Γραμμική Παλινδρόμηση Ελαχίστων Τετραγώνων

Data set που χρησιμοποίηθηκε κατά τις δοκιμές

- http://archive.ics.uci.edu/ml/datasets/Insurance+Company+Benchmark+%28COIL+2000%29

(τα αρχεία του συγκεκριμένου data set δεν ήταν στη μορφή που δέχεται το πρόγραμμα,όμως έγινε μετατροπή των αρχείων τα οποία βρίσκονται εδώ)

Υλοποίηση

Υπάρχουν 3 τάξεις συνολικά.Η Instance αναπαριστά τα features και την κατηγορία ενός παραδείγματος εκπαίδευσης ή αντικειμένου προς κατάταξη.Ουσιαστικά η αναπαράσταση γίνεται από ένα ArrayList<Integer> για τα features και μια int μεταβλητή που κρατάει την κατηγορία του Instance.Πρώτη τιμή που εισάγεται στα features είναι το 1 αφού το x0 είναι πάντα 1.Στην κλάση υπάρχουν κατάλληλες συναρτήσεις get και set για πρόσβαση στη δομή ArrayList αλλά και στην κατηγορία του Instance.Η InstancePool παριστά ένα σύνολο από Instance αντικείμενα κάνοντας χρήση μιας δομής ArrayList<Instance>.Επιπλέον,η InstancePool διαβάζει από το FileInputStream και κατασκευάζει τα αντικείμενα Instance.Αυτό συμβάινει στη συνάρτηση createInstances.Το αρχείο πρέπει να είναι στη κατάλληλη μορφή,δηλαδή κάθε γραμμή αναπαριστά ένα Instance και στη γραμμή αυτή πρέπει όλα τα features να χωρίζονται από κόμμα.Το τελευταίο στοιχείο της γραμμής δείχνει την κατηγορία του Instance.Τέλος,η τάξη LSR είναι αυτή που υλοποιεί την γραμμική παλινδρόμηση ελαχίστων τετραγώνων.

Η μέθοδος train έχει σκοπό να βρει τα κατάλληλα βάρη αναλύοντας τα παραδείγματα εκπαίδευσης με σκοπό να μιμηθεί την γραμμική συνάρτηση που ακολουθούν τα δεδομένα.Η evaluate έχει σκοπό να επιστρέψει τιμές για κάθε ένα από τα Instance του InstancePool που δίνεται ως όρισμα,χρησιμοποιώντας τα βάρη που δημιουργήθηκαν από την συνάρτηση train.Η randomiseWeights δημιουργεί το διάνυσμα βαρών(αναπαρίσταται με ένα ArrayList<Double>) και του αναθέτει μικρές τυχαίες θετικές τιμές (από 0 έως 1 με τη βοήθεια της Math.random()).Η fValue αποτιμά την τιμή του Instance χρησιμοποιώντας τα features του και το διάνυσμα βαρών.Η errorEval αποτιμά το σφάλμα δοθέντων των τιμών fval (αποτέλεσμα συνάρτησης fValue) και της τιμής της σωστής απόκρισης(κατηγορίας του Instance).Η updateWeights ενημερώνει το διάνυσμα βαρών σύμφωνα με τη λογική του αλγορίθμου(θα αναλυθεί ακολούθως).

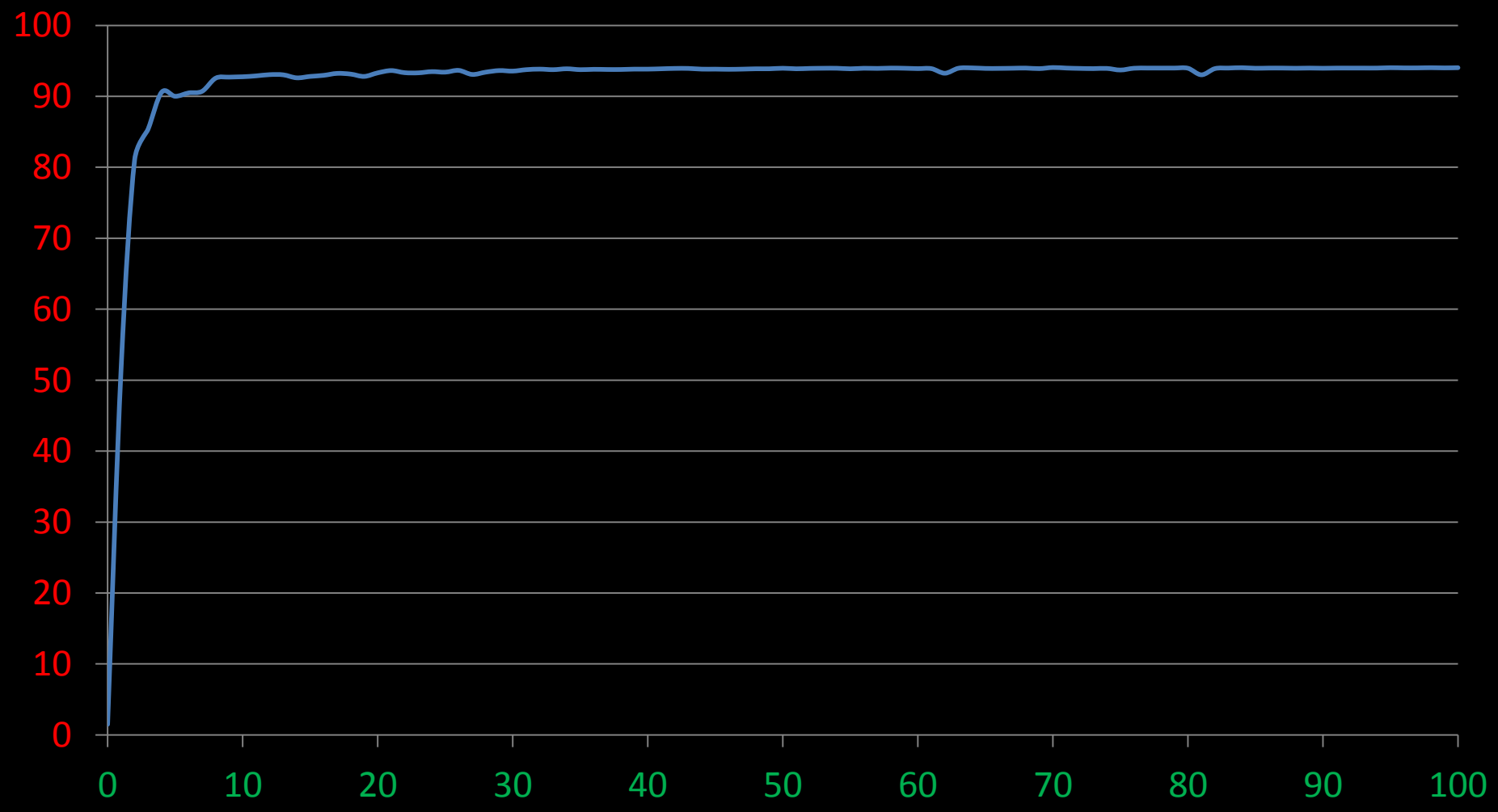
Σχετικά με τον αλγόριθμο

Για την υλοποίηση της γραμμικής παλινδρόμησης ελαχίστων τετραγώνων θα μπορούσε να είχε χρησιμοποιηθεί είτε ο αλγόριθμος κατάβασης κλίσης (batch gradient descent),είτε ο αλγόριθμος στοχαστικής κατάβασης κλίσης(stochastic gradient descent).Προτιμήθηκε ο δεύτερος επειδή έχει μικρότερο υπολογιστικό κόστος(κυρίως στα μεγάλα data sets) και επιπλέον συγκλίνει αρκετά γρήγορα στο ελάχιστο σφάλμα.Έπειτα απο δοκιμές,οι παράμετροι h και tol (το μέτρο σύγκλισης του αλγορίθμου που δείχνει αν η διαφορά του σφάλματος μεταξύ των εποχών είναι αρκετά μικρή,όπου μια εποχή είναι μια επανάληψη του αλγορίθμου) πήραν τις τιμές 0.0001 και 0.001 αντίστοιχα.Αυτή η ανάθεση τιμών φάνηκε να είναι βέλτιστη,από υπολογιστική και ποιοτική σκοπιά,στα data sets που εξετάστηκαν.Τέλος,επιτυχής απόκριση θεωρήθηκε αν συμβαίνει η στρογγυλοποιημένη τιμή της fValue και η κατηγορία να συμπίπτουν.

Παραδείγμα χρήσης

✓Tic data set:

Στο συγκεκριμένο data set το πρόγραμμα έχει ακρίβεια 94% περίπου.Παρακάτω φαίνεται η καμπύλη μάθησης του αλγορίθμου(στον οριζόντιο άξονα το μέγεθος,εκφρασμένο σε % επί του συνόλου,των δεδομένων εκπαίδευσης και στον κατακόρυφο άξονα το ποσοστό ορθότητας στα δεδομένα ελέγχου):



Παρατηρούμε ότι ο αλγόριθμος βρίσκει πολύ γρήγορα και με ελάχιστο ποσοστό παραδειγμάτων από το training set τις σωστές τιμές στο test set.Συνεπώς,με ελάχιστο ποσοστό χρήσης του training set (~10%) μπορούμε να επιτύχουμε ποσοστό επιτυχίας 90% και άνω.Μετά από ένα σημείο,παύει να έχει σημασία το πλήθος των παραδειγμάτων εκπαίδευσης και ο αλγόριθμος συγκλίνει σε ένα ποσοστό επιτυχίας περί το 94%.Αυτό συμβαίνει γιατί στο συγκεκριμένο data set υπάρχουν σημαντικά περισσότερα Instances με κατηγορία 0.Έτσι ο αλγόριθμος μαθαίνει να αναγνωρίζει αυτά τα Instances αρκετά καλά και σε πολύ σύντομο διάστημα.