Πρόβλεψη με χρήση Naive Bayes

Εργασία στο μάθημα "Αλγόριθμοι: Σχεδίαση και Ανάλυση"

Διδάσκουσα: Κ. Παπακωνσταντινοπούλου

1 Εισαγωγή

Σε αυτή την εργασία θα εφαρμόσουμε μηχανική μάθηση και θα προσπαθήσουμε να κάνουμε πρόβλεψη σε δεδομένα που δεν έχουμε δει. Ειδικότερα, θα προσπαθήσουμε να προβλέψουμε την κουζίνα στην οποία ανήκουν διάφορες συνταγές που μας δίνονται.

2 Σύνολο Εκπαίδευσης

Το σύνολο εκπαίδευσής μας αποτελείται από 1794 συνταγές από 20 διαφορετικές κουζίνες. Για παράδειγμα, μια συνταγή από το σύνολο εκπαίδευσής μας φαίνεται παρακάτω.

```
{
    "id" : 2,
    "cuisine" : "greek",
    "ingredients" : [
        "minced garlic",
        "dried oregano",
        "red wine vinegar",
        "olive oil",
        "boneless chop pork",
        "lemon juice"
    ]
}
```

Αυτή η συνταγή είναι μια ελληνική συνταγή με έξι συστατικά. Τα συστατικά παίζουν το ρόλο των χαρακτηριστικών και σε κάθε συνταγή κάθε συστατικό είναι είτε παρόν είτε όχι. Υπάρχουν 2398 διαφορετικά χαρακτηριστικά (συστατικά) που εμφανίζονται στις συνταγές του συνόλου εκπαίδευσης μας και για διευκόλυνση μας δίνεται ένα αρχείο που απαριθμεί όλα αυτά τα χαρακτηριστικά. Στον πίνακα 1 φαίνεται η κατανομή των συνταγών στο σύνολο εκπαίδευσής μας.

2.1 Αρχεία και μορφή

Το zip αρχείο περιέχει τα αρχεία που περιγράφονται παρακάτω.

training.json και training.csv που περιέχει τα δεδομένα εκπαίδευσής μας. Κάθε γραμμή του json αρχείου περιέχει μια διαφορετική συνταγή σε μορφή json. Κάθε γραμμή του csv αρχείου περιέχει μια διαφορετική συνταγή σε μορφή csv. Μπορείτε να χρησιμοποιήσετε οποιοδήποτε από τα δύο αρχεία για να δώσετε είσοδο στον αλγόριθμό σας. Και τα δύο περιέχουν την ίδια πληροφορία, επομένως επιλέγετε αυτό που προτιμάτε να διαβάσετε με το πρόγραμμά σας.

Πίνακας 1: Ανάλυση των 1794 συνταγών του συνόλου εκπαίδευσής μας.

cuisine	recipes
brazilian	20
british	35
cajun_creole	73
chinese	124
filipino	38
french	122
greek	57
indian	124
irish	34
italian	324

cuisine	recipes
jamaican	30
japanese	66
korean	47
mexican	275
moroccan	38
russian	27
southern_us	192
spanish	49
thai	74
vietnamese	45

ingredients.json και ingredients.txt που περιέχουν τη λίστα των συστατικών που εμφανίζονται στις συνταγές του συνόλου εκπαίδευσης. Το ingredients.json είναι ένα μεγάλο json αρχείο με όλα τα συστατικά. Το ingredients.txt περιέχει την ίδια πληροφορία, με διαφορετική όμως μορφή: περιέχει το όνομα ενός μόνο συστατικού σε κάθε γραμμή. Μπορείτε να χρησιμοποιήσετε όποιο αρχείο προτιμάτε.

2.1.1 Μορφή του training.json

Κάθε γραμμή περιέχει μια διαφορετική συνταγή σε μορφή json, οπότε το αρχείο έχει τη μορφή που φαίνεται παρακάτω:

```
{"id":0,"cuisine":"greek","ingredients":["romaine lettuce","black olives","grape tomatoes","garlic","pepper","purple onion","seasoning","garbanzo beans","feta cheese crumbles"]}
{"id":1,"cuisine":"greek","ingredients":["ground pork","finely chopped fresh parsley","onions","salt","vinegar","caul fat"]}
{"id":2,"cuisine":"greek","ingredients":["minced garlic","dried oregano","red wine vinegar","olive oil","boneless chop pork","lemon juice"]}
...
...
...
```

2.1.2 Μορφή του training.csv

Κάθε γραμμή έχει διαφορετική συνταγή σε μορφή csv (τιμές διαχωρισμένες με κόμμα). Η πρώτη στήλη είναι η ταυτότητα της συνταγής (ένας ακέραιος αριθμός). Η δεύτερη στήλη δείχνει την κουζίνα στην οποία ανήκει η συνταγή. Όλες οι επόμενες στήλες της γραμμής περιέχουν τα συστατικά της συγκεκριμένης συνταγής. Το αρχείο έχει τη μορφή που φαίνεται παρακάτω.

```
0,"greek","romaine lettuce","black olives","grape tomatoes","garlic","pepper","purple onion","seasoning","garbanzo beans","feta cheese crumbles"
1,"greek","ground pork","finely chopped fresh parsley","onions","salt","vinegar","caul fat"
2,"greek","minced garlic","dried oregano","red wine vinegar","olive oil","boneless chop pork","lemon juice"
...
...
...
```

2.1.3 Μορφή του ingredients.json

Έχουμε ένα μόνο αντικείμενο json. Το αρχείο έχει τη μορφή που φαίνεται παρακάτω.

```
{"ingredients":["boneless chop pork","dried oregano","lemon juice","minced garlic","olive oil","red wine vinegar",...]}
```

2.1.4 Μορφή του ingredients.txt

Σε περίπτωση που το παραπάνω αρχείο με το ένα μεγάλο αντικείμενο json δεν είναι βολικό για εμάς, τα συστατικά μας δίνονται επίσης σε ένα αρχείο κειμένου, όπου σε κάθε γραμμή του έχουμε ένα μόνο συστατικό που περικλείεται σε διπλά εισαγωγικά. Το αρχείο έχει τη μορφή που φαίνεται παρακάτω.

```
"boneless chop pork"
"dried oregano"
"lemon juice"
"minced garlic"
"olive oil"
"red wine vinegar"
...
...
...
```

3 Υλοποίηση

Καλείστε να υλοποιήσετε τον αλγόριθμο μηχανικής μάθησης που μελετήσατε (naive bayes). Μπορείτε να χρησιμοποιήσετε επιπλέον βιβλιοθήκες που σας βοηθούν να υπολογίσετε ή να διαβάσετε πληροφορίες, π.χ. για αλγεβρικό χειρισμό των πινάκων, ανάγνωση αρχείων json ή κάτι παρόμοιο. Ωστόσο, πρέπει να υλοποιήσετε μόνοι σας τον πυρήνα του αλγορίθμου που θα χρησιμοποιήσετε.

Επιπλέον θα χρειαστεί να εφαρμόσετε 6-fold cross validation (δείτε Παράρτημα A – στα αγγλικά λόγω ορολογίας) και να σχολιάσετε την ακρίβεια γενίκευσης που παρατηρήσατε κάθε φορά καθώς και τη μέση ακρίβεια χρησιμοποιώντας αυτές τις έξι τιμές που έχετε αναφέρει.

4 Υποβολή

Πρέπει να υποβάλετε ένα αρχείο zip με όλο τον κώδικά σας και την αναφορά. Σε περίπτωση που το πρόγραμμα σας απαιτεί ένα makefile για το compile, θα πρέπει επίσης να συμπεριληφθεί το makefile. Είστε ελεύθεροι να επιλέξετε οποιαδήποτε γλώσσα προτιμάτε.

4.1 Αναφορά

Υποβάλετε ένα έγγραφο 1-2 σελίδων, περιγράφοντας συνοπτικά τη μέθοδο που χρησιμοποιήσατε για την εκπαίδευση και αναφέροντας την πολυπλοκότητά της, καθώς και μια σύντομη περίληψη των αποτελεσμάτων του 6-fold cross validation που πήρατε από το σύνολο εκπαίδευσης που σας δόθηκε. Ουσιαστικά μία ή δύο μικρές παραγράφους πρέπει να είναι αρκετές για να περιγράψουν τη μέθοδο σας. Επίσης μια μικρή παράγραφος με ένα απλό πίνακα πρέπει να αρκεί για την ακρίβεια γενίκευσης που λάβατε σε κάθε ένα από τα 6 τρεξίματα της μεθόδου 6-fold cross validation που εφαρμόσατε. Μην ξεχάσετε να αναφέρετε ξεκάθαρα την ελάχιστη και τη μέγιστη ακρίβεια γενίκευσης που λάβατε μετά από όλα τα 6 τρεξίματα, π.χ., γράφοντας αυτές τις δύο τιμές με έντονους χαρακτήρες. Επίσης, μην ξεχάσετε να αναφέρετε με σαφήνεια τη μέση ακρίβεια που παρατηρήσατε σε αυτά τα 6 τρεξίματα.

Φυσικά είστε ελεύθεροι να χρησιμοποιήσετε περισσότερο κείμενο αν θέλετε να αναφέρετε λεπτομέρειες για την τεχνική σας, αλλά κάτι τέτοιο δεν είναι απαραίτητο. Τέλος, σχολιάστε επίσης τον απαιτούμενο χρόνο για *εκπαίδευση και δοκιμή* [1] και τις προδιαγραφές του μηχανήματος όπου πραγματοποιήθηκε η εκπαίδευση (δηλαδή cpu και μνήμη).

[1] Μια χονδρική εκτίμηση στο πλησιέστερο δευτερόλεπτο ή λεπτό είναι αρκετή.

A Cross Validation

Cross validation is a method that allows us to estimate the performance of the hypotheses that we obtain on *unseen data*. The typical approach when we are given a data set for training, is to keep a subset of the training set on the side and not use it as part of the input of our machine learning algorithm. Then, when we derive a hypothesis using this restricted training set, we now examin how well our hypothesis predicts unseen data using the set that we kept initially on the side.

The Holdout Method. The holdout method is the simplest kind of cross validation. The data set is separated into two sets, called the *training set* and the *test set*. Our machine learning algorithm comes up with a hypothesis *using the training set only*. Then, the derived hypothesis is asked to predict the output values for the data in the test set (note that these are unseen examples). The correct predictions our hypothesis makes are accumulated and in the end we report the accuracy (fraction of correct predictions for the classification) our hypothesis has in the test set. In other words we use the formula,

$$\frac{\text{number of correct predictions}}{\text{number of examples in the test set}} \cdot 100\% \,.$$

The main drawback of this approach is that the reported accuracy may fluctuate a lot (in other words, it has high *variance*) depending on how one divides the original training set into the two smaller sets (the one actually used for training and the other one used for testing).

k-Fold Cross Validation. k-fold cross validation is the natural improvement over the holdout method and is the method that you need to use in this assignment with $k = 6$.

The original data set is divided into k disjoint sets and the holdout method is repeated k times. The idea is now that each one of the k times, one of the sets is going to be used for testing (predicting the accuracy) and the rest $k - 1$ sets will be used for training. In other words, every data point gets to be in a test set exactly once, and gets to be in a training set $k - 1$ times. The advantage of this method is that it matters less how the data gets divided; i.e., the variance of our estimated accuracy on unseen data (also called *generalization accuracy*) decreases as k increases. The drawback of the method is that it takes longer to run so that we can come up with a good estimate on the generalization accuracy of the method.

Typically, we do k-fold cross validation and report an estimate on the generalization accuracy of the method that we have used. Once this part is over, then we use all the original data for training (i.e., no test set) so that we can come up with a hypothesis to be used on *new* unseen data.

Leave-One-Out Cross Validation. Leave-one-out cross validation is k-fold cross validation when $k = N$, where N is the number of examples in the given data set. In other words, we come up with a hypothesis N separate times and we use each hypothesis for predicting the label of just one example. Again, we report the average accuracy among all those runs.