

Πανεπιστήμιο Πειραιώς
Τμήμα Ψηφιακών Συστημάτων
Πρόγραμμα Μεταπτυχιακών Σπουδών
ΠΜΣ: Ψηφιακές Επικοινωνίες και Δίκτυα



Μεταπτυχιακή Διπλωματική Εργασία

**ΑΝΑΠΤΥΞΗ «ΑΥΤΟΜΑΤΟΥ» ΣΥΣΤΗΜΑΤΟΣ ΠΡΟΒΛΕΨΕΩΝ
ΒΑΣΙΣΜΕΝΟ ΣΕ ΔΕΔΟΜΕΝΑ ΤΕΧΝΙΚΗΣ ΑΝΑΛΥΣΗΣ.**

Μακρίδης Γεώργιος

Απρίλιος 2015

Επιβλέπων Καθηγητής

Απόστολος Μηλιώνης, Καθηγητής

Πανεπιστήμιο Πειραιώς

Εξεταστική Επιτροπή

.....
Πανεπιστήμιο Πειραιώς

.....
Πανεπιστήμιο Πειραιώς

.....
Πανεπιστήμιο Πειραιώς

Πίνακας περιεχομένων

Ακρωνύμια	6
Περίληψη.....	8
Abstract	9
Εισαγωγή.....	10
Σκοπός	11
Αρχιτεκτονική Συστήματος	11
Δεδομένα.....	11
Αλγόριθμος Μηχανικής Μάθησης.....	12
Backend.....	12
Εφαρμογή Κινητών Συσκευών σε Πλατφόρμα Android	12
Οικονομική θεωρία - Τεχνική Ανάλυση	13
1.1 Θεωρία Επενδύσεων - Χαρτοφυλακίου	13
1.1.1 Μετοχές.....	13
1.1.2 Ομόλογα.....	14
1.1.3 Εμπορεύματα	14
1.1.4 Αγορές Ξένων Νομισμάτων.....	14
1.1.5 Παράγωγα	15
1.2 Επενδύσεις - Trading	15
1.2.1 Κατηγορίες Επενδύσεων.....	16
1.3 Σύγχρονη Θεωρία Χαρτοφυλακίου	18
1.3.1 Παραδοσιακή και Σύγχρονη Θεωρία Διαχείρισης Χαρτοφυλακίου.....	19
1.4 Χαρακτηριστικά Δεδομένα.....	20
1.4.1 Δεδομένα Τεχνικής Ανάλυσης.....	20
1.4.2 Η θεωρία της «τυχαίας πορείας» (random walk theory).....	21
1.4.3 Δείκτες Τεχνικής Ανάλυσης - Ερμηνεία.....	22

Μηχανική Μάθηση.....	40
2.1 Βασικές Έννοιες της Μηχανικής Μάθησης	40
2.2 Μοντελοποίηση Προβλήματος.....	41
2.2.1 Αναπαράσταση Δεδομένων	41
2.2.2 Κατηγοριοποίηση Μοντέλων M.M.....	43
2.2.3 Αξιολόγηση Απόδοσης και Overfitting	43
2.2.4 Κλιμάκωση Δεδομένων	45
2.3 Επιλογή Αλγορίθμου M.M.	46
2.3.1 Μηχανές Διανυσμάτων Υποστήριξης - Support Vector Machines (SVM)..	47
2.3.2. Adaptive Boosting (Ada-Boost) Algorithm.....	58
Βασικές Δομές – Εργαλεία.....	61
3.1 Google App Engine	61
3.2 Android.....	65
Υλοποίηση Συστήματος	71
4.1 Δομή – Αρχιτεκτονική Συστήματος	71
4.2 Δεδομένα	73
4.2.1 Βήμα 1 ^ο – 2 ^ο - Πηγή Δεδομένων	73
4.2.2 –Βήμα 3 ^ο - Προ-Επεξεργασία Δεδομένων.....	75
4.3 Επιλογή κατάλληλου Αλγορίθμου – Υλοποίηση	77
4.3.1 Βήμα 4 ^ο Feature selection	77
4.3.2 Βήμα 5 ^ο - Επιλογή Μεταβλητών (C και γ)	80
4.3.3 Βήμα 6 ^ο -7 ^ο – Εκπαίδευση SVM model.....	81
4.3.4 Daily Indicators Calculations.....	83
4.4 Android application	86
4.4.1 Κατασκευή εφαρμογής Android.....	88
Συμπεράσματα – Προτάσεις.....	102
5.1 Συμπεράσματα.....	102
Γενικά.....	102

Οικονομική Κρίση	102
Υλοποίηση	104
5.2. Contributors	106
Πηγές.....	107

Ακρωνύμια

A

ALF - (Alexander's Filter)

API - Application Programming Interface

C

CCI – Commodity Channel Index

E

EMA – Exponential Moving Average

G

GAE – Google App Engine

M

MPT – Modern Portofolio Theory

MACD –

MDLP – Minimum Description Length Principle

MFI – Money Flow Index

P

PaaS – Platform as a Service

PROC – Price rate of change

R

RS – Relative Strength

RNN - Recurrent neural network

RSI -

S

SVM – Support Vector Machine

SMA – Simple Moving Average

U

UI – User Interface

K

KM – Κινητός Μέσος

M

M.M. – Μηχανική Μάθηση

T

TNΔ – Τεχνητά Νευρωνικά Δίκτυα

X

XAA – Χρηματιστήριο Αξιών Αθηνών

Περίληψη

Σκοπός της εργασίας είναι να μελετηθεί και να κατασκευαστεί ένα πρωτότυπο σύστημα πρόβλεψης κίνησης μετοχών βασισμένο στη θεωρία Τεχνικής Ανάλυσης. Απώτερος σκοπός είναι το ανωτέρω σύστημα να πλαισιώνεται από μια εφαρμογή σε λειτουργικό σύστημα Android καθιστώντας το εναρμονισμένο με τις σύγχρονες τάσεις της τεχνολογίας. Αυτό θα επιτευχθεί μέσω της χρήσης σύγχρονων εργαλείων και βιβλιοθηκών. Η διαδικασία της πρόβλεψης στηρίζεται στη θεωρία Μηχανικής Μάθησης, ενώ η υλοποίηση και παρουσίαση του συστήματος θα πραγματοποιηθεί με τη βοήθεια του API Hosting Cloud της Google και του Android.

Πιο αναλυτικά, στα πρώτα δύο κεφάλαια παρατίθενται τόσο η Οικονομική Θεωρία Τεχνικής Ανάλυσης όσο και η θεωρητική προσέγγιση Αλγορίθμων Μηχανικής Μάθησης που θα αποτελεί τον πυρήνα του συστήματος. Στη συνέχεια, παρουσιάζεται και επεξηγείται η υλοποίηση του συστήματος δίνοντας βαρύτητα στον τρόπο υλοποίησης. Θα πρέπει να αναφέρουμε πως λόγω του αυξημένου βαθμού δυσκολίας του εγχειρήματος θεωρήθηκε ως δεδομένο μια συγκεκριμένη μετοχή του ΧΑΑ, καθώς η προσπάθεια γενίκευσης ενός μοντέλου για όλες τις προσφερόμενες μετοχές θα καθιστούσε την προσπάθεια αδύνατη. Τέλος, αφού αναλυθούν τα συμπεράσματα του εγχειρήματος, προτείνονται συγκεκριμένες θέσεις με ανάλογες μεθόδους οι οποίες υπόσχονται βελτιστοποίηση της ακρίβειας παρόμοιων συστημάτων.

Abstract

The purpose of the project is to study and construct a prototype stock motion prediction system based on Technical Analysis theory. The ultimate goal is, the above system is framed by an application on Android operating system, making it harmonized with modern technology trends. This will be achieved through the use of inspiring, contemporary styles of tools and libraries. The process of forecasting based on Machine Learning theory, while the implementation and presentation of the system will be developed in Google's API Hosting Cloud and Android.

Specifically, the first two chapters listed as the Economic theory Technical Analysis and the theoretical approach Algorithms Machine Learning that will be the core of the system. Then we introduce and explain the implementation of the system by giving emphasis on its implementation. It should be mentioned that due to the increased difficulty level of the task, we took for granted a particular share of ASE, as the attempt of generalization of the model for all the offered shares would make our effort impossible. Finally, we present some specific positions proposed by analogous methods that promise to optimize the accuracy of such systems.

Εισαγωγή

Η ραγδαία εξέλιξη και εμβάθυνση της σύγχρονης χρηματιστηριακής πραγματικότητας που είναι αποτέλεσμα παραγόντων όπως, η είσοδος νέων δυναμικών επιχειρήσεων στο χρηματιστήριο, η λειτουργία του χρηματιστηρίου παραγώγων και η διεθνοποίηση των οικονομιών, επιβάλλει την εφαρμογή συγχρόνων επενδυτικών εργαλείων στη διαχείριση χαρτοφυλακίου. Τα περισσότερα από τα εργαλεία αυτά βασίζονται σε τεχνικές μαθηματικής ανάλυσης και μοντελοποίησης και απαιτούν ισχυρούς υπολογιστές και εξειδικευμένες γνώσεις για την υποστήριξή τους, κάνοντας τα να απόμακρα στο μέσο επενδυτή. Σε αυτή την κατεύθυνση, πρόσφατα, πολλές έρευνες και μελέτες έχουν εκπονηθεί στην περιοχή της εφαρμογής Αλγόριθμων Μηχανικής Μάθησης για την ανάλυση και την πρόβλεψη των αλλαγών των τιμών μετοχών (δεικτών κ.α.). Πλέον γίνεται ολοένα και πιο επιτακτική στους επενδυτές, η χρήση κάποιου είδους Ευφυή Συστημάτων Συναλλαγών που τους βοηθούν στον υπολογισμό και πρόβλεψη συγκεκριμένων τιμών, συμβάλλοντας έτσι τους στη λήψη επενδυτικών αποφάσεων. Η προκύπτουσα αυτή τάση βελτιστοποίησης του χαρτοφυλακίου είναι μια διεπιστημονική προσπάθεια, η οποία, στις μέρες μας, προσεγγίζεται από τους τομείς της Θεωρίας της Επένδυσης και της Μηχανικής Μάθησης.

Δεν πρέπει να ξεχνάμε επίσης πως οι τιμές των μετοχών θεωρείται ότι είναι πολύ δυναμικές και ευπαθείς σε γρήγορες μεταβολές, αυτό εξηγείται λόγω του χαρακτήρα του χρηματοπιστωτικού τομέα και εν μέρει λόγω του συνδυασμού των γνωστών παραμέτρων (Τιμή Κλεισίματος προηγούμενων ημερών, κλπ) με άγνωστους πολιτικο-κοινωνικούς παράγοντες (όπως Αποτελέσματα Εκλογών, φήμες, κλπ). Η ταχύτητα πλέον που πρέπει κάποιος να παρακολουθεί τις εξελίξεις εδραιώνει την άποψη πως είναι πλέον απαραίτητη η χρήση Ευφυή Συστημάτων Συναλλαγών απλά προσθέτοντας στον τίτλο τους και τον όρο Αυτόματα.

Ένας «έξυπνος» επενδυτής θα προβλέψει την τιμή της μετοχής και θα τοποθετηθεί κατάλληλα. Αν και είναι πολύ δύσκολο να αντικατασταθεί η εμπειρία ενός επενδυτή, ένας ακριβής αλγόριθμος πρόβλεψης μπορεί να χρησιμοποιηθεί ως εργαλείο οδηγώντας σε υψηλά κέρδη για τις επιχειρήσεις επενδύσεων. Η λέξη ακρίβεια είναι η λέξη κλειδί στην παραπάνω φράση, υποδεικνύοντας μια άμεση σχέση ανάμεσα στην

ακρίβεια του αλγόριθμου πρόβλεψης και το κέρδος που προέκυψε από τη χρήση του. Πλέον η ακρίβεια τέτοιου είδους συστημάτων (Προβλέψεων – Συναλλαγών) μπορεί να αμφισβητείται από τμήμα της επιστημονικής κοινότητας καθώς λίγες έρευνες βλέπουν το φώς της δημοσιότητας. Αυτό είναι αναμενόμενο αν αναλογιστεί κανείς την φύση και τον σκοπό τέτοιων εγχειρημάτων.

Παρόλα αυτά εμείς θα προσπαθήσουμε να υλοποιήσουμε ένα τμήμα ενός τέτοιου συστήματος. Σε αυτή μας την προσπάθεια και με σκοπό να καταστήσουμε πιο εύχρηστο το σύστημα μας θα υλοποιήσουμε ένα τμήμα αυτού σε πλατφόρμα Android. Ο λόγος φυσικά δεν είναι άλλος από τη ραγδαία αύξηση χρήσης κινητών συσκευών όπως Smartphone και tablet, που όπως φαίνεται από πολλές έρευνες στο άμεσο μέλλον θα επισκιάσει τη χρήση των σταθερών υπολογιστών (desktop).

Σκοπός

Σκοπός της εργασίας είναι η δημιουργία ενός εργαλείου, χρήσιμο στους επενδυτές, οπού θα μπορεί να συσχετίζει τα διάφορα μεγέθη που καθορίζουν την κίνηση μιας μετοχής και θα προβλέπει την κίνηση της. Ενώ το σύστημα αυτό θα πρέπει να παρουσιαστεί με τέτοιο τρόπο που να το καθιστά φιλικό στο χρήστη.

Ο σκοπός αυτός μπορεί να αναλυθεί σε «υπό – σκοπούς»

- Ανάπτυξη ενός αλγόριθμου πρόβλεψης μεταβολής τιμών.
- Σωστή αρχιτεκτονική Server – Client
- Σωστή παρουσίαση και δομή μιας εφαρμογής Android

Αρχιτεκτονική Συστήματος

Η αρχιτεκτονική του συστήματος έχει καθοριστεί τόσο από τις απαιτήσεις του σκοπού όσο και από της ιδιαιτερότητες των εργαλείων που χρησιμοποιήσαμε κατά την υλοποίηση.

Δεδομένα

Η παρούσα εργασία πραγματεύεται κυρίως Δεδομένα Τεχνικής Ανάλυσης. Τα δεδομένα αυτά έχουν υποστεί μερική ανάλυση και επεξεργασία ώστε να είναι εκμεταλλεύσιμα από τον Αλγόριθμο Μηχανικής Μάθησης.

Αλγόριθμος Μηχανικής Μάθησης

Ο Αλγόριθμος Μηχανικής Μάθησης που επιλέχτηκε για την υλοποίηση των προβλέψεων, δίνοντας το ένα στοιχείο ευφυΐας στο σύστημα μας, είναι οι Μηχανές Διανυσμάτων Υποστήριξης - Support Vector Machine (SVM) ο οποίος χρησιμοποιείται ευρέως σε παρόμοια συστήματα. Για την βελτιστοποίηση της απόδοσης του επιλεγμένου Αλγορίθμου χρησιμοποιήθηκε και ένας Μετά-Αλγόριθμος (adaptive boost).

Backend

Στην σημερινή εποχή όπως αναφέραμε και νωρίτερα Αλγόριθμοι Μηχανικής Μάθησης έχουν βρει εφαρμογή σε πάρα πολλούς τομείς. Σημαντικός παράγοντας στην εξέλιξη αυτή έχει η ανάπτυξη του cloud - computing στις μέρες μας. Η τεχνολογία αυτή βασίζεται σε μεγάλο βαθμό στην επεξεργασία (και αποθήκευση) τεράστιου όγκου δεδομένων σε datacenters. Στην παρούσα εργασία θα γίνει χρήση του Platform-as-a-Service (PaaS) της Google, γνωστό και ως Google App Engine (GAE)[1].

Εφαρμογή Κινητών Συσκευών σε Πλατφόρμα Android

Τέλος θα παρουσιάσουμε τα αποτελέσματα μέσω μιας χρήσιμης εφαρμογής για κινητές συσκευές σε περιβάλλον Android. Θα πρέπει να σημειωθεί πως δεν θα δώσουμε ιδιαίτερη βαρύτητα στο Περιβάλλον του Χρήστη - UI (User Interface) καθώς κάτι τέτοιο δεν είναι μέσα στα πλαίσια του σκοπού του εργασίας.

Κεφάλαιο 1ο

Οικονομική Θεωρία - Τεχνική Ανάλυση

1.1 Θεωρία Επενδύσεων - Χαρτοφυλακίου

Οι χρηματοπιστωτικές αγορές είναι μηχανισμοί που επιτρέπουν το εμπόριο ρευστών και ανταλλάξιμων προϊόντων (assets) με σχετικά χαμηλό κόστος συναλλαγής. Οι χρηματοπιστωτικές αγορές διευκολύνουν την άντληση κεφαλαίων, τη μεταβίβαση των κινδύνων, τη μεταφορά της ρευστότητας και το διεθνές εμπόριο. Μερικές πιο σημαντικές παρουσιάζονται στον Πίνακα 1.

Market	Description
Bond Markets	Provides financing through bonds
Stock Markets	Provides financing through stocks
Commodity Markets	Facilitates trading of commodities
Money Markets	Provides short term debt financing and investment
Insurance Markets	Facilitates redistribution of various risks
Foreign Exchange Markets	Facilitates trading of foreign exchange (currency)
Derivatives Markets	Provide instruments for the management of financial risk

Πίνακας 1 Διάφορες χρηματοπιστωτικές αγορές

1.1.1 Μετοχές

Οι Μετοχές είναι το μέσο που αντιπροσωπεύει ένα μερίδιο της ιδιοκτησίας στα ίδια κεφάλαια μιας εταιρείας. Η Χρηματιστηριακή Αγορά (Stock Market) είναι ένας δημόσιος φορέας για την εμπορία των εν λόγω μετοχών της εταιρείας. Οι Μετοχές συχνά αποτελούν αντικείμενο διαπραγμάτευσης στο χρηματιστήριο τόσο για επενδυτικούς σκοπούς όσο και για κερδοσκοπικούς.

Όταν μια εταιρεία εμφανίζει κέρδη, με απόφαση του διοικητικού συμβουλίου επιλέγουν είτε να τα επανεπενδύσουν στην εταιρεία, είτε αποσύρουν το σύνολο ή ένα τμήμα τους αναδιανέμοντάς το στους μετόχους με τη μορφή μερισμάτων.

Τα προϊόντα (assets) δηλαδή οι μετοχές που αναφέρονται στη Χρηματιστηριακή Αγορά μπορούν να αλλάξουν δραστικά την αξία τους για πολλούς λόγους. Αρκεί να αναλογιστεί κανείς ότι, εξ ορισμού η αξία της εκάστοτε μετοχής ακολουθεί την αποτίμηση της εταιρίας. Ουσιαστικά, η εταιρεία αποτιμάται με βάση τις αναμενόμενες μελλοντικές αποδόσεις. Οι εκτιμήσεις οι οποίες επηρεάζουν και την αξία της εταιρίας, όπως προαναφέρθηκε, προκύπτουν από γεγονότα που συμβαίνουν και σχετίζονται με την εταιρεία ή με την αγορά όπου ανήκει. Πλέον τέτοια συμβάντα στην εποχή της ταχύτητας όπου ζούμε λαμβάνουν χώρα σε καθημερινό επίπεδο. Ωστόσο, υπάρχουν ορισμένες μεταβολές των τιμών που προέρχονται από προσαρμογές που είναι εξαιρέσεις στον κανόνα, και θα δημιουργήσουν προβλήματα για ένα αυτοματοποιημένο σύστημα επενδύσεων. Χαρακτηριστικά παραδείγματα είναι η καταβολή μερίσματος και η πτώχευση της εταιρείας.

1.1.2 Ομόλογα

Τα Αγορά Ομολόγων είναι παραδοσιακά μια επιλογή χαμηλής απόδοσης – χαμηλού κινδύνου. Αυτά χρησιμοποιούνται συχνά ως συμπλήρωμα στο κλείσιμο του επενδυτικού χαρτοφυλακίου και ως ένα ασφαλές καταφύγιο. Πρόσφατα έχουμε δει αρκετά υψηλές αποδόσεις ομολόγων από ορισμένες χώρες, αλλά αυτό είναι εκ νέου επηρεαζόμενο από μια ξαφνική αύξηση του κινδύνου.

1.1.3 Εμπορεύματα

Οι Αγορές Εμπορευμάτων (Commodity Markets) διαφέρουν πολύ ανάλογα με το εμπόρευμα που αναφέρονται. Κάθε μία από αυτές έχει τους δικούς της παραμέτρους που επηρεάζουν την τιμή. Αγορές όπως ο χρυσός, χρησιμοποιούνται ως ασφαλή καταφύγια, όταν άλλες πιο ριψοκίνδυνες αγορές είναι ευμετάβλητες. Εμπορεύματα όπως το πετρέλαιο ή ο χαλκός μερικές φορές θεωρούνται ως ένας δείκτης πρόβλεψης των μελλοντικών προοπτικών διαφόρων αγορών.

1.1.4 Αγορές Ξένων Νομισμάτων

Foreign Exchange Markets (γνωστή και ως Forex) ή συναλλαγές ξένων νομισμάτων είναι το εμπόριο μεταξύ διαφορετικών νομισμάτων. Αποτελεί μια από τις πιο δημοφιλείς αγορές και οι συναλλαγές είναι συνήθως κερδοσκοπικού χαρακτήρα.

1.1.5 Παράγωγα

Παράγωγο στα χρηματοοικονομικά ονομάζεται ένα συμβόλαιο, η αξία του οποίου εξαρτάται από την αξία κάποιου άλλου βασικότερου προϊόντος (υποκείμενο προϊόν). Ουσιαστικά, δηλαδή, πρόκειται για ένα αξιόγραφο, η τιμή του οποίου καθορίζεται με άμεσο τρόπο από την τιμή του υποκείμενου τίτλου. Σε κάθε τέτοιο συμβόλαιο υπάρχουν δύο αντισυμβαλλόμενοι. Ο ένας έχει τη θέση του αγοραστή (long position) ενώ ο άλλος έχει τη θέση του πωλητή (short position). Τα υποκείμενα προϊόντα από τα οποία προέρχεται ένα παράγωγο μπορεί να είναι είτε προϊόντα που τίθενται υπό διαπραγμάτευση σε μία οργανωμένη δευτερογενή αγορά, όπως ένα χρηματιστήριο, είτε προϊόντα που δεν τίθενται υπό διαπραγμάτευση σε οργανωμένες αγορές. Σε γενικές γραμμές, τα υποκείμενα προϊόντα μπορεί να είναι σχεδόν οτιδήποτε από εμπορεύσιμες μετοχές και ομόλογα μέχρι αγροτικά προϊόντα (π.χ. σιτάρι) και μέταλλα (π.χ. χρυσός).

1.2 Επενδύσεις - Trading

Τα περισσότερα προϊόντα (assets) στον κόσμο διαπραγματεύονται. Όπου υπάρχει το εμπόριο, μπορεί να υπάρξει κερδοσκοπία. Όπως λέει και μια ρήση, η κερδοσκοπία είναι μια οικονομική ενέργεια που δεν υπόσχεται την ασφάλεια της αρχικής επένδυσης μαζί με την επιστροφή επί του αρχικού κεφαλαίου [2]. Αυτό αντιτίθεται στη φιλοσοφία των επενδύσεων. Η κερδοσκοπία συχνά αποκαλείται ενεργός διαπραγμάτευση και γίνεται συνήθως σε σύντομο χρονικό ορίζοντα.

Μια ανησυχία που ενδέχεται να προκύψει είναι κοινωνικού χαρακτήρα. Μερικοί ισχυρίζονται ότι η καθαρή κερδοσκοπία είναι απλά μια μορφή του θεσμοθετημένου παρασιτισμού που δεν φέρνει καμία αξία για την κοινωνία. Ωστόσο, αυτό δεν λαμβάνει υπόψη μια ευρύτερη προοπτική. Οι επιχειρήσεις χρειάζονται κεφάλαια για να αναπτυχθούν. Λίγοι επενδυτές είναι πρόθυμοι να κλειδώσουν τα κεφάλαιά τους για τόσο πολλά χρόνια, μέχρι η εταιρεία να τα χρειαστεί. Σε μια ρευστή χρηματοπιστωτική αγορά, αυτό μπορεί να μετριαστεί προσφέροντας στους επενδυτές τη δυνατότητα να απαλλαγούν από την έκθεσή τους στην εταιρεία σε κάθε δεδομένη στιγμή. Η κερδοσκοπία αυξάνει τη ρευστότητα των οικονομικών αγορών,

προσελκύοντας έτσι επενδυτές να αναπτύξουν επιχειρήσεις που χρειάζονται χρόνο για να ωριμάσουν. Με τη σειρά της η κερδοσκοπία, ασκεί πίεση για τις υπάρχουσες επιχειρήσεις να παραμείνουν επικερδείς και καινοτόμες.

1.2.1 Κατηγορίες Επενδύσεων

Υπάρχουν πολλοί διαφορετικοί τύποι επενδυτών που δραστηριοποιούνται σε διάφορες αγορές. Η πιο συνηθισμένη κατηγοριοποίηση βασίζεται στο χρονικό πλαίσιο τους. Εναλλακτικά είναι δυνατή η κατηγοριοποίηση με βάση το είδος των στρατηγικών που εφαρμόζουν. Χαρακτηριστικό παράδειγμα η διαφορά μεταξύ συναλλαγών βασισμένων σε Τεχνική και Θεμελιώδη Ανάλυση.

Στη Θεμελιώδη Ανάλυση λαμβάνονται υπόψη παράγοντες βασισμένοι στα λογιστικά στοιχεία της κάθε εταιρείας, ενώ στην Τεχνική Ανάλυση η εκτίμηση βασίζεται στην υπόθεση ότι, τα ιστορικά στοιχεία των τιμών, έχοντας υποστεί κατάλληλη ανάλυση, μπορούν να προβλέψουν τις μελλοντικές τιμές.

Η Θεμελιώδης Ανάλυση είναι εγγενώς πολύ περισσότερο στατική από ότι η Τεχνική Ανάλυση καθώς τα αποτελέσματα της εταιρείας δημοσιεύονται μόνο μία φορά ανά τρίμηνο. Στη Θεμελιώδη Ανάλυση επίσης συγκαταλέγονται και κάποια πιο ποιοτικά στοιχεία (π.χ. ειδήσεις) καθιστώντας συνεπώς, πιο δύσκολη την άμεση ανάλυση τους. Η ανάλυση των Τεχνικών δεδομένων είναι καθαρά ποσοτική, και έτσι μπορεί να αντιμετωπιστεί από τους υπολογιστές. Συνυπολογίζοντας τις αυξημένες υπολογιστικές δυνατότητες των Η/Υ στις μέρες μας, τα εντυπωσιακά αποτελέσματα της εν λόγω προσέγγισης δε μας προκαλούν έκπληξη.

Η κατηγοριοποίηση βάσει της χρονικής διάρκειας μας δίνει τις παρακάτω κατηγορίες οι οποίες αναπαρίστανται γραφικά στην Εικόνα 1:

Position Trading

Στην κατηγορία αυτή εντάσσονται όλες οι θέσεις που κατέχονται περισσότερο από μερικούς μήνες. Αυτή είναι ένα πολύ αξιόπιστη στρατηγική, αλλά με περιορισμένες δυνατότητες επιστροφής [3]. Επειδή κάποιος έχει αρκετό χρόνο για να κάνει μια διεξοδική ανάλυση του προϊόντος (asset). Οι αλγοριθμικές μέθοδοι δεν έχουν ιδιαίτερο πλεονέκτημα εδώ, μπορούν όμως να χρησιμοποιηθούν ως συμπληρωματικές κατευθύνσεις. Αυτή η κατηγορία περιλαμβάνει επίσης ένα είδος πιο ενεργών επενδυτών, οι οποίοι δραστηριοποιούνται με διαφορετικό τρόπο από

αυτόν που περιγράφηκε νωρίτερα. Η ειδοποιός διαφορά είναι ότι συμμετέχουν ενεργά στη διοίκηση της εταιρείας.

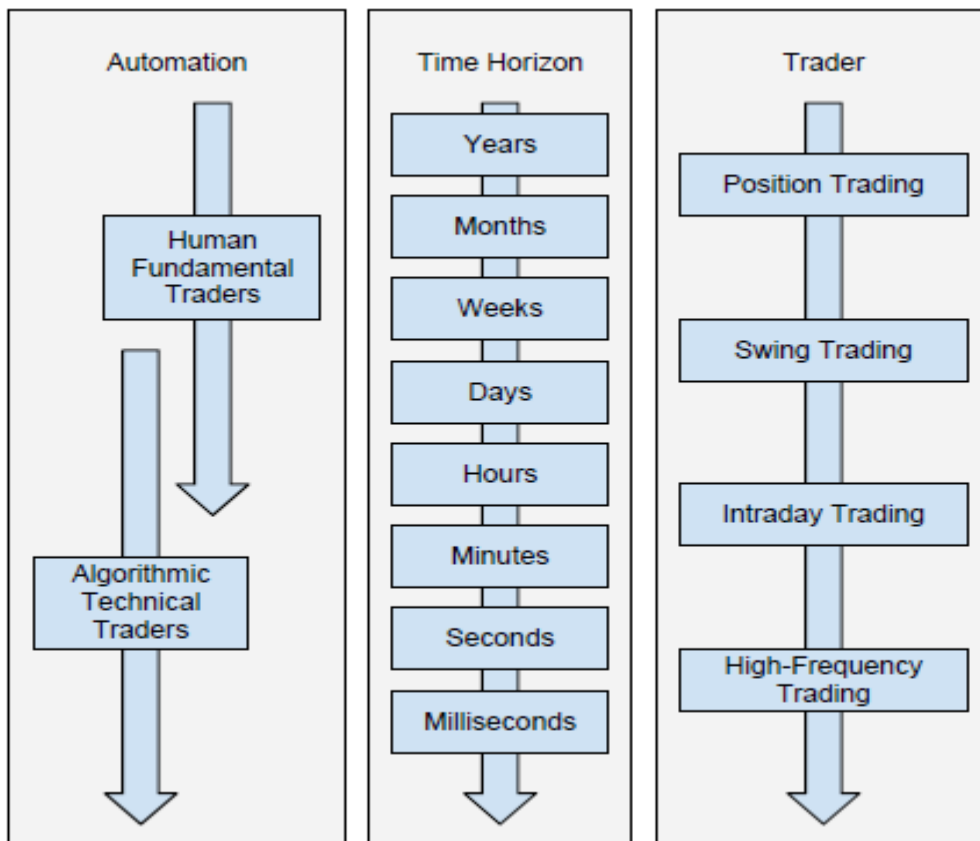
Swing Trading

Οι θέσεις κρατούνται για ημέρες ή εβδομάδες. Περιλαμβάνουν στρατηγικές, όπως “Swing Trading” και πιο βραχυπρόθεσμες στρατηγικές Θεμελιώδους Ανάλυσης. Αυτό είναι γενικά το συντομότερο χρονικό πλαίσιο στο οποίο κάποιος συναντά επενδύσεις βασισμένες σε Θεμελιώδη Ανάλυση. Αλλαγές την τιμή του προϊόντος σε μικρότερη κλίμακα από «ημέρες» θεωρούνται συχνά θόρυβος σε αυτού του είδους την συναλλαγή.

Intraday Trading

Ενδοημερήσιες συναλλαγές είναι αυτές όπου οι επενδυτές «κλείνουν» τις θέσεις τους πριν το πέρας της συνεδρίασης. Αυτού του τύπου οι επενδύσεις είναι κυρίως κερδοσκοπικού χαρακτήρα και ως επί το πλείστον γίνονται βασισμένες σε μεθόδους Τεχνικής Ανάλυσης. Τελευταία παρατηρείται το φαινόμενο οι επενδυτές να τις συμπληρώνουν και με Θεμελιώδεις τεχνικές. Οι συναλλαγές αυτού του τύπου γίνονται το βασίλειο των μηχανών καθώς μπορούν να κάνουν χρήση περισσότερων δεδομένων από ότι ένας άνθρωπος.

Τέλος, μια νέα (και αρκετά διαδεδομένη) κατηγορία είναι η Low-Latency Trading (χαμηλού χρόνου απόκρισης). Αποτελείται από σύνολο απλών αλγορίθμων το οποίο ως μοναδικό πλεονέκτημα έχει την άμεση πρόσβαση στην αγορά. Ένα παράδειγμα μιας στρατηγικής χαμηλού χρόνου απόκρισης είναι οι Αλγόριθμοι που ερευνούν ειδήσεις. Σε αυτό το είδος των συναλλαγών η κατοχή των ταχύτερων αλγορίθμων με τη χαμηλότερη λανθάνουσα κατάσταση αποτελεί στρατηγικό πλεονέκτημα. Αυτή η δραστηριότητα απαιτεί τεράστιο κεφαλαίο λόγω των απαιτήσεων υλικών και μέσων. Υπάρχουν λίγοι ερασιτέχνες σε αυτόν τον τομέα και οι περισσότεροι φορείς είναι μεγάλες τράπεζες ή επενδυτικοί οίκοι [4].



Εικόνα 1 Διάφοροι τύποι επενδύσεων και η σχέση τους

1.3 Σύγχρονη Θεωρία Χαρτοφυλακίου

Η Σύγχρονη Θεωρία Χαρτοφυλακίου (MPT), που θεσπίστηκε από τον Harry Markowitz το 1952 [5], είναι μια θεωρία που επιχειρεί να δημιουργήσει χαρτοφυλάκια προϊόντων (assets) ισορροπώντας μεταξύ απόδοσης και κινδύνου. Θεμελιώδης στοιχείο της ανωτέρω θεωρίας είναι η εφικτότητα της μεγιστοποίησης της αναμενόμενης απόδοσης του χαρτοφυλακίου στα πλαίσια όπου δίνεται ένας συγκεκριμένος κίνδυνος, ή ισοδύναμα η ελαχιστοποίηση του κινδύνου για ένα δεδομένο επίπεδο απόδοσης.

Μία από τις βασικές ιδέες της MPT είναι ότι ένα χαρτοφυλάκιο δεν βασίζεται στις επιδόσεις των επιμέρους προϊόντων, αλλά στην συνολική εικόνα της απόδοσης τους. Το κλασικό μαθηματικό μοντέλο του MPT χρησιμοποιεί τη μέση απόδοση ως ένα μέτρο της αναμενόμενης απόδοσης και τη διακύμανση της ως μέτρο κινδύνου.

Με τον όρο διαχείριση χαρτοφυλακίου εννοούμε τις απαραίτητες ενέργειες που ο κάθε επενδυτής πρέπει να πραγματοποιήσει, για κάθε χαρτοφυλάκιο που δημιουργεί, έτσι ώστε να διασφαλιστεί το κεφάλαιο το οποίο έχει επενδυθεί. Ορίζεται ως οι

διαδικασίες και οι ενέργειες συνδυασμού διαφόρων προϊόντων (assets) σε ένα χαρτοφυλάκιο, το οποίο δημιουργείται ανάλογα με τις ανάγκες του κάθε επενδύτη. Στις ενέργειες αυτές περιλαμβάνεται φυσικά η παρακολούθηση του χαρτοφυλακίου αυτού και η αποτίμηση της απόδοσης του.

Η διαχείριση Χαρτοφυλακίου περιλαμβάνει τα παρακάτω τρία στάδια:

1. Ανάλυση αξιόγραφων: Στο στάδιο αυτό εξετάζονται από τα διαθέσιμα χρεόγραφα αυτά τα οποία προβλέπεται να έχουν μεγαλύτερη απόδοση.
2. Ανάλυση Χαρτοφυλακίου: Στο στάδιο αυτό προβλέπεται η συνολική απόδοση ενός χαρτοφυλακίου (συνδυασμός χρεογράφων) και οι πιθανότητες κινδύνου του.
3. Επιλογή Χαρτοφυλακίου: Στο στάδιο αυτό, από τα χαρτοφυλάκια τα οποία ελαχιστοποιούν τον κίνδυνο σε σχέση με την απόδοσή τους, επιλέγεται ένα που θα ταιριάζει στα ιδιαίτερα χαρακτηριστικά του επενδυτή. Τα χαρακτηριστικά ενός επενδυτή εξαρτώνται από το πόσα χρήματα θέλει να επενδύσει και από το χρονικό διάστημα που θέλει να επενδύσει.

1.3.1 Παραδοσιακή και Σύγχρονη Θεωρία Διαχείρισης Χαρτοφυλακίου

Η παραδοσιακή διαχείριση χαρτοφυλακίου έχει να κάνει κυρίως με την δημιουργία ενός αποτελεσματικού χαρτοφυλακίου το οποίο θα απαρτίζεται από μια ευρεία ποικιλία χρεογράφων. Οι διαχειριστές παραδοσιακών χαρτοφυλακίων αποστρέφονται τον κίνδυνο για αυτό και επιθυμούν να επενδύουν σε γνωστές εταιρείες για τρεις λόγους.

- Ο πρώτος λόγος είναι ότι επειδή αυτές οι εταιρείες είναι γνωστές στην αγορά ως επιτυχημένες, μια επένδυση σε αυτές θεωρείται λιγότερο επικίνδυνη από την επένδυση σε λιγότερο γνωστές επιχειρήσεις.
- Ο δεύτερος λόγος είναι ότι οι έμπειροι διαχειριστές, επιδιώκουν να επενδύουν σε μεγάλες εταιρείες επειδή τα χρεόγραφα αυτών των επιχειρήσεων είναι περισσότερο ρευστά και διατίθενται σε μεγάλες ποσότητες.
- Ο τρίτος και τελευταίος λόγος έχει να κάνει με το γεγονός ότι οι διαχειριστές των παραδοσιακών χαρτοφυλακίων προτιμούν τις γνωστές εταιρείες επειδή λόγω της φήμης τους είναι ευκολότερο να πείσουν τους πελάτες να επενδύσουν σε αυτές.

Η μοντέρνα θεωρία χαρτοφυλακίου, χρησιμοποιεί αρκετά βασικά στατιστικά μέτρα για την ανάπτυξη ενός σχεδίου για το χαρτοφυλάκιο. Η στατιστική διαφοροποίηση

αποτελεί σημαντικό παράγοντα στην επιλογή χρεογράφων για την δημιουργία ενός χαρτοφυλακίου.

Παρόλο που η διαχείριση χαρτοφυλακίου είναι από τα πιο σημαντικά θέματα στο κόσμο των επενδύσεων, δεν θα κάνουμε περαιτέρω ανάπτυξη της θεωρίας διαχείρισης καθώς δεν αποτελεί εξεταζόμενο μέρος της ανά χείρας εργασίας.

1.4 Χαρακτηριστικά Δεδομένα

Στην παρούσα παράγραφο θα ακολουθήσει μια ανάλυση-επεξήγηση κάποιων Δεδομένων τα οποία όπως θα δούμε και στη συνέχεια θα αποτελέσουν τον ακρογωνιαίο λίθο του συστήματος που πραγματεύεται η εν λόγω εργασία. Ο συνδυασμός των δεδομένων που θα αναλυθούν ουσιαστικά θα είναι τα κριτήρια αποφάσεων στα οποία θα στηριχτεί το Σύστημα Προβλέψεων μας.

1.4.1 Δεδομένα Τεχνικής Ανάλυσης

Οι Τεχνικοί Αναλυτές υποστηρίζουν πως είναι δυνατόν να προβλεφτούν οι μελλοντικές τιμές σε μια αγορά, αρκεί κανείς να μελετήσει και να αναλύσει με συγκεκριμένες μεθόδους το διάγραμμα τιμής (price chart) κάθε μετοχής. Δηλαδή το μόνο στοιχείο που μας αρκεί για την πρόβλεψη τιμών είναι να γνωρίζουμε τις παρελθούσες τιμές χωρίς να λαμβάνουμε υπόψη άλλες οικονομικές μεταβλητές και οικονομικά δεδομένα εταιρειών και κλάδων, όπως υποστηρίζουν οι θεωρητικοί της Θεμελιώδους Ανάλυσης. Η θεωρία της Τεχνικής Ανάλυσης βασίζεται σε τρεις υποθέσεις.

- Η πρώτη παραδοχή είναι ότι οι τιμές των μετοχών, δηλαδή η αγορά, προεξοφλεί όλα τα μελλοντικά γεγονότα. Έτσι στην τρέχουσα τιμή μιας μετοχής μπορούν και ενσωματώνονται οικονομικοί, λογιστικοί, ψυχολογικοί, πολιτικοί κ.α. παράγοντες.
- Η δεύτερη παραδοχή είναι ότι οι τιμές πολύ συχνά και για μεγάλα ή μικρά χρονικά διαστήματα ακολουθούν κάποια τάση, δηλαδή κάποια συγκεκριμένη κατεύθυνση.
- Τέλος πολλοί επενδυτές πιστεύουν ότι η ιστορία επαναλαμβάνεται. Η τεχνική ανάλυση πολλές φορές χρησιμοποιεί και επαναληπτικές συμπεριφορές των τιμών για να πάρει κάποια σήματα για την μελλοντική συμπεριφορά τους.

Ενώ οι περισσότεροι ακαδημαϊκοί είναι σκεπτικοί απέναντι στην Τεχνική Ανάλυση πολλοί επαγγελματίες εξακολουθούν να το υπολογίζουν ως αναπόσπαστο μέρος στην ανάλυση τους [6,7]. Ο Irwin και ο Park διαπίστωσαν το 2003 ότι 56 από 95 σύγχρονες μελέτες έδειξαν ότι η Τεχνική Ανάλυση παράγει θετικά αποτελέσματα [8]. Εντούτοις, η Τεχνική Ανάλυση εξακολουθεί να θεωρείται σε μεγάλο βαθμό ψευδοεπιστήμη από τον ακαδημαϊκό χώρο.

Η Τεχνική Ανάλυση χωρίζεται γενικά σε ποσοτική ανάλυση και χαρτογράφηση.

Ως ποσοτική ανάλυση λογίζεται η στατιστική ανάλυση των δεδομένων της αγοράς. Τα εν λόγω ποσοτικά, Τεχνικά, οικονομικά δεδομένα είναι ιδανικά για την αλγοριθμική ανάλυση την οποία θα ακολουθήσουμε και στην εργασία μας. Η χαρτογράφηση, από την άλλη πλευρά, προσπαθεί να εφαρμόσει οπτικές αναπαραστάσεις, ώστε να βρούμε επαναλαμβανόμενα μοτίβα, εφαρμόζοντας κατά μία έννοια, την ανθρώπινη διαίσθηση. Στον Πίνακα 2 παρουσιάζονται μερικές ποσοτικές μέθοδοι που εφαρμόζονται στην Τεχνική Ανάλυση.

Indicator	Full Name
SMA	Simple Moving Average
EMA	Exponential Moving Average
RSI	Relative Strength Index
MACD	Moving Average Convergence/divergence

Πίνακας 2 Ποσοτικοί Δείκτες Τεχνικής Ανάλυσης

1.4.2 Η θεωρία της «τυχαίας πορείας» (random walk theory)

Οι οπαδοί της θεωρίας της τυχαίας πορείας των τιμών (random walk theory) υποστηρίζουν πως οι διακυμάνσεις που παρατηρούμε καθημερινά στις τιμές των μετοχών οφείλονται σε καθαρά τυχαίους παράγοντες με αποτέλεσμα να μην μπορούμε να προβλέψουμε, με κανένα τρόπο, το πώς θα κινηθούν οι τιμές αυτές στο μέλλον. Με άλλα λόγια θεωρούν πως οι μέθοδοι της Τεχνικής Ανάλυσης διαγραμμάτων δεν έχει τίποτα να προσφέρει στον επενδυτή που θέλει να προβλέψει το μέλλον. Η θεωρία της «τυχαίας πορείας» βασίζεται με την σειρά της στην υπόθεση της αποτελεσματικής αγοράς (efficient market theory), η οποία στηρίζεται στον εξής λογικό συνειρμό: οι χρηματοοικονομικές πληροφορίες δημιουργούνται τυχαία και εφόσον οι τιμές ανταποκρίνονται τέλεια στις νέες πληροφορίες, αυτό συνεπάγεται ότι

οι τιμές κυμαίνονται τυχαία. Τόσο οι Τεχνικοί όσο και οι Θεμελιώδεις αναλυτές δεν δέχονται τη θεωρία της τυχαίας πορείας απορρίπτοντας έτσι την υποκείμενη θεωρία της αποτελεσματικής αγοράς. Ναι μεν οι χρηματιστηριακές πληροφορίες έρχονται τυχαία, η αντίδραση όμως της αγοράς δεν είναι ακαριαία: πρώτα θα αντιδράσουν οι «έξυπνοι επενδυτές» και ύστερα το πλήθος των μικροεπενδυτών.[25]

1.4.3 Δείκτες Τεχνικής Ανάλυσης - Ερμηνεία

Τα εργαλεία της Τεχνικής Ανάλυσης είναι οι δείκτες. Ένας δείκτης είναι ένας μαθηματικός υπολογισμός που στηρίζεται στην τιμή της μετοχής ή τον όγκο συναλλαγών της ή και στα δύο. Οι ειδικοί Τεχνικοί δείκτες παρέχουν ενδείξεις που αφορούν:

1. Το ξεκίνημα μιας νέας τάσης των τιμών (ανοδική ή καθοδική)
2. Την ταχύτητα και την ορμή αυτής της τάσης
3. Πόσο «ώριμη» είναι η τάση και πότε έχουμε τερματισμό της τάσης
4. Εάν η τάση πρόκειται να αναστραφεί.

Παρακάτω θα αναλυθούν κάποιοι από τους βασικότερους δείκτες Τεχνικής Ανάλυσης. Στην ανάλυση θα αναφερθούμε κυρίως στην πληροφορία που μπορούμε να εξάγουμε από κάθε δείκτη και συγκεκριμένους κανόνες συναλλαγών που μπορεί να προκύπτουν από τη χρήση αυτών.

Κινητοί μέσοι όροι – Moving Average

Οι απλούστεροι και ίσως δημοφιλέστεροι δείκτες τάσης των τιμών είναι οι Κινητοί Μέσοι. Με τον Κινητό Μέσο καταφέρνουμε να απαλείψουμε τις έντονες διακυμάνσεις της τιμής της μετοχής. Με αυτό τον τρόπο ο δείκτης μας δίνει την ομαλοποιημένη τάση (smoothed trend) της μετοχής. Ο Κινητός Μέσος παρέχει καθαρή εικόνα της τάσης της τιμής καθώς αυτή ξεκινάει, εξελίσσεται και ωριμάζει. Επίσης με τη χρήση του μπορούμε να αφαιρέσουμε την τάση από τις τιμές (detrending) και να μελετήσουμε την κυκλική συμπεριφορά της μετοχής.

Βασική παράμετρος του δείκτη αυτού είναι η περίοδος υπολογισμού του. Μπορεί να υπολογιστεί με βάση την τιμή ανοίγματος, κλεισίματος, υψηλότερη ημέρας ή χαμηλότερη ημέρας. Συνήθως ο Κινητός Μέσος υπολογίζεται πάνω στη τιμή κλεισίματος που δείχνει την τιμή στην οποία τείνει να ισορροπήσει η αγορά στο τέλος κάθε ημερήσιας συνεδρίασης.

Επίσης ο Κινητός Μέσος έχει και μια άλλη ερμηνεία που αφορά περισσότερο την ψυχολογία των επενδυτών. Είναι ένας δείκτης που μας δίνει τη μέση τιμή κτήσης της μετοχής της χρονικής περιόδου που αναφέρεται. Άρα όταν η τιμή κλεισίματος βρίσκεται πάνω από τον Κινητό Μέσο, η μέση τιμή κτήσης της μετοχής είναι χαμηλότερη από την τρέχουσα και άρα οι επενδυτές (αυτοί που τοποθετήθηκαν κατά το μελετώμενο χρονικό διάστημα) διατηρούν κλίμα ευφορίας και αισιοδοξίας από την άνοδο των κερδών τους. Το αντίθετο συμβαίνει όταν η τρέχουσα τιμή κλεισίματος βρίσκεται χαμηλότερα από τον Κινητό Μέσο, δηλαδή από την πρόσφατη μέση τιμή κτήσης της μετοχής. Το κλίμα είναι σαφώς αρνητικό με τους επενδυτές να υφίστανται ζημίες κατά μέσο όρο.

Έχουν δημιουργηθεί συγκεκριμένοι κανόνες συναλλαγών που προκύπτουν από τη χρήση των Κινητών Μέσων, οι οποίοι είναι οι εξής:

- Η ανοδική διάσπαση του ΚΜ από την τιμή κλεισίματος δίνει σήμα ενάρξεως ανοδικής τάσης και αγοράς της μετοχής.
- Η πτωτική διάσπαση του ΚΜ από την τιμή κλεισίματος σηματοδοτεί έναρξη πτωτικής τάσης και μας δίνει το σήμα πώλησης της μετοχής.
- Η μεταβολή της κλίσης του ΚΜ επιβεβαιώνει την αναστροφή της τάσης. Το σήμα αυτό λειτουργεί μόνο ως επιβεβαίωση του γεγονότος αντιστροφής, διότι έρχεται καθυστερημένα λόγω της χρονικής υστέρησης του ΚΜ.
- Όσο μεγαλύτερος ο αριθμός των ημερών στον ΚΜ, τόσο πιο μακροχρόνια είναι η τάση που αναλύει και τόσο πιο αξιόπιστα είναι τα σήματα αγοραπωλησιών που μας δίνει.
- Ο ΚΜ δεν προβλέπει την τάση, απλά ακολουθεί την τάση. Πρόκειται για ένα είδος «κινητής» γραμμής τάσης.

Όπως αναφέραμε το βασικότερο σημείο για την ερμηνεία του δείκτη αυτού είναι η περίοδος υπολογισμού του. Πώς διαλέγουμε όμως την περίοδο αυτή; Η απάντηση στο ερώτημα αυτό έχει σίγουρα να κάνει με το χρονικό ορίζοντα για τον οποίο περιμένουμε από τις επενδύσεις μας να αποδώσουν. Για παράδειγμα ο Κινητός Μέσος των 200 ημερών χρησιμοποιείται μόνο αν επενδύουμε με βλέψεις στο μακρινό μέλλον. Αν διαπραγματευόμαστε μετοχές καθημερινά, τότε ο Κινητός Μέσος των 15 ή 25 ημερών είναι πιο χρήσιμος.

Απλός Κινητός Μέσος - SMA (Simple Moving Average)

Ο Απλός Κινητός Μέσος υπολογίζεται αθροίζοντας την τιμή κλεισίματος του υποκείμενου προϊόντος για ένα συγκεκριμένο αριθμό χρονικών περιόδων και στη συνέχεια διαιρώντας το σύνολο αυτό με τον αριθμό των χρονικών περιόδων. Με άλλα λόγια, αυτή είναι η μέση τιμή της μετοχής κατά τη διάρκεια μιας ορισμένης χρονικής περιόδου.

Βραχυπρόθεσμοι SMA (υπολογιζόμενοι για μικρές χρονικές περιόδους) ανταποκρίνονται γρήγορα στις μεταβολές της τιμής του προϊόντος, ενώ οι μακροπρόθεσμοι μέσοι όροι αργούν να αντιδράσουν.

Θα πρέπει να τονίσουμε ότι η στάθμιση που δίδεται σε κάθε ημερήσια τιμή είναι ίση.

Ο Σταθμισμένος Κινητός Μέσος

Δίνει μεγαλύτερη βαρύτητα στις πρόσφατες τιμές από ότι στις παλαιότερες. Για να το πετύχουμε αυτό πολλαπλασιάζουμε την τιμή κάθε ημέρας με ένα ειδικό βάρος το οποίο είναι ανάλογα αυξανόμενο όσο πιο μεταγενέστερη είναι η ημερομηνία. Στον υπολογισμό αυτό η τιμή κλεισίματος της δέκατης ημέρας (στην περίπτωση ενός μέσου όρου δέκα ημερών) θα πρέπει να πολλαπλασιαστεί με δέκα, η ένατη ημέρα με εννιά, η όγδοη με οκτώ, φτάνοντας με τον τρόπο αυτό μέχρι την πρώτη. Το σύνολο στη συνέχεια διαιρείται από το άθροισμα των πολλαπλασιαστών (δηλαδή με τον αριθμό πενήντα πέντε στο παράδειγμα του μέσου όρου των δέκα ημερών). Παρόλα αυτά η μέθοδος αυτή περιλαμβάνει τιμές που καλύπτουν μόνο το μήκος του κινητού μέσου όρου αυτού καθ' εαυτού.

Εκθετικός Μέσος - EMA (exponential moving average)

Ένας τύπος κινούμενου μέσου όρου που είναι παρόμοιο με ένα Σταθμισμένο Κινούμενο Μέσο, με τη διαφορά ότι ο Εκθετικός Μέσος έχει την ιδιότητα της σταδιακής συρρίκνωσης της σημασίας των παλαιών δεδομένων, αλλά ποτέ την εξ ολοκλήρου αφαίρεση του.

Αυτός ο τύπος κινούμενου μέσου όρου αντιδρά γρηγορότερα στις πρόσφατες αλλαγές τιμών από ότι ένα Απλό Κινούμενο Μέσο Όρο.

Οι EMA 12 και 26 ημερών είναι οι πιο δημοφιλείς βραχυπρόθεσμοι μέσοι, ενώ σε γενικές γραμμές, οι EMA 50 και 200 ημερών χρησιμοποιούνται ως σήματα μακροπρόθεσμων τάσεων.

Τομή κινητών μέσων όρων (Moving average crossover)

Ο δείκτης Τομής Κινητών Μέσων χρησιμοποιεί δύο Κινητούς Μέσους με σημαντικά διαφορετικές περιόδους (π.χ., 20 και 50 μέρες).

Εάν ο σύντομος μέσος όρος ξεπερνά τον αργό (ο αργός μέσος όρος είναι ο πιο ομαλός από τους δύο). Το σημείο αυτό είναι σημείο αγοράς. Στο σημείο όπου είναι σαφές ότι ο σύντομος μέσος όρος είναι χαμηλότερα από τον αργό είναι σημείο πώλησης.

Δείκτης Σχετικής Δύναμης - Relative Strength

Μια τεχνική που συγκρίνει την απόδοση μιας μετοχής με εκείνη του συνόλου της αγοράς. Η Σχετική Ισχύς υπολογίζει ποιες επενδύσεις είναι οι πιο ισχυροί ερμηνευτές, σε σύγκριση με τη συνολική αγορά, και συνιστά τις συγκεκριμένες επενδύσεις για την αγορά

Υπάρχουν περισσότεροι από έναν τρόποι για τον υπολογισμό της Σχετικής Ισχύος μιας επένδυσης. Μια επένδυση βασισμένη σε RS διαθέτει τόσο στρατηγική εισόδου όσο και εξόδου. Επενδυτές που χρησιμοποιούν αυτήν την τεχνική ως στόχο να πωλούν τις μετοχές τους από τη στιγμή που αρχίζουν να εμφανίζονται αδύναμες. Η Σχετική Ισχύς μπορεί επίσης να εφαρμοστεί σε πιο σύνθετες στρατηγικές όπως τα ζεύγη συναλλαγών.

Ζώνες Bollinger - Bollinger Bands

Στη διάρκεια των περιόδων υψηλών διακυμάνσεων των τιμών συνίσταται να οριοθετούμε κάποιες περιοχές μέσα στις οποίες αναμένεται να ταλαντωθούν οι τιμές, έτσι ώστε να φαίνεται η στήριξη και η αντίσταση τους. Αυτές οι λωρίδες παρέχουν ψυχολογικά επίπεδα εγκλωβισμού των τιμών αλλά και συνθήκες για αναμονή βίαιων αντιδράσεων (ανοδικών ή καθοδικών).

Οι λωρίδες του Bollinger αποτελούνται από ένα απλό κινητό μέσο 20 ημερών (όπως έχει συστήσει ο εμπνευστής τους John Bollinger) ο οποίος περιστοιχίζεται από δύο λωρίδες. Η λωρίδα που βρίσκεται πάνω από τον ΚΜ20 προκύπτει εάν προσθέσουμε σε αυτόν δύο τυπικές αποκλίσεις των τιμών. Η λωρίδα στο κάτω μέρος του κινητού μέσου προκύπτει όταν αφαιρέσουμε από αυτόν δύο τυπικές αποκλίσεις των τιμών. Αυτές οι δύο γραμμές τυπικών αποκλίσεων αποτελούν το άνω και κάτω όριο των λωρίδων του Bollinger.

Όταν η μεταβλητότητα και οι διακυμάνσεις της μετοχής αυξάνονται, τότε το πλάτος των λωρίδων του Bollinger αυξάνει καθώς μεγαλώνει η τυπική απόκλιση των τιμών. Η κατεύθυνση των λωρίδων Bollinger εξαρτάται από την ίδια την κατεύθυνση του κινητού μέσου των 20 ημερών. Η συμπεριφορά των εν λόγω λωρίδων εξαρτάται από την συμπεριφορά της τάσης των τιμών. Μέσα σε μια ισχυρά ανοδική τάση οι λωρίδες του Bollinger έχουν περιορισμένο πλάτος και η κατεύθυνση τους είναι ανοδική. Όταν η τάση ωριμάζει και εμφανίζει κόπωση, οι λωρίδες ανοίγουν καθώς αυξάνεται η μεταβλητότητα των τιμών. Σε ισχυρά πτωτική τάση, οι λωρίδες είναι στενές και πτωτικές. Καθοδικές λωρίδες με αύξηση του πλάτους τους, σημαίνουν εξασθένηση πτωτικής τάσης.

Οι κανόνες συναλλαγών που βασίζονται στις λωρίδες του Bollinger είναι οι εξής:

- Ανοδική διάσπαση του άνω ορίου Bollinger από την τιμή κλεισίματος, δείχνει ισχυρή ανοδική τάση και δίνει σήμα αγοράς της μετοχής.
- Εάν η τιμή διασπάσει ανοδικά το άνω όριο Bollinger, αλλά στη συνέχεια σχηματίσει κορυφή κάτω από αυτό, αναμένεται υποχώρηση της τιμής και αναστροφή της προϋπάρχουσας ανοδικής τάσης.
- Καθοδική διάσπαση του κάτω ορίου Bollinger σηματοδοτεί πτώση της τιμής, άρα πουλάμε τη μετοχή.
- Εάν η τιμή διασπάσει καθοδικά το κάτω όριο αλλά στη συνέχεια δημιουργήσει πυθμένα πάνω από το όριο τότε θα επακολουθήσει ανάκαμψη.
- Αποτυχία της τιμής να διασπάσει το κάτω όριο σημαίνει πιθανή έναρξη κίνησης της μέχρι το άνω όριο και το αντίθετο για την αποτυχία διάσπασης του άνω ορίου.
- Μια ανοδική κίνηση της τιμής προς το άνω όριο Bollinger θα συναντήσει αντίσταση σε αυτό.
- Μια πτωτική κίνηση της τιμής προς το κάτω όριο Bollinger θα βρει στήριξη στο όριο αυτό.
- Κάθε προσπάθεια της τιμής να κινηθεί νευρικά από το άνω όριο προς το κάτω θα συναντήσει τουλάχιστον βραχυχρόνια στήριξη στον κινητό μέσο των 20 ημερών και κάθε αναπήδηση της τιμής από το κάτω όριο προς το πάνω θα βρει αντίσταση στον ίδιο κινητό μέσο.
- Μετά από κάθε δραματική μείωση του εύρους των λωρίδων Bollinger θα αναμένουμε βίαιη αντίδραση των τιμών, η κατεύθυνση της οποίας μπορεί να προβλεφθεί από άλλους τεχνικούς δείκτες.

- Οι λωρίδες Bollinger τείνουν να αυξήσουν το πλάτος τους περισσότερο σε περιοχές κορυφών των τιμών παρά σε περιοχές βάσης.

Η ερμηνεία και η χρήση των λωρίδων Bollinger αποτελεί περισσότερο μια τέχνη παρά μια επιστημονική μέθοδο παραγωγής σημάτων αγοραπωλησιών. Αυτό σημαίνει, συνεπώς, πως η οι κανόνες συναλλαγών και η συμπεριφορά των ζωνών ερμηνεύονται διαφορετικά στην πράξη μεταξύ διαφορετικών αναλυτών. Η χρήση ± 2 τυπικών αποκλίσεων για τον προσδιορισμό του εύρους τους δεν είναι δεσμευτική, καθώς σε μετοχές μικρής μεταβλητότητας μπορεί να χρησιμοποιηθεί ± 1 τυπική απόκλιση.

Ορμή - Momentum

Ο δείκτης Momentum δείχνει την ταχύτητα της μεταβολής (ή ρυθμός μεταβολής). Συνήθως, ο δείκτης Ορμής συγκρίνει την πιο πρόσφατη τιμή κλεισίματος σε μια προηγούμενη τιμή κλεισίματος, αλλά μπορεί επίσης να χρησιμοποιηθεί με βάση άλλους δείκτες (όπως τους κινητούς μέσους όρους).

Ο δείκτης αυτός ουσιαστικά μετράει το πόσο αλλάζει η τιμή της μετοχής σ' ένα καθορισμένο χρονικό διάστημα. Έχει επίπεδο ισορροπίας το 100, στο οποίο η ορμή των τιμών θεωρείται μηδέν, γεγονός που υποδεικνύει πως στην αγορά δεν υπάρχει υπερβάλλουσα ζήτηση ή προσφορά του προϊόντος.

Η σημασία του επιπέδου ισορροπίας είναι διττή:

1. Το επίπεδο αυτό αποτελεί ένα έναυσμα για εκκίνηση νέας κατεύθυνσης τιμών με την έννοια πως όποτε ο δείκτης διαπερνά το επίπεδο των 100 από κάτω προς τα πάνω παίρνουμε σήμα ανόδου της μετοχής και το αντίθετο για την καθοδική διάσπαση του 100.
2. Το επίπεδο του 100 θεωρείται «ισορροπία» με την έννοια πως ο δείκτης δεν μπορεί να αποκλίνει για μεγάλο χρονικό διάστημα και για μεγάλη απόσταση «τιμών» από το επίπεδο αυτό και άρα τείνει πάντα να επιστρέφει στο επίπεδο αυτό.

ALF (Alexander's Filter)

Ένας δείκτης που αναφέρεται την ποσοστιαία άνοδο ή την πτώση της τιμής της μετοχής σε μια δεδομένη χρονική περίοδο.

Όπως ισχύει και στους άλλους δείκτες ορμής, η αγορά πρέπει να βρίσκεται σε έντονα τασική κίνηση ώστε να δώσει τα πιο αξιόπιστα σήματα.

Ταλαντωτές - Oscillators

Οι ταλαντωτές είναι εξαιρετικά χρήσιμοι σε μη-τασικές αγορές όπου οι τιμές κυμαίνονται σε ένα οριζόντιο εύρος τιμών, δημιουργώντας μία κατάσταση αγοράς όπου τα περισσότερα συστήματα που ακολουθούν την τάση απλά δεν λειτουργούν τόσο καλά. Οι ταλαντωτές μπορούν να παρέχουν στον Τεχνικό Αναλυτή ένα εργαλείο που μπορεί να τον βοηθήσει να καρπωθεί κέρδος από αυτές τις περιοδικές πλευρικές και ατασικές κινήσεις της αγοράς.

Η τιμή του ταλαντωτή παρόλα αυτά δεν περιορίζεται στα οριζόντια όρια διαπραγμάτευσης. Χρησιμοποιούμενα σε συνδυασμό με γραφήματα τιμών κατά την διάρκεια εξέλιξης μιας τάσης, ο ταλαντωτής γίνεται ένα χρήσιμο εργαλείο με το να ειδοποιεί τον συναλλασσόμενο για βραχυπρόθεσμες υπερβολές της αγοράς, που κοινώς ονομάζονται υπεραγορασμένες ή υπερπουλημένες συνθήκες.

Ανάλογα με την μέθοδο που ακολουθείται, η γραμμή της μέσης τιμής είναι συνήθως η γραμμή του μηδενός. Μερικοί ταλαντωτές έχουν επίσης άνω και κάτω εύρος τιμών από το 0 έως το 100.

Ως γενικός κανόνας είναι ότι όταν ο ταλαντωτής πλησιάζει μία ακραία τιμή τόσο στο άνω εύρος όσο και στο κάτω της λωρίδας, τότε αυτό σημαίνει ότι η κίνηση της τρέχουσας τιμής έχει κινηθεί πολύ έντονα προς μια κατεύθυνση και σε λίγο χρονικό διάστημα που το αναμενόμενο είναι να υπάρξει μία διόρθωση ή ένα είδος συσσώρευσης. Ως ένας άλλος γενικός κανόνας είναι ότι ο συναλλασσόμενος θα πρέπει να είναι σε θέση αγοραστή όταν η γραμμή του ταλαντωτή είναι προς τον πυθμένα της λωρίδας και πωλητή όταν η γραμμή βρίσκεται προς το ανώτερο όριο.

Οι τρεις πιο σημαντικές χρήσεις ενός ταλαντωτή είναι οι παρακάτω:

- Όταν η τιμή του αγγίζει μια ακραία τιμή κοντά στο άνω ή στο κάτω όριο της λωρίδας. Η αγορά λέγεται ότι έχει υπεραγοραστεί όταν βρίσκεται στην άνω ακραία τιμή ή υπερπουληθεί όταν βρίσκεται στην κάτω ακραία τιμή. Αυτό προειδοποιεί ότι η τάση τιμών έχει υπερεκταθεί και μπορεί εύκολα να αλλάξει.
- Η διάσχιση της γραμμής της μέσης τιμής μπορεί να δώσει ένα σημαντικό σήμα συναλλαγής προς την κατεύθυνση της τάσης των τιμών.

Κινητός Μέσος Σύγκλισης / Απόκλισης (MACD)

Ο Κινητός Μέσος Σύγκλισης / Απόκλισης (MACD) είναι ένας δείκτης που χρησιμοποιεί τη διαφορά ανάμεσα σε ένα γρήγορο και αργό Εκθετικό Κινητό Μέσο (EMA). Η γραμμή σήματος MACD, όπως λέγεται, προέρχεται από ένα άλλο EMA του MACD. Το τελικό ιστόγραμμα MACD είναι η διαφορά της γραμμής MACD και της γραμμής σήματος MACD.

Ο δείκτης αυτός δίνεται εάν αφαιρέσουμε τον εκθετικό κινητό μέσο των 26 ημερών από τον αντίστοιχο κινητό μέσο των 12 ημερών. Όμως, επειδή χρειαζόμαστε ταχύτερα σήματα αγοραπωλησιών από αυτά που δίνει η απλή διάσπαση της γραμμής του μηδενός από τον MACD, χρησιμοποιούμε τον εκθετικό κινητό μέσο των 9 ημερών του MACD τον οποίο ονομάζουμε γραμμή σήματος (signal line). Ένα χαρακτηριστικό σήμα είναι όταν ο MACD διασπάσει ανοδικά τη γραμμή σήματος του, αυτό είναι σήμα αγοράς της μετοχής, ακόμη και εάν ο MACD τη δεδομένη στιγμή βρίσκεται κάτω από το μηδέν. Τα σήματα που παίρνουμε με τη χρήση του MACD συνήθως προηγούνται των σημάτων του Κινητού Μέσου. Ως αποτέλεσμα, ο MACD προσφέρει τα καλύτερα των δύο κόσμων: ακόλουθη τάση και ορμή. Ο MACD διακυμαίνεται πάνω και κάτω από τη γραμμή μηδέν.

Για την ερμηνεία του συγκεκριμένου δείκτη υπάρχουν τρεις διαφορετικοί τρόποι:

1) Τομές του "σηματοδότη" με το δείκτη Απόκλισης - Σύγκλισης

Ο βασικός κανόνας είναι να πουλάμε όταν ο δείκτης (οι πολλαπλές κατακόρυφες γραμμές) πέφτει κάτω από το επίπεδο του "σηματοδότη" (η συνεχής γραμμή), και να αγοράζουμε όταν συμβαίνει το αντίστροφο. Μία άλλη δημοφιλής μέθοδος είναι να αγοράζουμε (πουλάμε) όταν ο δείκτης (οι πολλές γραμμές) ξεπερνάει (πέφτει κάτω από), το μηδέν.

2) Συνθήκες υπερ-αγοράς / υπερ-πώλησης

Ο δείκτης είναι επίσης χρήσιμος ως εκφραστής του πόσο υπερ-αγορασμένη ή υπερ-πουλημένη είναι μια μετοχή. Όταν ο βραχυπρόθεσμος μέσος όρος (με περίοδο 12) απομακρύνεται απότομα από τον μακροπρόθεσμο (με περίοδο 26), που σημαίνει ότι οι πολλαπλές κατακόρυφες γραμμές έχουν πορεία προς τα πάνω, είναι πιθανό η μετοχή να ετοιμάζεται για διόρθωση (αντιστροφή).

3) Αποκλίσεις

Όταν ο δείκτης Σύγκλισης-Απόκλισης αποκλίνει σε πορεία από την τιμή της μετοχής, αυτό είναι ένδειξη ότι η τωρινή κατεύθυνση της μετοχής είναι πιθανό να αντιστραφεί.

Μια αρνητική απόκλιση (bearish divergence) συμβαίνει όταν ο δείκτης πέφτει όλο και πιο χαμηλά, ενώ οι τιμές δεν ακολουθούν την ίδια πορεία. Μια θετική απόκλιση (bullish divergence) συμβαίνει όταν ο δείκτης ανεβαίνει όλο και πιο ψηλά, ενώ η μετοχή όχι. Και οι δύο αυτές αποκλίσεις θεωρούνται πιο σημαντικές όταν συμβαίνουν σε επίπεδα υπερ-αγοράς ή υπερ-πώλησης.

Δείκτης Ροής Χρημάτων - MFI (Money Flow Index)

Ο Δείκτης Money Flow Index (MFI) είναι ένας ταλαντωτής. Για τον υπολογισμό του χρησιμοποιείται τόσο η τιμή όσο και ο όγκος των συναλλαγών. Δημιουργήθηκε από τον Gene Quong και Avrum Soudack. Είναι επίσης γνωστός ως σταθμισμένος RSI.

Η τιμή του Money Flow Index είναι θετική όταν οι τυπικές τιμές αυξάνονται (πίεση αγοράς) και αρνητική όταν οι τυπικές τιμές μειώνονται.

Μία αναλογία των θετικών και αρνητικών MFI, συνδεδεμένη σε μια φόρμουλα RSI μπορεί δημιουργήσει έναν ταλαντωτή που κινείται μεταξύ 0-100. Ως ταλαντωτής ορμής συνδεδεμένος με τον όγκο, ο MFI είναι η πλέον κατάλληλος για τον εντοπισμό ανατροπών και ακραίων τιμών με μια ποικιλία των σημάτων

Ως όγκο-σταθμισμένη εκδοχή του RSI, ο Δείκτης Money Flow (MFI), μπορεί να ερμηνευθεί παρόμοια με τον RSI. Επειδή ο όγκος προστίθεται στο μίγμα, ο MFI θα ενεργήσει λίγο διαφορετικά από ότι ο RSI. Έχει παρατηρηθεί ότι η ενσωμάτωση του όγκου μπορεί να αυξήσει το προβάδισμα στο χρόνο σε σχέση με τον RSI.

Ο MFI χρησιμοποιείται για τον εντοπισμό μη βιώσιμων ακραίων τιμών. Συνήθως, MFI πάνω από 80 το υποκείμενο προϊόν θεωρείται υπερτιμημένο και MFI κάτω των 20 το προϊόν θεωρείται υποτιμημένο. Ισχυρές τάσεις μπορεί να παρουσιάσουν ένα πρόβλημα για αυτά τα κλασικά επίπεδα. Οι Quong και Soudack πρότειναν την επέκταση αυτών των ακρότατων για να τύχουν περαιτέρω σήματα. Μια κίνηση πάνω από 90 είναι πραγματικά υπερ-αγορασμένη και μια κίνηση κάτω των 10 είναι ερμηνεύεται ως υπερ-πουλημένη. Κινήσεις άνω των 90 και κάτω των 10 είναι σπάνια γεγονότα που υποδηλώνουν ότι μια κίνηση των τιμών δεν είναι βιώσιμη και δεν θα κρατήσει πολύ.

Ταλαντωτής τιμής (Price Oscillator)

Ο Ταλαντωτής Τιμής δείχνει τη διαφορά μεταξύ δύο κινητών μέσων όρων μιας μετοχής. Η διαφορά αυτή μπορεί να εκφραστεί σε μονάδες μέτρησης ίδιες με την τιμή της μετοχής ή σε ποσοστά. Ο Ταλαντωτής Τιμής είναι σχεδόν ίδιος με το δείκτη

Σύγκλισης- Απόκλισης (MACD), μόνο που στον Ταλαντωτή Τιμής ο τεχνικός αναλυτής μπορεί να χρησιμοποιήσει όποιους μέσους όρους θέλει, ενώ στο δείκτη Σύγκλισης-Απόκλισης αυτοί οι μέσοι όροι είναι των 12 και 26 ημερών.

Ο ταλαντωτής τιμής (Price Oscillator) εκφράζεται ως η ποσοστιαία διαφορά ενός απλού βραχυχρόνιου κινητού μέσου και ενός απλού μεσοπρόθεσμου ΚΜ. Ο βραχυχρόνιος ΚΜ μπορεί να πάρει τιμές από 5-21 ημέρες και ο μεσοπρόθεσμος από 60-90 ημέρες. Όμως όπως και στο MACD, η διαφορά των δύο κινητών μέσων δε χρησιμοποιείται από μόνη της αλλά σε συνδυασμό με τον απλό κινητό της μέσο που παίρνει τιμές από 45-60 ημέρες.

Ο ταλαντωτής Price Oscillator εκφράζεται ως ποσοστό επί τις εκατό και «ταλαντώνεται» γύρω από το επίπεδο του μηδενός.

Τυπικά μια ανάλυση που χρησιμοποιεί μέσους όρους υποδεικνύει σημεία αγοράς όταν ένας σύντομος μέσος όρος ξεπερνάει κάποιον αργό, και σημεία πώλησης όταν ο αργός ξεπερνάει το σύντομο. Σε αυτά τα πλαίσια κινείται και η ερμηνεία αυτού του δείκτη.

Ο ταλαντωτής των τιμών υπόκειται στο ίδιο μειονέκτημα με τον MACD, δηλαδή έχει σταθερό χρονικό ορίζοντα. Βέβαια θεωρείται πιο μακροχρόνιος δείκτης από τον MACD και λιγότερο ευαίσθητος, ως προς απότομες μεταβολές των τιμών. Ο ταλαντωτής των τιμών, όπως και ο δείκτης MACD δίνουν πιο έγκυρα σήματα σε περίοδο τάσης των τιμών, και αυτή την τάση πρέπει να ακολουθούμε.

Δείκτης καναλιού άξιας (Commodity Channel Index - CCI)

Ο δείκτης Commodity Channel είναι ένας ταλαντωτής που μετρά την απόκλιση της τιμής μιας μετοχής από το στατιστικό της μέσο όρο. Αρχικά η μέθοδος εφαρμόστηκε διεθνώς στην Τεχνική Ανάλυση των συμβολαίων futures και αργότερα έγινε δημοφιλής στην ανάλυση μετοχών. Ο CCI εντοπίζει το ξεκίνημα και τη λήξη των κυκλικών διακυμάνσεων επιτρέποντας την εκμετάλλευση των επενδυτικών τους ευκαιριών.

Τα στάδια υπολογισμού του Commodity Channel Index είναι τα εξής:

Ο δείκτης commodity channel δεν έχει προκαθορισμένα όρια ταλάντωσης όπως άλλοι ταλαντωτές, αλλά η πλειοψηφία των τιμών του βρίσκονται μέσα στη ζώνη του ± 100 .

Υπάρχουν δυο βασικές τεχνικές αγοραπωλησιών με τον ταλαντωτή Commodity Channel Index:

- Όταν ο ταλαντωτής περάσει σε υπερ-αγορασμένη περιοχή και γυρίσει διασπώντας καθοδικά τη γραμμή του +100, τότε επίκειται διορθωτική πτώση της τιμής. Όταν από την υπερ-πουλημένη ζώνη του διασπάσει ανοδικά τη γραμμή του -100, επίκειται άνοδος της τιμής. Αυτή η τεχνική είναι αξιόπιστη μόνο μέσα σε ζώνες συναλλαγών.
- Τα πιο αξιόπιστα όμως σήματα του ταλαντωτή είναι οι αρνητικές και θετικές αποκλίσεις. Όταν σχηματίσει θετική απόκλιση μέσα σε υπερ-πουλημένη περιοχή, τότε επίκειται αναστροφή της καθοδικής τάσης της μετοχής. Ο σχηματισμός αρνητικής απόκλισης σε υπερ-αγορασμένη περιοχή είναι αξιόπιστο σήμα αντιστροφής της ανοδικής τάσης της αγοράς.

Η πρώτη τεχνική αγοραπωλησιών του ταλαντωτή συνίσταται κυρίως για πολύ επιθετικούς επενδυτές. Η τεχνική των αρνητικών και θετικών αποκλίσεων είναι πιο αξιόπιστη και μπορεί να εφαρμοστεί και από πιο συντηρητικούς επενδυτές.

Συγκεκριμένα όταν ο ταλαντωτής διασπάσει καθοδικά τη γραμμή του +100 αφού πρώτα έχει σχηματίσει αρνητική απόκλιση σε υπέρ-αγορασμένη περιοχή, πουλάμε τη μετοχή με την προϋπόθεση όμως ότι εάν ο ταλαντωτής ξανά διασπάσει ανοδικά τη γραμμή χωρίς να μπορέσει να υποχωρήσει σε υπερ-πουλημένη περιοχή, τότε θα επανατοποθετηθούμε στη μετοχή.

Αντίστοιχα όταν θέλουμε να τοποθετηθούμε στην έναρξη μιας ανοδικής τάσης, περιμένουμε να σχηματιστεί θετική απόκλιση του ταλαντωτή μέσα σε υπερ-πουλημένη περιοχή και στη συνέχεια να διασπάσει ανοδικά τη γραμμή του -100. Εάν όμως ο δείκτης πισωγυρίσει χωρίς να έχει την δύναμη να μπει σε υπερ-αγορασμένη περιοχή και ξανά διασπάσει καθοδικά τη γραμμή του -100, θα πρέπει να ρευστοποιήσουμε. Αυτό συνήθως είναι ένα σημάδι προέκτασης της πτωτικής τάσης και με τον τρόπο αυτό θα καταφέρουμε να μπούμε στην αγορά σε πιο ευνοϊκή τιμή, λαμβάνοντας ένα νέο σήμα ανοδικής διάσπασης του -100. Εάν μετά την αγορά ο δείκτης εισέλθει σε υπερ-αγορασμένη περιοχή, θα πρέπει να πληροί την προϋπόθεση της αρνητικής απόκλισης για να ρευστοποιήσουμε.

Δείκτης σχετικής δύναμης (Relative Strength Index - RSI)

Ο δείκτης σχετικής ισχύος (RSI) αναπτύχθηκε από τον J. Welles Wilder, Jr. το 1978. Ως δείκτης ορμής, ο RSI μετράει την ταχύτητα της κίνησης των τιμών. Στο μοντέλο αυτό, οι τιμές γενικά θεωρούνται ελαστικές στο ότι μπορούν να μετακινηθούν κατά περιορισμένη μόνο απόσταση από το μέσο όρο πριν αντιδράσουν ή οπισθοχωρήσουν.

Η ταχεία άνοδος των τιμών οδηγεί σε υπερ-αγορασμένες καταστάσεις και η ταχεία πτώση σε υπερ-πουλημένες καταστάσεις. Η κλίση και οι τιμές του RSI είναι ευθέως ανάλογες προς την ταχύτητα και το μέγεθος της κίνησης των τιμών και είναι άκρως χρήσιμες για τον εντοπισμό υπερ-αγορασμένων και υπερ-πουλημένων καταστάσεων.

Ο δείκτης RSI είναι ο πιο δημοφιλής ίσως ταλαντωτής και χρησιμεύει κυρίως στο να παρέχει προειδοποιητικά σήματα για α) πιθανή εξάντληση της ανοδικής ή καθοδικής τάσης των τιμών, β) ενδείξεις επερχόμενης βίαιης αντιστροφής της αγοράς και γ) αξιόπιστα σήματα αγοραπωλησιών μέσα σε ζώνες συναλλαγών.

Το όνομα του δείκτη ίσως είναι παραπλανητικό διότι δείχνει να έχει το ρόλο σύγκρισης της απόδοσης της μετοχής με το γενικό δείκτη. Ένα πιο σωστό όνομα θα ήταν δείκτης εσωτερικής δύναμης. Στην πραγματικότητα εξετάζει πόσο ενισχυμένη φαίνεται να είναι η τρέχουσα τιμή της μετοχής σε σχέση με την ορμή που έχει αποκτήσει στις τελευταίες της συνεδριάσεις. Από εκεί μπορούμε να καταλάβουμε εάν η μετοχή βρίσκεται χαμηλά ή ψηλά σε σχέση με τα επίπεδα που έχει αγγίξει στο παρελθόν και άρα εάν θεωρείται υπερ-αγορασμένη ή υπερ-πουλημένη.

Όσον αφορά τον αριθμό ημερών που χρησιμοποιείται στον RSI, ο Wilder στην αρχική παρουσίαση του δείκτη πρότεινε το 14. Αρκετοί αναλυτές χρησιμοποιούν εναλλακτικά τις 9 ημέρες για πιο βραχυχρόνιες αναλύσεις. Σήμερα οι περίοδοι των 9 και 25 ημερών είναι εξίσου δημοφιλείς.

Το χαρακτηριστικό του RSI είναι πως πρόκειται για ένα δείκτη που ταλαντώνεται μεταξύ του 100 και του μηδενός και έχει σαφή όρια υπερτίμησης και υποτίμησης του. Όταν ο δείκτης παίρνει τιμές μεγαλύτερες του 70 θεωρείται υπερ-αγορασμένος, ενώ όταν πέφτει κάτω από το 30 θεωρείται υπερ-πουλημένος. Όταν ο δείκτης κινείται μεταξύ του 30 και του 70 θεωρείται πως βρίσκεται σε ουδέτερη ζώνη και δεν παρουσιάζει ενδιαφέρον, ενώ το επίπεδο ισορροπίας του δείκτη είναι το 50.

Οι βασικότερες μέθοδοι αγοραπωλησιών με το δείκτη RSI είναι οι ακόλουθοι:

- Κάθε φορά που ο RSI διασπά τη γραμμή του 70 καθοδικά δίνεται σήμα πώλησης της μετοχής, ενώ όταν ο δείκτης διασπά τη γραμμή του 30 ανοδικά δίνεται σήμα αγοράς. Όμως οι απλές διασπάσεις των υπερ-αγορασμένων - υπερ-πουλημένων γραμμών του RSI δεν είναι ιδιαίτερα αξιόπιστα σήματα στην πράξη, καθώς ο δείκτης παρασύρεται σε πολλαπλές διασπάσεις σε περιόδους τάσης οι οποίες δεν αντικατοπτρίζουν αγοραστικές ευκαιρίες ή κατάλληλες τιμές για ρευστοποίηση αλλά προέρχονται από την κεκτημένη ταχύτητα και την υπερβάλλουσα ορμή των τιμών.

- Σχηματισμοί κορυφής και βάσης. Όταν ο RSI σχηματίζει κορυφή πάνω από το 70 έχουμε ένδειξη δημιουργίας κορυφής στις τιμές και μάλιστα αντιμετωπίζουμε το ενδεχόμενο η αγορά να στραφεί προς τα κάτω με καθοδική αντίδραση. Όταν ο RSI σχηματίζει κοιλότητα κάτω από το 30 έχουμε ένδειξη δημιουργίας βάσης στις τιμές και ετοιμαζόμαστε για πιθανή αναστροφή της αγοράς προς τα πάνω.
- Αποκλίσεις κορυφής (top failure swings) και αποκλίσεις βάσης (bottom failure swings). Απόκλιση κορυφής εντοπίζεται όταν ο δείκτης σχηματίζει υψηλή κορυφή πάνω από το 70 και στη συνέχεια σχηματίζει δεύτερη κορυφή επίσης πάνω από το 70 αλλά χαμηλότερα από την πρώτη, ενώ στην επακόλουθη κάθοδο του κάτω από το 70 διασπά το χαμηλό του μεταξύ των δύο αυτών κορυφών. Το top failure swing αποτελεί ένδειξη ισχυρής αδυναμίας της μετοχής. Απόκλιση βάσης εμφανίζεται όταν ο δείκτης σχηματίζει χαμηλό πυθμένα κάτω από το 30 και στη συνέχεια δεύτερο πυθμένα επίσης κάτω από το 30 αλλά υψηλότερα από τον πρώτο, ενώ στην ακόλουθη άνοδο του δείκτη πάνω από το 30 διασπά το υψηλό μεταξύ των δύο αυτών πυθμένων. Το bottom failure swing του RSI αποτελεί ένδειξη συγκέντρωσης ισχύος της μετοχής. Τόσο οι σχηματισμοί αποκλίσεων κορυφής όσο και οι σχηματισμοί αποκλίσεων βάσης είναι βραχυχρόνια φαινόμενα συνήθους διάρκειας λίγων εβδομάδων.
- Γεωμετρικοί σχηματισμοί (chart formations). Ο RSI έχει το χαρακτηριστικό να εμφανίζει αρκετά συχνά σχηματισμούς κορυφής, βάσης ή συνέχισης όπως ακριβώς και οι τιμές μέσα στο bar chart. Πολλές φορές ενδέχεται ο RSI να εμφανίζει σχηματισμούς που δεν είναι ορατοί στο bar chart της μετοχής και μας θέτει σε αυξημένη ετοιμότητα για πιθανή αντίδραση της αγοράς. Οι σχηματισμοί κορυφής είναι περισσότερο αξιόπιστοι όσο πιο ψηλά βρίσκονται μέσα στην υπερτιμημένη ζώνη, ενώ οι σχηματισμοί βάσης είναι περισσότερο αξιόπιστοι όσο χαμηλότερα βρίσκονται μέσα στην υποτιμημένη ζώνη.
- Επίπεδα στήριξης / αντίστασης. Μπορούμε να ορίσουμε επίπεδα αντίστασης του RSI από τις αδιάσπαστες κορυφές που έχει σχηματίσει πάνω από το 70 και τα οποία αντιστοιχούν σε επίπεδα αντίστασης των ιδίων των τιμών. Η διάσπαση ενός επιπέδου αντίστασης της τιμής, της οποίας έχει προηγηθεί διάσπαση της αντίστασης του RSI θεωρείται σαφώς πιο αξιόπιστο σήμα αγοράς της μετοχής.
- Θετικές και αρνητικές αποκλίσεις. Όταν ο RSI σχηματίζει αρνητική απόκλιση από τις τιμές και μάλιστα σε υπερ-αγορασμένη περιοχή, τότε εμφανίζεται η ένδειξη εξάντλησης και αναστροφής της ανοδικής τάσης των τιμών. Όταν ο RSI σχηματίζει θετική απόκλιση από τις τιμές και μάλιστα σε υπερ-πουλημένη περιοχή τότε έχουμε ενδείξεις επίθεσης των αγοραστών. Οι

αρνητικές και θετικές αποκλίσεις διαφέρουν από τα top / bottom failure swings καθώς είναι πιο μακροχρόνια φαινόμενα.

- Γραμμές τάσης. Όπως ακριβώς γίνεται και στις τιμές, μπορούμε να σχεδιάσουμε γραμμές ανοδικής και καθοδικής τάσης στον RSI και να χρησιμοποιήσουμε τα σήματα διάσπασης των γραμμών τάσης για να προετοιμαστούμε προκειμένου να λάβουμε θέσεις στην αγορά. Οι θέσεις ενεργοποιούνται με τα σήματα που παίρνουμε από τις ίδιες τις τιμές και τη διάσπαση των δικών τους γραμμών τάσης.
- Εφαρμογή κινητών μέσων στον RSI. Προκειμένου να απαλειφθεί η επίδραση των έντονων και βραχύβιων διακυμάνσεων των τιμών στον RSI, συχνά προστίθεται στο διάγραμμα του δείκτη και ένας κινητός του μέσος (των 3, 5 ή 7 ημερών). Όταν ο RSI διασπά καθοδικά τον κινητό του μέσο μέσα σε υπερ-αγορασμένη περιοχή, τότε δίνεται σήμα πώλησης και όταν διασπά ανοδικά τον κινητό του μέσο σε υπειλημμένη περιοχή, τότε δίνεται σήμα αγοράς.

Στοχαστικός δείκτης (Stochastics - K%D)

Ο στοχαστικός δείκτης επινοήθηκε από τον αναλυτή George Lane ο οποίος βασίστηκε στην ιδέα πως κατά τη διάρκεια μιας ανοδικής τάσης οι τιμές συνήθως κλείνουν κοντά στο μέγιστο του πρόσφατου εύρους τιμών, υποδηλώνοντας διάθεση να διασπούν διαρκώς ανοδικά αυτό το εύρος τιμών. Όταν όμως η ανοδική τάση πλησιάζει σε επίπεδα ωρίμανσης και εξάντλησης, τότε οι τιμές απομακρύνονται διαρκώς από το μέγιστο του πρόσφατου εύρους τιμών. Το αντίθετο φαίνεται να συμβαίνει όταν οι τιμές βρίσκονται σε καθοδική τάση.

Ο δείκτης αυτός, μπορεί να αποτελέσει πολύτιμο εργαλείο για τον εντοπισμό επικείμενων κορυφών και πυθμένων, βοηθώντας έτσι το χρονισμό των αγοραπωλησιών κοντά στα σημεία αντιστροφής. Μετράει την τοποθέτηση μιας τρέχουσας τιμής μέσα σε ένα πρόσφατο εύρος διακύμανσης. Αποτελείται από δύο επιμέρους δείκτες τον %D και τον %K.

Ο δείκτης %D είναι ο πιο σημαντικός καθώς παρέχει τα σήματα αγοραπωλησιών του στοχαστικού δείκτη.

Ο δείκτης %K ταλαντώνεται μεταξύ του 100 και του 0 και θεωρείται υπερ-αγορασμένος όταν βρεθεί πάνω από το 70 και υπερ-πουλημένος όταν πέσει κάτω από το 30. Δηλαδή εάν η τιμή κινηθεί κοντά στο άνω όριο του πρόσφατου εύρους τιμών, τότε ο %K θα πάρει τιμές μεγαλύτερες του 70. Εάν η τιμή τείνει να πλησιάσει το

ελάχιστο του πρόσφατου εύρους τιμών της, τότε ο %K θα πάρει τιμές μικρότερες του 30.

Ο δείκτης %D είναι πιο αργός από τον %K και ταλαντώνεται και αυτός μεταξύ του 100 και του μηδενός με τις ίδιες υπερ-αγορασμένες και υπερ-πουλημένες περιοχές τιμών. Ο %D είναι μια ομαλοποιημένη έκδοση του %K. Ο τύπος υπολογισμού του %D είναι ο ακόλουθος:

Υπάρχουν αρκετοί τρόποι ερμηνείας του στοχαστικού δείκτη.

- Έχουμε σήμα αγοράς όταν ο δείκτης (ο "%K" ή ο "%D") πέφτει κάτω από κάποιο επίπεδο (20 για παράδειγμα), και μετά ξεπερνάει αυτό το επίπεδο. Πουλάμε όταν ο δείκτης ξεπερνάει κάποιο επίπεδο (80 π.χ.) και μετά πέφτει από αυτό το επίπεδο.
- Αγοράζουμε όταν η γραμμή "%K" ξεπερνάει την γραμμή "%D" και πουλάμε όταν συμβαίνει το αντίστροφο.
- Ψάχνουμε για αποκλίσεις. Για παράδειγμα, όταν η μετοχή κάνει καινούρια ρεκόρ και ο Στοχαστικός δείκτης δεν ξεπερνάει προηγούμενα ύψη του.

Ο δείκτης % Williams

Ο ταλαντωτής Williams %R είναι παρόμοιος με τον στοχαστικό ταλαντωτή με τη διαφορά ότι συγκρίνει την τρέχουσα τιμή κλεισίματος με την υψηλότερη ενδοημερήσια τιμή του πρόσφατου εύρους τιμών. Είναι ένας δείκτης που σχετίζεται με την ορμή μιας μετοχής.

Ο Williams %R ταλαντώνεται μεταξύ του -100 και του 0. Υπερ-αγορασμένη περιοχή τιμών θεωρείται η ζώνη πάνω από το -20 και υπερ-πουλημένη η ζώνη χαμηλότερα του -80. Ο ταλαντωτής Williams %R δε χρησιμοποιείται τόσο για να δίνει άμεσα σήματα αγοραπωλησιών όσο για τον εντοπισμό κορυφών και πυθμένων των πραγματικών τιμών.

Η κίνηση του δείκτη αυτού προηγείται της κίνησης της μετοχής, δηλαδή αν η μετοχή είναι υπερ-αγορασμένη ο δείκτης αρχίζει και πέφτει πριν αρχίσει η πτώση της μετοχής, ενώ αν είναι υπερ-πουλημένη αρχίζει και ανεβαίνει πριν από την άνοδο της μετοχής. Γι' αυτόν ακριβώς το λόγο είναι καλή ιδέα να χρησιμοποιεί κανείς τον δείκτη αυτό σε συνδυασμό με άλλα εργαλεία της τεχνικής ανάλυσης, πριν αποφασίσει να αγοράσει ή να πουλήσει μετοχές.

Ο συγκεκριμένος δείκτης δεν ενδείκνυται για επενδυτές με μακροχρόνιους στόχους.

Δείκτης συσσώρευσης - διανομής (Accumulation – Distribution Index)

Ο δείκτης Συσσώρευσης-Διανομής (Larry Williams 1972) είναι ένας δείκτης που συνδυάζει αλλαγές στην τιμή της μετοχής και του όγκου συναλλαγών. Αυτός ο δείκτης στηρίζεται στην υπόθεση ότι όσο μεγαλύτερος όγκος συναλλαγών συνοδεύει την κίνηση μιας μετοχής, τόσο πιο σημαντική είναι αυτή η κίνηση (ανοδική ή καθοδική)

Ο δείκτης Συσσώρευσης-Διανομής είναι ουσιαστικά μια παραλλαγή του πιο δημοφιλούς δείκτη που λέγεται Όγκος Ισορροπίας. Και οι δύο αυτοί δείκτες προσπαθούν να επιβεβαιώσουν αλλαγές της τιμής μιας μετοχής συγκρίνοντας τον όγκο συναλλαγών με τον οποίο έγιναν αυτές οι αλλαγές.

Όταν ο δείκτης Συσσώρευσης-Διανομής κινείται προς τα πάνω, δείχνει ότι η μετοχή συσσωρεύεται (αγοράζεται), καθώς ο περισσότερος όγκος συναλλαγών συνοδεύεται από ανοδική πορεία της μετοχής. Όταν ο δείκτης κινείται προς τα κάτω, δείχνει ότι η μετοχή διανέμεται (πωλείται), καθώς ο περισσότερος όγκος συναλλαγών συνοδεύεται από καθοδική πορεία της μετοχής.

Όταν υπάρχει απόκλιση ανάμεσα στο δείκτη Συσσώρευσης-Διανομής και την τιμή της μετοχής, αυτό μπορεί να προμηνύει άμεση αλλαγή (αντιστροφή). Με τον όρο απόκλιση εννοούμε ότι ο δείκτης ανεβαίνει και η τιμή της μετοχής πέφτει ή ότι ο δείκτης πέφτει και η μετοχή ανεβαίνει.

Η αντιστροφή που επίκειται μπορεί να σημαίνει ότι αν μεν η μετοχή πέφτει και ο δείκτης ανεβαίνει, η μετοχή θα σταματήσει να πέφτει και θα αρχίσει την πορεία προς τα πάνω, ενώ αν η μετοχή ανεβαίνει και ο δείκτης πέφτει, σύντομα η μετοχή θα πάρει την κατηφόρα.

Ταλαντωτής Chaikin

Ο Marc Chaikin βελτιώνοντας μια προθύστερη ιδέα του απλού αλλά μη λειτουργικού δείκτη συσσώρευσης / διανομής όγκου, επινόησε τον ταλαντωτή συσσώρευσης / διανομής όγκου (accumulation / distribution volume oscillator) στις αρχές της δεκαετίας του '70. Ο ταλαντωτής αυτός που είναι γνωστός ως ταλαντωτής του Chaikin A/D θεωρεί πως συσσωρεύεται όγκος σε μια μετοχή όταν η τιμή της κλείσει πάνω από τη μέση ενδοσυνεδριακή τιμή ημέρας. Αυτό μπορεί να συμβεί σε ανοδικές, σταθερές ή και πτωτικές ημέρες. Αντίστοιχα όταν η τιμή κλείσει κάτω από τη μέση

ενδοσυνεδριακή τιμή ημέρας, τότε ο όγκος διανέμεται (δηλαδή τα κεφάλαια των επενδυτών απομακρύνονται από τη μετοχή).

Όταν η τιμή του ταλαντωτή Chaikin αυξάνεται τότε δείχνει συσσώρευση κεφαλαίων στη μετοχή, ενώ όταν μειώνεται δείχνει διανομή όγκου από τη μετοχή.

Επομένως ο ταλαντωτής Chaikin A/D μετρά ουσιαστικά το ρυθμό συσσώρευσης ή διανομής όγκου από μια μετοχή. Ο ταλαντωτής έχει επίπεδο ισορροπίας το μηδέν και θεωρείται υπερ-αγορασμένος όταν ανεβαίνει πάνω από το επίπεδο αυτό, ενώ υπερ-πουλημένος όταν βρίσκεται χαμηλότερα από το μηδέν.

Ως προς την ερμηνεία του υπάρχουν δυο τρόποι:

- Ένας από τους τρόπους που μπορεί να χρησιμοποιήσει κανείς τον Ταλαντωτή Chaikin, είναι να ψάξει για αποκλίσεις ανάμεσα στην τιμή της μετοχής και την τιμή του Ταλαντωτή Chaikin.
- Αλλαγές στην κατεύθυνση του Ταλαντωτή Chaikin. Αν μια μετοχή βρίσκεται σε ανοδική (καθοδική) πορεία και ο Ταλαντωτής Chaikin περνάει από αρνητικές (θετικές) τιμές σε θετικές (αρνητικές), αυτό είναι ένα καλό σημείο για να αγοράσει (πουλήσει) κανείς μετοχές.

Ο ίδιος ο Chaikin προτείνει να χρησιμοποιείται ο ταλαντωτής του σε συνδυασμό με έναν κινητό μέσο όρο 21 περιόδων της τιμής της μετοχής. Αν υπάρχει θετική (αρνητική) αλλαγή στην κατεύθυνση του ταλαντωτή (όπως περιγράφεται παραπάνω), πρέπει η μετοχή να βρίσκεται πάνω (κάτω) από τον κινητό μέσο όρο της, για να επιβεβαιώνεται η απόφαση μας να αγοράσουμε (πουλήσουμε, αντίστοιχα).

Ρυθμός μεταβολής τιμής (Price rate of change)

Ο δείκτης Ρυθμού Μεταβολής Τιμής δείχνει τη διαφορά ανάμεσα στην πιο πρόσφατη τιμή της μετοχής και την τιμή της x-περιόδου πριν. Η διαφορά αυτή μπορεί να εκφραστεί σε μονάδες ή ποσοστά. Ο δείκτης Ορμής δείχνει ακριβώς την ίδια διαφορά εκφρασμένη ως αναλογία.

Είναι γενικά αποδεκτό ότι οι τιμές των μετοχών ακολουθούν περιοδικές-κυματοειδείς κινήσεις. Οι περιοδικές αυτές αυξομειώσεις είναι το αποτέλεσμα των μεταβαλλόμενων προσδοκιών εκείνων που επενδύουν. Ο Ρυθμός Μεταβολής Τιμής δείχνει αυτή την κυματοειδή κίνηση μετρώντας το ποσό μεταβολής της τιμής σε μια καθορισμένη χρονική περίοδο. Όταν οι τιμές ανεβαίνουν, ανεβαίνει και ο Ρυθμός Μεταβολής Τιμής, όταν οι τιμές πέφτουν, πέφτει και ο δείκτης. Όσο πιο μεγάλες είναι

οι αλλαγές στην τιμή της μετοχής τόσο μεγαλύτερες είναι και οι αλλαγές του Ρυθμού Μεταβολής Τιμής.

Η περίοδος που χρησιμοποιείται για τον υπολογισμό του Ρυθμού Μεταβολής Τιμής μπορεί να ξεκινάει από 1 μέρα μέχρι 200, αλλά μπορεί να χρησιμοποιηθεί και μεγαλύτερη περίοδος. Οι πιο δημοφιλείς περίοδοι που χρησιμοποιούνται είναι αυτοί των 12 και 25 ημερών.

Ωστόσο, όπως ακριβώς συμβαίνει με όλους τους άλλους δείκτες υπερ-αγορασμένων ή υπερ-πωλημένων μετοχών, είναι σωστό να περιμένει κανείς τη διόρθωση (αντιστροφή) της μετοχής (προς τα κάτω ή προς τα πάνω) πριν βιαστεί να πουλήσει ή να αγοράσει. Μια υπερ-αγορασμένη μετοχή μπορεί να παραμείνει υπερ-αγορασμένη για μεγάλο χρονικό διάστημα. Μερικές φορές μια σημαντικά υπερ-αγορασμένη (ή υπερ-πουλημένη) μετοχή δηλώνει ότι η τωρινή τάση (κατεύθυνση) θα συνεχιστεί.

Κεφάλαιο 2ο

Μηχανική Μάθηση

Μια από τις περιοχές της Τεχνητής Νοημοσύνης που ενέπνευσε ζωηρό ενδιαφέρον στους επιστήμονες του κλάδου - και όχι μόνο - απ' αρχής της θεμελίωσης της τη δεκαετία του 1980 ήταν εκείνη της Μηχανικής Μάθησης (Machine Learning). Αντικείμενο της αποτελεί η κατασκευή προγραμμάτων, ικανών να προσαρμόζονται τη λειτουργία τους με σκοπό τη βελτίωση της απόδοσή τους, μέσω της εμπειρίας που αποκτούν είτε κατά την δημιουργία τους είτε κατά την εκτέλεσή τους. Δεν θα μπορούσε βέβαια να διατυπωθεί ο ισχυρισμός ότι η επιστήμη έχει φθάσει σε σημείο να ερμηνεύσει τη διαδικασία μάθησης του ανθρώπου, πόσο μάλλον να την αποτυπώσει σε πρόγραμμα εκτελέσιμο από υπολογιστή. Αυτό που έχει επιτευχθεί όμως είναι η δημιουργία αλγορίθμων οι οποίοι μπορούν να αυτοματοποιήσουν την κατασκευή ευφών συστημάτων χρησιμοποιώντας δεδομένα εκπαίδευσης. Το γεγονός αυτό μαρτυρεί μια πληθώρα επιστημονικών αλλά και εμπορικών εφαρμογών που έχουν αναπτυχθεί τα τελευταία χρόνια, επωφελούμενοι από την χρήση των παραπάνω αλγορίθμων.

Σκοπός της ενότητας αυτής είναι μια σύντομη παρουσίαση των θεμελιωδών εννοιών της Μηχανικής Μάθησης, των οικογενειών στις οποίες κατατάσσονται οι πιο γνωστές μέθοδοι Μηχανικής Μάθησης, αλλά και των αλγορίθμων που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας αυτής

2.1 Βασικές Έννοιες της Μηχανικής Μάθησης

Επιχειρώντας τη διατύπωση ενός πιο τυπικού ορισμού από αυτού που προηγήθηκε, θα μπορούσαμε να πούμε πως:

Ένα πρόγραμμα μαθαίνει από την εμπειρία E που αποκτά κατά την εκτέλεση ενός συνόλου διεργασιών Δ , εφόσον η απόδοσή του A βελτιώνεται με την αξιοποίηση της εμπειρίας E . [Mitchell 1997].

Από τον παραπάνω ορισμό καθίσταται σαφές πως για να ορίσουμε τυπικά ένα πρόβλημα μάθησης, θα πρέπει να προσδιορίσουμε: το σύνολο των διεργασιών που θα πρέπει να εκτελεστούν, το μέτρο της απόδοσης που επιδιώκουμε να αυξήσουμε, καθώς και το είδος της εμπειρίας που απαιτείται για την αύξηση αυτή. Για να γίνει ο ορισμός αυτός πιο κατανοητός, παραθέτουμε ως παράδειγμα τη μοντελοποίηση του προβλήματος της πρόβλεψης κίνησης των μετοχών, με το οποίο θα ασχοληθούμε εκτενώς στη συνέχεια:

- Επιτελούμενη Διεργασία Δ: Πρόβλεψη κίνησης μετοχών (Άνοδος – Πτώση) βασισμένη σε δείκτες Τεχνικής Ανάλυσης.
- Μέτρο Απόδοσης A: Το ποσοστό των σωστών προβλέψεων.
- Εμπειρία E: Ένα σύνολο δεικτών Τεχνικής Ανάλυσης που έχει ήδη ταξινομηθεί.

2.2 Μοντελοποίηση Προβλήματος

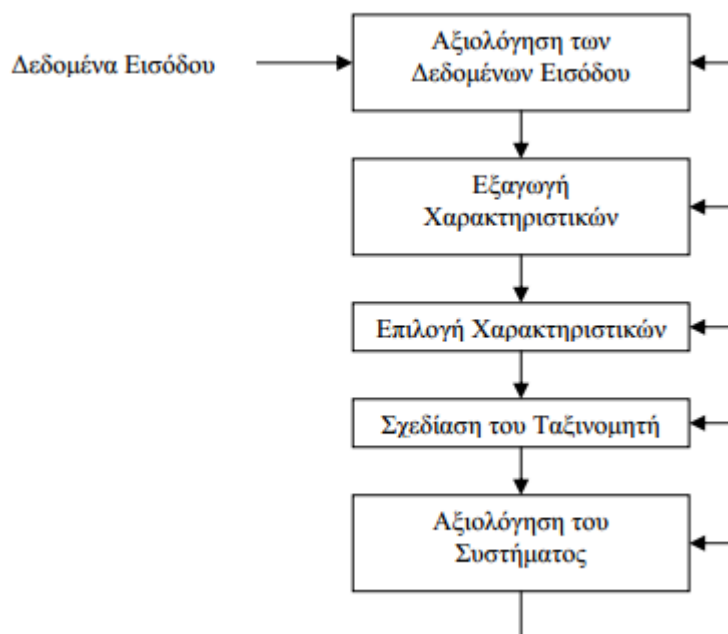
Στην παρούσα παράγραφο θα επιχειρήσουμε να καταστήσουμε κατανοητές κάποιες βασικές έννοιες της Μηχανικής Μάθησης, ενώ παράλληλα θα παρουσιάσουμε κάποια απαραίτητα βήματα για την υλοποίηση ενός Συστήματος M.M.

2.2.1 Αναπαράσταση Δεδομένων

Το πρώτο (και ίσως το σημαντικότερο) βήμα στην σχεδίαση ενός συστήματος M.M. συνίσταται στον προσδιορισμό της «γνώσης» που θα παραχωρηθεί στο σύστημα κατά την εκπαίδευσή του. Θα πρέπει να σημειωθεί πως μία πολύ διαδεδομένη έκφραση που χρησιμοποιείται σε παρόμοια συστήματα είναι «garbage in, garbage out (GIGO)», η οποία ερμηνευμένη πολύ απλά σημαίνει πως εάν εκπαιδεύσεις το σύστημα με «σκουπίδια», θα σου επιστρέψει «σκουπίδια». Η «γνώση», όπως την αναφέραμε παραπάνω δεν είναι τίποτα άλλο από ένα σύνολο παραδειγμάτων. Μια από τις πλέον συνήθεις μορφές αναπαράστασης των παραδειγμάτων είναι η διανυσματική: κάθε παράδειγμα εκπαίδευσης (sample), ή στιγμιότυπο (instance) όπως έχει επικρατήσει να αποκαλείται, αποτελείται από ένα σύνολο χαρακτηριστικών (attributes ή features), μετρήσιμων δηλαδή ποσοτήτων που έχουν επιλεγεί με τέτοιο τρόπο, ώστε το σύστημα M.M. να μπορέσει να εκμαιεύσει από αυτά τη γνώση που χρειάζεται. Ανάλογα με το είδος της πληροφορίας των διαφόρων παραδειγμάτων

εκπαίδευσης, μπορεί να είναι συνεχείς (continuous) αριθμοί, ή ονομαστικοί (nominal), παρουσιάζόμενοι ως ένα προκαθορισμένο σύνολο διακριτών τιμών, αριθμητικών ή συμβολικών, όπως στην περίπτωση της εφαρμογής που πραγματεύεται η παρούσα εργασία.

Επόμενο στάδιο αποτελεί ο τρόπος με τον οποίο θα διαχειριστεί το «έφυρες» σύστημα τη γνώση που αποκοιμίζει κατά την εκπαίδευση, με σκοπό την αποδοτικότερη λειτουργία του. Υιοθετώντας το μοντέλο της διανυσματικής αναπαράστασης, αναζητούμε ουσιαστικά μια συνάρτηση f' η οποία προσεγγίζει όσο το δυνατόν περισσότερο μια ιδανική συνάρτηση με την οποία δύναται να μοντελοποιηθεί το πρόβλημα, τη συνάρτηση στόχου f . Οι δύο συναρτήσεις έχουν ελεύθερη μεταβλητή ένα τυχαίο διάνυσμα x , πεδίο ορισμού το χώρο των στιγμιότυπων και σύνολο τιμών το οποίο καθορίζεται από την εκάστοτε εφαρμογή. Έτσι, η επίλυση ενός προβλήματος M.M. ανάγεται στην ελαχιστοποίηση της συνάρτησης κόστους, όπου συνάρτηση κόστους συνήθως ορίζεται η απόκλιση της f' από την f . Συνοπτικά, τα βασικά στάδια της σχεδίασης ενός συστήματος M.M. φαίνονται στο Εικόνα 2. Για λόγους πληρότητας, απεικονίζεται και το στάδιο της Επιλογής Χαρακτηριστικών (Attribute Selection), το οποίο θα παρουσιαστεί αναλυτικά στο Κεφάλαιο 2.3



Εικόνα 2 Δομή – Αρχιτεκτονική ενός Συστήματος Μηχανικής Μάθησης

2.2.2 Κατηγοριοποίηση Μοντέλων Μ.Μ.

Ανάλογα με το είδος της γνώσης που παρέχεται για εκπαίδευση, οι μέθοδοι Μ.Μ. κατηγοριοποιούνται σε δύο μεγάλες κατηγορίες:

- στη μάθηση με επίβλεψη (supervised learning),
- και τη μάθηση χωρίς επίβλεψη (unsupervised learning).

Στην πρώτη περίπτωση, το σύστημα κατά τη διάρκεια μάθησης «καθοδηγείται» από τα παραδείγματα εκπαίδευσης υπό την έννοια ότι υποδεικνύουν στον αλγόριθμο την τιμή της συνάρτησης στόχου. Στην κατηγορία αυτή ανήκουν τα προβλήματα ταξινόμησης (classification), στα οποία το αποτέλεσμα της κατηγοριοποίησης αποτελεί χαρακτηριστικό των στιγμιότυπων εκπαίδευσης και η τιμή του για κάθε στιγμιότυπο συμπεριλαμβάνεται στα δεδομένα εκπαίδευσης. Στην κατηγορία προβλημάτων ταξινόμησης εντάσσεται και το πρόβλημα που πραγματεύεται η παρούσα εργασία. Αντίθετα, στην περίπτωση της μάθησης χωρίς επίβλεψη, ο αλγόριθμος καλείται να ανακαλύψει σχηματισμούς και σχέσεις μεταξύ των στιγμιότυπων εκπαίδευσης, διαχωρίζοντας τα χωρίς κάποια γνωστά κριτήρια. Παραδείγματα προβλημάτων που εντοπίζονται σε αυτήν την περιοχή είναι: η ομαδοποίηση οντοτήτων (clustering), όπου επιχειρείται ομαδοποίηση των στιγμιότυπων σε κλάσεις τις οποίες θα συμπεράνει ο αλγόριθμος, στηριζόμενος στην ομοιότητα των μελών κάθε ομάδας.

2.2.3 Αξιολόγηση Απόδοσης και Overfitting

Ολοκληρώνοντας τη σύντομη αυτή αναφορά στις θεμελιώδεις έννοιες της Μ.Μ., κρίνεται σκόπιμη η παρουσίαση ενός συνόλου παραγόντων καθοριστικής σημασίας για τη σχεδίαση ενός αποδοτικού συστήματος Μ.Μ. Από τους πλέον σημαντικούς θεωρείται η ικανότητα γενίκευσης (generalization ability), η ικανότητά του δηλαδή να χρησιμοποιεί την «γνώση» που απέκτησε μετά την τροφοδότηση του με τα στιγμιότυπα εκπαίδευσης με αποτελεσματικότητα επί του συνόλου των άγνωστων στιγμιότυπων. Συνήθως, ισχύει ότι, εφόσον η υπόθεση στην οποία κατέληξε το σύστημα μετά την εκπαίδευσή του ισχύει και ερμηνεύει την πλειοψηφία των στιγμιότυπων εκπαίδευσης, τότε θα είναι σε μεγάλο βαθμό συνεπής και με τα άγνωστα στιγμιότυπα που θα κληθεί να διαχειρισθεί κατά την κανονική λειτουργία του. Θα πρέπει να δοθεί ιδιαίτερη προσοχή ωστόσο στο να μην παρερμηνευθεί η

παραπάνω πρόταση, καθώς η περίπτωση μιας υπόθεσης η οποία είναι απόλυτα συνεπής με την πλειοψηφία των στιγμιότυπων εκπαίδευσης ενέχει τον κίνδυνο της εκμάθησης των πλέον ασήμαντων λεπτομερειών που είναι δυνατόν να παρατηρηθούν σ' αυτά, με αποτέλεσμα η απόδοση του συστήματος να παρουσιάζεται ιδιαίτερα χαμηλή στα μη παρατηρημένα στιγμιότυπα. Το παραπάνω φαινόμενο είναι γνωστό ως φαινόμενο του υπερταιριάσματος (overfitting). Επιχειρώντας έναν πιο τυπικό ορισμό του φαινομένου, θα λέμε πως: Μια υπόθεση h υπερταιριάζει (overfits) των δεδομένων εκπαίδευσης όταν υπάρχει μια διαφορετική υπόθεση h' τέτοια ώστε, η h να παρουσιάζει μικρότερο σφάλμα από την h' επί των δεδομένων εκπαίδευσης, ενώ ταυτόχρονα η h' να παρουσιάζει μικρότερο σφάλμα από την h επί ολόκληρης της κατανομής των δεδομένων (εκπαίδευσης και ελέγχου). Καθοριστική σημασία στην ποιότητα του σώματος εκπαίδευσης φέρει η ορθή επιλογή των χαρακτηριστικών τα οποία θα συστήσουν το χώρο του προβλήματος, καθώς μέσω αυτών καθίσταται δυνατή η αναπαράσταση της γνώσης που απαιτείται για την αποδοτική λειτουργία του συστήματος. Επίσης, απαραίτητη θεωρείται, όπως ήδη αναφέραμε, η ομοιότητα της κατανομής των στιγμιότυπων εκπαίδευσης με την κατανομή των στιγμιότυπων που θα συναντήσει το σύστημα κατά τη διάρκεια της λειτουργίας του. Ο παράγοντας αυτός ωστόσο σχετίζεται άμεσα με το μέγεθος του σώματος εκπαίδευσης, το οποίο αποτελεί ένα περιορισμένο δείγμα του χώρου του προβλήματος. Ιδιαίτερη προσοχή θα πρέπει επίσης να δοθεί στην ορθότητα των δεδομένων που χρησιμοποιούνται για την εκπαίδευση, καθώς δεν είναι σπάνια η περίπτωση παρείσφρησης σφαλμάτων στις τιμές των χαρακτηριστικών κάποιων στιγμιότυπων. Το φαινόμενο αυτό ονομάζεται θόρυβος (noise) και συναντάται κατά τη συλλογή δεδομένων από πειραματικές μετρήσεις, αλλά και γενικότερα, όπου επεμβαίνει ο ανθρώπινος παράγοντας στη δημιουργία δεδομένων εκπαίδευσης. Όπως είναι αναμενόμενο, η εκτεταμένη παρουσία θορύβου οδηγεί στον αποπροσανατολισμό του αλγορίθμου μάθησης και στην επιλογή μιας υποβέλτιστης υπόθεσης, ελάχιστα συνεπούς με τα δεδομένα ελέγχου. Αυτό ωστόσο δε συμβαίνει και στην περίπτωση που παρουσιάζεται θόρυβος της ίδιας μορφής και στα δεδομένα αξιολόγησης, οπότε χρειάζεται να μοντελοποιηθεί και αυτός, καθώς ανάγεται πλέον σε «χαρακτηριστικό» του συγκεκριμένου προβλήματος μάθησης. Παρόμοιο φαινόμενο είναι και αυτό της απουσίας τιμών (missing values), περιπτώσεων δηλαδή όπου οι τιμές ορισμένων χαρακτηριστικών δεν αναφέρονται για κάποια από τα στιγμιότυπα. Όσον αφορά τέλος στους

παράγοντες εκείνους που υποβοηθούν το μηχανισμό που υιοθετείται για την προσέγγιση της συνάρτησης στόχου, συνοψίζονται στην Αρχή της Ελάχιστης Περιγραφής (Minimum Description Length Principle ή MDL), σύμφωνα με την οποία: Η καλύτερη θεωρία που είναι δυνατόν να εξαχθεί από ένα σώμα δεδομένων είναι η μικρότερη δυνατή, η οποία απαιτεί συνάμα τη μικρότερη ποσότητα πληροφορίας που μοντελοποιεί τις εξαιρέσεις που είναι σχετικές με τη θεωρία αυτή. Η παραπάνω αρχή, γνωστή και ως «το ξυράφι του Occam» (Occam's Razor), υποστηρίζει ουσιαστικά πως οι απλούστερες θεωρίες είναι προτιμότερες από τις πιο περίπλοκες. Στην περίπτωση της M.M., είναι άμεσα συνυφασμένη με την ικανότητα της γενίκευσης που πρέπει να διακρίνει μια εξαγόμενη υπόθεση, καθώς έχει παρατηρηθεί ότι προσεγγίσεις που είναι επιρρεπείς στον κίνδυνο του υπερταϊριάσματος τείνουν να παράγουν ιδιαίτερα πολύπλοκες υποθέσεις, υποβαθμίζοντας έτσι την απόδοση του συστήματος.

2.2.4 Κλιμάκωση Δεδομένων

Η κλιμάκωση των χαρακτηριστικών (features) των δεδομένων εκπαίδευσης είναι μια μέθοδος που χρησιμοποιείται για την τυποποίηση της κλίμακας των ανεξάρτητων μεταβλητών ή τα χαρακτηριστικών των δεδομένων. Πραγματοποιείται γενικά κατά το στάδιο της προ-επεξεργασίας των δεδομένων, ένα στάδιο που προηγείται χρονικά της εκπαίδευσης του συστήματος.

Δεδομένου ότι το εύρος των τιμών των χαρακτηριστικών των δεδομένων ποικίλει σε μεγάλο βαθμό, αντιλαμβανόμαστε πως σε ορισμένους αλγόριθμους Μηχανικής Μάθησης, οι αντικειμενικές συναρτήσεις είναι μοντελοποιημένες κατά τέτοιο τρόπο ώστε να δίνουν μεγαλύτερο βάρος στα χαρακτηριστικά οι τιμές των οποίων είναι μεγαλύτερες κατά απόλυτη τιμή. Ως εκ τούτου, το σύνολο όλων των χαρακτηριστικών θα πρέπει να ομαλοποιείται, έτσι ώστε κάθε χαρακτηριστικό συνεισφέρει ομοιόμορφα στην συνάρτηση κόστους. Οι μέθοδοι που ακολουθούνται στη διαδικασία της κλιμάκωσης των χαρακτηριστικών είναι δύο:

Επανακλιμάκωση - Rescaling

Η απλούστερη μέθοδος είναι η μεταφορά του εύρους τιμών των χαρακτηριστικών να σε ένα κοινό σύνολο π.χ. $[0, 1]$ ή $[-1, 1]$. Επιλέγοντας το εύρος-στόχο εξαρτάται από τη φύση των δεδομένων. Ο γενικός τύπος δίνεται ως εξής:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

όπου το x είναι μία αρχική τιμή ενός χαρακτηριστικού, και το x' είναι η κοινωνικοποιημένη τιμή.

Κανονικοποίηση - Standardization

Στη Μηχανική Μάθηση, μπορούμε να χειριστούμε διαφόρων τύπων δεδομένα, π.χ. ακουστικά σήματα, τις τιμές των εικονοστοιχείων για τα δεδομένα εικόνας και τα δεδομένα αυτά μπορεί να περιλαμβάνουν πολλές διαστάσεις. Η μέθοδος αυτή έχει ως σκοπό το σύνολο τιμών του κάθε χαρακτηριστικού των δεδομένων να έχει μηδενική μέση τιμή και η διακύμανση του να ισούται με μονάδα. Η γενική μέθοδος υπολογισμού είναι αρχικά να προσδιοριστεί η μέση τιμή και η τυπική απόκλιση της εκάστοτε κατανομής για κάθε χαρακτηριστικό. Στη συνέχεια αφαιρούμε τη μέση τιμή από κάθε χαρακτηριστικό και τέλος διαιρούμε τις προκύπτουσες τιμές με την τυπική απόκλιση του αναφερόμενου χαρακτηριστικού.

2.3 Επιλογή Αλγορίθμου M.M.

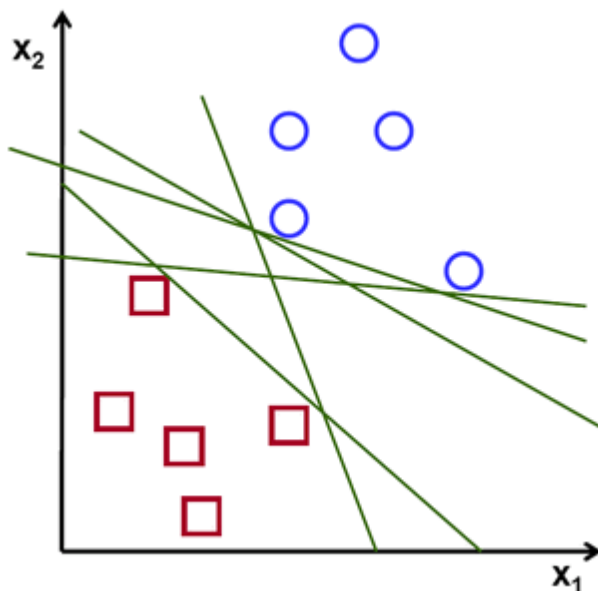
Όπως παρουσιάζεται σε [13], μετά από σύγκριση διαφόρων αλγορίθμων M.M. (όπως SVM, νευρωνικά δίκτυα, κ.α.) στο πεδίο της πρόβλεψης χρηματοοικονομικών στοιχείων το καλύτερο αποτέλεσμα σε παρόμοια εργασία προκύπτει κάνοντας χρήση μετά- Αλγόριθμου Adaptive Boost μετά την εφαρμογή του C-SVC Αλγόριθμου. Όπως αναφέραμε και σε προηγούμενο κεφάλαιο η σύγκριση έγκειται σε κάποιο μέτρο απόδοσης το οποίο συνήθως αναφέρεται ως ακρίβεια και ορίζεται ως η ποσοστιαία αναλογία σωστών προβλέψεων της υπόθεσης μας σε κάποιο σύνολο ελέγχου δεδομένων. Τα αποτελέσματα στη συγκεκριμένη μελέτη παρουσιάζουν ακρίβεια της τάξεως 64.32% +/- 3,99%.

Επίσης στη μελέτη [9] στην οποία συγκρίνονται αλγόριθμοι M.M. όπως RNN, SVM και APMA, αποδεικνύεται ότι ο αλγόριθμος SVM παρουσιάζει τον μεγαλύτερο βαθμό απόδοσης. Επίσης η μεθοδολογία SVM βρέθηκε ότι απαιτεί πολύ λιγότερα δεδομένα εκπαίδευσης από το δίκτυο Elman.

Λόγω αυτών των ικανοποιητικών αποτελεσμάτων SVM και SVR χρησιμοποιούνται ως παράγοντες πρόβλεψης χρονοσειρών σε πολλές περιπτώσεις. Βάση των ανωτέρω επιλέξαμε τελικά και εμείς την υλοποίηση με χρήση Μετά – Αλγόριθμου Adaptive Boost μετά την εφαρμογή του C-SVC Αλγόριθμου.

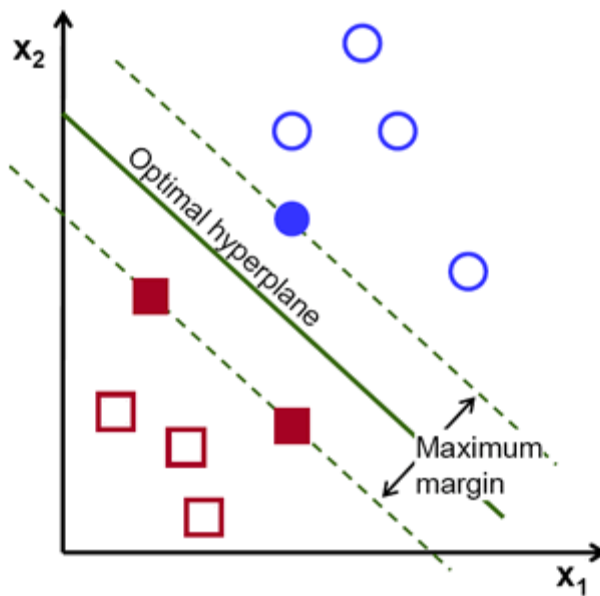
2.3.1 Μηχανές Διανυσμάτων Υποστήριξης - Support Vector Machines (SVM)

Έστω ότι πρέπει να διαχωρίσουμε δύο σύνολα στοιχείων όπως αυτά που φαίνονται στην Εικόνα 3.



Εικόνα 3 Διάφοροι γραμμικοί Classifiers

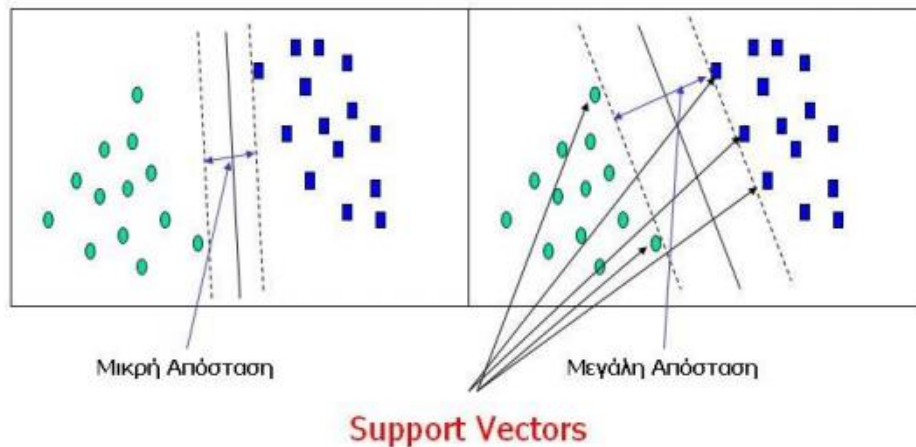
Στην παραπάνω εικόνα μπορείτε να δείτε ότι υπάρχουν πολλαπλοί ταξινομητές (classifiers) που προσφέρουν μια λύση στο πρόβλημα. Εάν ένας classifier περνά πολύ κοντά στα στοιχεία, σημαίνει ότι είναι ευαίσθητος από άποψη θορύβου και δεν μπορεί να γενικευτεί. Ως εκ τούτου, ο στόχος μας πρέπει να είναι να βρεθεί τη γραμμή που περνά όσο το δυνατόν πιο μακριά από όλα τα σημεία, όπως παρατηρείται στη Εικόνα 4.



Εικόνα 4 Classifier με μέγιστη απόσταση

Οι αλγόριθμοι Support Vector Machines (SVMs) [10], είναι ένα σύνολο μεθόδων εκμάθησης που χρησιμοποιούνται για προβλήματα ταξινόμησης. Η ιδέα των SVM είναι να δημιουργηθεί ένα υπερεπίπεδο, τέτοιο ώστε η απόσταση διαχωρισμού μεταξύ των παραδειγμάτων να μεγιστοποιείται. Τα διανύσματα των πιο κοντινών στοιχείων στο υπερεπίπεδο αυτό είναι τα υποστηρικτικά διανύσματα (support vectors) όπως αυτά φαίνονται στην Εικόνα 5. Είναι φανερό ότι η γραμμή στο δεξί σχήμα διαχωρίζει πολύ καλύτερα τα στοιχεία από αυτήν στο αριστερό σχήμα

Το πλεονέκτημα της μεθόδου είναι ότι επιτυγχάνονται καλές επιδόσεις στα προβλήματα ταξινόμησης χωρίς να ενσωματώνεται γνώση από τον τομέα του προβλήματος, ενώ παράλληλα η ικανότητά τους να μαθαίνουν είναι ανεξάρτητη από τις διαστάσεις του χώρου χαρακτηριστικών. Τα SVM μετράνε την πολυπλοκότητα των υποθέσεων με βάση την απόσταση που μπορούν να διαχωρίσουν τα στοιχεία, και όχι με βάση τον αριθμό των χαρακτηριστικών.



Εικόνα 5 Support Vectors

Τα συστήματα ταξινόμησης που βασίζονται στον αλγόριθμο αυτό αποτελούν σήμερα μια από τις δημοφιλέστερες προσεγγίσεις στο χώρο της χρηματιστηριακής πρόβλεψης μετοχών, λόγω της ευρωστίας, της αποτελεσματικότητας και της ταχύτητας που επιδεικνύουν, αλλά και της ικανότητάς τους να παράγουν μη γραμμικές επιφάνειες απόφασης, καθιστώντας έτσι υπολογιστικά εφικτή την επίλυση ενός μεγάλου αριθμού πρακτικών προβλημάτων μάθησης που δεν μπορούν να αντιμετωπιστούν από γραμμικά μοντέλα.

Για την πληρέστερη και ταυτόχρονα πιο κατανοητή παρουσίαση, θα εξετάσουμε συνοπτικά τη μεθοδολογία των SVM, εφαρμοζόμενη σε γραμμικώς διαχωρίσιμα δεδομένα και έπειτα θα κάνουμε μία σύντομη αναφορά στις διαφορές όπως αυτή εφαρμόζεται σε μη-γραμμικώς διαχωρίσιμα προβλήματα.

Τα SVMs μετασχηματίζουν τον χώρο εισόδου τους (input space) σε έναν χώρο χαρακτηριστικών (feature space) υψηλότερης διάστασης, στον οποίο κατασκευάζεται ένας γραμμικός ταξινομητής (linear classifier). Ο γραμμικός ταξινομητής βασίζεται σε μία συνάρτηση απόφασης (decision function) της μορφής:

$$f_{\vec{w},b}(\vec{x}) = \vec{w} \cdot \vec{x} + b$$

Ας θεωρήσουμε αρχικά ένα δοσμένο σύνολο παραδειγμάτων (samples) δεδομένων ή όπως καλείται data set.

$$S = \{(x_1, y_1), \dots, (x_l, y_l)\}$$

Οι μεταβλητές y_1, \dots, y_l που βρίσκονται σε κάθε sample καλούνται ετικέτες και παίρνουν τις τιμές 1 ή -1 ανάλογα με το αν το δεδομένο βρίσκεται στην πρώτη ή στη δεύτερη κατηγορία. Στη συνέχεια ορίζουμε τη συνάρτηση $g(x)$ όπου:

$$g(x) = \langle w, \phi(x) \rangle + b$$

Η συνάρτηση αυτή όπως βλέπουμε καθορίζεται από ένα διάνυσμα w που περιέχει τα βάρη και μια σταθερά b . Στη συνέχεια ορίζουμε μια συνάρτηση $m(S, g)$ η οποία ονομάζεται συνάρτηση περιθωρίου και δίνεται από τον παρακάτω τύπο:

$$m(S, g) = \min_{|s|=l} y_i g(x_i) \geq \gamma$$

Η συνάρτηση $m(S, g)$ δείχνει ότι οι δύο κατηγορίες δεδομένων μπορούν να διαχωριστούν από ένα υπερ-επίπεδο με ένα περιθώριο γ η καθε μία από αυτό. Αυτό ακριβώς το set δεδομένων καλείται γραμμικώς διαχωριζόμενο με περιθώριο γ . Όπως αναπαρίσταται στην Εικόνα 4. Το υπερ-επίπεδο συμβολίζεται στην Εικόνα 4 με την μεσαία από τις τρεις ευθείες γραμμές, ενώ οι άλλες δύο απέχουν απόσταση γ από αυτή.

Η σταθερότητα της μεθόδου, όπως είναι και διαισθητικά κατανοητό, είναι αυξανόμενη όσο μεγαλύτερο είναι το περιθώριο γ . Όμως, κάτι τέτοιο δεν μας εγγυάται απαραίτητα και την ευρωστία του (robustness). Έτσι σκοπός μας είναι να βρούμε μια γραμμική συνάρτηση η οποία θα μεγιστοποιεί το περιθώριο γ .

Το πρόβλημα αυτό της μεγιστοποίησης του γ το αντιμετωπίζουμε μαθηματικά όπως ακριβώς και προηγουμένως χρησιμοποιώντας τους τελεστές Lagrange. Έτσι προκύπτει ο παρακάτω αλγόριθμος που είναι γνωστός ως Hard margin SVM.

Είσοδος	training set $S = \{(x_1, y_1), \dots, (x_l, y_l)\}$, $\delta > 0$
Διαδικασία βελτιστοποίησης Με περιορισμούς	Βρείτε τη λύση a^* του προβλήματος βελτιστοποίησης : $W(a) = - \sum_{i,j=1}^l a_i a_j y_i y_j \kappa(x_i, x_j)$ $\sum_{i=1}^l y_i a_i = 0, \sum_{i=1}^l a_i = 1$ and $0 \leq a_i, i=1, \dots, l$
4	$\gamma^* = \sqrt{-W(a^*)}$
5	Επιλέξτε i έτσι ώστε $0 < a_i^*$
6	$b = y_i (\gamma^*)^2 - \sum_{j=1}^l a_j^* y_j \kappa(x_i, x_j)$
7	$f(.) = \text{sgn} \left(\sum_{j=1}^l a_j^* y_j \kappa(x_j, \cdot) + b \right)$;
8	$w = \sum_{j=1}^l y_j a_j^* \phi(x_j)$
Έξοδος	Διάνυσμα βαρών w, βέλτιστη λύση a^*, περιθώριο γ^* και συνάρτηση f που υλοποιεί την κατηγοριοποίηση με βάση το υπερ-επίπεδο

Ακολουθώντας τα βήματα του αλγορίθμου υπολογίζουμε αρχικά το διάνυσμα \mathbf{a} που μεγιστοποιεί τη συνάρτηση $W(\mathbf{a})$ και άρα και το περιθώριο γ . Οι συνιστώσες του διανύσματος \mathbf{a} είναι μη μηδενικές μόνο για εκείνα τα \mathbf{x}_i από το training set τα οποία απέχουν από το υπερεπίπεδο που διαχωρίζει τις δύο κατηγορίες απόσταση γ . Αυτό αποδεικνύεται από τη σχέση

$$a_i^* [y_i (\langle \mathbf{w}^*, \phi(x_i) \rangle + b^*) - \gamma^*] = 0$$

καθώς για τις μη μηδενικές τιμές των a_i μηδενίζεται το περιεχόμενο της αγκύλης. Άρα

$$y_i (\langle \mathbf{w}^*, \phi(x_i) \rangle + b^*) = \gamma^*$$

Όλες οι υπόλοιπες συνιστώσες του διανύσματος \mathbf{a} είναι μηδέν και δε μας απασχολούν. Τα δεδομένα x_i για τα οποία τα αντίστοιχα a_i είναι μη μηδενικά όπως

και στο προηγούμενο πρόβλημα ονομάζονται support vectors. Αυτά γεωμετρικά βρίσκονται πολύ κοντά στην διαχωριστική ευθεία.

Παρατηρούμε ότι η συνάρτηση $W(a)$ περιέχει τον όρο

$$G = (y_i y_j k(x_i, x_j))_{i,j=1}^l$$

για τον οποίο αποδεικνύεται ότι ισχύει:

$$\beta' G \beta = \sum_{i,j=1}^l \beta_i \beta_j y_i y_j k(x_i, x_j) = \left\langle \sum_{i=1}^l \beta_i y_i \phi(x_i), \sum_{j=1}^l \beta_j y_j \phi(x_j) \right\rangle = \left\| \sum_{i=1}^l \beta_i y_i \phi(x_i) \right\|^2 \geq 0$$

Με βάση τη θεωρία των πινάκων αυτή είναι ικανή συνθήκη ώστε να οδηγηθούμε στο συμπέρασμα ότι ο πίνακας G είναι θετικά ημιορισμένος. Αυτό σημαίνει ότι το πρόβλημα της βελτιστοποίησης είναι κυρτό και άρα θα υπάρχει μία και μόνο μία λύση του, δηλαδή το διάνυσμα \mathbf{a} που υπολογίζεται είναι μοναδικό. Το γεγονός αυτό αποτελεί μια σημαντική διαφορά της μεθόδου των support vector machines έναντι των νευρωνικών δικτύων όπου συναντώνται συχνά τοπικά ελάχιστα.

Ο αλγόριθμος Hard SVM που περιγράψαμε μπορεί να εφαρμοστεί με επιτυχία μόνο όταν τα δεδομένα είναι εύκολα διαχωρίσιμα. Αντιμετωπίζει όμως σοβαρά προβλήματα όταν τα δεδομένα περιέχουν και θόρυβο που είναι και η πιο συνηθισμένη κατάσταση σε πραγματικά προβλήματα. Για το λόγο αυτό αναπτύχθηκε ένας άλλος αλγόριθμος που ονομάζεται Soft margin SVM και ακριβώς επειδή έχει σαφώς καλύτερα αποτελέσματα στα προβλήματα κατηγοριοποίησης από τον Hard είναι και αυτός που χρησιμοποιήσαμε στα πειράματά μας. Ο αλγόριθμος αυτός είναι πιο εύρωστος από τον προηγούμενο και έτσι μπορεί να ανέχεται κάποιο θόρυβο καθώς και μεμονωμένα δεδομένα απομακρυσμένα από το σύνολο χωρίς να αλλάζει σημαντικά το αποτέλεσμα. Αυτό επιτυγχάνεται συνδυάζοντας το περιθώριο γ με το διάνυσμα ξ . Το διάνυσμα αυτό ορίζεται ως

$$\xi_i = \xi((y_i, x_i), \gamma, g) = (\gamma - y_i g(x_i)),$$

και συχνά αναφέρεται ως slack variable καθώς επιφέρει αλλαγές στους περιορισμούς που υπάρχουν στο πρόβλημα βελτιστοποίησης το οποίο τελικά μετατρέπεται στο ακόλουθο:

$$\min_{w,b,\gamma,\xi} -\gamma + C \sum_{i=1}^l \xi_i$$

με τους εξής περιορισμούς:

$$y_i (\langle w, \phi(x_i) \rangle + b) \geq \gamma - \xi_i$$

$$\xi_i \geq 0$$

$$i = 1, \dots, l$$

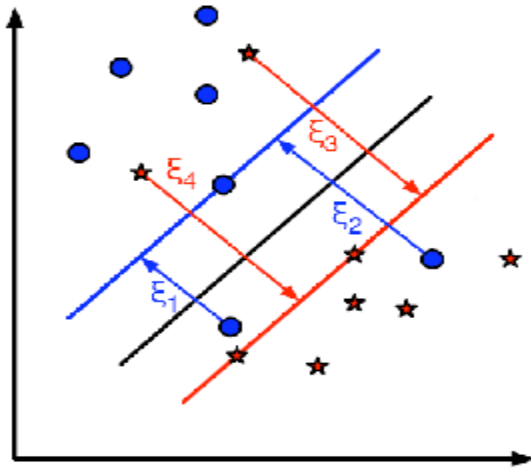
$$\|w\|^2 = 1$$

Η παράμετρος C ελέγχει το κατά πόσο το περιθώριο επηρεάζεται από τη μεταβλητή ξ και αντίστροφα και καθορίζει ουσιαστικά τις τιμές τους. Οι τιμές της είναι άπειρες, υποχρεωτικά όμως μεγαλύτερες από το $1/l$ όπου l ο αριθμός των δειγμάτων προς εκπαίδευση. Ομοίως με πριν, το πρόβλημα της βελτιστοποίησης επιλύεται μαθηματικά με τους τελεστές Lagrange και καταλήγουμε στον παρακάτω αλγόριθμο:

Είσοδος	training set $S = \{(x_1, y_1), \dots, (x_l, y_l)\}$, $\delta > 0$, $C \in [1/l, \infty]$
Διαδικασία βελτιστοποίησης Με περιορισμούς	Βρείτε τη λύση a^* του προβλήματος βελτιστοποίησης: $W(a) = - \sum_{i,j=1}^l a_i a_j y_i y_j \kappa(x_i, x_j)$ $\sum_{i=1}^l y_i a_i = 0, \sum_{i=1}^l a_i = 1 \text{ and } 0 \leq a_i \leq C, i=1, \dots, l$
4	$\lambda^* = \frac{1}{2} (\sum_{i,j=1}^l y_i y_j a_i^* a_j^* \kappa(x_i, x_j))^{1/2}$
5	Επιλέξτε i, j έτσι ώστε $-C < a_i^* y_i < 0 < a_j^* y_j < C$, $i=1, \dots, l$
6	$b^* = -\lambda^* (\sum_{k=1}^l a_k^* y_k \kappa(x_k, x_i) + \sum_{k=1}^l a_k^* y_k \kappa(x_k, x_j))$
7	$\gamma^* = 2\lambda^* \sum_{k=1}^l a_k^* y_k \kappa(x_k, x_j) + b^*$
8	$f(\cdot) = \text{sgn} \left(\sum_{j=1}^l a_j^* y_j \kappa(x_j, \cdot) + b^* \right);$
9	$w = \sum_{j=1}^l y_j a_j^* \phi(x_j)$
Έξοδος	Διάνυσμα βαρών w , βέλτιστη λύση a^* , περιθώριο γ^* και συνάρτηση f που υλοποιεί την κατηγοριοποίηση με βάση το υπερ-επίπεδο

Στο πρώτο μέρος του αλγορίθμου και πάλι υπολογίζουμε το διάνυσμα \mathbf{a} το οποίο μεγιστοποιεί τη συνάρτηση $W(\mathbf{a})$ λαμβάνοντας υπόψη και τους περιορισμούς που υπάρχουν δίνοντας μοναδική λύση. Παρατηρούμε ότι στους περιορισμούς αυτούς περιλαμβάνεται και η παράμετρος C η οποία περιορίζει τις τιμές που μπορούν να πάρουν οι συνιστώσες a_i που θα υπολογιστούν. Στο δεύτερο μέρος του αλγορίθμου και με βάση τις τιμές των a_i που υπολογίστηκαν ορίζονται οι μεταβλητές λ, b και γ . Πιο συγκεκριμένα για τον υπολογισμό των b και γ καλούμαστε να επιλέξουμε κάποια τιμή για τα i και j και κατ'επέκταση για τα x_i και x_j τέτοια ώστε τα αντίστοιχα a_i και a_j πολλαπλασιασμένα με την αντίστοιχη ετικέτα y_i, y_j να είναι μεγαλύτερα από το $-C$ και μικρότερα από το 0 για τα πρώτα και μεγαλύτερα από το μηδέν και μικρότερα από το C για τα δεύτερα

Εν κατακλείδι, θα πούμε λίγα λόγια για την περίπτωση μη-γραμμικών προβλημάτων. Στα προβλήματα αυτά, τα σημεία του συνόλου δεδομένων που αντιστοιχούν σε διαφορετικές ετικέτες διαχωρίζονται με μία μη-γραμμική καμπύλη, όπως φαίνεται στην Εικόνα 6.



Εικόνα 6 Μη-γραμμικά διαχωρίσιμα δεδομένα

Σε αυτή την περίπτωση, χρησιμοποιούνται οι λεγόμενες συναρτήσεις πυρήνα (kernel functions) που συμβολίζονται με Φ και μετασχηματίζουν τα δεδομένα σε ένα χώρο υψηλότερης διάστασης όπου τα δεδομένα μπορούν να διαχωριστούν με υπερ-επίπεδο

Για μία δεδομένη αντιστοίχιση Φ από το χώρο εισόδου σε ένα χώρο υψηλότερης διάστασης, ορίζεται ο πυρήνας δύο αντικειμένων \mathbf{x} και \mathbf{x}' ως το εσωτερικό γινόμενο των προβολών (των δύο αυτών αντικειμένων) στο χώρο υψηλότερης διάστασης. Πιο φορμαλιστικά, ισχύει:

$$K(\vec{x}, \vec{x}') = \vec{\Phi}(\vec{x}) \cdot \vec{\Phi}(\vec{x}')$$

Για την εκτέλεση της διαδικασίας σε μη-γραμμικά προβλήματα, αρκεί ο υπολογισμός του πυρήνα $K(\mathbf{x}, \mathbf{x}')$. Αυτό συμβαίνει γιατί η συνάρτηση απόφασης αποκτάει τη μορφή:

$$f(\vec{x}) = \sum_{i=1}^N a_i^* \cdot K(\vec{x}_i, \vec{x}) + b^*$$

Με τις τιμές a_i i στο $\{1, 2, \dots, N\}$ να λαμβάνουν τέτοιες τιμές που να μεγιστοποιούν τη σχέση:

$$Q(\vec{\alpha}) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \cdot \sum_{i=1}^N \sum_{j=1}^N \alpha_i \cdot \alpha_j \cdot y_i \cdot y_j \cdot K(\vec{x}_i, \vec{x}_j)$$

Με τους εξής περιορισμούς:

$$\sum_{i=1}^N \alpha_i \cdot y_i = 0$$

και

$$0 \leq \alpha_i \leq C \text{ για } i \in \{1, 2, \dots, N\}$$

Πώς θα πρέπει να επιλεγεί η παράμετρος C ; Είναι προφανές ότι η απάντηση στο ερώτημα αυτό εξαρτάται από το πώς κατανέμεται η κατάρτιση των δεδομένων. Αν και δεν υπάρχει γενική απάντηση, είναι χρήσιμο να ληφθούν υπόψη οι κανόνες:

Μεγάλες τιμές του C δίνουν λύσεις με λιγότερα λάθη ταξινόμησης, αλλά σε μικρότερο ποσοστό.

Μικρές τιμές του C δίνουν λύσεις με μεγαλύτερη ευρωστία και περισσότερα λάθη ταξινόμησης.

Οι διάφοροι τύποι πυρήνων (kernel) που χρησιμοποιούνται ανάλογα την εφαρμογή παρουσιάζονται παρακάτω.

- Γραμμικοί (Linear):

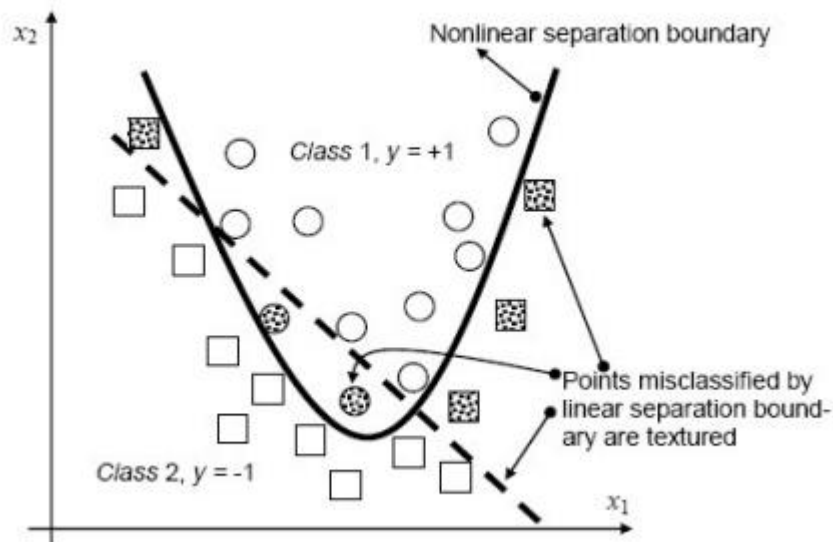
$$K(\vec{x}, \vec{x}') = \vec{x} \cdot \vec{x}' + \gamma,$$

όπου η παράμετρος γ καθορίζεται από τον χρήστη. Αυτός ο τύπος πυρήνα έχει εφαρμογή μόνο σε γραμμικώς διαχωρίσιμα προβλήματα.

- Πολυωνυμικοί (Polynomial):

$$K(\vec{x}, \vec{x}') = (\vec{x} \cdot \vec{x}' + 1)^p$$

όπου η παράμετρος p καθορίζεται από τον χρήστη. Στην Εικ. 6 παρουσιάζεται η περίπτωση εφαρμογής SVM γραμμικού πυρήνα και πολυωνυμικού πυρήνα σε μη-γραμμικό πρόβλημα. Είναι σαφής η μεγαλύτερη διαχωριστική ικανότητα που παρουσιάζει το SVM με πολυωνυμικό πυρήνα.

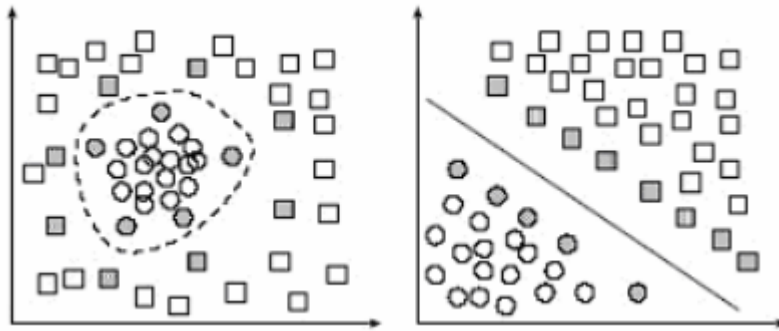


Εικόνα 7 Διαχωρισμός μη-γραμμικού προβλήματος με χρήση SVMs γραμμικού (διακεκομμένη γραμμή) και πολυωνυμικού πυρήνα (συνεχής γραμμή)

- Ακτινωτοί (Radial Basis Functions):

$$K(\vec{x}, \vec{x}') = e^{-\frac{\|\vec{x} - \vec{x}'\|^2}{2\sigma^2}},$$

όπου η παράμετρος σ καθορίζεται από τον χρήστη. Παράδειγμα ενός τέτοιου πυρήνα φαίνεται στην Εικ. 7



Εικόνα 8 Διαχωρισμός μη-γραμμικού προβλήματος με χρήση SVMs κυκλικού πυρήνα

- Σιγμοειδείς (Sigmoid):

$$K(\vec{x}, \vec{x}') = \tanh(\kappa \cdot \vec{x} \cdot \vec{x}' + \theta)$$

2.3.2. Adaptive Boosting (Ada-Boost) Algorithm

Όπως ήδη αναφέραμε σε προηγούμενη ενότητα, η αποδοτικότητα ενός μοντέλου που παράγεται από αλγόριθμους Μηχανικής Μάθησης καθορίζεται τόσο από το μέγεθος και την ποιότητα του training set, όσο και από την καταλληλότητα του χρησιμοποιούμενου αλγορίθμου μάθησης, παράγοντες οι οποίοι είναι κατά γενική ομολογία δύσκολο να προσδιορισθούν. Μια εναλλακτική προσέγγιση επιδιώκει να αυξήσει την αξιοπιστία ενός συστήματος Μ.Μ., επιστρατεύοντας την «εμπειρία» περισσότερων του ενός μοντέλων, από τον κατάλληλο συνδυασμό των οποίων προκύπτει η τελική απόφασή, αναφορικά με ένα άγνωστο στιγμιότυπο του προβλήματος. Στην περιοχή αυτή της Μ.Μ, η οποία ονομάζεται Μέτα-Μάθηση (Meta-Learning), συγκαταλέγεται και η μεθοδολογία που εφαρμόσαμε.

Συγκεκριμένα, χρησιμοποιήσαμε μια μέθοδο ενίσχυσης (boosting) γνωστό και ως Adaptive-Boost, ένας μετα-αλγόριθμος Μηχανικής Μάθησης, που διατυπώθηκε από Yoan Freund και Robert Schapire [11]. Ο αλγόριθμος ενίσχυσης ξεκινά με την ανάθεση ίσης βαρύτητας σε όλες τις περιπτώσεις των δεδομένων εκπαίδευσης. Στη συνέχεια ο αλγόριθμος εκμάθησης σχηματίζει ένα ταξινομητή για αυτό το σύνολο των δεδομένων. Εν συνεχεία επαναπροσδιορίζει τα βάρη στα δεδομένα σύμφωνα με την απόδοση του ταξινομητή του. Τα βάρη των σωστά διαβαθμισμένων δεδομένων μειώνεται, ενώ αυτών που ταξινομήθηκαν λανθασμένα αυξάνεται. Αυτό παράγει ένα σύνολο δεδομένων αποτελούμενο από εύκολες περιπτώσεις με χαμηλό βάρος, και μια

σειρά από δύσκολες περιπτώσεις με υψηλό βάρος. Στην επόμενη επανάληψη, ο ταξινομητής έχει κατασκευαστεί και τα βάρη των δεδομένων υπολογίζονται εκ νέου, τα οποία, κατά συνέπεια, επικεντρώνεται στην ταξινόμηση των δύσκολων περιπτώσεων έτσι ώστε να προβλεφθούν σωστά. Μετά τη διαδικασία ενημέρωσης των βαρών, αυτά κανονικοποιούνται έτσι ώστε το άθροισμά τους να παραμένει ίδιο όπως ήταν πριν. Μετά από όλες τις επαναλήψεις υπολογίζεται η τελική τιμή της υπόθεσης.

Ο ψευδο-κώδικας για αλγόριθμο Ada-Boost παρατίθεται παρακάτω στην Εικ. 8.

Adaboost Algorithm

Adaboost = ADaptive BOOSTing

Adaboost(*examples*, L , T)

1. $D_1(i) \leftarrow 1/m$ for each of m *examples*
2. **for** $t \leftarrow 1$ to T
3. $h_t \leftarrow$ apply L to *examples* using distribution D_t
4. $\epsilon_t \leftarrow$ sum of $D_t(i)$ for *examples* misclassified by h_t
5. $\beta_t \leftarrow (1 - \epsilon_t)/\epsilon_t$
6. $D_{t+1}(i) \leftarrow \begin{cases} D_t(i)\beta_t & \text{if } h_t \text{ misclassifies example } i \\ D_t(i) & \text{otherwise} \end{cases}$
7. normalize D_{t+1} so it sums to 1
8. **return** h_1, \dots, h_T and β_1, \dots, β_T

Adaboost Details

- D_t is a distribution, an assignment of a probability to each example.
- L should be sensitive to the distribution. If not, we can take a bootstrap sample using the distribution.
- If $\epsilon_t \geq .5$, then set T to $t - 1$ and break from loop. What if $\epsilon_t = 0$?
- After line 7, sum of D_{t+1} for misclassified exs. = sum for correctly classified examples = $1/2$. This forces h_{t+1} to be different from h_t .
- Classification is by plurality voting with base classifier h_t getting $\ln \beta_t$ votes.

Κεφάλαιο 3ο

Βασικές Δομές – Εργαλεία

Σε αυτήν την ενότητα αρχικά θα αναφέρουμε επιγραμματικά τις βασικές τεχνολογίες και πλατφόρμες που χρησιμοποιήσαμε και στη συνέχεια θα αναλύσουμε λεπτομερώς την κατασκευή του συστήματος μας.

3.1 Google App Engine

Ο όρος cloud computing (υπολογιστικό νέφος) χρησιμοποιείται μεταφορικά για το internet και βασίζεται στον τρόπο με τον οποίο το internet απεικονίζεται στα διαγράμματα δικτύων υπολογιστών. Απεικονίζεται ως ένα σύννεφο(cloud) και σύμφωνα [20] "Το εικονίδιο του σύννεφου αντιπροσωπεύει όλα αυτά τα πράγματα που κάνουν το δίκτυο να δουλεύει". Στην ουσία ο όρος νεφοϋπολογιστική ή Cloud computing ερμηνεύεται πλέον ως ένα μοντέλο για την παροχή καθολικής, εύκολης και ανάλογα με τις ανάγκες και τη ζήτηση, πρόσβαση σε ένα σύνολο παραμετροποιήσιμων υπολογιστικών παροχών (π.χ., δίκτυα, σέρβερ, αποθηκευτικού χώρου, εφαρμογών και υπηρεσιών) που μπορούν να διαχειριστούν και να προσφερθούν, απαιτώντας ελάχιστη ή και καθόλου παρέμβαση από τον χρήστη ή τον πάροχο.

Τα μοντέλα υπηρεσιών που υπάρχουν, είναι τρία. Το κάθε ένα από αυτά εξυπηρετεί διαφορετικές ανάγκες.

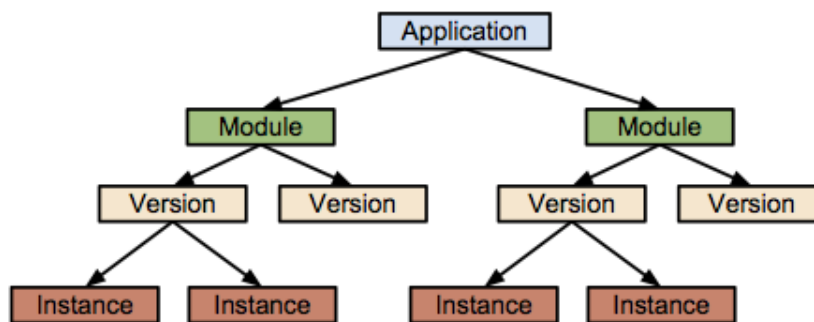
- Το Λογισμικό ως μια Υπηρεσία (Software as a Service-SaaS),
- η Πλατφόρμα ως Υπηρεσία (Platform as a Service-PaaS)
- και η Υποδομή ως Υπηρεσία (Infrastructure as a Service-IaaS).

Στην παρούσα εργασία θα γίνει χρήση του Platform-as-a-Service (PaaS) της Google, γνωστό και ως Google App Engine (GAE)[1]. Το GAE είναι μια πλατφόρμα για φιλοξενία και ανάπτυξη διαδικτυακών εφαρμογών σε datacenters που διαχειρίζεται η Google. Είναι δωρεάν μέχρι ενός συγκεκριμένου σημείου χρησιμοποιούμενων

υπολογιστικών πόρων. Μετά την υπέρβαση του ορίου αυτού υπάρχουν extra χρεώσεις για επιπλέον χρήση αποθηκευτικού χώρου, υπολογιστικής ισχύος, κτλ ανάλογα με τις ανάγκες του κάθε χρήστη. Αξίζει να σημειωθεί ότι η πλήρης αποδοχή της Google ως κορυφαίας εταιρίας στο χώρο της διαδικτυακής αναζήτησης δεν μεταφράζεται άμεσα σε αυτόματη ηγεσία στον τομέα του cloud computing, γεγονός που, εκτός των άλλων, αποδεικνύει και τον τεράστιο ανταγωνισμό που αναπτύσσεται μεταξύ των εταιριών που δραστηριοποιούνται στο χώρο.

Το “platform as service” μοντέλο χρησιμοποιείται πιο πολύ για δημιουργία web interfaces, web εφαρμογών κλπ. Ένα σημαντικό πρόβλημα που υπάρχει με αυτό το μοντέλο είναι ότι αυτή η εφαρμογή που δημιουργούμε βασίζεται σε ένα συγκεκριμένο framework και υπάρχει πιθανότητα αν θελήσουμε να την μεταφέρουμε σε άλλο πάροχο cloud υπηρεσιών αυτή να μη λειτουργεί σωστά.

Η βασική αρχιτεκτονική μιας GAE εφαρμογής παρουσιάζεται στην Εικόνα 10.



Εικόνα 10 Αρχιτεκτονική – Δομή μιας Εφαρμογής σε GAE

Όσον αφορά την παραμετροποίηση στον κώδικα του Backend μας στο Google App Engine, πρέπει να αναφέρουμε πως παρουσιάζει διαφορές ανάμεσα σε ελεύθερες και εμπορικές (επί πληρωμή) εφαρμογές. Οι ουσιαστικότερες ρυθμίσεις είναι αυτές που έχουν να κάνουν με τους πόρους του συστήματος που θα υποστηρίξει την εφαρμογή μας. Οι βασικότερες εξ’ αυτών αφορούν τα χαρακτηριστικά λειτουργίας των διαφόρων instances, όπως φαίνονται σημειωμένες στην εικόνα 11, όπου παρατίθεται τμήμα του αρχείου appengine.xml στο οποίο λαμβάνουν χώρα όλες οι παραμετροποιήσεις, πιο συγκεκριμένα έχουμε στην περίπτωση αυτή επιλέξει να ενεργοποιούνται instances τύπου B2, (χαρακτηριστικά των διαφόρων τύπων μπορείτε

να δείτε στην Εικόνα 11). Επιπλέον σημαντική είναι η επιλογή του τρόπου Scaling, του τρόπου δηλαδή το σύστημα θα ανταποκρίνεται σε αυξομειώσεις στις απαιτήσεις πόρων.

```
<?xml version="1.0" encoding="utf-8"?>
  <appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
    <application>unipi-makridis</application>
    <module>default</module>
    <version>1</version>

    <!--
      Allows App Engine to send multiple requests to one instance in
      parallel:
    -->
    <threadsafe>true</threadsafe>
    <!-- Configure java.util.logging -->
    <system-properties>
      <property name="java.util.logging.config.file" value="WEB-
      INF/logging.properties"/>
    </system-properties>
    <sessions-enabled>true</sessions-enabled>
    <async-session-persistence enabled="true" />

    <instance-class>B2</instance-class>
    <basic-scaling>
      <max-instances>20</max-instances>
      <idle-timeout>1m</idle-timeout>
    </basic-scaling>

  </appengine-web-app>
```

Εικόνα 11 τμήμα του αρχείου appengine.xml

Instance Class	Memory Limit	CPU Limit	Cost per Hour per Instance
B1	128 MB	600 Mhz	\$0.05
B2	256 MB	1.2 Ghz	\$0.10
B4	512 MB	2.4 Ghz	\$0.20
B4_1G	1024 MB	2.4 Ghz	\$0.30
B8	1024 MB	4.8 Ghz	\$0.40
F1	128 MB	600 Mhz	\$0.05
F2	256 MB	1.2 Ghz	\$0.10
F4	512 MB	2.4 Ghz	\$0.20
F4_1G	1024 MB	2.4 Ghz	\$0.30

Εικόνα 12 Διάφοροι τύποι instances με τα χαρακτηριστικά τους και το κόστος

Σε αυτό το σημείο θα πρέπει να σημειωθεί πως το Google App Engine δεν ενδείκνυται για μεγάλες υπολογιστικές πράξεις (καθώς έχει 10-λεπτά χρονικό περιορισμό για την ολοκλήρωση μιας λειτουργίας). Επίσης λόγω των αυξημένων υπολογιστικών απαιτήσεων του συστήματος απαιτήθηκε η χρήση <instance-class> B8 κατά τη διάρκεια των δοκιμών μας.

Δεν πρέπει να παραβλέψουμε να αναφέρουμε τη βάση δεδομένων που υποστηρίζει το GAE (Datastore API) η οποία είναι αντικειμενοστραφής σε αντίθεση με την γνωστή σε όλους SQL που είναι σχεσιακή. Ενώ έχουμε κάνει χρήση και του αντίστοιχου συστήματος αποθήκευσης αρχείων το blobstore API.

Datastore API

Ο κύριος χώρος αποθήκευσης δεδομένων είναι ο High Replication Datastore (HRD) , στον οποίο τα δεδομένα αναπαράγονται σε πολλαπλά κέντρα δεδομένων χρησιμοποιώντας ένα σύστημα που βασίζεται στον αλγόριθμο Paxos. Το ανωτέρω μοντέλο παρέχει υψηλό επίπεδο διαθεσιμότητας στις εφαρμογές τις οποίες υποστηρίζει.

Η Datastore αποθηκεύει αντικείμενα δεδομένων γνωστά ως οντότητες. Μία οντότητα έχει μία ή περισσότερες ιδιότητες, που ονομάζονται Properties υποστηρίζοντας πολλούς τύπους δεδομένων. Κάθε οντότητα προσδιορίζεται από το είδος της, το οποίο κατατάσσει την υποκείμενη οντότητα με τέτοιο τρόπο ώστε να διευκολύνεται η αναζήτηση της με τη χρήση των ερωτήσεων (queries), και ένα κλειδί που χαρακτηρίζει μοναδικά στο είδος της. Η Datastore μπορεί να εκτελέσει πολλαπλές

λειτουργίες σε μία μόνο συναλλαγή. Περισσότερες λεπτομέρειες μπορεί να αναζητήσει κάποιος από την επίσημη βιβλιογραφία της Google [22].

Blobstore API

Η Blobstore API εξυπηρετεί την αποθήκευση αντικειμένων δεδομένων, που ονομάζεται blobs, επιτρέποντας την αποθήκευση αρχείων μεγαλύτερα από το επιτρεπόμενο μέγεθος για τα αντικείμενα στην υπηρεσία Datastore. Η υπηρεσία είναι χρήσιμη για την εξυπηρέτηση μεγάλων αρχείων, όπως βίντεο ή αρχεία εικόνας. Τα blobs συνήθως δημιουργούνται από τη μεταφόρτωση ενός αρχείου από ένα αίτημα κάποιας εξωτερικής εφαρμογής ή ακόμη και από τον ίδιο το server (όπως στη δική μας περίπτωση) .

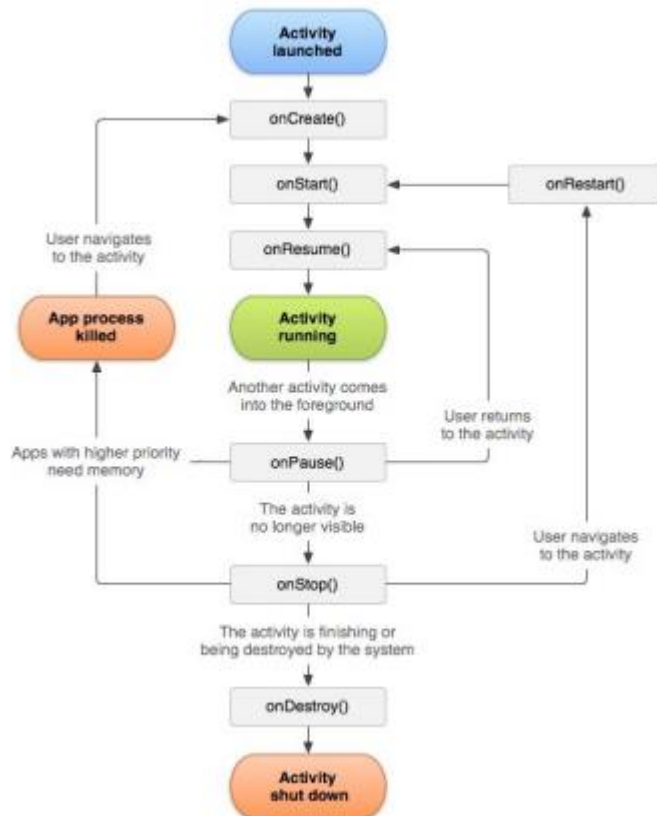
3.2 Android

Το Android είναι ένα λειτουργικό σύστημα, βασισμένο στο Linux, για φορητές συσκευές όπως smartphones και tablets. Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού.

Μια μεγάλη κοινότητα προγραμματιστών κατασκευάζουν εφαρμογές, οι οποίες εκμεταλλεύονται τις δυνατότητες που προσφέρουν οι κινητές συσκευές (υπηρεσίες εντοπισμού, εκτέλεση διεργασιών παρασκηνίου, και πάρα πολλές ακόμη δυνατότητες οι οποίες βασίζονται στα APIs που είναι διαθέσιμα) επεκτείνοντας τη λειτουργικότητα τους. Οι εφαρμογές γράφονται σε μια προσαρμοσμένη έκδοση της JAVA χρησιμοποιώντας το SDK του Android. Το Android είναι η πρώτη σε πωλήσεις παγκοσμίως πλατφόρμα για smartphones καθώς υποστηρίζεται από μεγάλες εταιρείες παραγωγής κινητών συσκευών.

Η δομή των εφαρμογών Android είναι τέτοια που ευνοείται η επαναχρησιμοποίηση δομικών συστατικών, και επίσης επιτρέπεται η χρήση των δυνατοτήτων τις μίας εφαρμογής από άλλες εφαρμογές, βέβαια κάτω από τις προδιαγραφές ασφάλειας. Τα σημαντικότερα δομικά στοιχεία του πλαισίου εφαρμογών είναι:

- Σύστημα προβολών (View System) – αποτελεί ένα εκτενές σύνολο από αντικείμενα του γραφικού περιβάλλοντος, τα οποία χρησιμοποιούνται κατά το σχεδιασμό μιας εφαρμογής. Παραδείγματα προβολών είναι οι λίστες (listView, recyclerView), το πλέγμα (GridView), πεδία εισαγωγής κειμένου, κουμπιά, κλπ
- Πάροχος Περιεχομένου (Content Provider) – δίνει τη δυνατότητα στις εφαρμογές να μοιράζονται ή να ανταλλάσσουν δεδομένα μιας συγκεκριμένης μορφής η οποία ορίζεται από τον πάροχο. Παραδείγματα δεδομένων, είναι οι επαφές χρήστη και οι βάσεις δεδομένων των εφαρμογών.
- Διαχειριστής Πόρων (Resource Manager) – παρέχει πρόσβαση σε υλικό το οποίο δεν είναι σε μορφή κώδικα όπως πχ, εικόνες, αρχεία xml, πίνακες χαρακτήρων, κλπ
- Διαχειριστής Ειδοποιήσεων (Notification Manager) – δίνει στις εφαρμογές πρόσβαση στις υπηρεσίες ειδοποιήσεων χρήστη. Τέτοιες είναι οι ειδοποιήσεις στη notification bar, τα toast μηνύματα στο κάτω μέρος της οθόνης, η δόνηση του κινητού και η ενεργοποίηση της οθόνης, κλπ
- Διαχειριστής Δραστηριοτήτων (Activity Manager) – διαχειρίζεται τον κύκλο ζωής των δραστηριοτήτων και παρέχει δυνατότητα πλοήγησης από δραστηριότητα σε δραστηριότητα κρατώντας αποθηκευμένα στη μνήμη τη σειρά εκτέλεσης αυτών. Στο σχεδιάγραμμα (Εικόνα 13) φαίνεται λεπτομερώς ο κύκλος ζωής κάθε δραστηριότητας.



Εικόνα 13 Activity life cycle

Στο εσωτερικό μιας εφαρμογής του Android

Κάθε εφαρμογή αποτελείται από ένα σύνολο αρχείων και φακέλων, τα οποία αφού γίνουν compile μας δίνουν το αρχείο .apk. Το εκτελέσιμο αυτό αρχείο αποτελεί την ουσιαστικά το αρχείο εγκατάστασης της εφαρμογής το οποίο μπορούμε να εγκαταστήσουμε στις συσκευές μας.

Ξεκινώντας, η κάθε εφαρμογή αποτελείται όπως είπαμε από πολλά αρχεία δομημένα σε φακέλους. Όλες οι εφαρμογές πρέπει να έχουν ένα μοναδικό όνομα πακέτου (package name) το οποίο χρησιμοποιείται από το λειτουργικό σύστημα για αναγνώριση της εφαρμογής.

Μια εφαρμογή μπορεί να αποτελείται από πολλά υποπακέτα, εφόσον αυτό είναι απαραίτητο λόγω της πολυπλοκότητας της εφαρμογής, αλλά μόνο από ένα κύριο. Στη συνέχεια θα παρουσιάσουμε τα βασικότερα σημεία (αρχεία) ενός android project επικεντρώνοντας κυρίως στο ρόλο που έχουν στην δημιουργία και τη λειτουργία της εφαρμογής.

Το αρχείο AndroidManifest.xml

Κάθε project εφαρμογής περιέχει το αρχείο AndroidManifest. Σε αυτό καταχωρούνται οι σημαντικότερες πληροφορίες της εφαρμογής για χρήση από το λειτουργικό σύστημα. Κάποιες από αυτές τις πληροφορίες είναι:

- Το όνομα του πακέτου της εφαρμογής
- Το όνομα που φαίνεται στον χρήστη
- Η έκδοση των APIs που χρησιμοποιούνται
- Ο αριθμός έκδοσης της εφαρμογής
- Οι άδειες χρήσης που ζητάει η εφαρμογή
- Όλες οι δραστηριότητες, πάροχοι περιεχομένου, υπηρεσίες, κλπ, που περιέχει και χρησιμοποιεί η εφαρμογή.

Όπως αντιλαμβανόμαστε πρόκειται για πολύ σημαντικό αρχείο και αποτελεί κύριο συστατικό κάθε εφαρμογής.

Οι φάκελοι src & res

Στον φάκελο src (εκ του source) περιέχονται τα αρχεία της Java όλων των Activities, Services, Content Providers, βοηθητικά αρχεία, κλπ. και αποτελεί τον βασικό φάκελο στο project στον οποίο εκτελείται η βασική δουλειά του προγραμματιστή. Φυσικά αυτό μπορεί να αλλάξει σε πιο πολύπλοκα project τα οποία απαραίτητα θα ακολουθούν κάποιο design pattern βάσει του οποίου τυπικά η αρχιτεκτονική των αρχείων ίσως παρουσιάζει κάποιες διαφορές.

Ο φάκελος res (εκ του resources) περιέχει όλα τα αρχεία εικόνας, κειμένου, xml layout, κλπ τα οποία χρησιμοποιούνται από τις Activities που βρίσκονται στον φάκελο src. Με άλλα λόγια εδώ καταχωρούνται όλο το γραφικό πλαίσιο της εφαρμογής.

Φυσικά δεν βρίσκονται όλα τα αρχεία πόρων, σε έναν φάκελο, αλλά είναι χωρισμένα και ταξινομημένα σε υπο-φακέλους ανάλογα με το είδος τους. Συνηθισμένοι υπό-φάκελοι του κύριου φακέλου res, είναι ο φάκελος drawable ο οποίος περιέχει τα αρχεία εικόνας (.png, .jpg, .gif) τα οποία χρησιμοποιεί η εφαρμογή μας, ο φάκελος layout ο οποίος περιέχει όλα τα αρχεία xml τα οποία ορίζουν τα διάφορα layouts που υπάρχουν στην εφαρμογή, και τέλος ο φάκελος values στον οποίο αποθηκεύονται όλοι οι πόροι κειμένου που χρησιμοποιούνται στην εφαρμογή.

Δομικά Μέρη μιας Εφαρμογής

Παραπάνω αναφέραμε ότι όλα τα δομικά μέρη της εφαρμογής πρέπει να αναφέρονται αναλυτικά στο αρχείο AndroidManifest.xml, ποια είναι όμως αυτά τα δομικά μέρη και πια η λειτουργία του καθενός;

- Δραστηριότητες (Activities) – Πρόκειται ίσως για το κύριο δομικό στοιχείο μιας εφαρμογής. Δραστηριότητα είναι μια οθόνη διεπαφής χρήστη (GUI) και προβολής πληροφοριών. Κάθε εφαρμογή έχει τόσες Activities όσες και οι διαφορετικές οθόνες οι οποίες εμφανίζονται στον χρήστη. Όλες οι δραστηριότητες συνεργάζονται μεταξύ τους για να δώσουν στον χρήστη μια συνολική εμπειρία χρήσης της εφαρμογής.
- Προθέσεις (Intents) – Οι δραστηριότητες επικοινωνούν και εναλλάσσουν την λειτουργία τους μέσω των Intents. Ουσιαστικά τα Intents εξασφαλίζουν την μετάβαση από την μία δραστηριότητα σε μια άλλη και επίσης χρησιμοποιούνται για ανταλλαγή δεδομένων. Η ανταλλαγή δεδομένων, μπορεί να γίνει είτε μεταξύ των Activities μιας εφαρμογής, είτε από τη μία εφαρμογή στην άλλη. Παραδείγματος χάρη μπορούμε μέσω ενός Intent μπορεί να εκκινήσει ένας browser ώστε να μας ανοίξει απευθείας ένα url.
- Υπηρεσίες (Services) – Πρόκειται για λειτουργίες της εφαρμογής οι οποίες είναι σχεδιασμένες να τρέχουν στο παρασκήνιο και να επιστρέφουν αποτελέσματά ακόμη και όταν η εφαρμογή δεν είναι στο προσκήνιο. Πχ μια εφαρμογή media player μπορεί μέσω μιας υπηρεσίας να συνεχίσει να παίζει μουσική ακόμη και όταν το κύριο παράθυρο της εφαρμογής δεν βρίσκεται στο προσκήνιο.
- Πάροχος Περιεχομένου (Content Providers) - Η ανταλλαγή δεδομένων από μια εφαρμογή στην άλλη όπως είπαμε παραπάνω μπορεί να γίνει μέσω ενός Intent, ένας πάροχος περιεχομένου όμως έχει πιο σύνθετη λειτουργία. Οι content providers μιας εφαρμογής διαχειρίζονται συγκεκριμένα δεδομένα της εφαρμογής τα οποία έχει ορίσει ο προγραμματιστής κατά την κατασκευή του. Συνηθισμένα δεδομένα τα οποία μοιράζονται μέσω Content Providers, είναι οι βάσεις δεδομένων SQLite μιας εφαρμογής, και οι επαφές του χρήστη.
- Δέκτες Μετάδοσης (Broadcast Receivers) – Πρόκειται για ένα είδος υπηρεσίας η οποία αντιλαμβάνεται κάποια γεγονότα του συστήματος και αναλαμβάνει να ενημερώσει το σύστημα ή τις υπόλοιπες εφαρμογές. Ο σκοπός τους είναι διπλός καθότι μπορούν και να ενημερωθούν για κάποιο συμβάν από άλλες εφαρμογές, αλλά

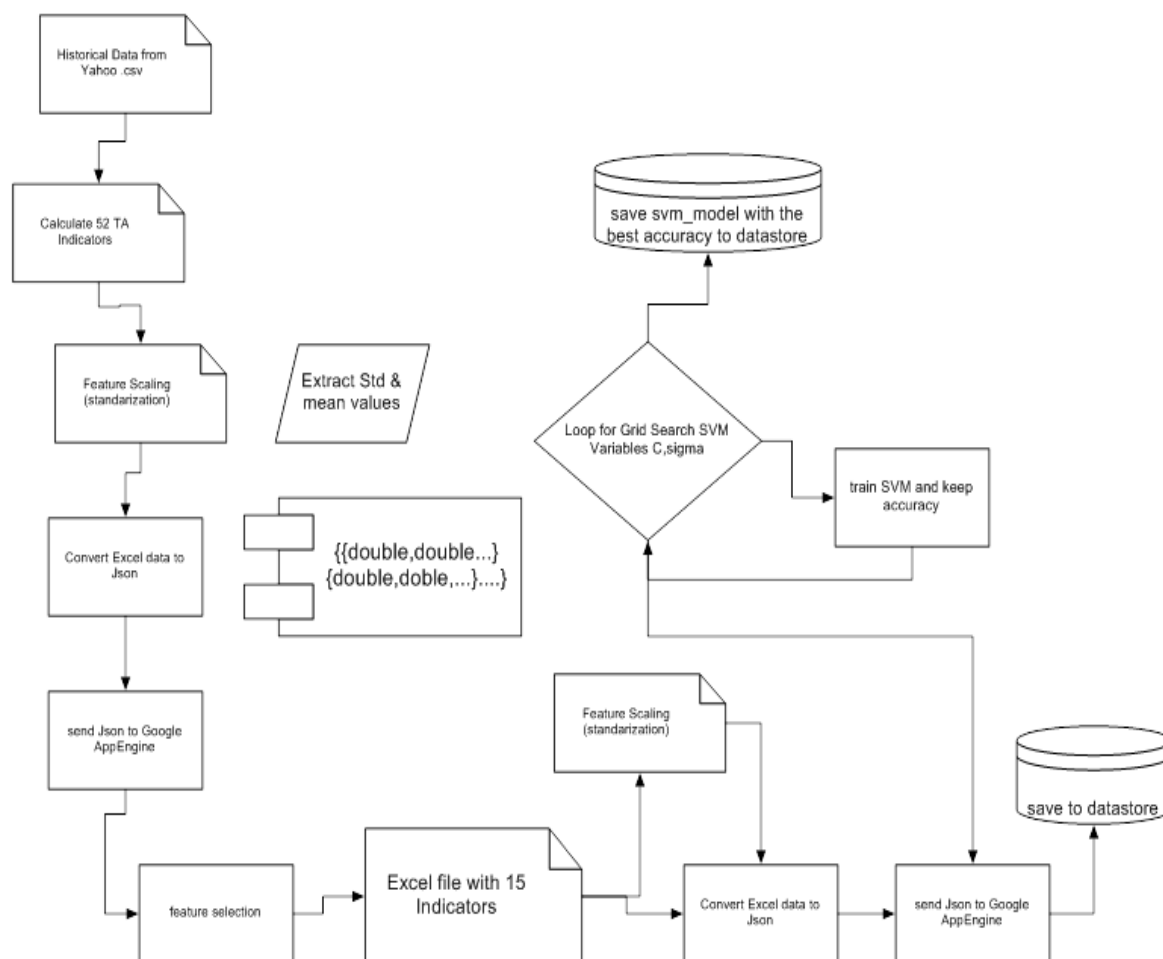
και να ειδοποιήσουν τις υπόλοιπες εφαρμογές και το σύστημα για κάποιο συμβάν που τις ενεργοποίησε. Δεν έχουν γραφικό περιβάλλον αλλά μπορούν να προβάλουν ειδοποίηση στον χρήστη μέσω της μπάρας ειδοποιήσεων. Συνήθως χρησιμοποιούνται ως διαμεσολαβητές μεταξύ των Activities και των Services μιας εφαρμογής.

Κεφάλαιο 4ο

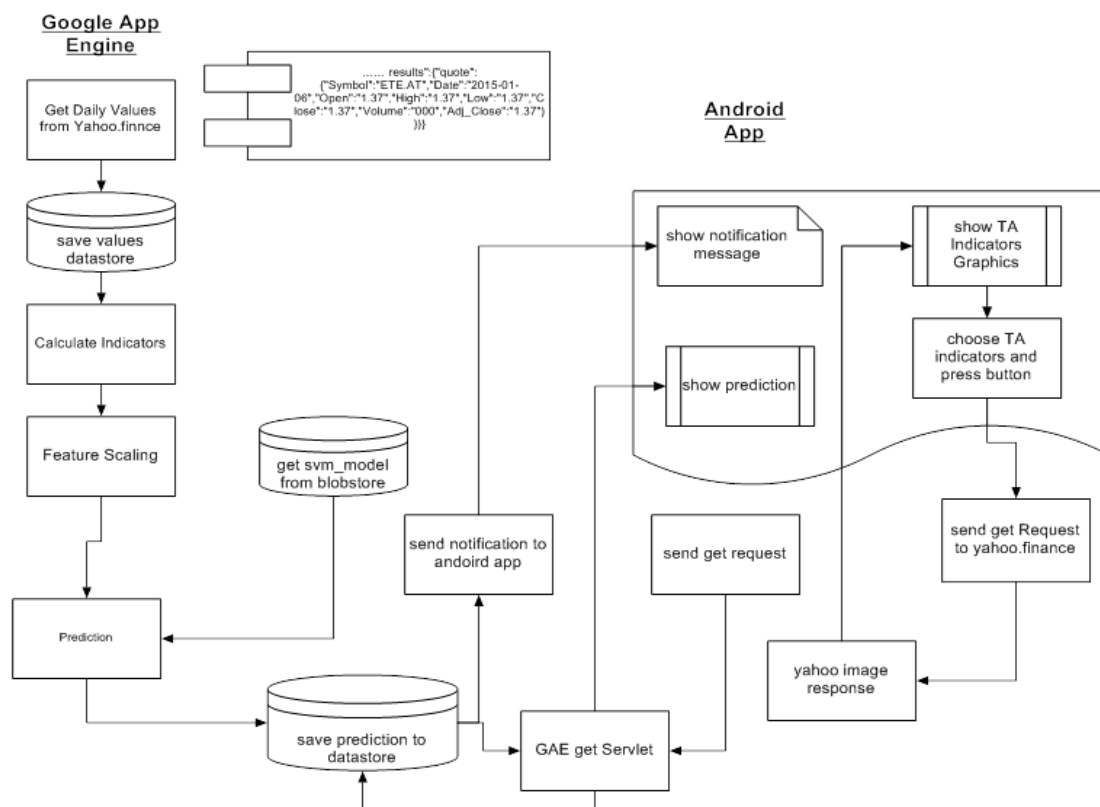
Υλοποίηση Συστήματος

4.1 Δομή – Αρχιτεκτονική Συστήματος

Όπως αναφέρθηκε ανωτέρω, για την υλοποίηση χρησιμοποιήσαμε δεδομένα μιας μετοχής. Η μετοχή που επιλέξαμε είναι αυτή της ΕΘΝΙΚΗΣ ΤΡΑΠΕΖΑΣ (ETE) του δείκτη FTE20 του ΧΑΑ. Η γενική φιλοσοφία της υλοποίησης αναπαρίσταται στις Εικόνες 14 και 15.



Εικόνα 14 Σχεδιάγραμμα ροής για το train του classifier



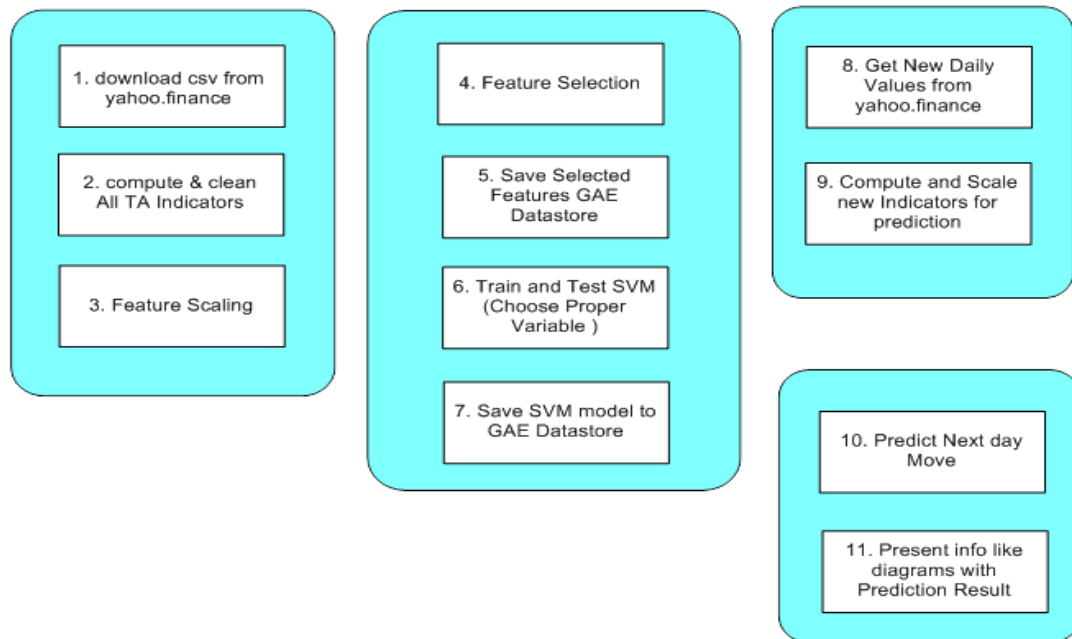
Εικόνα 15 Σχεδιάγραμμα πρόβλεψης μεμονωμένου παραδείγματος

Πιο συγκεκριμένα στο σχέδιο της Εικόνας 7 φαίνεται η ροή των διαδικασιών από την λήψη του αρχείου csv, το οποίο περιέχει τις βασικές τιμές για τον υπολογισμό των τιμών Τεχνικής Ανάλυσης, μέχρι και την αποθήκευση του svm_model στο Google AppEngine σε μορφή .txt. Το svm_model είναι ουσιαστικά η γνώση που αποκτά το σύστημα μας μετά την «εκπαίδευση», το αποτέλεσμα με το οποίο μπορούμε να παράγουμε προβλέψεις σε συνδυασμό με τα «άγνωστα» δεδομένα. Προφανώς τα ενδιάμεσα στάδια περιλαμβάνουν διεργασίες προ-επεξεργασίας δεδομένων όπως feature Selection, Feature scaling, Training καθώς και αποθήκευση κάποιων δεδομένων στη βάση δεδομένων του GAE (datastore) με σκοπό τη χρήση τους σε μεταγενέστερες διεργασίες κ.α.

Αντίστοιχα στην Εικόνα 8. Παρουσιάζονται τα βήματα που ακολουθήσαμε για την υλοποίηση των καθημερινών προβλέψεων με ορίζοντα ενός μήνα. Τα βήματα αυτά περιλαμβάνουν όλη τη διαδικασία από την εξαγωγή των καθημερινών τιμών της μετοχής που προαναφέραμε αποσκοπώντας στον υπολογισμό των νέων τιμών Δεικτών Τεχνικής Ανάλυσης με σημείο αναφοράς την πιο πρόσφατη ημερομηνία τα

οποία, στην ουσία αποτελούν τις εισόδους στον Αλγόριθμο πρόβλεψης, μέχρι και τον υπολογισμό (πρόβλεψη) της κίνησης της μετοχής σε ορίζοντα ενός μήνα και την απεικόνιση αυτού σε μια εφαρμογή Android.

Όλα τα ανωτέρω βήματα (των Εικόνων 14 και 15) παρουσιάζονται επιγραμματικά στην Εικόνα 16. Ενώ αναλύονται λεπτομερώς στο Κεφάλαιο 3.2.



Εικόνα 16 Βήματα υλοποίησης συστήματος

4.2 Δεδομένα

Όπως αναφέραμε στο Κεφάλαιο 2.2.1 το σημαντικότερο στοιχείο καθοριστικής σημασίας για την απόδοση του κάθε αλγορίθμου Μηχανικής Μάθησης είναι η αξιοπιστία και η κατάλληλη επεξεργασία των δεδομένων εκπαίδευσης

4.2.1 Βήμα 1^ο – 2^ο - Πηγή Δεδομένων

Αρχικά τα δεδομένα που θα χρησιμοποιηθούν σαν παραδείγματα και θα αποτελέσουν το σώμα των Δεδομένων μας το οποίο αργότερα θα κατατμηθεί σε σώμα «εκπαίδευσης» και «έλεγχου απόδοσης» στον Αλγόριθμο μας, είναι διαθέσιμα στο site της YAHOO [23]. Όπως φαίνεται στην Εικόνα 10.

finance.yahoo.com/q/hp?s=ETE.AT+Historical+Prices

YAHOO! FINANCE

Search Finance Search Web

Finance Home My Portfolio My Quotes News Market Data Yahoo Originals Business & Finance Personal Finance CNBC Contributors

Enter Symbol Look Up Fri, Mar 6, 2015, 3:04AM EST - US Markets open in 6 hrs and 26 mins Report an Issue

Dow ↑0.21% Nasdaq ↑0.32%

More On ETE.AT

QUOTES Summary Order Book Options

Historical Prices

Set Date Range

Start Date: Jan 3 2000 Eg. Jan 1, 2010

End Date: Mar 6 2015

Get Prices

Prices

Date	Open	High	Low	Close	Volume	Adj Close*
Mar 5, 2015	1.24	1.35	1.23	1.34	7,674,300	1.34
Mar 4, 2015	1.30	1.32	1.18	1.25	10,048,500	1.25
Mar 3, 2015	1.26	1.31	1.25	1.30	12,188,700	1.30
Mar 2, 2015	1.29	1.30	1.19	1.19	13,866,700	1.19

Εικόνα 17 Δεδομένα από Yahoo finance

Η μορφή που αποκτήσαμε τα δεδομένα είναι .csv. Ο αριθμός των δεδομένων είναι αρχικά ήταν περίπου 3500, καλύπτοντας ημερήσιες ιστορικές τιμές από το 2000. Σε αυτό το σημείο χωρίσαμε στις εξής στήλες DATE OPEN CLOSE HIGH LOW VOLUME ADJ. CLOSE όπως φαίνεται στην Εικόνα 11. Ανωτέρω διεργασία έγινε με χρήση του Microsoft Excel και δεν αποτελεί τμήμα του αυτοματοποιημένου συστήματος.

A	B	C	D	E	F	G	H
	Date	Open	High	Low	Close	Volume	
	03-01-00	73.86	76.34	73.77	74.71	49300	
	04-01-00	73.98	73.98	70.99	71.81	60800	
	05-01-00	66.78	70.4	66.78	69.01	44000	
	06-01-00	68.98	68.98	68.98	68.98	0	
	07-01-00	69.65	70.25	68.22	68.71	37800	
	10-01-00	69.11	69.68	68.41	68.65	20800	
	11-01-00	68.61	68.61	67.1	67.43	14000	
	12-01-00	66.88	72.6	62.92	66.79	20400	
	13-01-00	67.02	67.92	66.47	67.02	19700	
	14-01-00	67.98	69.49	67.71	68.77	35200	
	17-01-00	70.16	70.85	69.04	69.37	42100	
	18-01-00	69.49	69.49	68.13	68.61	18200	
	19-01-00	65.93	67.65	65.72	65.99	46800	
	20-01-00	66.48	67.05	65.42	65.66	34500	
	21-01-00	65.81	66.41	65.54	65.84	15300	
	24-01-00	66.05	67.32	65.15	66.2	40000	
	25-01-00	65.03	66.3	64.37	65.39	26100	
	26-01-00	66.22	66.37	64.35	64.35	29100	

Εικόνα 18 Excel με αρχικές τιμές ETE

Οι 3500 τιμές που προαναφέραμε φαντάζουν λίγες για μία υλοποίηση αλγορίθμου Μηχανικής Μάθησης οπότε αποφασίσαμε να χρησιμοποιήσαμε όμοια δεδομένα από

τις μετοχές Τραπεζικού Κλάδου που ανήκουν στον δείκτη ftse20. Πιο συγκεκριμένα χρησιμοποιήσαμε τιμές από ALPHA, ΤΡΑΠΕΖΑ ΠΕΙΡΑΙΩΣ και EUROBANK εκτιμώντας ότι παρουσιάζουν όμοια χρηματιστηριακή συμπεριφορά.

Έχοντας τις παραπάνω τιμές δημιουργήσαμε τους δείκτες Τεχνικής Ανάλυσης τους οποίους θα χρησιμοποιήσουμε για να εκπαιδεύσουμε το σύστημα μας. Οι δείκτες και οι τρόπος υπολογισμού απορρέει από την παράγραφο 1.4.3 Οι δείκτες αυτοί φαίνονται στον πίνακα 3.

Open	EMA 50	MOMENTUM 10	%K 7	EMA 26	Slope 20 High
High	ALF 5	MFI 14	%K 8	MACD	
Low	ALF 20	Upper Bollinger	%K 9	MACD EMA 9	
Close	ALF 50	Lower Bollinger	%K 10	CCI	
Volume	RSI 14	%R 6	PROC 12	High price average 2	
SMA 5	MOMENTUM 3	%R 7	PROC 13	Low price average 2	
SMA 20	MOMENTUM 4	%R 8	PROC 14	Slope 3 High	
SMA 50	MOMENTUM 5	%R 9	PROC 15	Slope 4 High	
EMA 5	MOMENTUM 8	%R 10	WCL	Slope 5 High	
EMA 20	MOMENTUM 9	%K 6	EMA 12	Slope 10 High	

Πίνακας 3 Υπολογιζόμενοι Δείκτες Τεχνικής Ανάλυσης

Οι υπολογισμοί έγιναν με χρήση του Microsoft Excel. Τόσο οι τιμές όσο και οι συναρτήσεις υπολογισμού φαίνονται στο συνημμένο αρχείο Microsoft Excel με όνομα ptixiaki.xls

4.2.2 –Βήμα 3^ο - Προ-Επεξεργασία Δεδομένων

Ακολούθως και αφού έχουμε συγκεντρώσει τα αρχικά δεδομένα μας από τα οποία αποτελείται το data set μας, πρέπει τα δεδομένα αυτά να τα επεξεργαστούμε κατάλληλα με σκοπό την καλύτερη εκμετάλλευση από τον αλγόριθμο μας. Όπως προαναφέραμε το στάδιο αυτό καλείται ως προ-επεξεργασία δεδομένων και περιλαμβάνει τα κατωτέρω βήματα:

Standarixzation

Έχοντας υπολογίσει τις πραγματικές τιμές των δεικτών Τεχνικής Ανάλυσης, το επόμενο βήμα για την υλοποίηση του αλγόριθμου είναι όπως προαναφέραμε σε παράγραφο 2.2.4 η κλιμάκωση των δεδομένων. Σε αυτή τη περίπτωση θα πραγματοποιήσουμε κλιμάκωση Standarization.

Η ανωτέρω εργασία έγινε σε περιβάλλον της Matlab, ενώ ο κώδικας που χρησιμοποιήθηκε για το Standarization παρατίθεται παρακάτω:

```
X_norm = X;
mu = zeros(1, size(X, 2)); (mean value matrix initialization)
sigma = zeros(1, size(X, 2)); (standar deviation matrix initialization)

mu=mean(X); (mean value calculation for each feature)
sigma=std(X); (std value calculation for each feature)
for i=1:size(X,2)
X_norm(:,i)=(X(:,i)-mu(1,i))/sigma(1,i); (X scaled calculation)
End
```

Μπορούμε εύκολα να διακρίνουμε ότι το προκύπτον σώμα δεδομένων δημιουργείται αφαιρώντας την mean value και εν συνεχεία διαιρώντας με τη διακύμανση, υποδεικνύοντας τη μέθοδο Standarization.

Προ – επεξεργασία Ετικέτας

Σε αυτό το σημείο υπολογίσαμε και σημειώσαμε με 0 την πτώση σε ένα μήνα (22 ημέρες λειτουργίας του χρηματιστηρίου) και με 1 την αντίστοιχη άνοδο. Όπως φαίνεται σε συνημμένο αρχείο ptixiaki.xls στη τελευταία στήλη.

Imputation of missing values

Για διάφορους λόγους, πολλά πραγματικά σύνολα δεδομένων περιέχουν τιμές που λείπουν, συχνά κωδικοποιούνται ως κενά, NaN ή άλλα σύμβολα κράτησης θέσης. Τέτοια σύνολα δεδομένων, ωστόσο είναι ασυμβίβαστα με Αλγορίθμους Μηχανικής Μάθησης. Μια βασική στρατηγική που χρησιμοποιούν για ελλιπή σύνολα δεδομένων είναι να απορρίψει ολόκληρες γραμμές ή / και στήλες που περιέχουν τις τιμές που λείπουν. Ωστόσο, αυτό έρχεται με το τίμημα της απώλειας δεδομένων, τα οποία μπορεί να είναι πολύτιμα (αν και ημιτελή). Μια καλύτερη στρατηγική είναι να γίνει εκτίμηση των ελλειπουσών τιμών, δηλαδή, να τους συναγάγει από το γνωστό μέρος

των δεδομένων. Μία στρατηγική κάλυψης τέτοιων σημείων είναι είτε χρησιμοποιώντας τη μέση τιμή, το μέσο όρο ή την πιο συχνή τιμή της γραμμής ή της στήλης στην οποία βρίσκονται οι τιμές που λείπουν.

4.3 Επιλογή κατάλληλου Αλγορίθμου – Υλοποίηση

Στα πλαίσια της ανά χείρας εργασίας, δοκιμάσαμε σε περιβάλλον Matlab τρεις αλγόριθμους. Προφανώς η μοντελοποίηση μας κατηγοριοποιεί το πρόβλημα ως πρόβλημα ταξινόμησης (classification) και οι αλγόριθμοι που δοκιμάσαμε είναι οι Logistic Regression – ANN (Artificial Neural Networks) – SVM (Support Vector Machines).

Ακολουθεί ο συγκριτικός πίνακας με τις αποδόσεις των ανωτέρω αλγορίθμων.

Μέθοδος	Ακρίβεια στο Training Set	Ακρίβεια στο Test Set	Παρατηρήσεις
Logistic Regression	61.464596	59.833333	
Neural Networks	65.739813	63.366667	input layer:12 , hidden layer: 30, iterations:1000
SVM	66.34	64.233333	C:1, σ:1

Θα πρέπει να σημειώσουμε πως οι Αλγόριθμοι που δοκιμάστηκαν, προέρχονται από την επίλυση των ασκήσεων του Μαθήματος Machine Learning από το Πανεπιστήμιο του Stanford [29], μέσω της ηλεκτρονική πλατφόρμα coursera. Επομένως θα πρέπει να σημειωθεί πως οι αλγόριθμοι που χρησιμοποιήθηκαν ίσως δεν ήταν πλήρως βελτιστοποιημένοι τόσο ως προς την απόδοση όσο και ως προς την ταχύτητα.

Τα αποτελέσματα των δοκιμών συμφωνούν με αυτά τις παραγράφου 2.3.. Οπότε και καταλήξαμε στην υλοποίηση σε Java αλγορίθμου SVM με Adaptive Boost. Για την υλοποίηση χρησιμοποιήσαμε τις εξής βιβλιοθήκες:

- ▶ Library for machine learning - Java ML [27]
- ▶ Library for machine Learning - Java Encog [28]
- ▶ Library for TA Indicators – Java TA [29]

4.3.1 Βήμα 4^ο Feature selection

Έχοντας τελειώσει με την προ-επεξεργασία των δεδομένων μας, έχει προκύψει ένα νέο data set το οποίο εκπροσωπεί τα ίδια χαρακτηριστικά με το αρχικό μόνο που οι

τιμές τους είναι κατάλληλα τροποποιημένες. Επόμενο στάδιο είναι η επιλογή χαρακτηριστικών, επίσης γνωστή ως επιλογή χαρακτηριστικού υποσυνόλου (FSS), ή επιλογή χαρακτηριστικού (Χαρακτηριστικό Επιλογής). Αποτελεί μια διαδικασία-μέθοδο για την επιλογή ενός υποσυνόλου από το σύνολο των χαρακτηριστικών της εισόδου, για να κάνει το μοντέλο πιο αποτελεσματικό. Ο όρος «αποτελεσματικό» ερμηνεύεται ως εξής: κατά την πρακτική εφαρμογή Αλγορίθμων Μηχανικής Μάθησης, η ποσότητα των χαρακτηριστικών είναι συνήθως πολύ μεγάλη, στα οποία μπορεί να υπάρχουν χαρακτηριστικά που δεν παρέχουν πληροφορία ή τα χαρακτηριστικά μπορεί να έχουν εξάρτηση μεταξύ τους. Η διαδικασία αυτή μπορεί να μειώσει τον αριθμός των χαρακτηριστικών βελτιώνοντας τόσο την ακρίβεια όσο και την ταχύτητα του μοντέλου. Μία δημοφιλής μέθοδος επιλογής είναι η συσχέτιση του εκάστοτε χαρακτηριστικού με την έξοδο του μοντέλου.

Στη περίπτωση μας, η υλοποίηση της αναφερόμενης επιλογής έγινε με χρήση της βιβλιοθήκης Java ML. Ενώ επίσης δημιουργήσαμε ένα java application με το οποίο διαβάζαμε τα δεδομένα από το αρχείο Excel και τα μετατρέπαμε σε JSONObject χρησιμοποιώντας τη Java XL. Το JSONObject μέσω HTTP POST Request το στέλναμε στο GAE. Η διαδικασία αυτή ήταν απαραίτητη καθώς ο διαθέσιμος stand alone υπολογιστής δεν διέθετε τους απαραίτητους πόρους έτσι ώστε να πραγματοποιήσει τέτοιους υπολογισμούς.

Στην Εικόνα 13 μπορούμε να δούμε τη μορφή του JSONObject με τις τιμές των δεικτών το οποίο όπως αναφέραμε αποστέλλεται στο backend (GAE).

```
{“data”:{[CLOSE[0],CLOSE[1],...CLOSE[N]},{OPEN[0],OPEN[1],...,OPEN[N]].....[SLOPE[0],SLOPE[1],...,SLOPE[N]],[OUT[0],OUT[1]....,OUT[N]]}}
```

Εικόνα 19 μορφή JSONObject που αποστέλλουμε στο Backend. Όπου N ο αριθμός των δεδομένων.

Ο κώδικας του java app για τη μετατροπή από Excel σε JSONObject παρατίθεται στην Εικόνα 20.

```

/**
 * reads columns from MS Excel inputFile and creates each column to a JSONArray
 * @throws IOException
 */
public void read() throws IOException {
    File inputWorkbook = new File(inputFile);
    Workbook w;
    try {
        w = Workbook.getWorkbook(inputWorkbook);
        // Get the first sheet
        Sheet sheet = w.getSheet(0);
        JSONArray columnsArray = new JSONArray();
        JSONArray rowsArray = new JSONArray();
        //create JSONArray for each feature (Excel Column)
        for (int j = 0; j < sheet.getColumns(); j++) {
            rowsArray = new JSONArray();
            for (int i = 0; i < sheet.getRows(); i++) {
                NumberCell numCell = (NumberCell) sheet.getCell(j,
i);
                rowsArray.put(numCell.getValue());
            }
            columnsArray.put(rowsArray);
        }
        //sends json to Servlet
        sendData(columnsArray);
    } catch (BiffException e) {
        e.printStackTrace();
    } catch (JSONException e) {
        e.printStackTrace();
    }
}
}

```

Εικόνα 20 Κώδικας για δημιουργία JSONObject από EXCEL

Ενώ το Servlet το οποίο υποδεχόταν την κλήση στο GAE (Η κλήση γίνεται στο URL /featureselection), υπολογίζει τη βαθμολογία συσχέτισης του κάθε χαρακτηριστικού με την ετικέτα. Συγκεκριμένα ο έλεγχος συσχέτισης με του κάθε χαρακτηριστικού με την έξοδο έγινε με χρήση της βιβλιοθήκης JavaML όπως φαίνεται και σε Εικόνα 21.

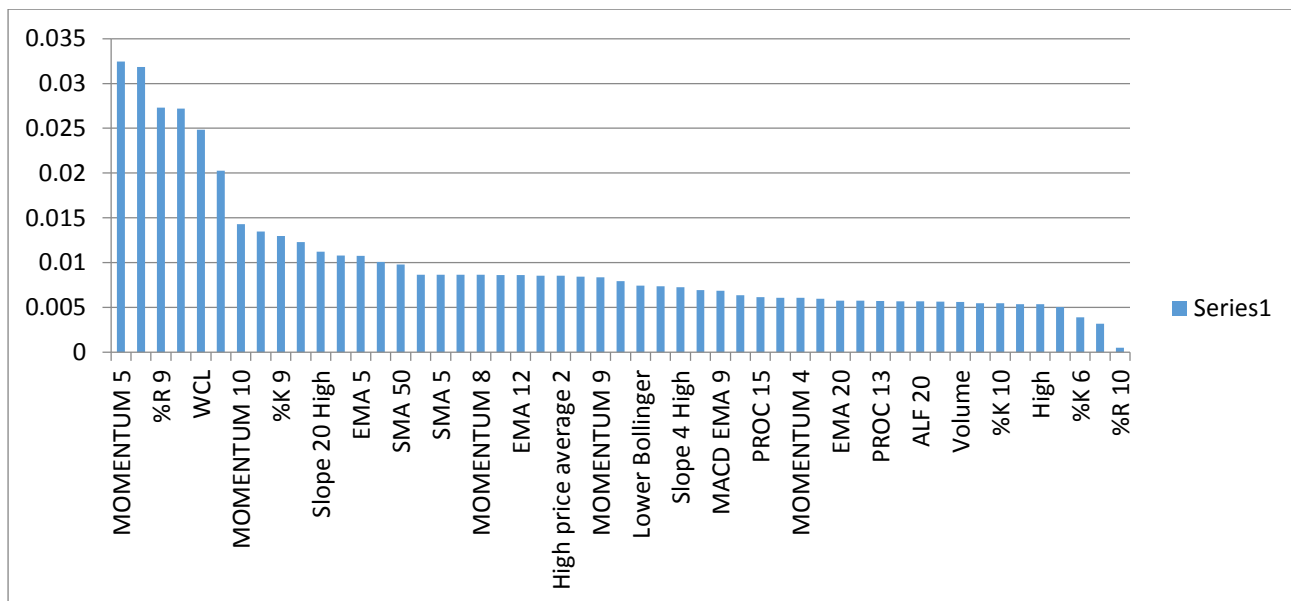
```

GainRatio ga = new GainRatio();
/* Apply the algorithm to the data set */
ga.build(featuresTraining);
/* Print out the score of each attribute */
for (int i = 0; i < ga.noAttributes(); i++) {
    System.out.println(ga.score(i));
}

```

Εικόνα 21 Υπολογισμός βαθμολογίας συσχέτισης

Στην Εικόνα 22 μπορούμε να διακρίνουμε την βαθμολογία (score) του κάθε δείκτη τεχνικής ανάλυσης σε σχέση με την έξοδο μας.



Εικόνα 22 Βαθμολογία συσχέτισης

Από το σύνολο των 51 μεταβλητών και δεικτών Τεχνικής Ανάλυσης επιλέξαμε τις δεκαπέντε (15) με τη μεγαλύτερη βαθμολογία, όπως αυτές παρουσιάζονται στον Πίνακα 4.

Low	MOMENTUM 10	PROC 12
SMA 50	%R 7	PROC 14
EMA 5	%R 9	WCL
EMA 50	%K 8	Low price average 2
MOMENTUM 5	%K 9	Slope 20 High

Πίνακας 4 Επιλογή από διαδικασία feature selectio

4.3.2 Βήμα 5^ο - Επιλογή Μεταβλητών (C και γ)

Εν συνεχεία ακολουθεί η διαδικασία επιλογής των δύο μεταβλητών C και γ του Αλγορίθμου SVM. Σε κάθε περίπτωση υλοποίησης ενός SVM Classifier συνιστάται η «greedy» αναζήτηση σε μεγάλη κλίμακα για την εύρεση των C και γ που βελτιστοποιούν την ακρίβεια του αλγορίθμου. Με απλά λόγια για την επιλογή των κατάλληλων τιμών για τις δύο εξεταζόμενες μεταβλητές είναι απαραίτητο η επανάληψη της διαδικασίας εκπαίδευσης και ελέγχου του SVM αλγορίθμου, συνδυάζοντας διάφορες τιμές των δύο μεταβλητών κάθε φορά σημειώνοντας ποιος συνδυασμός επιφέρει την καλύτερη απόδοση του αλγορίθμου.

Παρακάτω παρατίθεται ο Πίνακας 4 των τιμών C και gamma σε συνδυασμό με την ακρίβεια που πετύχαμε στο μοντέλο μας με κάθε συνδυασμό. Η επιλογή των

ανωτέρω μεταβλητών έγινε λαμβάνοντας υπόψη όλες τις παραμέτρους και όχι μόνο τις 15 που επιλέξαμε με την Επιλογή Χαρακτηριστικών.

C	gamma	ΑΚΡΙΒΕΙΑ ΣΕ TEST DATASET
0.1	0.001	0.6165
0.1	0.01	0.645
0.1	0.1	0.637
0.1	1	0.665
0.1	10	0.665
1	0.001	0.6395
1	0.01	0.6345
1	0.1	0.6345
1	1	0.672
1	10	0.6665
10	0.001	0.655
10	0.01	0.623
10	0.1	0.6545
10	1	0.6715
10	10	0.667
100	0.001	0.645
100	0.01	0.6275
100	0.1	0.6575
100	1	0.6685
100	10	0.667

Πίνακας 5 Ακρίβεια αλγορίθμου σε συνδυασμό με ζεύγη τιμών C και γ

4.3.3 Βήμα 6^ο -7^ο – Εκπαίδευση SVM model

Εν συνεχεία και αφού έχουμε επιλέξει τις ζητούμενες μεταβλητές C και gamma, χρησιμοποιώντας το ίδιο java application της παραγράφου 4.3.1 Και με την ίδια φιλοσοφία αποστέλλουμε στο API μόνο τις 15 ακολουθίες αριθμών που έχουν επιλέγει από τη διαδικασία Επιλογής χαρακτηριστικών. Σε αυτό το σημείο πρέπει να τονιστεί ότι το data set μας που αποτελείται πλέον από 13500 δείγματα, το χωρίζουμε σε Training και Testing με τιμές 11500 και 2000 αντίστοιχα. Αυτή η ενέργεια έχει ως σκοπό την εξέταση της ακρίβειας του Αλγορίθμου μας σε άγνωστα προς αυτό δείγματα.

Η εκπαίδευση λαμβάνει χώρα στο backend και ο κώδικας του Servlet που διαχειρίζεται την ανωτέρω διεργασία της εκπαίδευσης είναι στην Εικόνα 23.

```

//trainn SVM
svm_model model = runSVM(featuresTraining, labelTraining, fea-
turesTesting, labelTesting);
double accuracy = testSVM(featuresTesting, labelTesting ,model);
try {
    // Saving svm.model to datastore
    svm_save_model2("dd", model);
} catch (IOException e) {
    e.printStackTrace();
}

```

Εικόνα 23 Train SVM

Όπως παρατηρείται μετά την τελική εκπαίδευση (train) αποθηκεύεται το svm.model στο Blobstore. Σε αυτή την περίπτωση χρειάστηκε να τροποποιήσουμε τη native μέθοδο της βιβλιοθήκης `svm_save_model` έτσι ώστε να γίνεται αποθήκευση σε μορφή text (.txt) αρχείου στο Blobstore του GAE. Ανωτέρω αλλαγή φαίνεται στην Εικόνα 24.

```

// Get a file service
FileService fileService = FileServiceFactory.getFileService();
// Create a new Blob file with mime-type "text/plain"
AppEngineFile file = fileService.createNewBlobFile("text/plain");
// Open a channel to write to it
boolean lock = false;
FileWriteChannel writeChannel = fileService.openWriteChannel(file, true);
PrintWriter out = new PrintWriter(Channels.newWriter(writeChannel, "UTF8"));

```

Εικόνα 24 τροποποίησης μεθόδου `svm_save_model`

Σε αυτό το στάδιο πέραν της εκπαίδευσης του αλγορίθμου κρίνεται απαραίτητο να αποθηκεύσουμε και τις τιμές τις διακύμανσης και της μέσης τιμής με τις οποίες πραγματοποιήσαμε τη διαδικασία της κλιμάκωσης (Standariztion) όπως αναφέρθηκε στην παράγραφο 3.2.2στο backend. Οι τιμές αυτές είναι αναγκαίες για την κλιμάκωση των άγνωστων δεδομένων όπου θα αντλούμε σε καθημερινή βάση από το API της YAHOO και θα αποτελούν τα δεδομένα μας για την παραγωγή καθημερινών προβλέψεων.

Πέρα από τις τιμές mean και std πρέπει να αποθηκεύσουμε στο βάση δεδομένων (datatstore) του backend και τις τελευταίες τιμές του dataset καθώς το απαιτεί ο υπολογισμός κάποιων Δεικτών Τεχνικής Ανάλυσης για την εκτίμηση των

καθημερινών μεταβλητών με τις οποίες θα κάνουμε την εκάστοτε πρόβλεψη για τον επόμενο μήνα.

Η αποστολή των τιμών αυτών ακολουθεί την ίδια λογική με ένα παρόμοιο java application όπως αναφέρθηκε σε προηγούμενη παράγραφο.

4.3.4 Daily Indicators Calculations

Έχοντας αποθηκεύσει όλες τις τιμές που χρειαζόμαστε στο backend μπορούμε να περάσουμε στο επόμενο βήμα υλοποιώντας τις καθημερινές προβλέψεις.

Για αυτό αρχικά θα χρειαστούμε την καθημερινές τιμές

- DATE
- OPEN
- CLOSE
- HIGH
- LOW
- VOLUME

Για να τις αποκτήσουμε αρκεί μια κλήση στο API της YAHOO. Πιο συγκεκριμένα ένα Get HttpRequest στο URL της Εικόνας 25.

```
https://query.yahooapis.com/v1/public/yql?q=select%20*%20from%20yahoo.finance.historical
data%20where%20symbol%20%3D%20%22ETE.AT%22%20and%20startDate%20%3D%20%2220
15-01-06%22%20and%20endDate%20%3D%20%222015-01-
06%22&format=json&diagnostics=true&env=store%3A%2F%2Fdatatables.org%2Falltableswithk
eys&callback=
```

Εικόνα 25 URL Yahoo API για ημερήσιες τιμές

Στο ανωτέρω URL έχουμε επισημάνει τα σημεία όπου πρέπει να τοποθετηθούν, με τη σειρά που παρουσιάζονται, ο κωδικός της μετοχής, η ημερομηνία έναρξης και λήξης. Προφανώς οι δύο ημερομηνίες ταυτίζονται με την ημέρα όπου γίνεται η κλήση με σκοπό να πάρουμε τις ημερήσιες τιμές.

Η απάντηση στην παραπάνω κλήση είναι ένα JSONObject το οποίο όπως μπορείτε να δείτε στην Εικόνα 26 περιλαμβάνει τις ζητούμενες τιμές.

```
..... results": {"quote": {"Symbol": "ETE.AT", "Date": "2015-01-06", "Open": "1.37", "High": "1.37", "Low": "1.37", "Close": "1.37", "Volume": "000", "Adj_Close": "1.37" }}}}
```

Εικόνα 26 Απάντηση από Yahoo Finance API

Αφού αποκτήσουμε τις ανωτέρω τιμές κάνοντας κατάλληλο parse το JSONObject τις αποθηκεύουμε στο datastore. Χρησιμοποιώντας τόσο τις τιμές που αναφέραμε όσο και αυτές των προηγούμενων ημερών τι οποίες ομοίως είχαμε αποθηκεύσει στη βάση δεδομένων, υπολογίζουμε τους δείκτες Τεχνικής Ανάλυσης της πιο πρόσφατης ημέρας. Ο υπολογισμός αυτός υλοποιήθηκε εξ ολοκλήρου μέσω της βιβλιοθήκης java TA.

Επόμενο βήμα που ακολουθεί είναι να πραγματοποιήσουμε κλιμάκωση στα νέα δεδομένα έτσι ώστε να είναι αποδεκτά ως είσοδοι, από τον Αλγόριθμο μας. Η κλιμάκωση τύπου Standardization πραγματοποιείται με τη χρήση των τιμών mean και std που έχουμε αποθηκεύσει, όπως αναφέρεται σε Παράγραφο 4.3.3..

Τέλος ανακτώντας τόσο τα κατάλληλα δεδομένα από τη βάση και με τη χρήση του svm_model που είχαμε αποθηκεύσει στο blobstore (βλ. Παράγραφο 4.3.3) μπορούμε να έχουμε την πρόβλεψη μας για τον επόμενο μήνα όπως φαίνεται στον κώδικα της Εικόνας 27.

```

public class DailyPredictionServlet extends HttpServlet {
    Key stockKey;
    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse response) throws
ServletException {
        stockKey = DatastoreHelper.getStockKeyFromDatastore("ete");
        Text path = DatastoreHelper.getSVMModelPathFromDatastore(stockKey,
PropertyConstants.PATH);
        svm_model model=null;
        try {
            //get svm model from blobstore
            model = SVMHelper.load_model(path);
        } catch (IOException e) {
            e.printStackTrace();
        }
        double prediction;
        if (model!=null) {
            //calculate prediction
            prediction = predictionResult(model);
            //save prediction result to datastore
            DatastoreHelper.savePredictionToDatastore(stockKey, prediction);
        }
    }

    private double predictionResult(svm_model model){
        String property[] =
{PropertyConstants.LOW,PropertyConstants.SMA_50,PropertyConstants.EMA_5,PropertyConstan
ts.EMA_50
,PropertyConstants.MOMENTUM_5,PropertyConstants.MOMENTUM_10,PropertyConstants.R_7,Prope
rtyConstants.R_9,
PropertyConstants.K_8,PropertyConstants.K_9,PropertyConstants.PROC_12,PropertyConstants
.PROC_14,PropertyConstants.WCL,
PropertyConstants.LOWPRICEAVG_2,PropertyConstants.SLOPE_20_H};
        svm_node[] x=new svm_node[property.length];
        for(int i=0;i<property.length;i++){
            x[i]=new svm_node();
            x[i].index = i;
            x[i].value =
NormalizeDailyValuesHelper.normalizeValue(stockKey,property[i]);
        }
        double prediction = svm.svm_predict(model,x);
        return prediction;
    }
}

```

Εικόνα 27

Όλες οι ενέργειες που περιγράφηκαν στην παρούσα παράγραφο πρέπει να γίνονται κάθε μέρα. Για τον παραπάνω λόγο αντί για να καλούμε τα συγκεκριμένα Servlets με χρήση κάποιου timer, υλοποιήσαμε τη λειτουργία αυτή με τη χρήση των cron jobs που παρέχει το Google App Engine όπως φαίνεται στην Εικόνα 28. Συγκεκριμένα οι διεργασίες που λαμβάνουν χώρα συνοπτικά είναι οι κατωτέρω:

- ▶ Cron Job – Every day at 19:00
- ▶ Java - TA lib

- ▶ Feature Scaling (χρησιμοποιώντας mean και std που έχουμε αποθηκεύσει)
- ▶ Prediction every day Cron job at 19:30
- ▶ Load svm model from Blobstore for prediction

Application: unipi-makridis [High Replication] Try the new Cron Jobs page.

Main

- [Dashboard](#)
- [Instances](#)
- [Logs](#)
- [Versions](#)
- [Backends](#)
- [Cron Jobs](#)**
- [Task Queues](#)
- [Quota Details](#)

Data

- [Datastore Indexes](#)
- [Datastore Viewer](#)
- [Datastore Statistics](#)
- [Blob Viewer](#)
- [Prospective Search](#)

Below are scheduled tasks (cron jobs) for the application. Cron jobs are defined in [cron](#).
Learn more about [cron](#).

< Prev 10 Next 10 >

Cron Job	Schedule/Last Run/Last Status
/dailycalculateindicators Calculate and save indicators	every day 19:00 (Europe/Athens) 2015/03/07 19:00:00 on time Su
/dailyprediction Prediction	every day 19:30 (Europe/Athens) 2015/03/07 19:30:00 on time Su
/dailystockvalues Update new values	every day 18:00 (Europe/Athens) 2015/03/07 18:00:00 on time Fai

Εικόνα 28 Cron Jobs σε GAE

4.4 Android application

Όπως μπορεί να καταλάβει κάποιος, το μεγαλύτερο κομμάτι της εφαρμογής δε θα μπορούσε να υλοποιηθεί σε ένα κινητό (σε μια εφαρμογή android). Ο λόγος φυσικά είναι οι σχετικά περιορισμένες υπολογιστικές δυνατότητες που έχουν οι συγκεκριμένες συσκευές. Οι ανωτέρω τεχνικοί περιορισμοί μας οδήγησαν στην επιλογή της απλής απεικόνισης του αποτελέσματος σε εφαρμογή της πλατφόρμας android. Για να επιτευχθεί αυτό μας αρκεί μια κλήση Http Get στο backend μας όπως φαίνεται σε Εικόνα 28. Η ανωτέρω κλήση μας επιστρέφει το αποτέλεσμα της τελευταίας πρόβλεψης μαζί με την ημερομηνία που πραγματοποιήθηκε.

```

public interface Api {

    public static final String ROOT_URL = "http://unipi-makridis.appspot.com";
    public static final String GET_PREDICTION_URL = "/getlastprediction";

    @GET(GET_PREDICTION_URL )
    void searchBooksByTitle(Callback<PredictionResponse> callback);

}

```

Εικόνα 29 Κλήση για λήψη τελευταίας πρόβλεψης

Στα πλαίσια της απαίτησης μας η εφαρμογή να γίνει πιο ελκυστική στον χρήστη προσθέσαμε μια επιπλέον λειτουργία. Έτσι πέρα από την τελευταία πρόβλεψη, ο χρήστης έχει τη δυνατότητα να επιλέξει ανάμεσα σε δέκα από τους πιο δημοφιλείς δείκτες Τεχνικής Ανάλυσης και να δει τα διαγράμματα τους, όπως φαίνεται στην Εικόνα 25 .Η υλοποίηση αυτή έγινε και πάλι με χρήση του API της yahoo.

```

clickMe.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        Toast.makeText(context, "The button is clicked",
        Toast.LENGTH_LONG).show();

        String ta1 =
        YahooURL.getTA1(checkboxSMA50,checkboxSMA200,checkboxEMA50,checkboxEMA200,checkboxBooLi
        ngerBands,
                checkboxPSAR,checkboxSplits,checkboxVolume);
        String ta2 =
        YahooURL.getTA2(checkboxMACD,checkboxMFI,checkboxFastStoch,checkboxSlowStoch,checkboxWi
        ll R,
                checkboxRSI);

        imageUrl = YahooURL.createURL(stockName,"6","1","on","1",ta1,ta2);
        //Loading image from below url into imageView
        Picasso.with(context)
                .load(imageURL)
                .into(imageView);
    }
});

```

Εικόνα 30

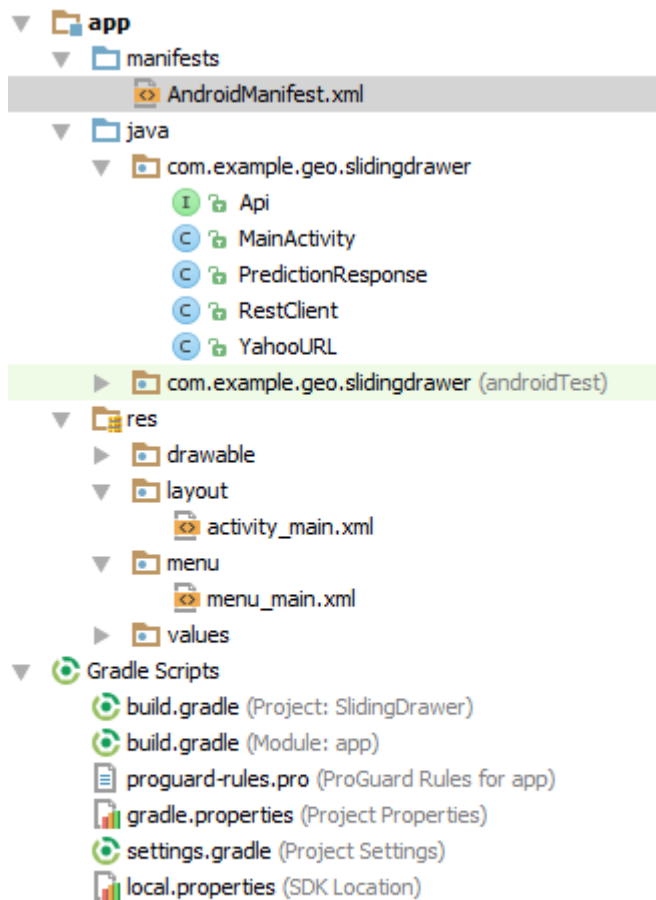


Εικόνα 31 Android app Layout

4.4.1 Κατασκευή εφαρμογής Android

Στην παρούσα ενότητα θα επεξηγήσουμε αναλυτικότερα τα αρχεία που περιλαμβάνονται στην εφαρμογή σε περιβάλλον android στα πλαίσια της εργασίας, παραλείποντας τη διαδικασία δημιουργίας νέου android project για λόγους οικονομίας. Όπως αναφέραμε σε προηγούμενη παράγραφο για τόσο για την android εφαρμογή όσο και για το backend, χρησιμοποιήσαμε το IDE Android Studio. Στην Εικόνα 32 μπορούμε να αποκτήσουμε μια γενική εικόνα από τα αρχεία της εφαρμογής μας τα οποία και θα αναλύσουμε στη συνέχεια. Αυτά είναι

- Gradle
- Manifest.xml
- Api
- MainActivity
- PredictionResponse
- RestClient
- YahooURL
- activity_main.xml



Εικόνα 32 Δέντρο των κλάσεων και αρχείων του app

Αρχείο Gradle

Στο Gradle, μπορούμε να παρατηρήσουμε διάφορα στοιχεία από τα οποία τα περισσότερα δημιουργούνται αυτόματα κατά την επιλογή δημιουργίας ενός νέου project. Σε αυτά περιλαμβάνονται το `minSdkVersion=15`, `targetSdkVersion=21` τα οποία αφορούν ποιες συσκευές (version android) θέλουμε να είναι συμβατές με την εφαρμογή μας. Το σημαντικότερο τμήμα του Gradle είναι ο ορισμός των εξωτερικών βιβλιοθηκών που θα κάνουμε χρήση. Στην περίπτωση μας χρησιμοποιούμε τις εξής:

```
'com.squareup.retrofit:retrofit:1.6.1'
```

```
'com.squareup.okhttp:okhttp-urlconnection:2.0.0'
```

```
'com.squareup.okhttp:okhttp:2.0.0'
```

```
'com.squareup.okio:okio:1.0.1'
```

```
'com.squareup.picasso:picasso:2.3.2'
```

Η διαφορά είναι πως πλέον δε χρειάζεται να κατεβάσουμε και να εισάγουμε τα αντίστοιχα jar αρχεία, καθώς το Gradle αυτόματα κάνει αυτή την εργασία για εμάς. Απαιτείται βέβαια σύνδεση στο διαδίκτυο για να ολοκληρωθεί η διεργασία αυτή.

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 21
    buildToolsVersion "21.1.2"

    defaultConfig {
        applicationId "com.example.geo.slidingdrawer"
        minSdkVersion 15
        targetSdkVersion 21
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'),
'proguard-rules.pro'
        }
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile 'com.android.support:appcompat-v7:21.0.3'
    compile 'com.squareup.retrofit:retrofit:1.6.1'
    compile 'com.squareup.okhttp:okhttp-urlconnection:2.0.0'
    compile 'com.squareup.okhttp:okhttp:2.0.0'
    compile 'com.squareup.okio:okio:1.0.1'
    compile 'com.squareup.picasso:picasso:2.3.2'
}
```

Αρχείο Manifest.xml

όπως αναφέρθηκε και στο 3.2 το αρχείο Manifest περιλαμβάνει όλες τις σημαντικότερες πληροφορίες της εφαρμογής για χρήση από το λειτουργικό σύστημα.

Μπορούμε να παρατηρήσουμε

`<uses-permission android:name="android.permission.INTERNET"/>`, με το οποίο ουσιαστικά ζητάει την άδεια η εφαρμογή να χρησιμοποιήσει τον πόρο της συσκευής που της παρέχει πρόσβαση στο internet. Επιπλέον των ανωτέρω μια άλλη πληροφορία που μπορούμε να αντλήσουμε είναι ότι στην εφαρμογή μας τρέχει μόνο ένα activity, το MainActivity.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.geo.slidingdrawer" >
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name"
            android:screenOrientation="landscape">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Το αρχείο activity_main.xml

Γενικά στα αρχεία .xml που βρίσκονται στην τοποθεσία .res/layout τοποθετούνται και διατάσσονται κατάλληλα τα διάφορα views που θα χρησιμοποιηθούν για την δημιουργία του γραφικού περιβάλλοντος της εφαρμογής. Στην περίπτωση μας έχουμε χρησιμοποιήσει τόσο RelativeLayout όσο και LinearLayout εμφωλευμένο στο πρώτο. Επιπλέον χρησιμοποιήσαμε views όπως κουμπιά (buttons), επιλογείς (checkboxes) και πεδία κειμένου (textviews). Έχουμε παραλείψει εσκεμμένα κάποιο τμήμα του αρχείου το οποίο επαναλαμβανόταν και δεν παρείχε κάποια πληροφορία. Παρατηρούμε ότι σε κάθε view εκχωρείται και ένα id (π.χ. `android:id="@+id/handle"`) το οποίο είναι μοναδικό για κάθε view και αυτό για να μπορούμε να καλέσουμε και να επεξεργαστούμε την λειτουργία κάποιων από αυτά.

Τέλος να επισημάνουμε πως το αρχείο θα πρέπει να γίνει inflate από κάποιο Activity, όπως θα δούμε στην πορεία.

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/text1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="For more info drag the button..."
        android:textColor="@color/red"
        android:textStyle="bold"
        android:textSize="18sp"/>

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/imageView"
        android:layout_alignTop="@+id/slidingDrawer"
        android:paddingLeft="120dp">
</ImageView>

    <SlidingDrawer
        android:id="@+id/slidingDrawer"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="30dp"
        android:content="@+id/content"
        android:handle="@+id/handle"
        android:orientation="vertical">

        <Button
            android:id="@+id/handle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="open TA Charts"
            android:textColor="@color/red"
            android:textStyle="bold"
            android:textSize="18sp"/>

        <LinearLayout
            android:id="@+id/content"
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:orientation="horizontal"
            android:background="@color/primary_dark">

            <LinearLayout
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:orientation="vertical" >

```

```

<TextView
    android:id="@+id/text2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Choose TA Indicators"
    android:textColor="@color/red"
    android:textStyle="bold"
    android:textSize="18sp"/>

<CheckBox
    android:id="@+id/checkboxSMA50"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="SMA50"
    android:checked="true"
    android:textColor="@color/red"
    android:textStyle="bold"
    android:textSize="18sp"
    android:paddingTop="20dp"/>
<CheckBox
    android:id="@+id/checkboxSMA200"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="SMA200"
    android:checked="false"
    android:textColor="@color/red"
    android:textStyle="bold"
    android:textSize="18sp"/>
<CheckBox
    android:id="@+id/checkboxEMA50"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="EMA50"
    android:checked="false"
    android:textColor="@color/red"
    android:textStyle="bold"
    android:textSize="18sp"/>
    -
    -
    -
    -

<Button
    android:id="@+id/click"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="update TA chart"
    android:textColor="@color/red"
    android:textStyle="bold"
    android:textSize="18sp"/>
</LinearLayout>
</LinearLayout>
</SlidingDrawer>

</RelativeLayout>

```

Το αρχείο MainActivity

Στη MainActivity όπως σε κάθε Activity πρέπει αρχικά, όπως αναφέραμε στην προηγούμενη παράγραφο, να κάνουμε inflate το αντίστοιχο layout. Στην περίπτωση μας έχουμε το activity_main.xml όπως φαίνεται παρακάτω:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
}
```

Στη συνέχεια ανακτούμε τα διάφορα views που περιλαμβάνονται στο layout με σκοπό να καθορίσουμε τις διάφορες λειτουργίες τους. Η μέθοδος που καλούμε να υλοποίηση την ανωτέρω ενέργεια είναι η *findViewById(int id)* η οποία δέχεται σαν όρισμα ένα id, όπως είδαμε και όταν αναλύσαμε το layout, κάθε view εκχωρείται ένα μοναδικό id, και επιστρέφει το αντίστοιχο view.

```
handle = (Button) findViewById(R.id.handle);
text1 = (TextView) findViewById(R.id.text1);
clickMe = (Button) findViewById(R.id.click);
drawer=(SlidingDrawer) findViewById(R.id.slidingDrawer);
imageView = (ImageView) findViewById(R.id.imageView);
checkboxSMA50 = (CheckBox) findViewById(R.id.checkboxSMA50);
checkboxSMA200 = (CheckBox) findViewById(R.id.checkboxSMA200);
checkboxEMA50 = (CheckBox) findViewById(R.id.checkboxEMA50);
checkboxEMA200 = (CheckBox) findViewById(R.id.checkboxEMA200);
checkboxBoolingerBands = (CheckBox) findViewById(R.id.checkboxBoolingerBands);
checkboxPSAR = (CheckBox) findViewById(R.id.checkboxP_SAR);
checkboxSplits = (CheckBox) findViewById(R.id.checkboxSplits);
checkboxVolume = (CheckBox) findViewById(R.id.checkboxVolume);
checkboxMACD = (CheckBox) findViewById(R.id.checkboxMACD);
checkboxMFI = (CheckBox) findViewById(R.id.checkboxMFI);
checkboxFastStoch = (CheckBox) findViewById(R.id.checkboxFAST_STOCH);
checkboxSlowStoch = (CheckBox) findViewById(R.id.checkboxSLOW_STOCH);
checkboxWill_R = (CheckBox) findViewById(R.id.checkboxWILL_R);
checkboxRSI = (CheckBox) findViewById(R.id.checkboxRSI);
```

Στη συνέχεια θα πρέπει να αναφέρουμε για μία ακόμη φορά τη χρήση του Retrofit Rest Client, βάση του οποίου πραγματοποιήσαμε όλες τις απαραίτητες κλήσεις

(HTTP Requests). Όπως παρατηρούμε έχουμε κάνει compile στο gradle τις αντίστοιχες βιβλιοθήκες.

```
compile 'com.squareup.retrofit:retrofit:1.6.1'
compile 'com.squareup.okhttp:okhttp-urlconnection:2.0.0'
compile 'com.squareup.okhttp:okhttp:2.0.0'
compile 'com.squareup.okio:okio:1.0.1'

compile 'com.squareup.picasso:picasso:2.3.2'
```

Η χρήση του RestClient φαίνεται παρακάτω:

```
RestClient.get().searchBooksByTitle(new Callback<PredictionResponse>() {

    @Override
    public void success(PredictionResponse predictionResponse, Response
response) {
        String prediction = getPrediction(predictionResponse);
        Toast.makeText(context, "To response girnaei", Toast.LENGTH_LONG).show
();
        text1.setText("Prediction for " + predictionResponse.getDay() + " is " +
prediction);
    }

    @Override
    public void failure(RetrofitError error) {

    }

});
```

Όπως παρατηρούμε είναι μια ασύγχρονη κλήση. Έχει δύο δυνατές καταστάσεις την success και την failure, με εύκολα κατανοητή ερμηνεία. Φαίνεται λοιπόν πως όταν έχουμε επιτυχή επιστροφή στην κλήση μας (response code = 200) και αφού μετατρέψουμε την απάντηση από int σε String καλώντας τη μέθοδο String prediction = getPrediction(predictionResponse); εμφανίζουμε το αποτέλεσμα σε ένα TextView.

Ακολούθως η διαδικασία κλήσης στο API της YAHOO με σκοπό την εμφάνιση των γραφημάτων Τεχνικής Ανάλυσης διαφόρων δεικτών γίνεται με τον ίδιο ακριβώς τρόπο, με τη μόνη διαφορά ότι χρησιμοποιούμε τον ImageLoader της βιβλιοθήκης αυτής.


```

clickMe.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        Toast.makeText(context, "The button is clicked",
Toast.LENGTH_LONG).show();

        String ta1 = YahooURL.getTA1
(checkboxSMA50, checkboxSMA200, checkboxEMA50, checkboxEMA200, checkboxBooLingerBan
ds,
        checkboxPSAR, checkboxSplits, checkboxVolume);
        String ta2 = YahooURL.getTA2
(checkboxMACD, checkboxMFI, checkboxFastStoch, checkboxSlowStoch, checkboxWill_R,
        checkboxRSI);

        imageUrl = YahooURL.createURL(stockName, "6", "1", "on", "1", ta1, ta2);
        //Loading image from below url into imageView
        Picasso.with(context)
            .load(imageURL)
            .into(imageView);
    }
});

```

Το αρχείο Api

Τα υπόλοιπα αρχεία είναι ουσιαστικά βοηθητικά για την πραγματοποίηση των δύο κλήσεων όπως αυτές αναφέρθηκαν νωρίτερα. Οι πληροφορίες που αντλούμε από αυτά είναι κυρίως τα URL στα οποία γίνονται οι κλήσεις π.χ. στο backend του GAE το URL όπως φαίνεται στο αρχείο Api είναι

<http://unipi-makridis.appspot.com/getlastprediction>

```

public interface Api {

    public static final String ROOT_URL = "http://unipi-makridis.appspot.com";
    public static final String GET_PREDICTION_URL = "/getlastprediction";

    @GET(GET_PREDICTION_URL)
    void searchBooksByTitle(Callback<PredictionResponse> callback);
}

```

Το αρχείο RestClient

Τα αρχεία RestClinet είναι ουσιαστικά η καρδιά του συστήματος κλήσεων του Retrofit.

```
public class RestClient {
    private static Api REST_CLIENT;
    private static String ROOT = Api.ROOT_URL;

    static {
        setupRestClient();
    }

    private RestClient() {}

    public static Api get() {
        return REST_CLIENT;
    }

    private static void setupRestClient() {
        RestAdapter.Builder builder = new RestAdapter.Builder()
            .setEndpoint(ROOT)
            .setClient(new OkClient(new OkHttpClient()))
            .setLogLevel(RestAdapter.LogLevel.FULL);

        RestAdapter restAdapter = builder.build();
        REST_CLIENT = restAdapter.create(Api.class);
    }
}
```

Το αρχείο YahooURL

Η κλάση YahooURL είναι καθαρά βοηθητική ουσιαστικά για την δημιουργία του URL όπου θα γίνει η κλήση με τις κατάλληλες παραμέτρους στο API της YAHOO με σκοπό να εμφανίσουμε τις ζητούμενες εικόνες Διαγραμμάτων Τεχνικών Δεικτών.

```

public static final String YAHOO_QUERY_BASE_URL = "http://chart.finance.yahoo.co
    public static final String STOCK_NAME_PREFIX = "s=";
    public static final String TIME_SPAN_PREFIX = "&t=";
    public static final String TYPE_PREFIX = "&q=";
    public static final String SCLAE_PEFIX = "&l=";
    public static final String SIZE_PREFIX = "&z=";
    public static final String TA_1_PREFIX = "&p=";
    public static final String TA_2_PREFIX = "&a=";
    public static final String NO_DATA = "no data";

    public static String createURL(String stockName, String timeSpan, String typ
scale, String size, String ta1, String ta2) {
        String URL = YAHOO_QUERY_BASE_URL + STOCK_NAME_PREFIX + stockName
            + TIME_SPAN_PREFIX + timeSpan + TYPE_PREFIX + type
            + SCLAE_PEFIX + scale + SIZE_PREFIX + size+TA_1_PREFIX+ta1
+TA_2_PREFIX+ta2;

        return URL;
    }

    public static String getTA1(CheckBox checkboxSMA50,CheckBox checkboxSMA200,
checkboxEMA50,CheckBox checkboxEMA200,CheckBox checkboxBoolingerBands, CheckBox
checkboxPSAR, CheckBox checkboxSplits, CheckBox checkboxVolume ){
        String ta1 = "";
        if (checkboxSMA50.isChecked()){
            ta1="m50";
        }
        if (checkboxSMA200.isChecked()){
            ta1=ta1+",m200";
        }
        if (checkboxEMA50.isChecked()){
            ta1=ta1+",e50";
        }
        if (checkboxEMA200.isChecked()){
            ta1=ta1+",e200";
        }
        if (checkboxBoolingerBands.isChecked()){
            ta1=ta1+",b";
        }
        if (checkboxPSAR.isChecked()){
            ta1=ta1+",p";
        }
        if (checkboxSplits.isChecked()){
            ta1=ta1+",s";
        }
        if (checkboxVolume.isChecked()){
            ta1=ta1+",v";
        }
        return ta1;
    }
}

```

```
public static String getTA2(CheckBox checkboxMACD, CheckBox checkboxMFI, Che
checkboxFastStoch, CheckBox checkboxSlowStoch, CheckBox checkboxWill_R,
                           CheckBox checkboxRSI){
    String ta2="";
    if (checkboxMACD.isChecked()){
        ta2="m26-12-9";
    }
    if (checkboxMFI.isChecked()){
        ta2=ta2+",f14";
    }
    if (checkboxFastStoch.isChecked()){
        ta2=ta2+",fs";
    }
    if (checkboxSlowStoch.isChecked()){
        ta2=ta2+",ss";
    }
    if (checkboxWill_R.isChecked()){
        ta2=ta2+",w14";
    }
    if (checkboxRSI.isChecked()){
        ta2=ta2+",r14";
    }
    return ta2;
}
```

Το αρχείο PredictionResponse

Το συγκεκριμένο αρχείο είναι μια απλή κλάση αντικειμένου της Java βάσει της οποίας το Retrofit κάνει parse την απάντηση της κλήσης μας από το Backend μας.

```
public class PredictionResponse {

    @SerializedName("id")
    private String id;

    @SerializedName("day")
    private String day;

    @SerializedName("prediction")
    private double prediction;

    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getDay() {
        return day;
    }
    public void setDay(String day) {
        this.day = day;
    }
    public double getPrediction() {
        return prediction;
    }
    public void setPrediction(double prediction) {
        this.prediction = prediction;
    }
}
```

Κεφάλαιο 5ο

Συμπεράσματα – Προτάσεις

5.1 Συμπεράσματα

Γενικά

Τα αποτελέσματα από αυτή την εργασία μας οδηγούν στο συμπέρασμα ότι οι μετοχές και η αγορά που μελετήθηκε παρουσιάζουν efficient συμπεριφορά. Καμία στρατηγική εμπορικών συναλλαγών βασισμένη στο μελετώμενο σύστημα δεν είναι σε θέση να ξεπεράσει την ομοιόμορφη buy-and-hold στρατηγική.

Οι μέθοδοι μηχανικής μάθησης που χρησιμοποιούνται στην παρούσα διατριβή, δείχνουν κάποια προβλεπτική ικανότητα, ιδίως SVM και SVR. Δυστυχώς, οποιαδήποτε κέρδη από ένα σύστημα προβλέψεων με ακρίβεια 65-70% μπορεί να αποτύχουν λόγω κυρίως του κόστους συναλλαγής. Οι αλγόριθμοι Μηχανικής Μάθησης δείχνουν στοιχεία τα οποία ίσως σε μεγαλύτερο αριθμό δεδομένων καταστούν ικανά να προβλέψουν με μεγαλύτερη ακρίβεια τις κινήσεις των μετοχών.

Οικονομική Κρίση

Η εργασία έχει έντονο οικονομικό χαρακτήρα, οπότε είναι αναπόφευκτη μια αναφορά στην οικονομική κρίση που μαστίζει τη σημερινή ελληνική κοινωνία. Η κρίση έχει δημιουργήσει τεράστια προβλήματα στη χώρα και η λέξη χρεοκοπία ακούγεται καθημερινά. Πολλές αναλύσεις έχουν επισημάνει τους παράγοντες που οδήγησαν σε αυτή, όπως το τεράστιο δημόσιο χρέος, τα υψηλά δημοσιονομικά ελλείμματα, την ανορθολογική διαχείριση των δημόσιων δαπανών κ.α.

Επιγραμματικά τα στάδια της Ελληνικής Οικονομίας στην πορεία προς την σημερινή κρίση είναι τα παρακάτω:

Το έλλειμμα πέφτει μέχρι το 1999 κάτω από 3%, πετυχαίνοντας τελικά η Ελλάδα να γίνει μέλος της ΟΝΕ.

Στα τέλη 2004, αποκαλύπτονται αποκρύψεις δαπανών με αποτέλεσμα την αναθεώρηση των ελλειμμάτων των προηγούμενων ετών, τη μείωση της αξιοπιστίας της χώρας και την τριετή επιτήρηση από την Ε.Ε.

Από τα τέλη του 2008 όμως, λόγω της παγκόσμιας οικονομικής κρίσης, η ελληνική οικονομία εκτροχιάζεται και το έλλειμμα, αλλά και το χρέος αρχίζουν να αυξάνονται με γρήγορους ρυθμούς.

Οκτώβριος,2009- Ιανουάριος,2010: ανακοινώνεται ότι το έλλειμμα για το 2009 θα εκτιναχτεί στο 12,5% από 6% που εκτιμούσε η προηγούμενη κυβέρνηση (το 2010 αναθεωρήθηκε στο 15,4%) με αποτέλεσμα οι διεθνείς οίκοι αξιολόγησης να υποβαθμίζουν την Ελληνική οικονομία.

Από τον Ιανουάριο του 2010 συνεχής άνοδος των spreads, ξεπερνώντας ακόμα και τις 1000 μονάδες τον Απρίλιο του 2010.

Φεβρουάριος, 2010: Πρώτο πακέτο οικονομικών μέτρων: Η κυβέρνηση ανακοίνωσε μέτρα για τον δημόσιο τομέα που περιλάμβαναν: πάγωμα μισθών, περικοπές επιδομάτων, υπερωριών και οδοιορικών.

Μάρτιος, 2010: Δεύτερο πακέτο μέτρων. Λήψη νέων σκληρών μέτρων, όπως: Μειώσεις 30% δώρων, 12% επιδομάτων Δημοσίου, 7% αποδοχών υπαλλήλων ΔΕΚΟ, ΟΤΑ, ΝΠΙΔ, αυξήσεις ΦΠΑ και φόρου βενζίνης κλπ

Απρίλιος-Μάιος, 2010: Μνημόνιο και Τρίτο πακέτο οικονομικών μέτρων: Η Ελλάδα δεν κατάφερε να βελτιώσει την θέση της στις διεθνείς αγορές, παρά την λήψη των μέτρων.

23 Απριλίου ο ανακοινώνεται η προσφυγή της χώρας στο μηχανισμό στήριξης (ONE,EKT,ΔΝΤ) και τη λήψη νέων μέτρων.

Είναι πλέον βέβαιο ότι η οικονομία της χώρας είναι σε άσχημη κατάσταση κάτι που αντικατοπτρίζεται έντονα και στο ΧΑΑ. Στην παρούσα εργασία το dataset που χρησιμοποιήσαμε περιλαμβάνει τιμές από το 2000 έως το έτος 2015. Αναπόφευκτα έχουμε συμπεριλάβει τιμές των μετοχών τόσο από το διάστημα 2000-2008-9 όσο και μετά από αυτό. Φάνηκε αρκετά ενδιαφέρον να δούμε τη συμπεριφορά του συστήματος μας ξεχωριστά στις δύο αυτές περιόδους. Σαν σημείο ορόσημο επιλέχτηκε η 23 Απριλίου 2010. Βάση των ανωτέρω τροφοδοτήσαμε το σύστημα μας με τα νέα dataset και προέκυψαν τα κατωτέρω αποτελέσματα:

	Περίοδος πριν την κρίση 2000-2010	Περίοδος κρίσης 2010- 2015
ακρίβεια πρόβλεψης σε training set	69.191489	63.7894
ακρίβεια πρόβλεψης σε test set	62.8	61.2

Η αρχική μας υπόθεση φαίνεται να επιβεβαιώνεται καθώς σημειώνεται μεγαλύτερο ποσοστό σωστών προβλέψεων στην περίοδο 2000-2010, καθώς αναμενόταν μεγαλύτερη ακρίβεια (καλύτερη απόδοση) σε ένα πιο ομοιογενές δείγμα και ποιο σταθερό, όπως αυτό της περιόδου 2000-2010, καθώς την πρόσφατη περίοδο (περίοδος έντονης κρίσης) που διανύουμε λαμβάνουν χώρα μεταβολές που δεν ακολουθούν συγκεκριμένους κανόνες και ενδεχομένως δεν μπορεί να τις «προβλέψει» κάποιος Τεχνικός Αναλυτής .

Παρατηρείται επιπλέον, πως την περίοδο 2000-2010 η ακρίβεια του συστήματος μας δεν ξεπερνάει αυτή που υπολογίσαμε στην περίοδο 2000-1015. Αυτό μπορεί να οφείλεται εν μέρει στην «ασταθή» οικονομική περίοδο 2008-2010 ή στον μικρότερο αριθμό δεδομένων – λόγος στον οποίο οφείλεται και η καλύτερη προσέγγιση των γνωστών στο σύστημα δεδομένων (training set).

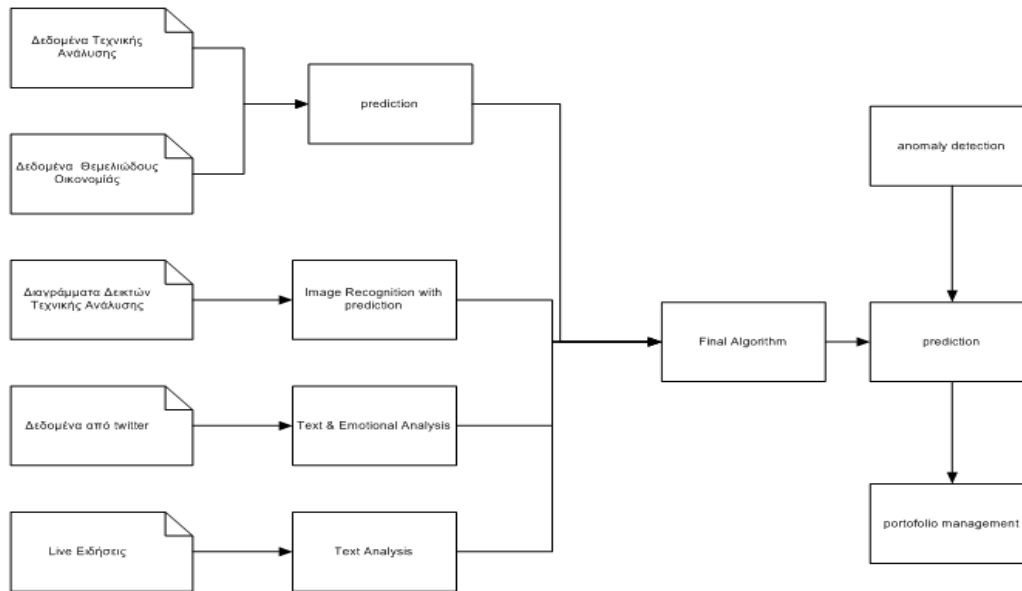
Παρόλα αυτά αν αναλογιστούμε πως σε εποχές, τόσο οικονομικής κρίσης όσο και οικονομικής σταθερότητας παρατηρούνται φαινόμενα έντονων – ακανόνιστων διακυμάνσεων και συμπεριφορές στις κινήσεις των μετοχών χωρίς αυτό να επιβάλλει αλλαγή στον τρόπο χρήσης της Τεχνικής Ανάλυσης. Επίσης θα πρέπει να σημειωθεί πως οι οποιοσδήποτε «απροσδόκητες» μεταβολές (συνήθως οφείλονται σε ειδήσεις) δεν συμβαίνουν τακτικά και προγραμματισμένα αλλά τυχαία. Σε αυτή την περίπτωση δύναται να αντιμετωπιστεί σαν λευκός θόρυβος ακολουθώντας κατανομή Gauss χωρίς να επηρεάζει εμφανώς το σύστημα μας.

Υλοποίηση

Συμπερασματικά αναφέρουμε τη συγκυριακή δυσκολία να κάνουμε host τη διαδικασία εκπαίδευσης του συστήματος σε ένα compute cloud και η «αναγκαστική» επιλογή να εργαστούμε στο πλαίσιο της παρούσης με το PaaS της Google το Google AppEngine. Εκτιμάται ως μια πολύ καλή επιλογή, τόσο για frontend όσο και backend υπηρεσίες για κάποιον που κάνει τα πρώτα του βήματα σε αυτό τον τομέα. Οσον αφορά τους χρόνους υλοποίησης των εργασιών παρουσιάζονται προσεγγιστικά παρακάτω .

Εργασία	Χρόνος Ολοκλήρωσης
Training and save svm_model	7500ms
Feature Selection	5000ms
Daily TA Index Calculation	2000ms

Τόσο τεχνικά όσο και θεωρητικά οι μελλοντικές εργασίες θα μπορούσαν να εξελίξουν την παρούσα με τρόπο που παρουσιάζεται στην Εικόνα 33.



Εικόνα 33 Ολοκληρωμένο Αυτοματοποιημένο Σύστημα Προβλέψεων

Διαφαίνεται ότι οι μεγάλες εταιρίες έχουν επικεντρωθεί στην προσπάθεια πρόβλεψης στον τομέα βασιζόμενες σε στοιχεία από κοινωνικά δίκτυα, όπως το twitter ή γενικότερα κάνοντας text analysis σε ειδήσεις. Αυτό θα μπορούσε να χρησιμοποιηθεί ως συναισθηματική ανάλυση των επενδυτών καθιστώντας τη πρόβλεψη πιο ακριβή. Φυσικά μια παρόμοια μέθοδος με αυτή που εφαρμόσαμε χρησιμοποιώντας δείκτες θεμελιώδους οικονομίας θα ήταν σίγουρα ένα κομμάτι μιας συνολικής εργασίας. Τέλος διαπιστώσαμε πως οι έμπειροι επενδυτές δεν εφαρμόζουν τους κανόνες Τεχνικής Ανάλυσης βασιζόμενοι μόνο στις πραγματικές τιμές, κυρίως η εκτίμηση τους βασίζεται σε συγκεκριμένα σχήματα (μορφές) στα διαγράμματα των δεικτών αυτών. Το ανωτέρω πρόβλημα είναι προσεγγίζεται πληρέστερα μέσω της ανάλυσης χρονοσειρών. Η ίδια διαδικασία θα μπορούσε να μεταφραστεί σε ανάλυση και επεξεργασία χρονοσειρών. Υπάρχουν τρεις βασικές λειτουργίες που πραγματοποιούνται στην ανάλυση χρονοσειρών. Στη μία περίπτωση, χρησιμοποιούνται μονάδες μέτρησης απόστασης για να καθορίσουν την ομοιότητα ανάμεσα σε διαφορετικές χρονοσειρές. Στη δεύτερη περίπτωση, εξετάζεται η δομή

της χρονοσειράς για να καθορίσει (και ίσως να κατηγοριοποιήσει) τη συμπεριφορά της. Μία τρίτη εφαρμογή θα μπορούσε να είναι η χρήση διαγραμμάτων χρονοσειρών για την πρόβλεψη μελλοντικών τιμών

Σε αυτή την περίπτωση θα μπορούσε να υλοποιηθεί ένα OCR σύστημα το οποίο θα αναγνωρίζει αυτές της μορφές

5.2. Contributors

Στην επιλογή θέματος, κάποιιοι θα έλεγαν ότι αυτό είναι ένα πολύ δύσκολο εγχείρημα (αν αναλογιστεί κανείς πως μεγάλες εταιρείες του κλάδου χρησιμοποιούν παρόμοιες μεθόδους, οποίες υποστηρίζονται από πολυάριθμες ομάδες επαγγελματιών) και θα οδηγήσει σε ένα έργο με λίγες ή και καθόλου εισφορές. Αν συλλογιστούμε ότι η θεωρητική προσέγγιση δεν είναι καινοτόμα καθώς είναι ένα γνωστικό αντικείμενο το οποίο διδάσκεται σε πολλά πανεπιστήμια, η υλοποίηση και η εφαρμογή του είναι κάτι που παρουσιάζει μεγάλο ενδιαφέρον. Στο γεγονός αυτό αν συνυπολογίσουμε ότι πολύ λίγα project σε Μηχανική Μάθηση έχουν υλοποιηθεί σε γλώσσα προγραμματισμού Java μπορούμε να καταλάβουμε την αξία και την προσφορά της δουλειάς μας

Πηγές

- [1]https://cloud.google.com/appengine/?gclid=Cj0KEQjwl_6oBRDHxNGz6ueJufMBEiQAvm_k_oAEtQrww_C7nDBhl80FtkKD-5wn54JOWjjHXCO7i-8aApPk8P8HAQ
- [2] Sidney Cottle Roger F. Murray Frank E. Block Benjamin Graham, David Le Fevre Dodd. Graham and Dodd's security analysis. McGraw-Hill Professional, 1951.
- [3] A. Schroeder. The Snowball: Warren Buett and the Business of Life. Bantam Books. Random House Publishing Group, 2009. ISBN 9780553384611.
URL <http://books.google.no/books?id=OdpWOjo7EeQC>.
- [4] Bruno Biais. High frequency trading, 2011. URL <http://www.eifr.eu/files/file8379010.pdf>.
- [5] Harry Markowitz. Portfolio selection. The Journal of Finance, 7(1):pp. 77{91, 1952. ISSN 00221082. URL <http://www.jstor.org/stable/2975974>.
- [6] J.D. Schwager. Getting Started in Technical Analysis. Getting Started In. John Wiley, 1999. ISBN 9780471295426. URL <http://books.google.no/books?id=dm6EvSzLYNAC>.
- [7] M P Taylor and H Allen. The use of technical analysis in the foreign exchange market. Journal of International Money and Finance, 11(3): 304{314, 1992. URL <http://linkinghub.elsevier.com/retrieve/pii/0261560692900483>.
- [8] Cheol-Ho Park and Scott H. Irwin. What do we know about the profitability of technical analysis? Journal of Economic Surveys, 21(4):786{826, 2007. ISSN 1467-6419. doi: 10.1111/j.1467-6419.2007.00519.x. URL <http://dx.doi.org/10.1111/j.1467-6419.2007.00519.x>.
- [9] U Thissen, R van Brakel, A.P de Weijer, W.J Melssen, and L.M.C Buydens. Using support vector machines for time series prediction. Chemometrics and Intelligent Laboratory Systems, 69(1-2):35 { 49, 2003. ISSN 0169-7439. doi: 10.1016/S0169-7439(03)00111-4. URL <http://www.sciencedirect.com/science/article/pii/S0169743903001114>.
- [10] http://docs.opencv.org/2.4.9/doc/tutorials/ml/non_linear_svms/non_linear_svms.html

- [11] Yoav Freund and Robert E. Schapire, “Experiments with a new boosting algorithm”, In Machine Learning: Proceedings of the Thirteenth International Conference, pages 148–156, 1996.
- [12] Vatsal H Shah: Machine learning techniques for stock prediction. www.vatsals.com
- [13] Brown, D. & Jennings, R. (1989), “*On Technical Analysis*”, The Review Financial Studies, Vol. 2, No 4, pp. 527-551.
- [14] Brown, C. (1999), “*Technical Analysis for the trading Professional*”, McGraw-Hill.
- [15] Caginalp, G. & Balevonich, D. (2003), “*Theoretical Foundation for Technical Analysis*”, Journal of Technical Analysis, Vol. 59, pp. 5-22.
- [16] Κουρούκλης, Χ. (1999), *Τεχνική ανάλυση : θεωρία και πρακτικές εφαρμογές*, εκδόσεις Metapublications, Αθήνα.
- [17] <http://www.investopedia.com/>
- [18] <http://stockcharts.com/>
- [19] http://vivliothmyy.ee.auth.gr/160/1/Support_Vector_Machines.pdf
- [20] Cloud computing: Μια πρακτική προσέγγιση των Anthony T, Velte, Toby J. Velte και Robert Elsenpeter
- [21] <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/6322/1/Mazaraki.pdf>
- [22] <https://cloud.google.com/appengine/docs/java/datastore/>
- [23] <http://finance.yahoo.com/>
- [24] <http://square.github.io/retrofit/>
- [25] [Random Walks in StockMarket Prices By EUGENE F. FAMA GRADUATE SCHOOL OF BUSINESS UNIVERSITY OF CHICAGO]
- [26] <https://www.coursera.org/course/ml>
- [27] <http://www.heatonresearch.com/encog>
- [28] <http://java-ml.sourceforge.net/>
- [29] <http://java-ml.sourceforge.net/>