Deep Learning – 2ο εξάμηνο

Εργασία

Μέρος Α: Ταξινομητές για δεδομένα πινάκων

Έχοντας ολοκληρώσει το μεταπτυχιακό σας, δίνετε συνέντευξη για μια θέση με αντικείμενο “data analytics & financial risk estimations associate” σε ένα χρηματοπιστωτικό ίδρυμα.

Έχετε πάει πολύ καλά στην αρχική συνέντευξη και στη συνέχεια ακολουθεί η πρακτική αξιολόγηση. Σας παραδίδουν ένα αρχείο excel που περιέχει χρηματοπιστωτικούς δείκτες και μερικές ακόμα πληροφορίες για μια σειρά από ελληνικές εταιρείες.

Τα δεδομένα σας είναι:

1. Οι δείκτες απόδοσης των εταιρειών (στήλες Α έως και Η)
2. Τρεις δυϊκοί δείκτες δραστηριοτήτων (στήλες Ι, J, Κ)
3. Η κατάσταση της εταιρείας (1 όλα καλά, 2 έχει κηρύξει χρεωκοπία)
4. Το έτος στο οποίο αφορούν τα ως άνω μεγέθη.

Η δοκιμασία σας είναι η ακόλουθη:

I. Να δημιουργήσετε το καλύτερο δυνατό μοντέλο ταξινόμησης που θα εντοπίζει τις εταιρείες εκείνες που θα χρεωκοπήσουν, παίρνοντας ως είσοδο τις τιμές στις στήλες Α έως Κ. Φυσικά, όλα τα μοντέλα που θα αναπτύξετε πρέπει να είναι υλοποιημένα σε Python και να παραδοθούν για αξιολόγηση από τους υπευθύνους.

Προσοχή στους ακόλουθους δύο περιορισμούς:

1. Το μοντέλο πρέπει να βρίσκει με ποσοστό επιτυχίας **τουλάχιστον 62%** τις εταιρείες που θα πτωχεύσουν.
2. Το μοντέλο πρέπει να βρίσκει με ποσοστό επιτυχίας **τουλάχιστον 70%** τις εταιρείες που **δεν** θα πτωχεύσουν.

Μετά από σκέψη, αποφασίζετε να συντάξετε κώδικα στον οποίο:

1. Διαβάζετε τα δεδομένα από το excel, τα κανονικοποιείτε, και τα χωρίζετε σε training/test sets (αγνοήστε τη στήλη Μ με το έτος).
2. Εκπαιδεύετε και αξιολογείτε τα ακόλουθα μοντέλα επιβλεπόμενης μάθησης πάνω στα set που δημιουργήσατε, δηλαδή τα:
    - Linear Discriminant Analysis
    - Logistic Regression
    - Decision Trees
    - k-Nearest Neighbors
    - Naïve Bayes
    - Support Vector Machines
    - Neural Networks

3. Περνάτε τα αποτελέσματα των πειραμάτων σε ένα αρχείο excel όπου κάθε γραμμή έχει τις ακόλουθες τιμές:

`Classifier Name | Training or test set| Number of training samples | Number of non-healthy companies in training sample | TP | TN | FP | FN | Precision | Recall | F1 score | Accuracy`

Με βάση τα αποτελέσματα πάνω στα test data, υπάρχει κάποιο μοντέλο που να ικανοποιεί τους περιορισμούς απόδοσης;

II. Επαναλάβετε το παραπάνω πείραμα ως εξής:

Αφού φτιάξετε τα training και test sets, ελέγξτε το training set. Αν η κατανομή είναι πάνω από 3 υγιείς επιχειρήσεις για κάθε χρεωκοπημένη, διαλέξτε με τυχαίο τρόπο όσες υγιείς εταιρείες χρειαστεί, ώστε η αναλογία στο training set να είναι 3 υγιείς / 1 χρεωκοπημένη.

Μέρος Β: Δημιουργία, εκπαίδευση, αξιολόγηση και αποθήκευση διαφορετικών αρχιτεκτονικών βαθέων δικτύων σε εικόνες

Σύγκριση μεταξύ διαφορετικών αρχιτεκτονικών. Θα αναπτύξετε και θα συγκρίνετε τις επιδόσεις μεταξύ δύο αρχιτεκτονικών: a) deep learning neural network (DNN) και b) convolutional neural network (CNN). Τα αποτελέσματα θα είναι πάνω στο mnist dataset. Πιο συγκεκριμένα:

1. Θα δημιουργήσετε ένα νέο Colab notebook με τίτλο ComparingBasicNNArchitectures.
2. Θα αξιοποιήσετε και θα τροποποιήσετε τις παραμέτρους / αρχιτεκτονική και ό,τι άλλο χρειαστεί στους κώδικες που υπάρχουν διαθέσιμοι στο eclass. Βασικές προϋποθέσεις:
    a. Τα μοντέλα να κάνουν χρήση του keras, όπως αυτό παρέχεται μέσω του TensorFlow.
    b. να γίνεται χρήση της συνάρτησης ReLu σε όλα τα hidden layers,
    c. να χρησιμοποιήσετε τουλάχιστον ένα dropout layer.
    d. Το output layer θα χρησιμοποιεί την softmax.
3. Θα εκπαιδεύσετε και θα αξιολογήσετε τα DNN και CNN πάνω στο mnist dataset, του οποίου το train set (από το keras.datasets) περιλαμβάνει 60000 εγγραφές. Βασικές προϋποθέσεις:
    a. Θα δημιουργήσετε έξι (6) μικρότερα datasets χρησιμοποιώντας το KFold ή το StratifiedKFold (διαθέσιμα στο sklearn.model_selection).
    b. Θα μεριμνήσετε ώστε η εκπαίδευση του μοντέλου (fit) να χρησιμοποιεί ένα συνόλου ελέγχου (validation set). Το validation set πρέπει να είναι ανεξάρτητο των train και test sets. Με το πέρας του training process, πρέπει να τυπώνονται στην οθόνη τα losses τόσο για το training όσο και για το validation set.
    c. Με το που ολοκληρωθεί το training process, για ένα μοντέλο, θα χρησιμοποιήσετε την εντολή .predict πάνω στο τρέχων training set (ένα εκ των 6) και στο test set (10000 samples, πάντα το ίδιο). Με βάση τα αποτελέσματα και αφού κάνετε τις οποίες τροποποιήσεις χρειαστούν στα output, θα υπολογίσετε διαφορά μεγέθη απόδοσης, που είναι σχετικά με προβλήματα classification.
    d. Τα αποτελέσματα κάθε πειράματος θα καταγράφονται σε μια γραμμή σε ένα pandas dataframe. Στο τέλος της εκτέλεσης θα έχετε ένα dataframe με 24 γραμμές (+1 για τους τίτλους). Κάθε γραμμή θα έχει τις ακόλουθες εγγραφές: Technique name [DNN/CNN]| Set [Train/Test]| Fold number [1, .., 6] / Accuracy | Precision | Recall | F1 score. Στο τέλος αποθηκεύστε το dataframe ως αρχείο ονόματι erotima1.csv.
4. Θα αποθηκεύσετε ένα DNN και ένα CNN ως αρχεία .h5. Τα DNN και CNN που αποθηκεύετε πρέπει να έχουν τα καλύτερα αποτελέσματα με βάση τις επιδόσεις στο test set, στο σύνολο των 6 πειραμάτων.

Συντάξτε μια αναφορά στην οποία θα παρουσιάσετε μια τελική πρόταση εξηγώντας ποια ήταν η καλύτερη τεχνική πάνω στα δεδομένα που επιλέξατε. Οι αρχιτεκτονικές των μοντέλων πρέπει να παρουσιαστούν ως γράφημα, χρησιμοποιώντας το power point, το Diagrams.net ή κάποιο αντίστοιχο εργαλείο. Φροντίστε να υπάγουν οι κατάλληλες γραφικές παραστάσεις, για την σύγκριση αποτελεσμάτων, και να συνοδεύονται από μία τουλάχιστον παράγραφο σχολιασμό.

Μέρος Γ: Αξιολόγηση καταλληλότητας διαφορετικών αρχιτεκτονικών αναδρομικών νευρωνικών δικτύων για ενεργειακό επιμερισμό

Σύγκριση μεταξύ διαφορετικών αρχιτεκτονικών. Θα αναπτύξετε και θα συγκρίνετε τις επιδόσεις μεταξύ τριών διαφορετικών αρχιτεκτονικών: a) recurrent neural network (RNN), b) long short-term memory network (LSTM) και c) Gated Recurrent Unit (GRU). Τα αποτελέσματα θα είναι πάνω στα δεδομένα που παρέχοντα στο eclass, και αξιοποιούν το "Coffee Machine Consumption" dataset. Η βασική ιδέα έχει ως εξής: δοθείσης μιας κυματομορφής (πχ. Κατανάλωση ρεύματος στο σπίτι τα τελευταία 120 λεπτά) υπολογίστε την αντίστοιχη κυματομορφή για μια συγκεκριμένη συσκευή στο σπίτι (π.χ. καφετιέρα), στο ίδιο διάστημα.

Περιγραφή του dataset: Συνολικά έχετε τέσσερα (4) txt αρχεία.

1. CoffeeMachinemaxAgg.txt: Περιλαμβάνει την μέγιστη τιμή συνολικής κατανάλωσης που παρατηρήθηκε στον ηλεκτρολογικό πίνακα του σπιτιού. Θα χρησιμοποιηθεί για την κανονικοποίηση των input values.
2. CoffeeMachinemaxApp.txt: Περιλαμβάνει την μέγιστη τιμή κατανάλωσης που παρατηρήθηκε στην καφετιέρα. Θα χρησιμοποιηθεί για την κανονικοποίηση των output values.
3. Input_Data.txt: Περιλαμβάνει *k* γραμμές x *m* τιμές/γραμμή. Κάθε γραμμή είναι η κατανάλωση σε watt, ανά ένα λεπτό, για ένα διάστημα *m* λεπτών. Οι τιμές αφορούν την συνολική κατανάλωση στο σπίτι, σε αυτό το διάστημα.
4. Output_Data.txt: Περιλαμβάνει *k* γραμμές x *m* τιμές/γραμμή. Κάθε γραμμή είναι η κατανάλωση σε watt, ανά ένα λεπτό, για ένα διάστημα *m* λεπτών. Οι τιμές αφορούν στην κατανάλωση της συγκεκριμένης συσκευής, σε αυτό το διάστημα.

Ζητούμενα άσκησης:

5. Θα δημιουργήσετε ένα νέο Colab notebook με τίτλο ComparingBasicRNNArchitectures.
6. Θα αξιοποιήσετε και θα τροποποιήσετε τις παραμέτρους / αρχιτεκτονική και ό,τι άλλο χρειαστεί στους κώδικες που υπάρχουν διαθέσιμοι στο eclass (βλέπε αρχείο seq2seqdemo.py) .
7. Ο κώδικας θα διαβάζει τα αρχεία με τις καταναλώσεις απευθείας από το google drive, χρησιμοποιώντας την βιβλιοθήκη numpy.
   Σημείωση: θα χρειαστεί να κάνετε mount το drive για να έχετε πρόσβαση. Αν δεν καταφέρετε να αξιοποιήσετε την numpy, δοκιμάστε εναλλακτικούς τρόπους.
8. Θα τυπώσετε τυχαία τρία plots σε κάθε ένα εκ των οποίων θα φαίνεται η συνολική κατανάλωση και η κατανάλωση της συσκευής, για μια τυχαία χρονική περίοδο.
   Σημείωση 1: Για να είναι πιο ευδιάκριτες οι τιμές στα plot θα έχετε δύο (2) y-axis (ένα primary – aggregated signal και ένα secondary – appliance signal) .
   Σημείωση 2: Ένα τουλάχιστον από τα plots θα πρέπει να δείχνει κατανάλωση και στην συσκευή.
9. Θα κανονικοποιήστε τις τιμές input και output και θα δημιουργήσετε δείκτες (indexes) που θα χωρίζουν το dataset σε train και test sets.
10. Θα εκπαιδεύσετε και θα αξιολογήσετε τα RNN, LSTM και GRU. Βασικές προϋποθέσεις:
    a. Η αρχιτεκτονική του μοντέλου είναι δική σας επιλογή. Έχετε το ελεύθερο να δοκιμάσετε αριθμό layer, ή οποία άλλη παράμετρο θεωρείτε σημαντική για την καλή επίδοση του μοντέλου. Οι εποχές εκπαίδευσης θα είναι τουλάχιστον 30.

b. Θα μεριμνήσετε ώστε η εκπαίδευση του μοντέλου (fit) να χρησιμοποιεί ένα συνόλου ελέγχου (validation set). Το validation set πρέπει να είναι ανεξάρτητο των train και test sets.
c. Κατά την διάρκεια του training, θα κάνετε χρήση του callbacks API ώστε η εκπαίδευση να σταματάει αν το validation error δεν μειωθεί για 5 συνεχόμενες εποχές.
d. Με το πέρας του training process, πρέπει να τυπώνονται στην οθόνη τα losses τόσο για το training όσο και για το validation set.

11. Με το που ολοκληρωθεί το training process, για ένα μοντέλο, θα χρησιμοποιήσετε την εντολή .predict πάνω στο τρέχων training set και στο test set. Με βάση τα αποτελέσματα θα υπολογίσετε διαφορά μεγέθη απόδοσης, που είναι σχετικά με προβλήματα regression (δηλ. RMSE, MAE και max error).
    Σημείωση 1: τα errors πρέπει να υπολογιστούν στα *denormalized* δεδομένα.
    Σημείωση 2: έστω ότι έχετε *n* test sequences. Θα υπολογίσετε *n* φορές τα RMSE, MAE και max error και θα κάνετε τα αντίστοιχα plots. Δηλαδή 1 plot ανά τύπο σφάλματος, για τις τρεις τεχνικές.
12. Επιλέξτε 4 γραμμές από τα test δεδομένα: δυο στις οποίες η συσκευή λειτουργούσε και δύο στις οποίες ήταν σβηστή. Τυπώστε τις αντίστοιχες γραφικές παραστάσεις (4 χωριστά plots), δείχνοντας την πραγματική κατανάλωση και τις προβλέψεις των τριών μοντέλων.

Συντάξτε μια αναφορά στην οποία θα παρουσιάσετε μια τελική πρόταση εξηγώντας ποια ήταν η καλύτερη τεχνική πάνω στα δεδομένα που επιλέξατε. Βασική ερώτηση: υπάρχει κάποιο μοντέλο που μπορεί να υπολογίσει αρκετά καλά την κατανάλωση της συγκεκριμένης συσκευής (καφετιέρα). Φροντίστε να υπάγουν οι κατάλληλες γραφικές παραστάσεις, για την σύγκριση αποτελεσμάτων, και να συνοδεύονται από μία τουλάχιστον παράγραφο σχολιασμού έκαστη.

Οδηγίες:

Α. Οι εργασίες είναι ατομικές.
Β. Οι εργασίες θα πρέπει να αναρτώνται στο eClass σε ένα αρχείο zip (όχι rar) εντός της προβλεπόμενης προθεσμίας. Δεν θα δοθεί παράταση.
Γ. Κάθε εργασία πρέπει να συνοδεύεται από:

- Τα **αρχεια .py** που περιέχουν τις απαντήσεις στα ερωτήματα
- Μια **αναφορά** σε pdf με τα ακόλουθα στοιχεία:
    - Εξώφυλλο: 1 σελίδα, περιλαμβάνει τα στοιχεία του/της φοιτητή/-τριας, όνομα μαθήματος, ημερομηνία, και λοιπά σχετικά στοιχεία.
    - Συγκεντρωτικός πίνακας περιεχομένων, εικόνων, και λοιπών γραφημάτων που παραθέτετε στην αναφορά.
    - Ενότητα 1 εισαγωγή: 1 σελίδα, περιγράφετε το πρόβλημα (*χωρίς* να αντιγράψετε αυτούσια την εκφώνηση της άσκησης)
    - Ενότητα 2 μέθοδοι που εφαρμόστηκαν: από 1 μέχρι 3 σελίδες, περιγράφετε τις μεθόδους που χρησιμοποιήσατε, πλεονεκτήματα και μειονεκτήματα που έχουν, παραθέτετε τα σχετικά link και αναφέρετε τυχόν παραδοχές που κάνατε.
    - Ενότητα 3 πειραματικά αποτελέσματα: περιγράφετε τα dataset που χρησιμοποιήσατε και παραθέτετε τα σχετικά αποτελέσματα. Φροντίστε να είναι ξεκάθαρο στο ποιο ερώτημα αναφέρεστε.
    - Ενότητα 4, συμπεράσματα: 1 σελίδα, με βάση τα αποτελέσματα τι προτείνετε, ποιο μοντέλο αποδίδει καλύτερα, τι θα μπορούσε να γίνει για περαιτέρω βελτίωση στην απόδοση.
    - Η αναφορά θα περιέχει γραφικές παραστάσεις κάθε είδους και πίνακες αξιολόγησης των αποτελεσμάτων που πρέπει να συνοδεύονται (έκαστο) από μια τουλάχιστον παράγραφο με σχολιασμό.

Φροντίστε ώστε:

- Ο κώδικας να συνοδεύεται απαραιτήτως από κατάλληλα σχόλια, στα αγγλικά.
- Να έχει γίνει συντακτικός και ορθογραφικός έλεγχος στην αναφορά που θα υποβάλετε.
- Οι προτάσεις να είναι κατανοητές και μικρές σε έκταση.
- Οι εικόνες να ***μην*** έχουν προκύψει από print screen. Αν το πρόγραμμα δημιουργεί μια εικόνα αποθηκεύστε την κανονικά (jpg ή png), πριν την χρησιμοποιήσετε.

- Οι γραφικές παραστάσεις να περιλαμβάνουν ονόματα στους άξονες και λεζάντα. Σκοπός είναι να γίνεται κατανοητό τι δείχνει, με μια ματιά.
- Αν κάτι δεν διευκρινίζεται, έχετε το δικαίωμα να κάνετε όποια υλοποίηση σας βολεύει.
- Οι βιβλιοθήκες που θα χρησιμοποιήσετε *πρέπει* να μπορούν να εγκατασταθούν μέσω του pip.
- Ο κώδικας *πρέπει* να τρέχει σε Google Colab.

Καταληκτική Ημερομηνία Παράδοσης: **09 Ιουλίου 2023**. Δεν θα δοθεί παράταση