έγινε ανά μέτρο ακριβείας και υπολογίστηκε η μέση κατάταξη ανά μοντέλο συνολικά. Τα αποτελέσματα συνοψίζονται ακολούθως.

| Nasa93 | Models | MAE | MMRE | MMER | MBRE | MIBRE | Μέση Κατάταξη |
|---|---|---|---|---|---|---|---|
| | EbA | 6 | 6 | 7 | 7 | 7 | 6,6 |
| | CART | 7 | 7 | 6 | 6 | 6 | 6,4 |
| | Bagging | 5 | 5 | 4 | 4 | 4 | 4,4 |
| | **Random Forest** | 3 | 3 | 3 | 3 | 3 | **3** |
| | **LSEbA** | 2 | 2 | 2 | 2 | 2 | **2** |
| | LSCART | 4 | 4 | 5 | 5 | 5 | 4,6 |
| | LSBagging | 8 | 8 | 8 | 8 | 8 | 8 |
| | **LSRandom Forest** | 1 | 1 | 1 | 1 | 1 | **1** |

**Πίνακας 7.11:** Συγκεντρωτικός πίνακας κατάταξης συνόλου δεδομένων Nasa93

Από τον παραπάνω πίνακα 7.11, η σειρά κατάταξης των τριών επικρατέστερων μεθόδων για το σύνολο δεδομένων Nasa93, είναι η εξής: LSRandom Forest, LSEbA και Random Forest.

### 7.3.3 Σύνολο Δεδομένων COCOMO81

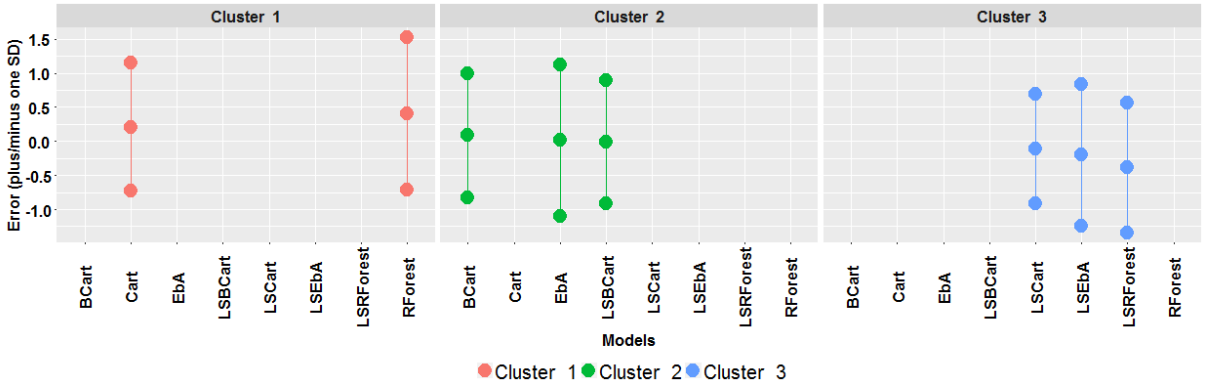
Το τρίτο σύνολο δεδομένων που τροφοδοτήθηκε στην εφαρμογή της παρούσας διατριβής ήταν το "Cocomo81". Αφότου υπολογίστηκαν οι εκτιμήσεις για όλα τα μοντέλα που είναι διαθέσιμα στην εφαρμογή και ορίζοντας τον αριθμό 3 για κ-πλησιέστερους γείτονες στα μοντέλα EbA και LSEbA, λάβαμε τις ακόλουθες γραφικές παραστάσεις για κάθε ένα από τα μέτρα ακριβείας.

Στην συνέχεια επιλέγοντας το μέτρο ακριβείας Absolute Error (AE) λάβαμε την ακόλουθη γραφική παράσταση:


**Διάγραμμα 7.9:** Γραφική απεικόνιση του μέτρου AE.

Στην προκειμένη περίπτωση παρατηρήθηκε ότι δημιουργήθηκαν 3 συστάδες και η καλύτερη συστάδα είναι η τρίτη. Η συγκεκριμένη εμπεριέχει τα ημι-παραμετρικά μοντέλα LSCart, LSEbA και LSRForest, με την LSRForest να είναι η αποδοτικότερη.

Στην συνέχεια επιλέχθηκε το μέτρο ακριβείας Magnitude Relative Error (MRE) και η γραφική του αναπαράσταση είναι η παρακάτω:




**Διάγραμμα 7.10:** Γραφική απεικόνιση του μέτρου MRE.

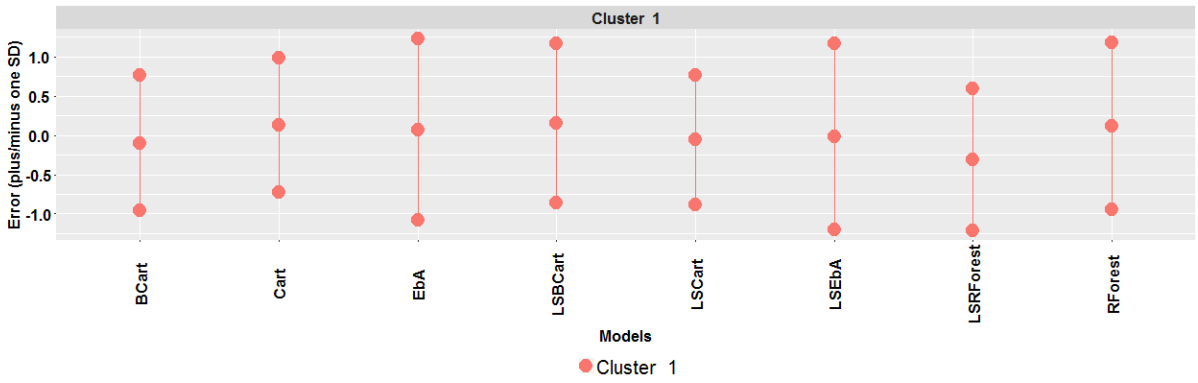
Όπως διαπιστώθηκε οι συστάδες που δημιουργήθηκαν ήταν και σε αυτήν την περίπτωση 3, με την Τρίτη συστάδα να εμπεριέχει τα αποδοτικότερα μοντέλα. Αυτά ήταν τα 3 ημι-παραμετρικά μοντέλα LSCart, LSEbA και LSRForest.

Το επόμενο μέτρο ακριβείας που επιλέχθηκε ήταν το Magnitude Relative Error to the Estimate (MER) όπου επέφερε τα παρακάτω:


**Διάγραμμα 7.11:** Γραφική απεικόνιση του μέτρου MER.

Εδώ παρατηρήθηκε η δημιουργία 2 συστάδων. Στην συστάδα νούμερο 3 συγκαταλέχθηκα τα μοντέλα με την καλύτερη απόδοση τα οποία ήταν τα BCart, RForest και LSRForest, με το ημι-παραμετρικό μοντέλο να παρουσιάζει την καλύτερη απόδοση όλων.

Τέλος επιλέχτηκαν τα μέτρα ακριβείας Balance Relative Error (BRE) και το Inverted Balance Relative Error (IBRE) και τα αποτελέσματα είναι τα παρακάτω:




**Διάγραμμα 7.12:** Γραφική απεικόνιση των μέτρων BRE/IBRE.

Σε αυτή την περίπτωση δημιουργήθηκε μία μόνο συστάδα με καλύτερο μοντέλο το ημι-παραμετρικό LSRForest.

Κατά αντιστοιχία για το εν λόγω μοντέλο πραγματοποιήθηκε αποτίμηση ταξινομώντας τα μοντέλα βάση των τιμών καθολικού σφάλματος από το καλύτερο προς το χειρότερο. Αυτό έγινε ανά μέτρο ακριβείας και υπολογίστηκε η μέση κατάταξη ανά μοντέλο συνολικά. Τα αποτελέσματα παρουσιάζονται στον πίνακα 7.12

| Cocomo81 | Models | MAE | MMRE | MMER | MBRE | MIBRE | Μέση Κατάταξη |
|---|---|---|---|---|---|---|---|
| | EbA | 5 | 5 | 4 | 5 | 5 | 4,8 |
| | CART | 7 | 7 | 5 | 7 | 7 | 6,6 |
| | **Bagging** | 6 | 6 | 3 | 2 | 2 | **3,8** |
| | Random Forest | 8 | 8 | 2 | 6 | 6 | 6 |
| | **LSEbA** | 2 | 2 | 7 | 4 | 4 | **3,8** |
| | **LSCART** | 3 | 3 | 6 | 3 | 3 | **3,6** |
| | LSBagging | 4 | 4 | 8 | 8 | 8 | 6,4 |
| | **LSRandom Forest** | 1 | 1 | 1 | 1 | 1 | **1** |

**Πίνακας 7.12:** Συγκεντρωτικός πίνακας κατάταξης συνόλου δεδομένων Cocomo81

Από τα παραπάνω διαφαίνεται η σειρά κατάταξης των επικρατέστερων μοντέλων που είναι η ξής: LSRandom Forest, η LSCART και στην τρίτη θέση η LSEbA με την Bagging.

### 7.3.4 Σύνολο Δεδομένων DESHARNAIS

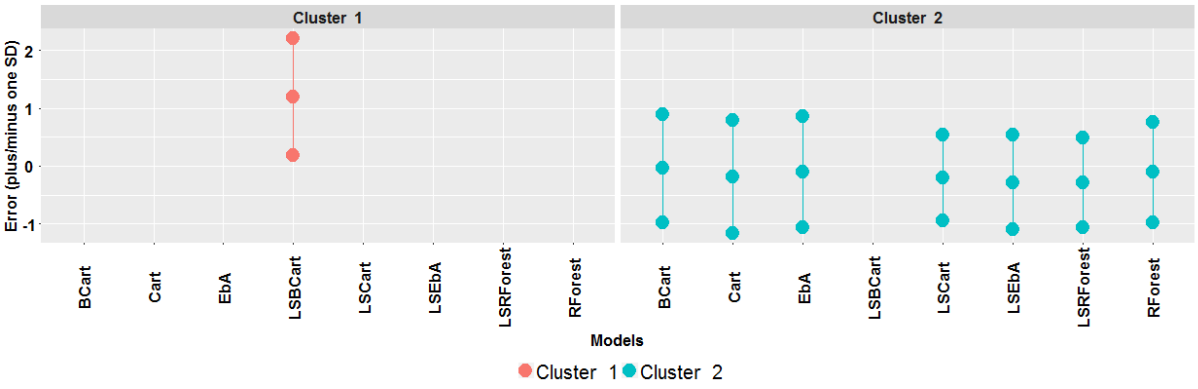
Το τέταρτο σύνολο δεδομένων που τροφοδοτήθηκε στην εφαρμογή της παρούσας διατριβής ήταν το “Deshrnais”. Αφότου υπολογίστηκαν οι εκτιμήσεις για όλα τα μοντέλα που είναι διαθέσιμα στην εφαρμογή και ορίζοντας τον αριθμό 9 για κ-πλησιέστερους γείτονες στα μοντέλα EbA και LSEbA, λάβαμε τις γραφικές απεικονίσεις που ακολουθούν.

Ακολούθως παρατίθενται τα αποτελέσματα που λάβαμε για το μέτρο ακριβείας Absolute Error (AE).


**Διάγραμμα 7.13:** Γραφική απεικόνιση του μέτρου AE.

Όπως παρατηρήθηκε δημιουργήθησαν 2 συστάδες και η συστάδα νούμερο 2 εμπεριείχε τα μοντέλα με την καλύτερη απόδοση. Από αυτά το ημι-παραμετρικό μοντέλο LSEbA είχε την καλύτερη απόδοση.

Στην συνέχεια επιλέγοντας το μέτρο ακριβείας Magnitude Relative Error (MRE) λάβαμε τα παρακάτω.




**Διάγραμμα 7.14:** Γραφική απεικόνιση του μέτρου MRE.

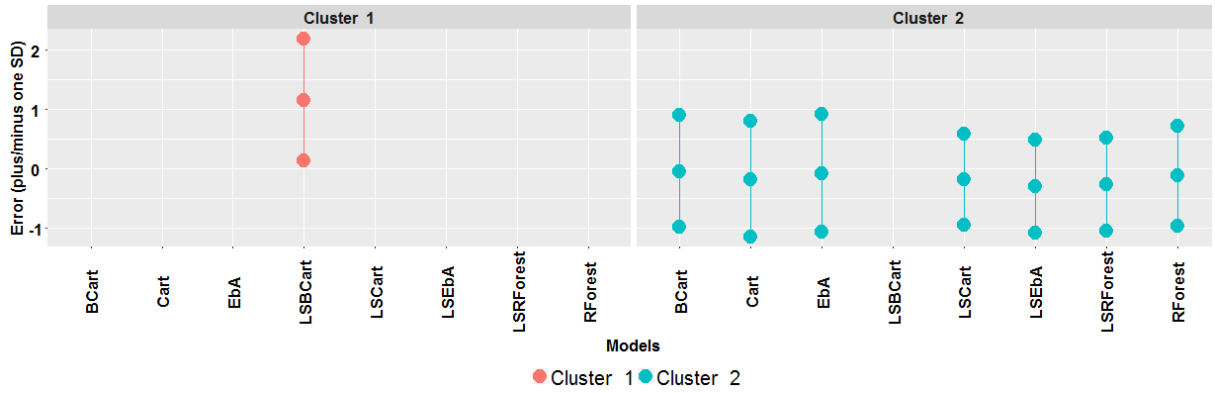
Στην προκειμένη περίπτωση δημιουργήθηκαν 2 συστάδες, στην δεύτερη συστάδα εμπεριέχονται τα μοντέλα με την καλύτερη απόδοση. Από αυτά τα 3 ημι-παραμετρικά μοντέλα παρουσίασαν την καλύτερη απόδοση.

Στην συνέχεια επιλέχθηκε το μέτρο ακριβείας Magnitude Relative Error to the Estimate (MER) και τα αποτελέσματα που λάβαμε είναι τα παρακάτω.


**Διάγραμμα 7.15:** Γραφική απεικόνιση του μέτρου MER.

Στην προκειμένη περίπτωση δημιουργήθηκαν 2 συστάδες, στην συστάδα νούμερο 2 εμπεριέχονται τα μοντέλα με την καλύτερη απόδοση. Το ημι-παραμετρικό μοντέλο LSEbA παρουσιάζει την καλύτερη απόδοση.

Τέλος επιλέχτηκαν τα μέτρα ακριβείας Balance Relative Error (BRE) και το Inverted Balance Relative Error (IBRE) και τα αποτελέσματα είναι τα παρακάτω:




**Διάγραμμα 7.16:** Γραφική απεικόνιση των μέτρων BRE/IBRE.

Σε αυτά τα δύο μέτρα ακριβείας δημιουργήθηκαν δύο συστάδες, στην δεύτερη συστάδα εμπεριέχεται το ημι-παραμετρικό μοντέλο LSEbA που παρουσιάζει την καλύτερη απόδοση όλων.

Ακολούθως για το σύνολο δεδομένων Desharnais παρατίθεται ο συγκεντρωτικός πίνακας με την μέση κατάταξη των μοντέλων.

| Desharnais | Models | MAE | MMRE | MMER | MBRE | MIBRE | Μέση Κατάταξη |
|---|---|---|---|---|---|---|---|
| | EbA | 6 | 6 | 8 | 6 | 6 | 6,4 |
| | **CART** | 4 | 4 | 2 | 4 | 4 | 3,6 |
| | Bagging | 7 | 7 | 7 | 7 | 7 | 7 |
| | Random Forest | 5 | 5 | 3 | 5 | 5 | 4,6 |
| | **LSEbA** | 1 | 2 | 1 | 1 | 1 | **1,2** |
| | **LSCART** | 3 | 3 | 5 | 3 | 3 | **3,4** |
| | LSBagging | 8 | 8 | 6 | 8 | 8 | 7,6 |
| | **LSRandom Forest** | 2 | 1 | 4 | 2 | 2 | **2,2** |

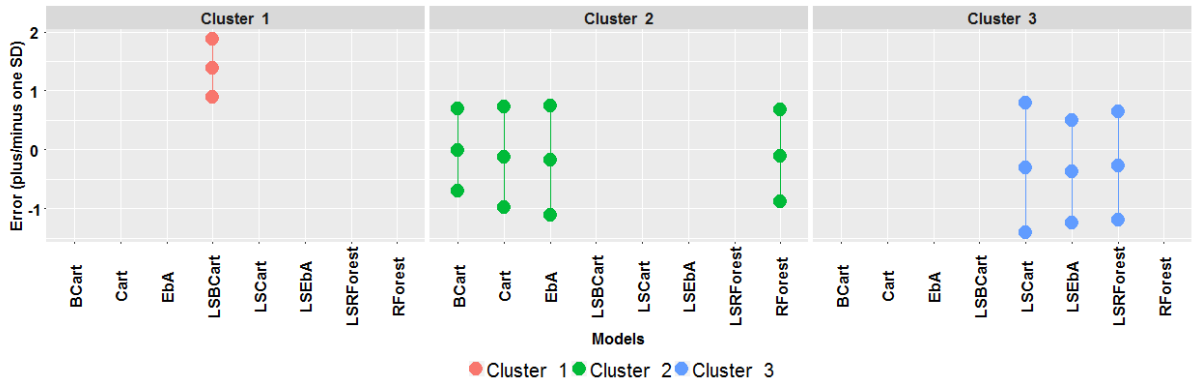
**Πίνακας 7.13:** Συγκεντρωτικός πίνακας κατάταξης συνόλου δεδομένων Desharnais

Η σειρά κατάταξης των τριών επικρατέστερων μεθόδων για το εν λόγω σύνολο είναι η εξής: LSEbA, LSRandom Forest και στην τρίτη θέση από κοινού η LSCART.

### 7.3.5 Σύνολο Δεδομένων MAXWELL

Το πέμπτο σύνολο δεδομένων που τροφοδοτήθηκε στην εφαρμογή ήταν το “Maxwell”. Αφότου υπολογίστηκαν οι εκτιμήσεις για όλα τα μοντέλα που είναι διαθέσιμα στην εφαρμογή και ορίζοντας τον αριθμό 6 για κ-πλησιέστερους γείτονες στα μοντέλα EbA και LSEbA, λάβαμε τις παρακάτω γραφικές απεικονίσεις.

Τα πρώτο το μέτρο ακριβείας που επιλέχθηκε ήταν το Absolute Error (AE) τα αποτελέσματα που λάβαμε είναι τα παρακάτω..




**Διάγραμμα 7.17:** Γραφική απεικόνιση του μέτρου AE.

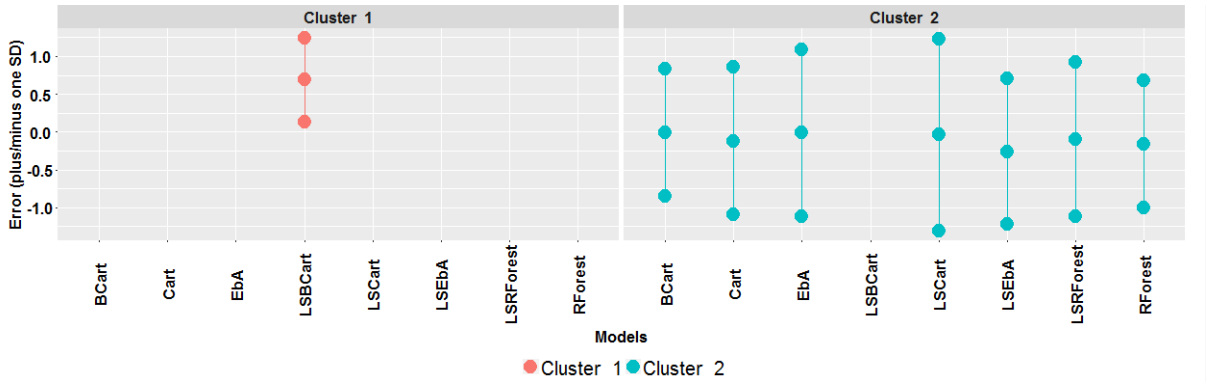
Όπως διαφαίνεται στην τρίτη συστάδα εμπεριέχονται τρία ημι-παραμετρικά μοντέλα που παρουσιάζουν την καλύτερη απόδοση.

Το επόμενο μέτρο που επιλέχθηκε ήταν το Magnitude Relative Error (MRE) τα αποτελέσματα που λάβαμε είναι τα παρακάτω.


**Διάγραμμα 7.18:** Γραφική απεικόνιση του μέτρου MRE.

Από τα παραπάνω δημιουργήθηκαν 3 συστάδες, στην τρίτη εμπεριέχονται τα τέσσερα μοντέλα με την καλύτερη απόδοση. Αξίζει να σημειωθεί ότι 3 από τα τέσσερα μοντέλα είναι ημι-παραμετρικά.

Το επόμενο μέτρο ακριβείας ήταν το Magnitude Relative Error to the Estimate (MER) και τα αποτελέσματα του είναι τα παρακάτω.




**Διάγραμμα 7.19:** Γραφική απεικόνιση του μέτρου MER.

Παρατηρήθηκε η δημιουργία δύο συστάδων στην δεύτερη εμπεριέχονται τα μοντέλα με την καλύτερη απόδοση. Το μοντέλο με την καλύτερη απόδοση είναι το ημι-παραμετρικό LSEbA.

Τέλος επιλέχτηκαν τα μέτρα ακριβείας Balance Relative Error (BRE) και το Inverted Balance Relative Error (IBRE) και τα αποτελέσματα είναι τα παρακάτω.


**Διάγραμμα 7.20:** Γραφική απεικόνιση των μέτρων BRE/IBRE.

Παρατηρήθηκε η δημιουργία 2 συστάδων, στην δεύτερη συστάδα εμπεριέχονται τρία ημι-παραμετρικά μοντέλα που παρουσιάζουν την καλύτερη απόδοση όλων.

Για το σύνολο δεδομένων Maxwell η σειρά κατάταξης των μοντέλων είναι η παρακάτω.

| Maxwell | Models | MAE | MMRE | MMER | MBRE | MIBRE | Μέση Κατάταξη |
|---|---|---|---|---|---|---|---|
| | EbA | 4 | 4 | 6 | 5 | 5 | 4,8 |
| | CART | 5 | 5 | 3 | 6 | 6 | 5 |
| | Bagging | 7 | 7 | 7 | 7 | 7 | 7 |
| | Random Forest | 6 | 6 | 2 | 4 | 4 | 4,4 |
| | **LSEbA** | 1 | 1 | 1 | 1 | 1 | **1** |
| | **LSCART** | 2 | 2 | 5 | 3 | 3 | **3** |
| | LSBagging | 8 | 8 | 8 | 8 | 8 | 8 |
| | **LSRandom Forest** | 3 | 3 | 4 | 2 | 2 | **2,8** |

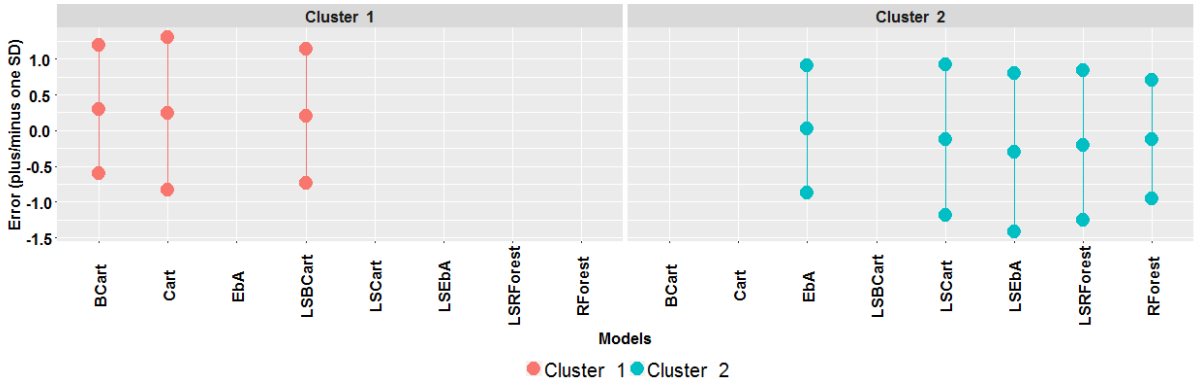
**Πίνακας 7.14:** Συγκεντρωτικός πίνακας κατάταξης συνόλου δεδομένων **Maxwell**

Οι τρεις επικρατέστερες μέθοδοι για το εν λόγω μοντέλο είναι η LSEbA, στην δεύτερη LSRandom Forest και στην τρίτη θέση η LSCART.

### 7.3.6 Σύνολο Δεδομένων MIYAZAKI

Το έκτο και τελευταίο σύνολο δεδομένων που τροφοδοτήθηκε στην εφαρμογή ήταν το "Miyazaki". Αφότου υπολογίστηκαν οι εκτιμήσεις για όλα τα μοντέλα που είναι διαθέσιμα στην εφαρμογή και ορίζοντας τον αριθμό 9 για κ-πλησιέστερους γείτονες στα μοντέλα EbA και LSEbA, λάβαμε τα αποτελέσματα των γραφικών αναπαραστάσεων που ακολούν.

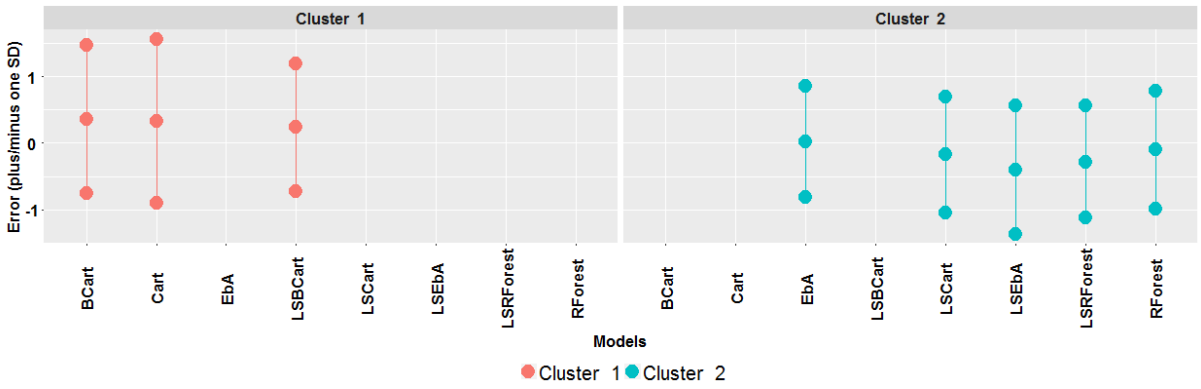
Το πρώτο μέτρο ακριβείας που επιλέχθηκε ήταν το Absolute Error (AE) και η γραφική αναπαράσταση είναι η παρακάτω:




**Διάγραμμα 7.21:** Γραφική απεικόνιση του μέτρου ακριβείας AE.

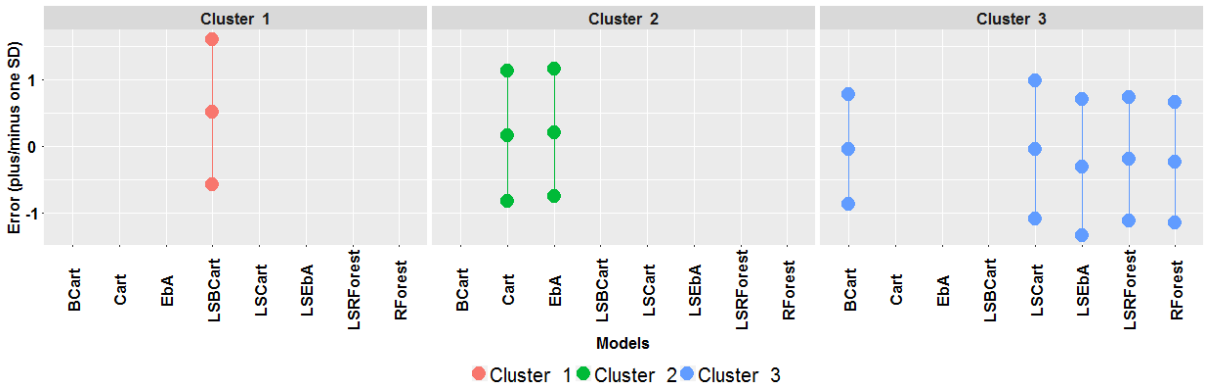
Δημιουργήθηκαν 2 συστάδες, στην συστάδα νούμερο 2 εμπεριέχονται 3 ημι-παραμετρικά μοντέλα που παρουσιάζουν την καλύτερη απόδοση.

Το επόμενο μέτρο ακριβείας που επιλέχθηκε ήταν το Magnitude Relative Error (MRE) τα αποτελέσματα που λάβαμε είναι τα παρακάτω.




**Διάγραμμα 7.22:** Γραφική απεικόνιση του μέτρου ακριβείας MRE.

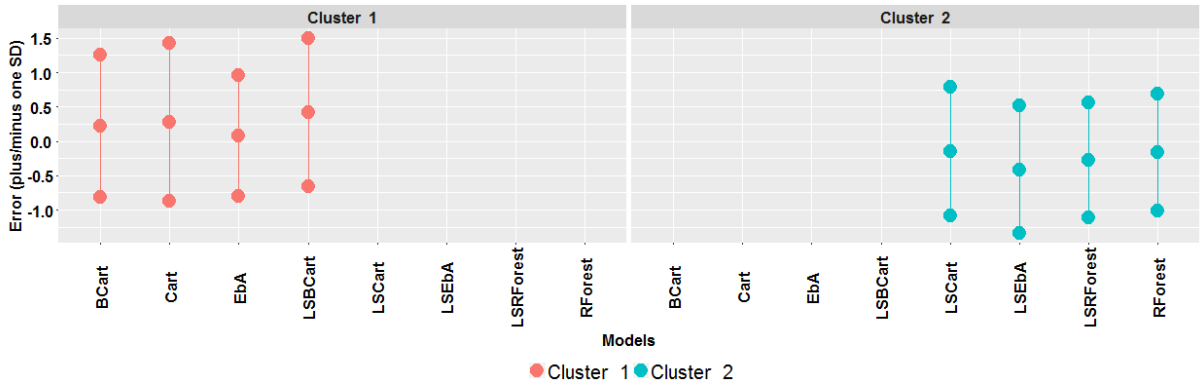
Το τρίτο μέτρο ακριβείας που επιλέχθηκε είναι το Magnitude Relative Error to the Estimate (MER) και η γραφική αναπαράσταση είναι η ακόλουθη:




**Διάγραμμα 7.23:** Γραφική απεικόνιση του μέτρου ακριβείας MRE.

Για το μέτρο ακριβείας MER, δημιουργήθηκαν 3 συστάδες, στην τρίτη εμπεριέχονται τα πέντε μοντέλα με την καλύτερη απόδοση, τα τρία από αυτά είναι ημι-παραμετρικά.

Τέλος επιλέχτηκαν τα μέτρα ακριβείας Balance Relative Error (BRE) και το Inverted Balance Relative Error (IBRE) και τα αποτελέσματα είναι τα παρακάτω.




**Διάγραμμα 7.24:** Γραφική απεικόνιση των μέτρων ακριβείας BRE/IBRE.

Δημιουργήθηκαν δύο συστάδες, στην δεύτερη συστάδα εμπεριέχονται τέσσερα μοντέλα δύο ημι-παραμετρικά κ δύο μη-παραμετρικά. Τα δύο ημι-παραμετρικά παρουσιάζουν την καλύτερη απόδοση όλων.

Για το σύνολο δεδομένων Maxwell η σειρά κατάταξης των μοντέλων είναι η παρακάτω.

| Miyazaki94 | Models | MAE | MMRE | MMER | MBRE | MIBRE | Μέση Κατάταξη |
|---|---|---|---|---|---|---|---|
| | EbA | 5 | 5 | 7 | 5 | 5 | 5,4 |
| | CART | 7 | 7 | 6 | 7 | 7 | 6,8 |
| | Bagging | 8 | 8 | 5 | 6 | 6 | 6,6 |
| | **Random Forest** | 4 | 4 | 2 | 3 | 3 | **3,2** |
| | **LSEbA** | 1 | 1 | 1 | 1 | 1 | **1** |
| | LSCART | 3 | 3 | 4 | 4 | 4 | 3,6 |
| | LSBagging | 6 | 6 | 8 | 8 | 8 | 7,2 |
| | **LSRandom Forest** | 2 | 2 | 3 | 2 | 2 | **2,2** |

**Πίνακας 7.15:** Συγκεντρωτικός πίνακας κατάταξης συνόλου δεδομένων **Miyazaki94**

Σύμφωνα με τον πίνακα 7.15, η σειρά κατάταξης των τριών επικρατέστερων μεθόδων για το σύνολο δεδομένων Miyazaki είναι η εξής: LSEbA, LSRandom Forest και Random Forest.

Από τα ανωτέρω φαίνεται πως το μη-παραμετρικό μέρος που χρησιμοποιείται στις ημι-παραμετρικές μεθόδους έχει θετική επίδραση στην ακρίβεια εκτίμησης του κόστους. Αυτό καθώς αξιοποιεί με επιτυχία την γραμμική σχέση που έχουν οι λογαριθμικοί μετασχηματισμοί του μεγέθους ενός έργου και τις προσπάθειας που απαιτείται για την ολοκλήρωσή του. Αυτό διαφαίνεται από το γεγονός πως τα ημι-παραμετρικά μοντέλα

αποδίδουν καλύτερα αποτελέσματα από τα αντίστοιχα μη-παραμετρικά (π.χ. το LSEbA σε σχέση με το EbA).

Στις περισσότερες περιπτώσεις που εξετάστηκαν τα ημι-παραμετρικά μοντέλα ταξινομήθηκαν στην συστάδα με τα καλά χαρακτηριστικά. Επομένως καταδεικνύεται ότι η συγκεκριμένη μεθοδολογία καταφέρνει να συσχετίσει αποτελεσματικά τις ανεξάρτητες με την εξαρτημένη μεταβλητή, ασχέτως του μοντέλου που χρησιμοποιείται για την αξιολόγηση του μη παραμετρικού στοιχείου.

# Κεφάλαιο 8
# Επίλογος

## 8.1 Συμπεράσματα

Στην παρούσα διατριβή μελετήθηκε το σφάλμα εκτίμησης του κόστους έργων λογισμικού μέσω της εξαγωγής αποτελεσμάτων με γραφικό τρόπο. Για τον λόγο αυτό δημιουργήθηκε μια εφαρμογή στο RStudio η οποία κάνει χρήση του αλγορίθμου Scott-Knott. Πρόκειται για την υλοποίηση μίας προσέγγισης (Mittas & Aggelis, 2013) που δίδει τη δυνατότητα να πραγματοποιείται η συγκριτική αποτίμηση της απόδοσης ορισμένων μοντέλων μέσω μεθοδολογιών που είναι γνωστές ως Πολλαπλές Συγκρίσεις (multiple comparisons). Σε αυτό το πλαίσιο, εξετάστηκε η προβλεπτική ικανότητα 8 μοντέλων σε 6 ευρέως διαδεδομένα σύνολα δεδομένων. Τούτο κατέδειξε πως αξιοποιώντας και μετασχηματίζοντας γνωστά μη-παραμετρικά μοντέλα σε μία ημι-παραμετρική μορφή μπορούμε να πετύχουμε καλύτερες προβλέψεις. Πιο συγκεκριμένα από τα αποτελέσματα που λήφθηκαν τα ημι-παραμετρικά μοντέλα παρουσίασαν καλύτερη απόδοση από τα αντίστοιχα μη-παραμετρικά τους σε ένα συγκεκριμένο σύνολο δεδομένων στην πλειονότητα των μέτρων ακριβείας. Επίσης καθότι αυτό παρατηρήθηκε σε όλα τα σύνολα δεδομένων και τα μέτρα ακριβείας που μελετήθηκαν θα μπορούσαμε να συμπεράνουμε ότι τα ημι-παραμετρικά μοντέλα έχουν συνολικά καλή απόδοση, με τα επικρατέστερα αυτών το LSEbA και το LSRandom Forest.

Επιπλέον, με την προσέγγιση που ακολουθήθηκε καταδεικνύεται πως μέσα από τον αλγόριθμο Scott-Knott επικυρώνεται η προβλεπτική ακρίβεια ενός συνόλου μοντέλων που δεν έχουν στατιστική σημαντική διαφορά μεταξύ τους σε μη αλληλεπικαλυπτόμενες ομάδες. Έτσι μπορεί να υπερκεραστεί το πρόβλημα του παρελθόντος που αναζητούσε απαραίτητα ένα επικρατέστερο μοντέλο και να θέσει την μελλοντική έρευνα σε νέες βάσεις. Εκτός αυτού παρέχεται πλέον η ευελιξία στον διαχειριστή έργου ή στον ερευνητή να επιλέγει ένα διαφορετικό μοντέλο κάθε φορά, από την επικρατέστερη ομάδα ανάλογα με τις ανάγκες που προκύπτουν. Τέλος θα

λέγαμε πως η φύση της εφαρμογής αυτής καθαυτής φέρει ένα βασικό πλεονέκτημα. Δεδομένου ότι βασίστηκε στο πακέτο λογισμικού Shinny, έχει το χαρακτηριστικό της μεταφερσιμότητας (portability) και μπορεί εύκολα να μεταφορτωθεί σε έναν κεντρικό διακομιστή web για ευκολότερη πρόσβαση από πληθώρα συσκευών και χωρίς την ανάγκη για εγκατάσταση βιβλιοθηκών ή εκτελέσιμων εφαρμογών σε κάποια προσωπική συσκευή.

Με την βοήθεια των ανωτέρω οι ερευνητές και οι διοικητές έργων είναι σε θέση να επιλέξουν μία ομάδα καταλληλότερων και αποδοτικότερων μοντέλων πρόβλεψης λογισμικού έχοντας επίσης ενδείξεις για το επικρατέστερο στην συγκεκριμένη ομάδα. Αυτό απαλλάσσει τους ενδιαφερόμενους από χρονοβόρες και δυσκίνητες διαδικασίες που όχι μόνο μπορεί να οδηγήσουν σε λανθασμένες αποφάσεις που αφορούν την ανάπτυξη ενός υπό μελέτη έργου, αλλά και στην ακύρωση του. Έτσι αποτελεί ένα πολύτιμο εργαλείο το οποίο μπορεί να υποβοηθήσει δραστικά στη λήψη αποφάσεων χωρίς να επαυξάνει το κόστος στην συνολικότερη διαδικασία.

## 8.2 Μελλοντικές Επεκτάσεις

Ως μελλοντικές επεκτάσεις στην παρούσα διατριβή θα μπορούσε να θεωρηθεί η μελέτη άλλων γνωστών μη-παραμετρικών μεθοδολογιών ως προς την απόδοση τους όταν εισάγεται σε αυτές ένα γραμμικό συστατικό (component) με σκοπό να μετασχηματιστούν σε ημι-παραμετρικές. Αυτό θα διεύρυνε το σύνολο των ημι-παραμετρικών μεθόδων που μπορούν να χρησιμοποιηθούν στο επιστημονικό πεδίο της ΕΚΛ, γεγονός που αυξάνει κατά επέκταση την ευελιξία των ερευνητών ή των διοικητών έργων ως προς την χρήση της εκάστοτε μεθόδου.

Εκτός αυτών, από πλευράς της εφαρμογής αυτής θα μπορούσε να επεκταθεί ώστε να αποδέχεται και να διαχειρίζεται αρχεία εισόδου διαφορετικών δομών και τύπων. Κάτι τέτοιο θα ευνοούσε την χρησιμοποίηση της εφαρμογής με δεδομένα εισόδου που θα προέρχονταν από διάφορες πηγές. Τέλος θα μπορούσε να εμπλουτιστεί με μηνύματα που αφορούν τα αποτελέσματα και δε τις καταλληλότερες μεθόδους, ώστε να είναι κατανοητά και από τον απλό χρήστη-διαχειριστή και όχι μόνο ερευνητές με εξειδικευμένες γνώσεις στατιστικής. Αυτό θα επέτρεπε την χρησιμοποίηση της στο

ευρύτερο επαγγελματικό περιβάλλον για την χρήση σε πραγματικές συνθήκες ανάπτυξης έργων.

# Βιβλιογραφία

Albrecht, A.J., & Gaffny, J.E. (1983) Software Function, Source Lines of Code and Development Effort Prediction: A Software Science Validation. IEEE Transactions on Software Engineering, EE-9 (6), pp. 639-648.

Angelis, L., Stamelos I. (2000) A Simulation Tool for Efficient Analogy Based Cost Estimation. Empirical Software Engineering. Vol. 5, No. 1. pp.35-68.

Anglin, M.P., & Gencay, R. (1996) Semi-parametric Estimation of a Hedonic Price Function. Journal of Applied Economics, Vol. 11, No. 6, pp. 633- 648.

Aronin, S.B., Bailey, W.J., Byun, S. J., Davis, A.G., Wolfe, L.C., Frazier, P.T., & Bronson, F.P. (2011) Expeditionary Combat Support System: Root Cost Analysis. IDA Paper P-4732 (Draft Final). Alexandria, VA: Institute for Defense Analyses.

Boehm, B.W., Clark, Horowitz, Brown, Reifer, Chulani, Madachy, R., & Steece, B. (2000a) Software cost estimation with COCOMO II. Prentice Hall PTR Upper Saddle River, NJ, USA

Boehm, B.W., Abts, C., & Chuani, S. (2000b) Software Cost Estimation Approaches: A survey. Journal of Software Engineering and Applications, 10, 824-842. Department of Computer Engineering, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia

Boehm, B. W. (1981) Software Engineering Economics. Vol. 197. Prectice-hall Englewood Cliffs (NJ).

Breiman, L. (2001) Random forests. Machine Learning. Vol. 45, No. 1, pp. 5-32. Statistics Department. University of California. Berkeley, CA.

Breiman, L. (1996) Bagging predictors. Machine Learning Vol. 24, pp. 123-140 Department of Statistics University of California Berkeley, California.

Brownlee, J. (2016) Parametric and Nonparametric Machine Learning Algorithms. http://machinelearningmastery.com/parametric-and-nonparametric-machine-learning-algorithms/

Burbeck, S. (1987) Applications Programming in Smalltalk-80. How to use Model–View–Controller (MVC). http://st-www.cs.illinois.edu/users/smarch/st-docs/*mvc*.html.

Cleveland, W.S. (1979) Robust locally Weighted Regression and Smoothing Scatterplots. Journal of the American Statistical Association, Vol. 74, No. 368. pp. 829-836.

Chambers, M.J. (1998) Programming with Data: A Guide to the S Language. Springe-Verlag New York, Inc Secaucus, NJ, USA

Committee, Controller and Auditor General presented to the BBC Trust's Finance and Compliance. (2011) The BBC's management of its Digital Media Initiative. London, UK: BBC Trust.

Conte, S.D., Dunsmore, H.E., & Shen, V.Y. (1986) Software Engineering Metrics and Models. Menlo Park, Calif. : The Benjamin/Cummings Publishing Company, Inc.

Currie, W. (1995) Management Strategy for IT: An International Perspective. Financial Times, Pitman Publishing, London.

Desharnais, J.M. (1988) Analyse Statistique de la Productivite des Projects de Development en Informatique a Partir de la Technique des Points de Fonction. MSC Thesis. Monteral Universite du Quebec.

Dysert, R.L. (2008) An Introduction to Parametric Estimating. AACE INTERNATIONAL TRANSACTIONS: p1.

Efron, B., & Tibshirani, R.J. (1993) An Introduction to the Bootstrap. London: Chapman & Hall, NY.

Flowers, S. (1996) Software Failure: Management Failure, John Wiley and Sons, Chichester. John Wiley, Chichester, 197 pp. ISBN 9780471951131

Gentleman, R. (2006) Individual Expertise profile of Robert Gentleman.

Hayes, A.F., & Cai, L. (2008) A New Test of Linear Hypotheses in OLS Regression Under Heteroscedasticity of Unknown Form. Journal of Educational and Behavioral Statistics, Vol. 33, No. 1, pp.21-40

Heemstra, J. F. (1992) Software Cost Estimation. Information and Software Technology, Vol. 34, No. 10, pp. 627-639. Elsevier Science Inc.

Idri, A., & Abran, A. (2000) Towards a Fuzzy Logy Based Measures for Software Projects Similarity. 6th MCSEAI'2000 – Maghrebian Conference on Computer Sciences, Fez, Morocco.

Jones, C.B. (1991) Applied Software Measurement: Assuring Productivity and Quality. New York, NY: McGraw Hill.

Jones, C.B. (1986) C Programming Productivity. IEEE-CS Press, Los Alamitos, Ca

Kitchenham, B.A. & Mendes, E. (2009) Why comparative effort prediction studies may be invalid. Proceedings of the ACM 5th International Conference on Predictor Models in Software Engineering, May. (PROMISE'09). ACM Press, New York, pp. 1-5

Kittlaus, H.B. & Clough, P.N. (2009) Software Product Management and Pricing. Key Success Factors for Software Organizations. Springer, Heidelberg NY.

Kohavi, R. (1995) A study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection. Proceedings of the 14th International Joint Conference on Artificial Intelligence, Vol. 2, pp. 1137-1143

MacDonell, S.G., & Gray, A.R. (1997) A Comparison of Modeling Techniques for Software Development Effort Prediction. International Conference on Neural Information

Processing and Intelligent Information Systems. Dunedin, New Zealand, Springer-Verlag, pp. 869 - 872.

Maxwell, K., Wassenhove Van N.L., & Dutta, S. (1996) Software Development Productivity of European Space, Military and Industrial Applications. Software Engineering, IEEE Transactions on, Vol. 22, No. 10, pp. 706 -708.

Mittas, N., Mamalikidis, I., & Angelis, L. (2015a) A framework for comparing multiple cost estimation methods using an automated visualization toolkit. Information and Software Technology 57, 310-328.

Mittas, N., Papatheocharous, E., Angelis, L., & Andreou, S.A. (2015b) The Journal of Systems and Software Integrating non-parametric models with linear components for producing software cost estimations. Elsevier Science Inc.

Mittas, N., & Aggelis, L. (2013) Ranking and Clustering Software Cost Estimation Models through a Multiple Comparisons Algorithm. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 39, 2013

Mittas, N. (2011) Evaluating the Performances of Software Cost Estimation Models through Prediction Intervals. Journal of Engineering Science and Technol-ogy Review 4 (3) (2011) 266–270.

Mittas, N., & Angelis, L. (2010) LSEbA: Least Squares Regression and Estimation by Analogy in a Semi-parametric Model for Software Cost Estimation, Empirical Software Engineering. Journal Empirical Software Engineering archive Volume 15 Issue 5, October 2010 Pages 523 - 555

Mittas, N., & Angelis, L. (2009) Bootstrap Prediction Intervals for a Semi-parametric Software Cost Estimation Model. In 35th Euromicro Conference on Software Engineering and Advanced. Applications. pp 293-299

Mittas, N., Angelis, L. (2008a) Partial regression error characteristic curves for the comparison of software cost prediction models. Proceedings of the Artificial Intelligence Techniques in Software Engineering (AISEW'08), July, 6–10.

Mittas, N. & Angelis, L. (2008b) Comparing cost prediction models by resampling techniques. Journal of Systems and Software, 8(5), pp. 616-632. Elsevier Science Inc.

Montgomery, C.D. (1991) Design and Analysis of Experiments. John Wiley & Sons. New York, NY

Miyazaki, Y., Terakado, M., Ozaki, K., & Nozaki, H. (1994) Robust Regression for Developing Software Estimation Models. Journal of Systems and Software, Vol. 27, pp. 3-16.

Miyazaki, Y., Takanou, A., Nozaki, H., Nakagawa, N., & Okada, K. (1991) Method to Estimate Parameter Values in Software Prediction Models. Information and Software Technology, Vol. 33, pp. 239-243.

NASA93, (2007) Dataset, http://promisedata.org/repository/#nasa93. (NASA93 2007)

Opitz, D., & Maclin, R. (1999) Popular ensemble methods: An empirical study. Journal of Artificial Intelligence Research Vol.11, pp. 169-198.

Poonam, P. (2013) Analysis of the Techniques for Software Cost Estimation. Third International Conference on Advanced Computing & Communication Technologies.

Putnam, L., & Myers, W. (1992) Measures for Excellence: Reliable Software on Time, NY: River Yourdon Press Computing Series, Prentice-Hall, Quantitative Software Management (QSM), Inc.

Robinson, P. (1988) Root-N-Consistent Semiparametric Regression. Econometrica, Vol. 56, No. 4, pp. 931-954. The Econometric Society.

Scott, J.A., & Knott, M. (1974) A cluster analysis method for grouping means in the analysis of variance. Biometrics, Vol. 30, No. 3, pp. 507–512. University of Auckland, Auckland, New Zealand and The London School of Economics and Political Science.

Shepperd, M., & Schofield, C. (1997) Estimating Software Project Effort Using Analogies. IEEE Transactions on Software Engineering, Vol. 23, No. 11, pp. 736-743.

Stamelos, I., Angelis, L., & Sakellaris, E. (2001) BRACE: Bootstrap based Analogy Cost Estimation. Automated support for an enhanced effort prediction method. Proceedings of the 12th European Software Control Metrics, pp. 17-23. London UK.

Shalizi C. (2009) Classification and Regression Tree: Tutorial, Course handout, Statistics Dept, Carnegie Mellon University.

Sommerville, I. (2004) Software Cost Estimation. Software Engineering, 7th edition. Chapter 26

Vernazi, J. (2011) Getting Started with RStudio. An Integrated Development Environment for R. Sebastopol, CA: O'Reilly Media.

Wittig G., & Finnie G. (1997) Estimating software development effort with connectionist models. Information and Software Technology Volume 39, Issue 7, 1997, Pages 469-476. School of Information Technology, Bond University, Gold Coast, Queensland, 4229, Australia

Zivadinovic, J., & Medic, Z. (2011) Methods of Effort Estimation in Software Engineering. International Symposium Engineering Management and Competitiveness, Serbia.

Φωκιανός, Κ., & Χαραλάμπους, Χ. (2010) Εισαγωγή στην R Πρόχειρες Σημειώσεις. Τμήμα Μαθηματικών & Στατιστικής. Πανεπιστήμιο Κύπρου