ΕΞΟΡΥΞΗ ΓΝΩΣΕΙΣ ΑΠΟ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ

Project

“The case of flight passengers prediction”

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
Τμήμα Πληροφορικής

Χειμερινό Εξάμηνο 2018-2019

Καθηγητής
Μιχάλης Βιρζιγιάννης

Ομάδα:
Αναγνωστάκη Ηρώ : 3140008
Βαγγελάκης Ορφέας : 3140018

Όνομα ομάδας στο Kaggle : “Hey there”

Στα πλαίσια της εργασίας του μαθήματος "Εξόρυξη Γνώσης από Βάσεις Δεδομένων και τον Παγκόσμιο Ιστό", δουλέψαμε πάνω σε ένα πρόβλημα κατηγοριοποίησης. Συγκεκριμένα, μας δόθηκε ένα σύνολο δεδομένων το οποίο αποτελείται από μερικές χιλιάδες πτήσεις, όπου κάθε πτήση περιγράφεται από ένα σύνολο μεταβλητών (αεροδρόμιο αναχώρησης, αεροδρόμιο άφιξης, κτλ). Κάθε πτήση χαρακτηρίζεται επίσης από μια μεταβλητή που σχετίζεται με τον αριθμό των επιβατών της πτήσης (π.χ. κάθε τιμή της μεταβλητής σχετίζεται με ένα εύρος πλήθους επιβατών). Για κάποιες πτήσεις, η τιμή της μεταβλητής είναι γνωστή, ενώ για άλλες όχι. Στόχος μας ήταν να προβλέψουμε την τιμή της μεταβλητής για τις πτήσεις για τις οποίες δεν είναι διαθέσιμη.

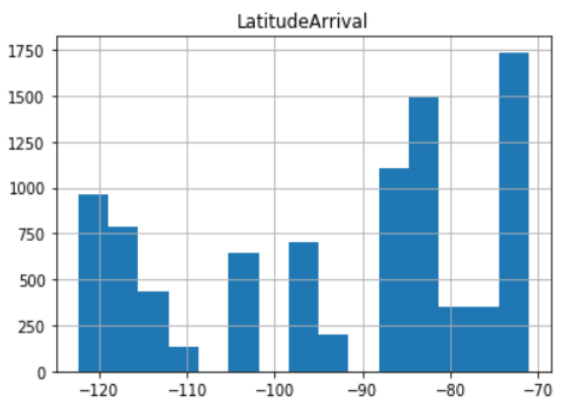
Αρχικά, στο πρόγραμμά μας “φορτώνουμε” τα αρχεία train και test που μας δόθηκαν από την εκφώνηση της εργασίας σε αντικείμενα pandas Dataframe. Αφού διαβάσουμε τα αρχεία μπορούμε πλέον να δούμε τις μεταβλητές που περιέχει το αρχείο train καθώς και τον τύπο τους: DateOfDeparture (object), Departure (object), CityDeparture (object), LongitudeDeparture (float64), LatitudeDeparture (float64), Arrival (object), CityArrival (object), LongitudeArrival (float64), LatitudeArrival (float64), WeeksToDeparture (float64), std_wtd (float64), PAX (int64). Το training set που δόθηκε αποτελείται από 8899 γραμμές και 12 στήλες. Το test set αποτελείται από 2229 γραμμές και 11 στήλες καθώς δεν περιέχει τη στήλη pax. Χρησιμοποιούμε επίσης την εντολή “describe” προκειμένου να αντλήσουμε περισσότερα στοιχεία για κάθε μεταβλητή ώστε να βγάλουμε κάποιο συμπέρασμα.

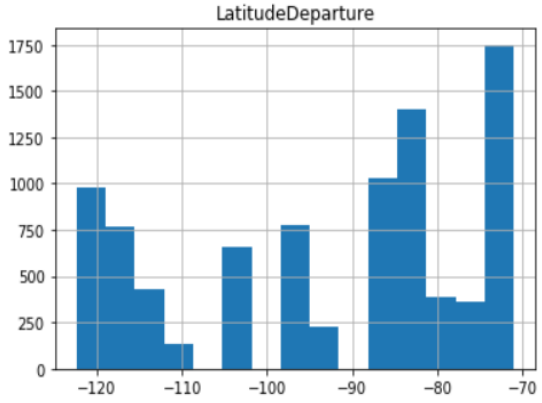
Στις πρώτες προσπάθειες πειραματιστήκαμε με τον αρχικό κώδικα που μας δόθηκε στο notebook. Ακολουθήσαμε τα βήματα όπως περιγράφονται δηλαδή πρώτα κάναμε drop όλες τις στήλες εκτός από τις Departure και Arrival και στα δυο set, χρησιμοποιήσαμε Label Encoder για να μετατρέψουμε τα αλφαριθμητικά σε αριθμητικές τιμές και τέλος ακολουθήσαμε τον αλγόριθμο Logistic Regression. Το σκορ για το πρόγραμμα αυτό ήταν 0.23.
Στη δεύτερη προσπάθεια προσθέσαμε τον One-Hot Encoder όπως περιγράφεται στο notebook για τις μεταβλητές Departure και Arrival και χρησιμοποιήσαμε πάλι Logistic Regression. Το σκορ που σημείωσε στην πλατφόρμα ήταν ελαφρώς βελτιωμένο και ίσο με 0.33.

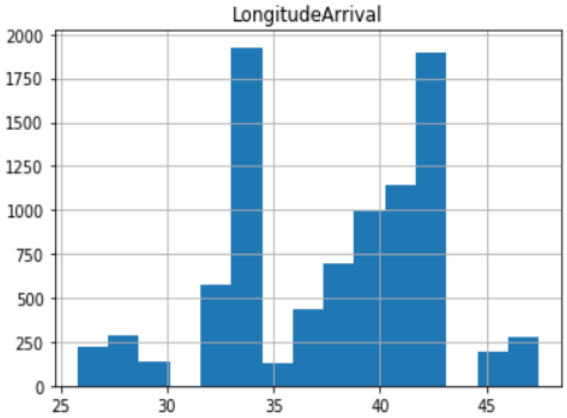
Έπειτα, δοκιμάσαμε και άλλους αλγόριθμους που δόθηκαν στα εργαστήρια με βάση τις δυο αυτές μεταβλητές αλλά είτε δεν βελτιωνόταν σημαντικά είτε μειωνόταν. Σκεφτήκαμε λοιπόν να λάβουμε υπόψιν μας και άλλες μεταβλητές. Αρχικά τις λάβαμε υπόψιν όλες και υπήρξε μια μικρή βελτίωση. Η επόμενη σκέψη μας ήταν να λάβουμε υπόψιν μας τις μεταβλητές ίδιου τύπου object και να “πετάξουμε” τις άλλες. Σε αυτή την προσπάθεια το σκορ μας βελτιώθηκε αισθητά. Δοκιμάσαμε σε αυτές τις μεταβλητές πάλι όλους τους αλγόριθμους που δόθηκαν στα εργαστήρια αλλά το καλύτερο σκορ το έδωσε ο αλγόριθμος Logistic Regression.

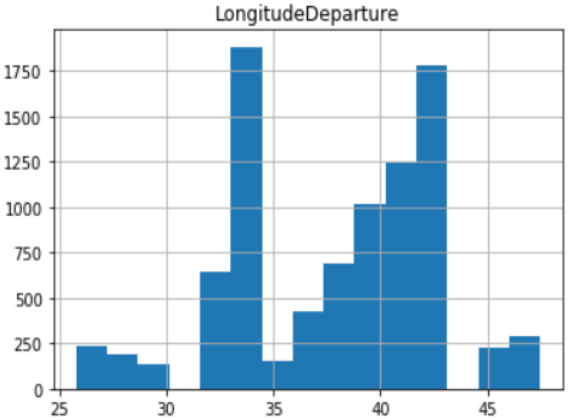
Έτσι συμπεράναμε ότι πρέπει να βρούμε έναν αποτελεσματικό τρόπο να συνδυάσουμε τα δεδομένα που μας δίνονται για αρχή και στη συνέχεια να βρούμε και έναν ακόμα πιο αποτελεσματικό και ακριβή αλγόριθμο.

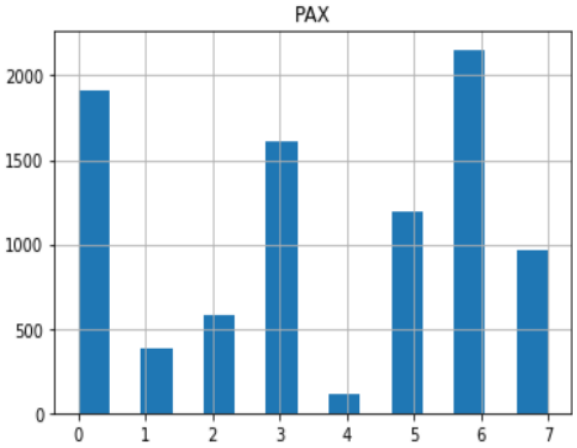
Χρησιμοποιήσαμε λοιπόν την εντολή hist για να εξάγουμε τα ιστογράμματα για τις μεταβλητές μας προκειμένου να εξάγουμε περαιτέρω συμπεράσματα. Τα ιστογράμματα παρουσιάζονται παρακάτω:

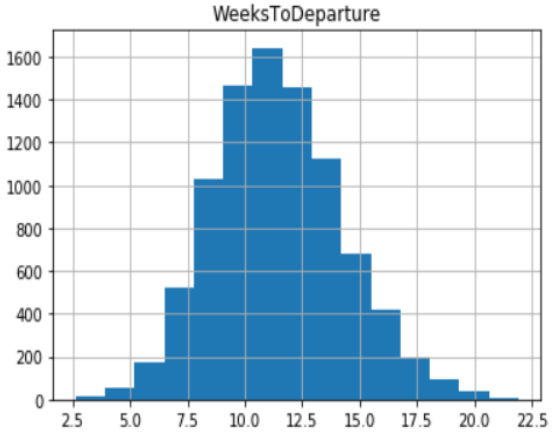


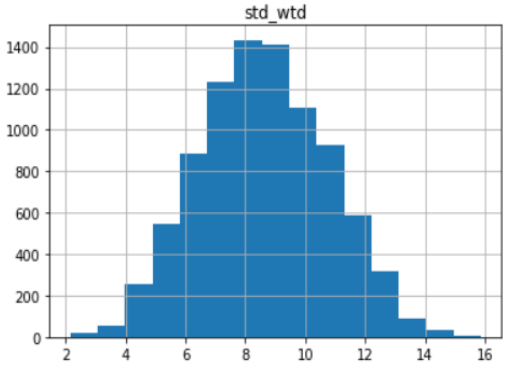












Από τα ιστογράμματα παρατηρήσαμε ότι το ιστόγραμμα της μεταβλητής LatitudeArrival μοιάζει πολύ με της LatitudeDeparture. Το ίδιο και της μεταβλητής LongitudeArrival με την LongitudeDeparture καθώς και της μεταβλητής WeeksToDeparture με της std_wtd.
Προσπαθήσαμε να βρούμε μια σχέση μεταξύ των μεταβλητών WeeksToDeparture και std_wtd αλλά δεν καταλήξαμε κάπου. Το ίδιο και για τις υπόλοιπες.

Προσπαθήσαμε με την χρήση του αλγόριθμου RandomForest να βρούμε ποιες μεταβλητές είναι πιο σημαντικές ώστε να αξιοποιήσουμε αυτές και να “πετάξουμε” τις υπόλοιπες. Ωστόσο το σκορ που πετύχαμε δεν ήταν ικανοποιητικό.

Τελικά καταλήξαμε στο να “σπάσουμε” το DateOfDeparture σε ένα dataframe με όνομα date και στήλες year, month, day και weekday και στα δύο set.
Στη συνέχεια χρησιμοποιήσαμε των LabelEncoder όπως στη αρχική ιδέα. Έπειτα, ενοποιήσαμε τον X_train πίνακα με τον date που φτιάξαμε. Μετά από πολλές δοκιμές, όπως φαίνεται και από τα σχόλια στον κώδικά μας, διαπιστώσαμε ότι ο καλύτερος συνδυασμός μεταβλητών είναι να κρατήσουμε όλες τις μεταβλητές εκτός από την “day” και την “week” και να έχουμε 505 iterations.
Χρησιμοποιούμε επίσης τον One-Hot Encoder.
Παρόμοια μετά από πολλαπλές δοκιμές καταλήξαμε στο να επιλέξουμε νευρωνικά δίκτυα και συγκεκριμένα τον classifier MLP. Έτσι το τελικό σκορ μας έφτασε το 0.64371.
Ο συνολικός χρόνος εκτέλεσης είναι 67,6235.