ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ **ΕΥΡΩΠΑΪΚΗ ΕΝΩΣΗ**

ΠΕΡΙΦΕΡΕΙΑ ΗΠΕΙΡΟΥ

ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ

ΔΙΑΧΕΙΡΙΣΗΣ

Ε.Π. ΠΕΡΙΦΕΡΕΙΑΣ ΗΠΕΙΡΟΥ

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ

“Ηπειρος” 2014-2020

ΚΩΔΙΚΟΣ ΠΡΑΞΗΣ (ΕΡΓΟΥ): ΗΠ1ΑΒ-00169

ΕΠΩΝΥΜΙΑ ΦΟΡΕΑ: Όμιλος Κολιού ΑΒΕΕ.

Εξαγωγές αγροτικών προϊόντων

Ακτινίδια - Πορτοκάλια - Μανταρίνια - Φράουλες

ΣΥΝΤΟΜΟΓΡΑΦΙΑ ΦΟΡΕΑ: Όμιλος Κολιού ΑΒΕΕ

Τίτλος Πρότασης

Σύγχρονο σύστημα αξιολόγησης της ποιότητας των ακτινιδίων, ιχνηλασιμότητα των παραγόμενων προϊόντων ακτινιδίου και ευφυής διαχείριση της αλυσίδας εφοδιασμού με βάση προηγμένες εφαρμογές Πληροφορικής ICT - FoodAware

ΙΑΝΟΥΑΡΙΟΣ, 2022

ΕΡΕΥΝΗΤΙΚΗ ΕΝΟΤΗΤΑ 6

ΠΑΡΑΔΟΤΕΟ: ΠΕ6.2

Τεχνική Έκθεση Ανάλυσης και απαιτήσεων της διαδικτυακής υπηρεσίας των αλγορίθμων εξόρυξης δεδομένων

ΠΕΡΙΛΗΨΗ

Η ακριβής πρόβλεψη της θερμοκρασίας του αέρα είναι μια δύσκολη διαδικασία, αλλά είναι και βοηθητική και, σε πολλές περιπτώσεις, απαραίτητη για την λήψη αποφάσεων στον τομέα της γεωργίας. Το παρόν παραδοτέο ασχολείται με την ανάπτυξη ενός μοντέλου που θα είναι ικανό να προβλέπει την θερμοκρασία του αέρα. Η πρόβλεψη της θερμοκρασίας του αέρα θα επιτρέψει την έγκαιρη λήψη αποφάσεων για πρόληψη ζημιάς στα ακτινίδια και την ασφάλεια των εργατών. Για την επίτευξη αυτού του στόχου, θα γίνει χρήση, ανάλυση και επεξεργασία χρονοσειρών. Οι χρονοσειρές, είναι δεδομένα που έχουν ως ένα από τα κύρια χαρακτηριστικά τον χρόνο. Για την δημιουργία του μοντέλου πρόβλεψης αξιοποιούνται σύγχρονες τεχνολογίες που εφαρμόζονται στο πεδίο της εξόρυξης δεδομένων και μηχανικής μάθησης, όπως η γλώσσα προγραμματισμού Python, η βιβλιοθήκη Keras και τα αναδρομικά νευρωνικά δίκτυα, τα οποία είναι αποτελεσματικά στην πρόβλεψη χρονοσειρών.

ΠΕΡΙΕΧΟΜΕΝΑ

1 Εισαγωγή 7

1.1 Σκοπός της εφαρμογής 7

2 Τεχνολογίες ανάπτυξης 9

2.1 Python 9

2.1.1 Ιστορική αναδρομή 10

2.1.2 Πεδία ανάπτυξης εφαρμογών Python 10

2.1.3 Χαρακτηριστικά της Python 11

2.1.4 Πλεονεκτήματα και μειονεκτήματα της Python 12

2.1.5 Πακέτα 13

2.1.6 NumPy 14

2.1.7 Pandas 16

2.1.8 Scikit-learn 19

2.1.9 Matplotlib 21

2.1.10 Seaborn 23

2.1.11 requests 23

2.2 Keras 25

2.2.1 Τί είναι το Keras 25

2.2.2 Πλεονεκτήματα του Keras 27

2.2.3 Δημιουργία μοντέλου με Keras 27

2.2.4 Tensorflow 28

2.2.5 Tensorflow vs Keras 29

2.3 Jupyter Notebook 30

2.4 Ημι-δομημένα δεδομένα 31

2.5 CSV 32

3 Δεδομένα 33
3.1 Χρονοσειρές 33
3.1.1 Naïve-SNaive 34
3.1.2 Arima-Sarima 35
3.1.3 Seasonal Decomposition 36
3.1.4 Αναδρομικά νευρωνικά δίτκυα 37
3.2 Διαδικασία Εξόρυξης Δεδομένων 37
3.2.1 Συλλογή των δεδομένων 38
3.2.2 Το σύνολο δεδομένων 39
3.2.3 Ορισμός προβλήματος και επιλογή χαρακτηριστικών 41
3.2.4 Προεπεξεργασία των δεδομένων 42
4 Υλοποίηση 48
4.1 Αναδρομικά Νευρωνικά Δίκτυα 48
4.1.1 LSTM 50
4.1.2 Αρχιτεκτονική LSTM 50
4.2 Εκπαίδευση 52
4.2.1 Δημιουργία συνόλου εκπαίδευσης 52
4.2.2 Μοντέλο 52
4.2.3 Επιλογή αλγορίθμου βελτιστοποίησης 53
4.3 Αποθήκευση και φόρτωση του μοντέλου 54
4.4 Πρόβλεψη 55
4.5 Αποτελέσματα 56
5 Βιβλιογραφία 58

1 Εισαγωγή

Η ακριβής πρόβλεψη της θερμοκρασίας αποτελεί ένα απαραίτητο συστατικό και στη λήψη αποφάσεων στη γεωργική παραγωγή. Ακραίες θερμοκρασίες μπορεί να προκαλέσουν ζημιές στα ακτινίδια, οι οποίες οδηγούν σε μείωση της παραγωγής και στη δημιουργία επικίνδυνων καιρικών συνθηκών εργασίας για αυτούς που εργάζονται στη φύση.

Η ανάλυση και η πρόβλεψη καιρικών συνθηκών είχε πάντα μεγάλο ενδιαφέρον. Στην εποχή μας, μια ακριβής πρόβλεψη του καιρού γίνεται όλο και πιο σημαντική σε πολλούς τομείς της κοινωνίας και της οικονομίας, της διαχείρισης της ασφαλούς κυκλοφορίας αεροπλάνων και πλοίων, της προστασίας από φυσικές καταστροφές και της γεωργίας.

Η πρόβλεψη θα επιτευχθεί με την χρήση τεχνικών εξόρυξης δεδομένων και μηχανικής μάθησης. Η μηχανική μάθηση, γνωστή και ως στατιστική μάθηση, είναι κλάδος της τεχνητής νοημοσύνης που αποσκοπεί στην δημιουργία αλγορίθμων για πρόβλεψη και λήψη αποφάσεων. Η έννοια της εξόρυξης δεδομένων με την μηχανική μάθηση είναι στενά συνδεδεμένη, και πολλές φορές χρησιμοποιούνται σαν συνώνυμα. Αυστηρώς ορισμένα, η εξόρυξη δεδομένων ασχολείται με την μελέτη δεδομένων, κυρίως μεγάλα δεδομένα, ενώ η μηχανική μάθηση στοχεύει στην εκμάθηση από τα δεδομένα που δίνονται στο μοντέλο, για την πρόβλεψη σε απαρατήρητα δεδομένα.

1.1 Σκοπός της εφαρμογής

Σκοπός της εφαρμογής είναι η δημιουργία ενός μοντέλου που μπορεί να προβλέπει την θερμοκρασία του αέρα, καθώς αποτελεί ένα σημαντικό παράγοντα στην παροχή μιας πρώτης προειδοποίησης τόσο για καύσωνες, όσο και για πολύ χαμηλές θερμοκρασίες. Με μια καλή πρόβλεψη, η χρήση μέτρων πρόληψης μπορεί να πραγματοποιηθεί εγκαίρως, με αποτέλεσμα να μειώνονται οι απώλειες ακτινιδίων και να εργάζονται με μεγαλύτερη ασφάλεια οι εργάτες.

Το μοντέλο που θα χρησιμοποιηθεί για την πρόβλεψη είναι ένα νευρωνικό δίκτυο LSTM, που βασίζεται στην αρχιτεκτονική ενός αναδρομικού νευρωνικού δικτύου. Για την δημιουργία του μοντέλου θα χρησιμοποιηθεί η γλώσσα προγραμματισμού Python και η βιβλιοθήκη Keras. Τέλος, τα δεδομένα που θα χρησιμοποιηθούν, παρέχονται από μια υπηρεσία ιστού, από την οποία συλλέγονται δεδομένα από τέσσερις διαφορετικούς σταθμούς, από Άρτα και έναν από Ιωάννινα.

2 Τεχνολογίες ανάπτυξης

2.1 Python

Η Python είναι μια σύγχρονη, γενικού σκοπού και, υψηλού επιπέδου γλώσσα προγραμματισμού. Είναι σχετικά εύκολη η αναγνωσιμότητα και η εκμάθηση του κώδικά της και είναι αρκετά εκφραστική, με αποτέλεσμα να είναι εύκολη η συγγραφή κώδικα σε Python, σε αντίθεση με άλλες γλώσσες προγραμματισμού, όπως η C++ και η Java. Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες.

Η Python έχει τη φήμη μιας γλώσσας φιλικής προς τους αρχάριους, αντικαθιστώντας την Java ως την πιο ευρέως χρησιμοποιούμενη εισαγωγική γλώσσα, επειδή χειρίζεται μεγάλο μέρος της πολυπλοκότητας για τον χρήστη, επιτρέποντας στους αρχάριους να επικεντρωθούν στην πλήρη κατανόηση των εννοιών του προγραμματισμού και όχι στις μικρολεπτομέρειες.

Η Python υποστηρίζει διάφορα προγραμματιστικά παραδείγματα, όπως αντικειμενοστραφή και συναρτησιακό προγραμματισμό. Είναι δυναμική γλώσσα προγραμματισμού (dynamically typed), δηλαδή επιτρέπει την αλλαγή τύπων των μεταβλητών κατά την διάρκεια εκτέλεσης του κώδικα και δεν απαιτείται η δήλωση τύπων μεταβλητών από τον προγραμματιστή. Επιπλέον, είναι μια διερμηνευόμενη (interpreted) γλώσσα προγραμματισμού, που σημαίνει ότι οι εντολές εκτελούνται μία προς μία, που την κάνει πιο ευέλικτη, αλλά και να υστερεί σε ταχύτητα σε σχέση με μεταγλωττιζόμενες γλώσσες όπως η C και η C++. Τέλος, υποστηρίζει συλλογή απορριμμάτων (garbage collection).

Οι διερμηνευτές της Python είναι διαθέσιμοι για εγκατάσταση σε μια πληθώρα από λειτουργικά συστήματα. Για Microsoft Windows υπάρχουν εκδόσεις 32 ή 64 bits, ενώ στα λειτουργικά συστήματα Linux και Mac OS X συνηθίζεται να είναι προ εγκατεστημένη. Με την χρήση εργαλείων από τρίτους, όπως το Py2exe και το Pyinstaller, ο κώδικας της Python μπορεί να μετατραπεί σε εκτελέσιμο αρχείο, επιτρέποντας την εκτέλεση του λογισμικού σε περιβάλλοντα, χωρίς να απαιτείται εγκατάσταση ενός διερμηνευτή της Python.

Εικόνα 2.1. Το Logo της γλώσσας προγραμματισμού Python

(Πηγή: https://wikipedia.org/wiki/Python#/media/File:Python_logo_and_wordmark.svg)

2.1.1 Ιστορική αναδρομή

Η Python δημιουργήθηκε από τον Ολλανδό Γκίντο βαν Ρόσσουμ (Guido van Rossum) το 1989 στο ερευνητικό κέντρο Centrum Wiskunde & Informatica (CWI) και κυκλοφόρησε για πρώτη φορά το 1991. Βασική έμπνευση για τον Γκίντο βαν Ρόσσουμ ήταν η γλώσσα προγραμματισμού ABC.

Η Python 2.0 κυκλοφόρησε το 2000 και το 2008 κυκλοφόρησε η έκδοση 3.0. Η Python 3.0 είναι η πρώτη γλώσσα προγραμματισμού που σπάει την προς τα πίσω συμβατότητα με προηγούμενες εκδόσεις, για να αντιμετωπιστούν κάποια λάθη που υπήρχαν σε παλαιότερες εκδόσεις και να γίνει ακόμα πιο απλή και σαφής η συγγραφή κώδικα σε αυτήν.

2.1.2 Πεδία ανάπτυξης εφαρμογών Python

Η Python μπορεί να χρησιμοποιηθεί για την ανάπτυξη ποικίλων εφαρμογών όπως:

- **Εφαρμογές ιστού:** Διαθέτει frameworks όπως Django, Streamlit, Flask τα οποία συμβάλλουν στο να γίνουν οι διαδικασίες εφαρμογές ιστού απλές και εύκολες.
- **Επιστημονικοί και αριθμητικοί υπολογισμοί:** Με τα πακέτα Python, όπως τα Pandas και NumPy, οι επιστημονικοί και αριθμητικοί υπολογισμοί μπορούν να γίνουν αποτελεσματικά.
- **Προγραμματισμός δικτύων:** Η Python διευκολύνει τη δημιουργία σεναρίων που αυτοματοποιούν τη διαμόρφωση πολύπλοκων δικτύων. Για τη δικτύωση που καθορίζεται από λογισμικό, είναι η πιο ευρέως χρησιμοποιούμενη γλώσσα προγραμματισμού.

- **Παιχνίδια και τρισδιάστατες εφαρμογές:** Η Python αποτελεί μια αξιόπιστη γλώσσα για τη δημιουργία ενός απλού τρισδιάστατου παιχνιδιού με τη χρήση του Pygame, πράγμα που την καθιστά αποτελεσματικό εργαλείο για την κατασκευή προτύπων.
- **Πρωτότυπα λογισμικού:** Η Python αποτελεί εξαιρετική γλώσσα για την ανάπτυξη πρωτοτύπων, δοκιμών και εργαλείων εντοπισμού σφαλμάτων.
- **Blockchain:** Παρόλο που μπορεί να υπάρχουν μερικές δημοφιλείς γλώσσες για την ανάπτυξη blockchain, όπως η JavaScript, η Java και άλλες, η Python αποδεικνύεται ως μια ισχυρή γλώσσα. Όπως και σε άλλες χρήσεις, η Python συνιστάται για την ανάπτυξη blockchain λόγω της υψηλής ευελιξίας και λειτουργικότητάς της που ενισχύεται από την ασφάλειά της.

2.1.3 Χαρακτηριστικά της Python

Ορισμένα από τα χαρακτηριστικά της Python είναι (Python - Overview, 2022):

- **Εύκολη εκμάθηση:** Η Python έχει λίγες λέξεις – κλειδιά, απλή δομή και σαφώς καθορισμένο συντακτικό επιτρέποντας έτσι την γρήγορη κατανόησή της.
- **Εύκολη ανάγνωση:** Ο κώδικας σε Python είναι πιο σαφής και ορατός στα μάτια.
- **Εύκολη συντήρηση:** Ο πηγαίος κώδικας σε Python είναι αρκετά εύκολος στη συντήρηση.
- **Ευρεία τυποποιημένη βιβλιοθήκη:** Ο κύριος όγκος της βιβλιοθήκης της Python είναι συμβατός με πολλαπλές πλατφόρμες όπως Unix, Macintosh και Windows.
- **Δια δραστική λειτουργία:** Υποστηρίζει μια λειτουργία που επιτρέπει τη δια δραστική δοκιμή και αποσφαλμάτωση αποσπασμάτων κώδικα.
- **Βάσεις δεδομένων:** Παρέχει διεπαφές σε όλες τις μεγάλες εμπορικές βάσεις δεδομένων.

- **Προγραμματισμός GUI:** Η Python υποστηρίζει εφαρμογές GUI που μπορούν να δημιουργηθούν και να μεταφερθούν σε άλλα συστήματα και πλατφόρμες.
- **Επεκτάσιμη:** Υπάρχει δυνατότητα προσθήκης χαμηλού επιπέδου modules στον διερμηνέα της Python επιτρέποντας την προσαρμογή των εργαλείων της ώστε να γίνουν πιο αποδοτικά.
- **Ευρεία υποστήριξη δεδομένων:** Παρέχει δυναμικούς τύπους δεδομένων πολύ υψηλού επιπέδου και υποστηρίζει δυναμικό έλεγχο τύπων.
- **Ενσωμάτωση:** Μπορεί εύκολα να ενσωματωθεί με C, C++, COM, ActiveX, CORBA και Java.

2.1.4 Πλεονεκτήματα και μειονεκτήματα της Python

Ορισμένα από τα πλεονεκτήματα της Python είναι τα εξής:

- Η Python είναι δωρεάν και ανοιχτή, ώστε ο καθένας να μπορεί να τη κατεβάσει και να τη χρησιμοποιήσει αμέσως.
- Αποτελεί γλώσσα προγραμματισμού υψηλού επιπέδου με σύνταξη που είναι παρόμοια στην αγγλική γεγονός που την καθιστά εύκολη επιλογή για την κατανόηση και εκμάθηση από αρχάριους.
- Επειδή ο κώδικας είναι απλός, η παραγωγικότητα είναι συγκριτικά υψηλότερη από άλλες γλώσσες προγραμματισμού.
- Σε περίπτωση εμφάνισης σφάλματος, η Python σταματά την κωδικοποίηση μέχρι να επιλυθεί το σφάλμα συμβάλλοντας έτσι στη δημιουργία κώδικα χωρίς σφάλματα.
- Είναι ανεξάρτητη από το σύστημα, πράγμα που σημαίνει ότι δεν χρειάζεται να αλλάξει ο κώδικας όταν γίνεται χρήση σε διαφορετικές πλατφόρμες.
- Διαθέτει πολυάριθμα πακέτα στην βιβλιοθήκη της βοηθώντας τους χρήστες να εργάζονται σε διάφορες εφαρμογές με ευκολία.

Ορισμένα από τα μειονεκτήματα της Python είναι τα εξής:

- Η διαδικασία εκτέλεσης είναι σχετικά πιο αργή.
- Οι δομές της Python χρειάζονται πρόσθετη μνήμη.

- Μπορεί να οδηγήσει σε σφάλματα κατά τη διάρκεια εκτέλεσης.
- Δεν είναι η καλύτερη επιλογή όταν αλληλοεπιδρά με βάσεις δεδομένων.
- Η επεξεργαστική ισχύ της είναι αργή σε σύγκριση με άλλες γλώσσες.

2.1.5 Πακέτα

Μια συνηθισμένη διαδικασία κατά την συγγραφή κώδικα Python, είναι η εγκατάσταση και χρήση πακέτων. Τα πακέτα είναι κώδικες, που παρέχουν υλοποιημένους αλγόριθμους, με αποτέλεσμα να μην χρειαστεί η υλοποίησή τους εκ νέου.

Μία από τις πιο βασικές βιβλιοθήκες της Python είναι η πρότυπη βιβλιοθήκη της Python (Python Standard Library). Η βιβλιοθήκη αυτή περιέχει το ακριβές συντακτικό, τη σημασιολογία και τα tokens της Python. Η τυπική βιβλιοθήκη της Python παίζει πολύ σημαντικό ρόλο. Χωρίς αυτήν, οι προγραμματιστές δεν μπορούν να έχουν πρόσβαση στις λειτουργίες της Python. Περιέχει ενσωματωμένες ενότητες που παρέχουν πρόσβαση σε βασικές λειτουργίες του συστήματος, όπως είναι οι λειτουργίες εισόδου / εξόδου και άλλες βασικές ενότητες. Οι περισσότερες από τις βιβλιοθήκες της Python είναι γραμμένες σε γλώσσα προγραμματισμού C. Η τυπική βιβλιοθήκη της Python αποτελείται από περισσότερες από 200 βασικές ενότητες. Όλα αυτά συνεργάζονται για να κάνουν την Python μια γλώσσα προγραμματισμού υψηλού επιπέδου.

Για την εύκολη εγκατάσταση πακέτων απαιτείται η χρήση ενός διαχειριστή πακέτων (package manager). Ο διαχειριστής πακέτων της Python είναι το PIP (Package Installer for Python). Το PIP επιτρέπει την αναζήτηση και εγκατάσταση πακέτων από το κεντρικό αποθετήριο πακέτων Python, που διατίθενται στο Ευρετήριο πακέτου Python. (PyPi,2022)

Τα πακέτα που χρησιμοποιούνται σε αυτό το έργο είναι τα εξής:

- NumPy
- Pandas
- Scikit-learn
- Matplotlib

- Seaborn
- requests

```
py -m pip install [options] <requirement specifier> [package-index-options] ...
py -m pip install [options] -r <requirements file> [package-index-options] ...
py -m pip install [options] [-e] <vcs project url> ...
py -m pip install [options] [-e] <local project path> ...
py -m pip install [options] <archive url/path> ...
```

Εικόνα 2.2. Εντολές εγκατάστασης πακέτων σε Windows. (pip install - pip documentation v22.1.2, 2022)

```
python -m pip install [options] <requirement specifier> [package-index-options] ...
python -m pip install [options] -r <requirements file> [package-index-options] ...
python -m pip install [options] [-e] <vcs project url> ...
python -m pip install [options] [-e] <local project path> ...
python -m pip install [options] <archive url/path> ...
```

Εικόνα 2.3. Εντολές εγκατάστασης πακέτων σε Unix/macOS. (pip install - pip documentation v22.1.2, 2022)

2.1.6 NumPy

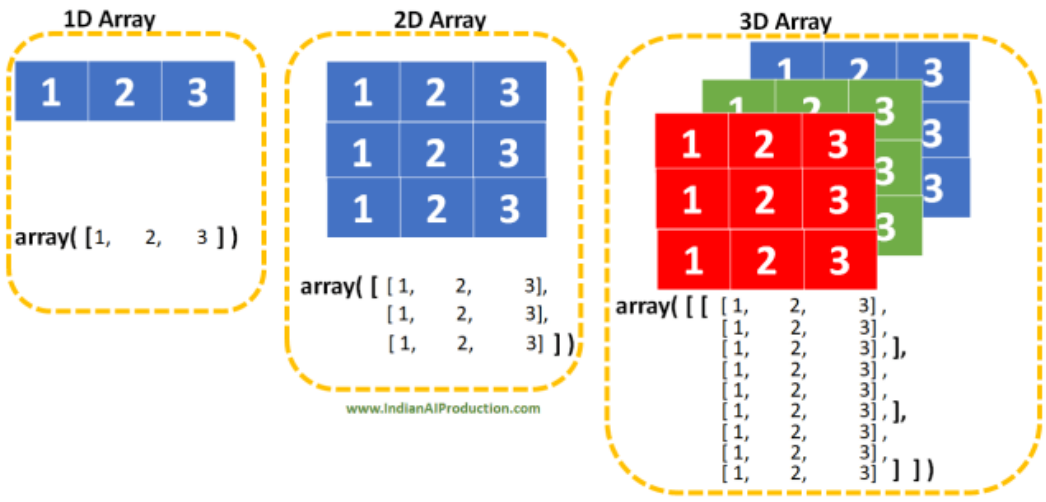
Το πακέτο NumPy (Numerical Python) παρέχει υποστήριξη για συναρτήσεις και δομές δεδομένων για την υποστήριξη μεγάλων, πολυδιάστατων μητρών (matrices) και τις απαραίτητες μαθηματικές πράξεις και τελεστές για τον χειρισμό τους. (What is NumPy? — NumPy v1.22 Manual, 2022)

Εικόνα 2.4. Το logo του πακέτου Numpy. (Πηγή: https://upload.wikimedia.org/wikipedia/commons/3/31/NumPy_logo_2020.svg)

Ένα σύνολο δεδομένων είναι μια συλλογή από δεδομένα. Η μορφή τους, συνήθως, είναι παρόμοια με έναν πίνακα μιας Βάσης Δεδομένων. Δηλαδή, οι στήλες αντιπροσωπεύουν τα χαρακτηριστικά- μεταβλητές των δεδομένων, ενώ οι σειρές τις εγγραφές. Η εφαρμογή αλγορίθμων Μηχανικής Μάθησης και Εξόρυξης Δεδομένων επιταχύνεται (και πολλές φορές απαιτείται να είναι) με πράξεις μητρών. Η Python είναι πολύ πιο αργή σε σχέση με το πακέτο Numpy. Υπάρχουν δύο κύριοι λόγοι για αυτό:

1. Το πακέτο Numpy είναι γραμμένο σε C, και ο μεταγλωττιζόμενος κώδικας C θα είναι πάντα πιο γρήγορος από την Python, η οποία είναι μια διερμηνευόμενη γλώσσα.
2. Το πακέτο Numpy έχει έναν μηχανισμό που προσδιορίζει τον τύπο δεδομένων και επιτρέπει μόνο ομογενή δεδομένα (σε αντίθεση με την απλή λίστα της Python), με αποτέλεσμα να υπάρχει μια επιπλέον βελτιστοποίηση στις πράξεις, καθώς μαθηματικές πράξεις μεταξύ ετερογενών δεδομένων είναι αποτελεσματικές. (NumPy: the absolute basics for beginners — NumPy v1.24.dev0 Manual, 2022)

Τέλος, τα πακέτα Scikit-learn, Matplotlib, Seaborn έχουν υποστήριξη για το πακέτο NumPy, ενώ το πακέτο Pandas είναι χτισμένο πάνω στο Numpy.



Εικόνα 2.5. Αναπαράσταση πολυδιάστατων πινάκων τύπου ndarray του πακέτου Numpy.

Η δομή δεδομένων 'ndarray' ή n-διάστατων πινάκων είναι η κύρια λειτουργικότητα του Numpy. Αυτοί οι πίνακες είναι ομοιογενείς και όλα τα στοιχεία του πίνακα πρέπει να είναι του ίδιου τύπου. Ένας πίνακα ndarray είναι ένα (συνήθως) σταθερού μεγέθους δοχείο το οποίο περιέχει στοιχεία ίδιου τύπου και μεγέθους. Ο αριθμός των διαστάσεων και των στοιχείων σε έναν πίνακα ορίζεται από το σχήμα του, το οποίο είναι μια πλειάδα από N μη αρνητικούς ακέραιους αριθμούς που καθορίζουν τα μεγέθη κάθε διάστασης. Ο τύπος των στοιχείων του πίνακα καθορίζεται από ένα ξεχωριστό αντικείμενο τύπου δεδομένων (dtype), ένα από τα οποία σχετίζεται με κάθε ndarray.

Με την χρήση της βιβλιοθήκης NumPy μπορούν να εκτελεστούν οι ακόλουθες λειτουργίες (What Is Numpy Used For In Python?, 2022):

- Πολλαπλασιασμός μεταξύ διανυσμάτων
- Πολλαπλασιασμός πινάκων – μητρών και πινάκων – διανυσμάτων
- Στοιχειομετρικές πράξεις σε διανύσματα και πίνακες (όπως πρόσθεση, αφαίρεση, πολλαπλασιασμός και διαίρεση με έναν αριθμό)
- Συγκρίσεις κατά στοιχείο ή κατά πίνακα
- Εφαρμογή στοιχειομετρικών συναρτήσεων σε ένα διάνυσμα – πίνακα (όπως pow, log και exp)
- Πράξεις που σχετίζονται με τη γραμμική άλγεβρα καθώς διαθέτει ενσωματωμένες συναρτήσεις για γραμμική άλγεβρα και παραγωγή τυχαίων αριθμών.
- Μετασχηματισμούς Fourier και ρουτίνες για χειρισμό σημάτων

2.1.7 Pandas

Το πακέτο pandas παρέχει συναρτήσεις και δομές δεδομένων για την εύκολη ανάγνωση δεδομένων από αρχεία (π.χ. csv) και χειρισμό αυτών. Η ονομασία της βιβλιοθήκης αυτής προέρχεται από το Pan (Panel) πίνακας και το Das (Data) δεδομένα. Επιτρέπει τον εύκολο χειρισμό δεδομένων που λείπουν (missing values), τη λήψη υποσυνόλου δεδομένων που πληρούν συγκεκριμένα

κριτήρια, τον χειρισμό χρονοσειρών κλπ. (pandas - Python Data Analysis Library, 2022)

Εικόνα 2.6. Το logo του πακέτου pandas. (Πηγή: https://upload.wikimedia.org/wikipedia/commons/e/ed/Pandas_logo.svg)

Το πακέτο Pandas παρέχει εξαιρετικά εξορθολογισμένες μορφές εκπροσώπησης δεδομένων. Βασική λειτουργία του πακέτου Pandas είναι η κλάση DataFrame. Το Pandas DataFrame είναι μια δισδιάστατη δομή δεδομένων με δυνατότητα αλλαγής μεγέθους δυνητικά ετερογενής, με πίνακες δεδομένων και με επισημασμένους άξονες ευθυγραμμίζοντας τα δεδομένα σε γραμμές και στήλες. Τα DataFrame αποτελούνται από τρία κύρια συστατικά τα δεδομένα, τις γραμμές και τις στήλες. Αυτό βοηθά στην καλύτερη ανάλυση και κατανόηση των δεδομένων. Η απλούστερη εκπροσώπηση δεδομένων συνεπάγεται καλύτερα αποτελέσματα. Επιπλέον, η χρήση του πακέτου Pandas συμβάλλει στη μείωση της διαδικασίας χειρισμού δεδομένων, με αποτέλεσμα να μπορούμε να επικεντρωθούμε περισσότερο στους αλγορίθμους ανάλυσης και επεξεργασίας δεδομένων. (Murugesan, 2022)

Τέλος, το πακέτο Pandas παρέχει ένα τεράστιο σύνολο σημαντικών εντολών και χαρακτηριστικών, που χρησιμοποιούνται για την εύκολη ανάλυση των δεδομένων. Μπορεί να χρησιμοποιηθεί για διάφορες εργασίες, όπως το φιλτράρισμα των δεδομένων σύμφωνα με ορισμένες προϋποθέσεις, ή την κατάτμηση και τον διαχωρισμό των δεδομένων σύμφωνα με τις προτιμήσεις κλπ.

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0.0	PG	25.0	6-2	180.0	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99.0	SF	25.0	6-6	235.0	Marquette	6796117.0
2	John Holland	Boston Celtics	30.0	SG	27.0	6-5	205.0	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28.0	SG	22.0	6-5	185.0	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8.0	PF	29.0	6-10	231.0	NaN	5000000.0

Εικόνα 2.7. Μορφή αποτύπωσης ενός συνόλου δεδομένων στο Pandas με χρήση DataFrame.

Οι κυριότεροι τομείς στους οποίους η βιβλιοθήκη Pandas είναι:

- Χρηματοοικονομικά
- Οικονομικά
- Ανάλυση

Το Pandas διευκολύνει την εκτέλεση πολλών από τις χρονοβόρες και επαναλαμβανόμενες εργασίες που σχετίζονται με την επεξεργασία και διαχείριση δεδομένων όπως:

- Καθαρισμός δεδομένων
- Συμπλήρωση δεδομένων
- Κανονικοποίηση δεδομένων
- Συγχωνεύσεις και ενώσεις
- Οπτικοποίηση δεδομένων
- Φιλτράρισμα των δεδομένων
- Εισαγωγή και διαγραφή στηλών δομών δεδομένων
- Αναδιαμόρφωση συνόλων δεδομένων
- Αντικείμενα DataFrame για χειρισμό δεδομένων με ενσωματωμένη ευρετηρίαση.
- Λειτουργίες χρονοσειρών όπως δημιουργία εύρους ημερομηνιών και μετατροπείς συχνοτήτων, στατιστικά στοιχεία κινούμενου παραθύρου, γραμμικές παλινδρομήσεις κινούμενου παραθύρου, μετατόπιση ημερομηνίας.
- Στατιστική ανάλυση
- Επιθεώρηση δεδομένων

- Φόρτωση και αποθήκευση δεδομένων

2.1.8 Scikit-learn

Το πακέτο Scikit-learn (γνωστό και ως sklearn) αναπτύχθηκε ως ένα ολοκληρωμένο πακέτο μηχανικής μάθησης με κάποιες πολύ σημαντικές αρχές σχεδιασμού. Το πακέτο έπρεπε να ήταν εύκολο στη χρήση ακόμα και για αρχάριους, ενώ ταυτόχρονα έπρεπε να έχει ευελιξία και πολλές επιλογές προσαρμογής και να είναι αποτελεσματικό. (Buitinck et al., 2022)

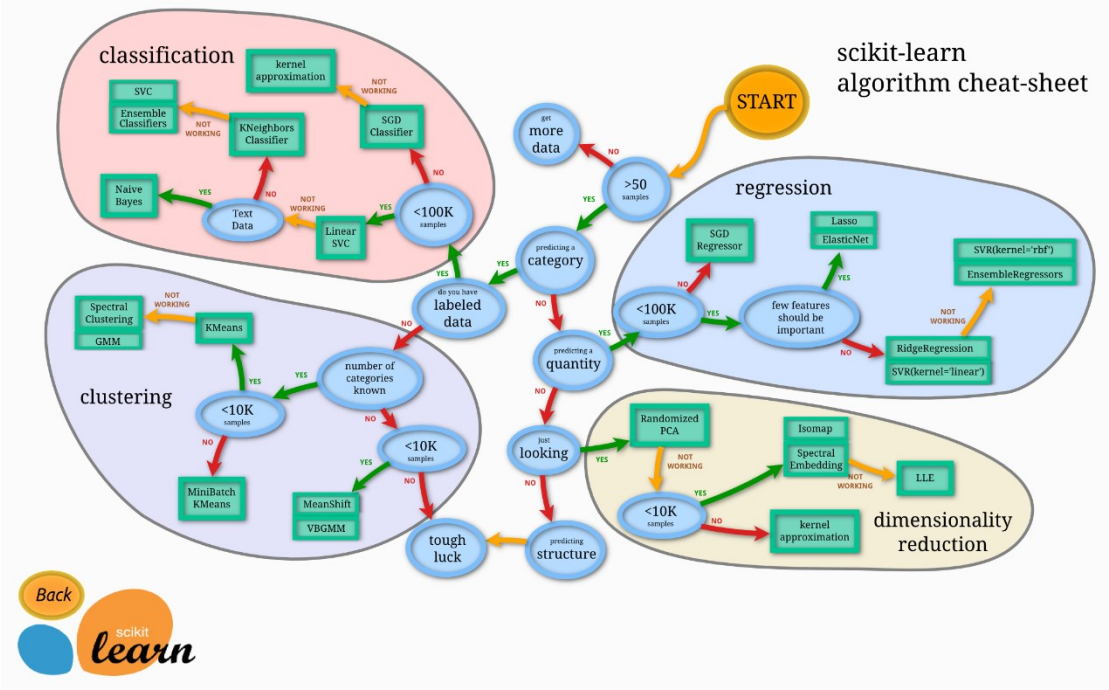
Η χρήση του πακέτου Scikit-learn μας επιτρέπει να επικεντρωθούμε στην δοκιμή, παραμετροποίηση και εύρεση των καλύτερων αλγορίθμων για ένα πρόβλημα, αφού η υλοποίηση των αλγορίθμων δίνεται από το πακέτο Scikit-learn.

Εικόνα 2.8. Το logo του πακέτου scikit - learn. (Πηγή: https://upload.wikimedia.org/wikipedia/commons/0/05/Scikit_learn_logo_small.svg)

Ορισμένα από τα χαρακτηριστικά της scikit – learn είναι (Loading..., 2022)

- **Υποστήριξη αλγορίθμων επιβλεπόμενης μάθησης:** Σχεδόν όλοι οι δημοφιλείς αλγόριθμοι επιβλεπόμενης μάθησης, όπως η γραμμική παλινδρόμηση, η μηχανή διανυσμάτων υποστήριξης (SVM), το δέντρο αποφάσεων κ.λπ. αποτελούν μέρος του scikit – learn.

- **Υποστήριξη μάθησης χωρίς επίβλεψη:** Διαθέτει επίσης όλους τους γνωστούς αλγορίθμους μάθησης χωρίς επίβλεψη, από ομαδοποίηση, ανάλυση παραγόντων, την ανάλυση κύριων παραγόντων (PCA) έως τα νευρωνικά δίκτυα χωρίς επίβλεψη.
- **Ομαδοποίηση (Clustering):** Το μοντέλο αυτό χρησιμοποιείται για την ομαδοποίηση δεδομένων χωρίς ετικέτες.
- **Μείωση διαστάσεων (Dimensionality Reduction):** Η μεθοδολογία αυτή χρησιμοποιείται για τη μείωση του αριθμού χαρακτηριστικών στα δεδομένα τα οποία θα χρησιμοποιηθούν στην περαιτέρω ανάλυση, οπτικοποίηση και επιλογή χαρακτηριστικών.
- **Διασταυρούμενη επικύρωση (Cross Validation):** Ελέγχει την ακρίβεια των εποπτευόμενων μοντέλων σε νέα δεδομένα.
- **Εξαγωγή χαρακτηριστικών (Feature extraction):** Χρησιμοποιείται για την εξαγωγή των χαρακτηριστικών από ένα σύνολο δεδομένων κειμένου ή εικόνας.
- **Επιλογή χαρακτηριστικών (Feature selection):** Χρησιμοποιείται κυρίως στα μοντέλα με επίβλεψη και συμβάλει στον προσδιορισμό των χρήσιμων χαρακτηριστικών για τη δημιουργία μοντέλων με επίβλεψη.
- **Μέθοδοι Ensemble:** Συμβάλει στον συνδυασμό των προβλέψεων πολλαπλών εποπτευόμενων μοντέλων.
- **Ανοιχτός κώδικας:** Αποτελεί βιβλιοθήκη ανοιχτού κώδικα και εμπορικά αξιοποιήσιμη με άδεια BSD.



Εικόνα 2.9. Αλγόριθμοι του Scikit-learn (Loading..., 2022)

2.1.9 Matplotlib

Η βιβλιοθήκη matplotlib χρησιμοποιείται για την δημιουργία γραφικών παραστάσεων και δισδιάστατων διαγραμμάτων με τη χρήση πηγαίου κώδικα (script) σε γλώσσα προγραμματισμού Python. Υποστηρίζει μια πολύ μεγάλη ποικιλία γραφικών παραστάσεων και διαγραμμάτων, δηλαδή ιστογράμματα, ραβδογράμματα, φάσματα ισχύος, διαγράμματα σφάλματος κ.λπ. Διαθέτει επίσης ένα module με το όνομα pyplot το οποίο διευκολύνει την διαδικασία δημιουργίας γραφικών παραστάσεων παρέχοντας τη δυνατότητα ελέγχου των στυλ γραμμών, των ιδιοτήτων γραμματοσειράς, της μορφοποίησης των αξόνων κ.λπ.. Συνήθως χρησιμοποιείται σε συνδυασμό με το NumPy για να παρέχει ένα περιβάλλον το οποίο αποτελεί μια αποτελεσματική εναλλακτική λύση ανοικτού κώδικα για το MatLab. Ένα από τα μεγαλύτερα οφέλη της οπτικοποίησης είναι ότι μας επιτρέπει την οπτική πρόσβαση σε τεράστιες ποσότητες δεδομένων σε εύπεπτα οπτικά στοιχεία. Παρέχει επίσης ένα αντικειμενοστραφές API που επιτρέπει την επέκταση της λειτουργικότητας για την τοποθέτηση των στατικών διαγραμμάτων σε εφαρμογές με τη χρήση διαφόρων διαθέσιμων εργαλείων γραφικών διεπαφών Python. (Matplotlib — Visualization with Python, 2022)

Εικόνα 1.10. Το logo του πακέτου matplotlib. (Πηγή: https://pythondiario.com/2017/12/visualizacion-de-datos-con-python-y.html)

Ορισμένα από τα χαρακτηριστικά της matplotlib είναι τα ακόλουθα (Kumar, 2022):

- Χρησιμοποιείται ως βιβλιοθήκη οπτικοποίησης δεδομένων για τη γλώσσα προγραμματισμού Python.
- Παρέχει εργαλεία που επιτρέπουν τη δημιουργία γραφικών παραστάσεων και σχημάτων προτύπων σε διάφορες μορφές εξαγωγής και σε διάφορα περιβάλλοντα (όπως pycharm, jupyter, notebook) σε όλες τις πλατφόρμες.
- Παρέχει τον πιο απλό και συνηθισμένο τρόπο γραφικής σχεδίασης δεδομένων στην Python.
- Παρέχει μια διαδικαστική διεπαφή που ονομάζεται Pylab, η οποία χρησιμοποιείται με σκοπό να λειτουργεί όπως το MATLAB (λογισμικό εφαρμογών επί πληρωμής που χρησιμοποιείται από επιστήμονες και ερευνητές).
- Διαθέτει τις ίδιες δυνατότητες σχεδιασμού με το MATLAB επιτρέποντας στους χρήστες να έχουν πλήρη έλεγχο των γραμματοσειρών, των γραμμών, των χρωμάτων, των στυλ και των ιδιοτήτων των αξόνων.
- Το Matplotlib σε συνδυασμό με το NumPy αποτελούν μια λύση ανοιχτού κώδικα παρόμοια με το MATLAB.
- Υποστηρίζεται από διάφορες άλλες βιβλιοθήκες και πακέτα τρίτων με αποτέλεσμα να επεκτείνονται οι δυνατότητες και λειτουργίες του. Για παράδειγμα το basemap και cartopy χρησιμοποιούνται για τη σχεδίαση γεωχωρικών δεδομένων ενώ τα seaborn και ggplot παρέχουν περισσότερες δυνατότητες για τη σχεδίαση.

- Διαθέτει εξαιρετικό τρόπο για την παραγωγή ποιοτικών στατικών οπτικοποιήσεων οι οποίες μπορούν να χρησιμοποιηθούν για δημοσιεύσεις και επαγγελματικές παρουσιάσεις.
- Αποτελεί μια βιβλιοθήκη πολλαπλών πλατφορμών που μπορεί να χρησιμοποιηθεί σε διάφορα script της Python σε οποιοδήποτε πυρήνα της Python (conda, jupyter, google colab) και διακομιστές εφαρμογών ιστού (Django, flask) και σε διάφορες εργαλειοθήκες GUI (Tkinter, PyQt).
- Μέσω της υποστήριξης διαφόρων συμβατών βιβλιοθηκών τρίτων παρέχει στον χρήστη ισχυρά εργαλεία για την οπτικοποίηση ποικίλων δεδομένων.

2.1.10 Seaborn

Το Seaborn είναι ένα πακέτο απεικόνισης δεδομένων που βασίζεται στο matplotlib. Στατιστικά διαγράμματα, όπως θερμοχάρτες και χρονοσειρές μπορούν να κατασκευαστούν εύκολα με το πακέτο Seaborn. (seaborn: statistical data visualization — seaborn 0.11.2 documentation, 2022)

2.1.11 requests

Το πακέτο requests είναι ένας πολύ συνηθισμένος τρόπος για την πραγματοποίηση HTTP αιτημάτων, μέσω Python. Καλύπτει την πολυπλοκότητα της διαδικασίας υποβολής αιτήσεων, με την βοήθεια ενός API, ώστε ο χρήστης να μπορεί να επικεντρωθεί στην αλληλεπίδραση με τις υπηρεσίες και να τραβάει δεδομένα για την εφαρμογή του. Υποστηρίζει επίσης την αποστολή επιπλέον πληροφοριών σε έναν διακομιστή ιστού μέσω παραμέτρων και επικεφαλίδων, την κωδικοποίηση των απαντήσεων του διακομιστή, την ανίχνευση σφαλμάτων και το χειρισμό ανακατευθύνσεων (Ronquillo, 2022).

Εικόνα 2.11.. Το logo του πακέτου requests. (Πηγή: https://upload.wikimedia.org/wikipedia/commons/a/aa/Requests_Python_Logo.png)

Οι μέθοδοι HTTP που υποστηρίζει η βιβλιοθήκη Requests είναι οι εξής (Ronquillo, 2022):

- **GET:** Η μέθοδος GET χρησιμοποιείται για την ανάκτηση πληροφοριών από τον συγκεκριμένο διακομιστή χρησιμοποιώντας ένα συγκεκριμένο URL.
- **POST:** Τα αιτήματα τύπου POST ζητούν από έναν διακομιστή να δεχτεί τα δεδομένα που περιέχονται στο σώμα του μηνύματος αίτησης πιθανότητα για να τα αποθηκεύσει.
- **PUT:** Η μέθοδος αυτή ζητά την αποθήκευση της περιεχόμενης οντότητας κάτω από το περιεχόμενο URL. Σε περίπτωση που το URL παραπέμπει σε έναν ήδη υπάρχοντα πόρο, τροποποιείται και εάν δεν παραπέμπει σε υπάρχοντα πόρο, τότε ο διακομιστής μπορεί να δημιουργήσει τον πόρο με αυτό το URL.
- **DELETE:** Η μέθοδος DELETE διαγράφει τον καθορισμένο πόρο.
- **PATCH:** Η μέθοδος PATCH χρησιμοποιείται για την τροποποίηση των δυνατοτήτων και πρέπει να περιέχει μόνο τις αλλαγές στον πόρο και όχι τον πλήρη πόρο.

- **HEAD:** Η μέθοδος αυτή ζητάει μια απάντηση πανομοιότυπη με εκείνη της αίτησης GET αλλά χωρίς το σώμα της απάντησης.

Ορισμένα από τα χαρακτηριστικά της βιβλιοθήκης Requests είναι (Requests: HTTP for Humans™ — Requests 2.28.1 documentation, 2022):

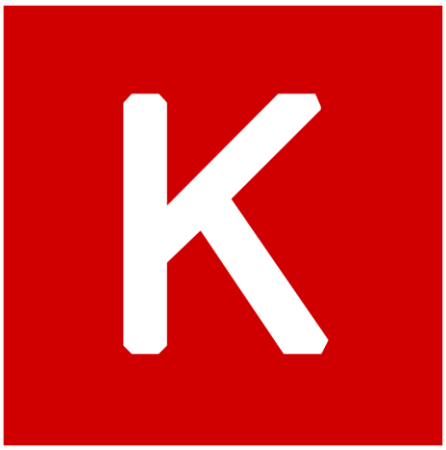
- Τμηματοποιημένες αιτήσεις
- Αυτόματη αποκωδικοποίηση περιεχομένου
- Επαλήθευση SSL μέσω προγράμματος περιήγησης
- Υποστήριξη .netrc
- Αυτόματη αποσυμπίεση
- Ανέβασμα αρχείων πολλαπλών τμημάτων
- Λήψεις μέσω ροής
- Συνεδρίες με υποστήριξη cookies
- Response με υποστήριξη Unicode
- Υποστήριξη μεσολάβησης – proxy HTTP(S)
- Τμηματοποιημένες αιτήσεις
- Connection Timeouts
- Υποστήριξη HTTP Keep Alive και connection pooling

2.2 Keras

2.2.1 Τί είναι το Keras

Η βαθιά μάθηση είναι ένας κλάδος της τεχνητής νοημοσύνης που ασχολείται με την επίλυση πολύ περίπλοκων προβλημάτων με μίμηση της λειτουργίας του ανθρώπινου εγκεφάλου. Στη βαθιά μάθηση, χρησιμοποιούνται νευρωνικά δίκτυα που χρησιμοποιούν πολλαπλούς τελεστές τοποθετημένους σε κόμβους για να βοηθήσουν στη διάσπαση του προβλήματος σε μικρότερα μέρη, τα οποία επιλύονται το καθένα ξεχωριστά. Ωστόσο, τα νευρωνικά δίκτυα μπορεί να είναι πολύ δύσκολο να εφαρμοστούν. Αυτό το πρόβλημα φροντίζει το Keras, ένα πλαίσιο βαθιάς μάθησης.

Το Keras είναι ένα API υψηλού επιπέδου, βαθιάς μάθησης που αναπτύχθηκε από την Google για την υλοποίηση νευρωνικών δικτύων. Είναι γραμμένο σε Python και χρησιμοποιείται για να διευκολύνει την υλοποίηση νευρωνικών δικτύων. Υποστηρίζει επίσης πολλαπλούς υπολογισμούς νευρωνικών δικτύων υποστήριξης.



Εικόνα 2.12. Το logo του Keras.

(Πηγή: https://en.wikipedia.org/wiki/Keras#/media/File:Keras_logo.svg)

Το Keras είναι σχετικά εύκολο στην εκμάθηση και την εργασία, επειδή παρέχει ένα frontend python με υψηλό επίπεδο αφαίρεσης, ενώ έχει την επιλογή πολλαπλών back-ends για υπολογιστικούς σκοπούς. Αυτό κάνει το Keras πιο αργό από άλλα πλαίσια βαθιάς μάθησης, αλλά εξαιρετικά φιλικό προς τους αρχάριους.

Το Keras επιτρέπει την εναλλαγή μεταξύ διαφορετικών back-ends. Τα πλαίσια που υποστηρίζονται από την Keras είναι:

- Tensorflow
- Theano
- PlaidML
- MXNet
- CNTK (Microsoft Cognitive Toolkit)

2.2.2 Πλεονεκτήματα του Keras

Κάποια από τα σημαντικότερα πλεονεκτήματα αυτού του πλαισίου παρουσιάζονται παρακάτω:

- Το Keras είναι ένα API που δημιουργήθηκε για να είναι εύκολο να το μάθουν οι άνθρωποι. Το Keras φτιάχτηκε για να είναι απλό. Προσφέρει συνεπή και απλά API, μειώνει τις ενέργειες που απαιτούνται για την εφαρμογή κοινού κώδικα και εξηγεί ξεκάθαρα το σφάλμα χρήστη.
- Ο χρόνος δημιουργίας πρωτοτύπων στο Keras είναι μικρότερος. Αυτό σημαίνει ότι οι ιδέες μπορούν να εφαρμοστούν και να αναπτυχθούν σε συντομότερο χρόνο. Το Keras παρέχει επίσης μια ποικιλία επιλογών ανάπτυξης ανάλογα με τις ανάγκες των χρηστών.
- Οι γλώσσες με υψηλό επίπεδο αφαίρεσης και ενσωματωμένων χαρακτηριστικών είναι αργές και η δημιουργία προσαρμοσμένων χαρακτηριστικών μπορεί να είναι δύσκολη. Ωστόσο, το Keras τρέχει πάνω από το TensorFlow και είναι σχετικά γρήγορο. Το Keras είναι επίσης βαθιά ενσωματωμένο με το TensorFlow, ώστε να διευκολύνει τη δημιουργία προσαρμοσμένων ροών εργασίας με ευκολία.
- Η ερευνητική κοινότητα του Keras είναι τεράστια και ιδιαίτερα ανεπτυγμένη. Η τεκμηρίωση και η διαθέσιμη βοήθεια είναι πολύ πιο εκτεταμένη από άλλα πλαίσια βαθιάς μάθησης.
- Το Keras χρησιμοποιείται εμπορικά από πολλές εταιρείες όπως οι Netflix, Uber, Square, Yelp, κ.λπ., οι οποίες έχουν αναπτύξει προϊόντα στον δημόσιο τομέα που έχουν κατασκευαστεί με χρήση Keras.

Εκτός από αυτό, το Keras έχει χαρακτηριστικά όπως:

- Λειτουργεί ομαλά τόσο σε CPU όσο και σε GPU.
- Υποστηρίζει σχεδόν όλα τα μοντέλα νευρωνικών δικτύων.
- Έχει αρθρωτό χαρακτήρα, γεγονός που το καθιστά εκφραστικό, ευέλικτο και κατάλληλο για καινοτόμο έρευνα.

2.2.3 Δημιουργία μοντέλου με Keras

Το παρακάτω διάγραμμα δείχνει τα βασικά βήματα που απαιτούνται για την κατασκευή ενός μοντέλου στον Keras:

Εικόνα 2.13. Κατασκευή Μοντέλου με Keras (Πηγή: https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-keras)

Ορισμός δικτύου: Σε αυτό το βήμα, ορίζονται τα διαφορετικά επίπεδα στο μοντέλο και οι συνδέσεις μεταξύ τους. Το Keras έχει δύο βασικούς τύπους μοντέλων: Διαδοχικά και Λειτουργικά μοντέλα. Μετά την επιλογή του τύπου μοντέλου, ορίζεται η ροή δεδομένων μεταξύ τους.

Μεταγλώττιση ενός δικτύου: Η μεταγλώττιση κώδικα σημαίνει τη μετατροπή του σε μια μορφή κατάλληλη για να κατανοήσει το μηχάνημα. Στο Keras, η μέθοδος model.compile() εκτελεί αυτή τη λειτουργία. Για να γίνει compile το μοντέλο, ορίζουμε τη συνάρτηση απώλειας που υπολογίζει τις απώλειες στο μοντέλο, τον βελτιστοποιητή που μειώνει την απώλεια και τις μετρήσεις που χρησιμοποιούνται για την εύρεση της ακρίβειας του μοντέλου.

Προσαρμογή του δικτύου: Χρησιμοποιώντας αυτό, το μοντέλο προσαρμόζεται στα δεδομένα μετά το compile. Αυτό χρησιμοποιείται για την εκπαίδευση του μοντέλου στα δεδομένα.

Αξιολόγηση του δικτύου: Αφού προσαρμοστεί το μοντέλο, πρέπει να γίνει η αξιολόγηση σφάλματος στο μοντέλο.

Προβλέψεις: Το model.predict() χρησιμοποιείται για να γίνουν προβλέψεις χρησιμοποιώντας το μοντέλο σε νέα δεδομένα.

2.2.4 Tensorflow

Το Tensorflow είναι ένα πλαίσιο βαθιάς μάθησης ανοιχτού κώδικα που αναπτύχθηκε από την Google και κυκλοφόρησε το 2015. Είναι γνωστό για την τεκμηρίωση και την υποστήριξη εκπαίδευσης, τις κλιμακούμενες επιλογές παραγωγής και ανάπτυξης, τα πολλαπλά επίπεδα αφαίρεσης και την υποστήριξη για διαφορετικές πλατφόρμες, όπως το Android.

Το TensorFlow είναι μια συμβολική βιβλιοθήκη μαθηματικών που χρησιμοποιείται για νευρωνικά δίκτυα και είναι η πλέον κατάλληλη για προγραμματισμό ροής δεδομένων σε μια σειρά εργασιών. Προσφέρει πολλαπλά επίπεδα αφαίρεσης για μοντέλα κατασκευής και εκπαίδευσης.

Αποτελώντας μια πολλά υποσχόμενη και ταχέως αναπτυσσόμενη είσοδο στον κόσμο της βαθιάς μάθησης, το TensorFlow προσφέρει ένα ευέλικτο, ολοκληρωμένο σύστημα κοινοτικών πόρων, βιβλιοθηκών και εργαλείων που διευκολύνουν τη δημιουργία και την ανάπτυξη εφαρμογών μηχανικής μάθησης. Επίσης, όπως αναφέρθηκε προηγουμένως, το TensorFlow έχει υιοθετήσει το Keras, γεγονός που κάνει τη σύγκριση των δύο να φαίνεται προβληματική.

Το Keras είναι ενσωματωμένο στο TensorFlow και μπορεί να χρησιμοποιηθεί για γρήγορη εκμάθηση σε βάθος, καθώς παρέχει ενσωματωμένες μονάδες για όλους τους υπολογισμούς νευρωνικών δικτύων. Ταυτόχρονα, ο υπολογισμός που περιλαμβάνει τανυστές, γραφήματα υπολογισμού, συνεδρίες κ.λπ. μπορεί να προσαρμοστεί χρησιμοποιώντας το Tensorflow Core API, το οποίο δίνει απόλυτη ευελιξία και έλεγχο της εφαρμογής και επιτρέπει την εφαρμογή ιδεών σε σχετικά σύντομο χρονικό διάστημα.

2.2.5 Tensorflow vs Keras

Το TensorFlow είναι μια πλατφόρμα ανοιχτού κώδικα, μια βιβλιοθήκη για πολλαπλές εργασίες μηχανικής μάθησης, ενώ το Keras είναι μια βιβλιοθήκη νευρωνικών δικτύων υψηλού επιπέδου που τρέχει πάνω από το TensorFlow. Και τα δύο παρέχουν API υψηλού επιπέδου που χρησιμοποιούνται για εύκολη κατασκευή και εκπαίδευση μοντέλων, αλλά το Keras είναι πιο φιλικό προς τον χρήστη επειδή είναι ενσωματωμένο στην Python.

Οι ερευνητές στρέφονται στο TensorFlow όταν εργάζονται με μεγάλα σύνολα δεδομένων και ανίχνευση αντικειμένων και χρειάζονται εξαιρετική λειτουργικότητα και υψηλή απόδοση. Το TensorFlow εκτελείται σε Linux, MacOS, Windows και Android. Το πλαίσιο αναπτύχθηκε από την Google Brain και χρησιμοποιείται επί του παρόντος για τις ανάγκες έρευνας και παραγωγής της Google.

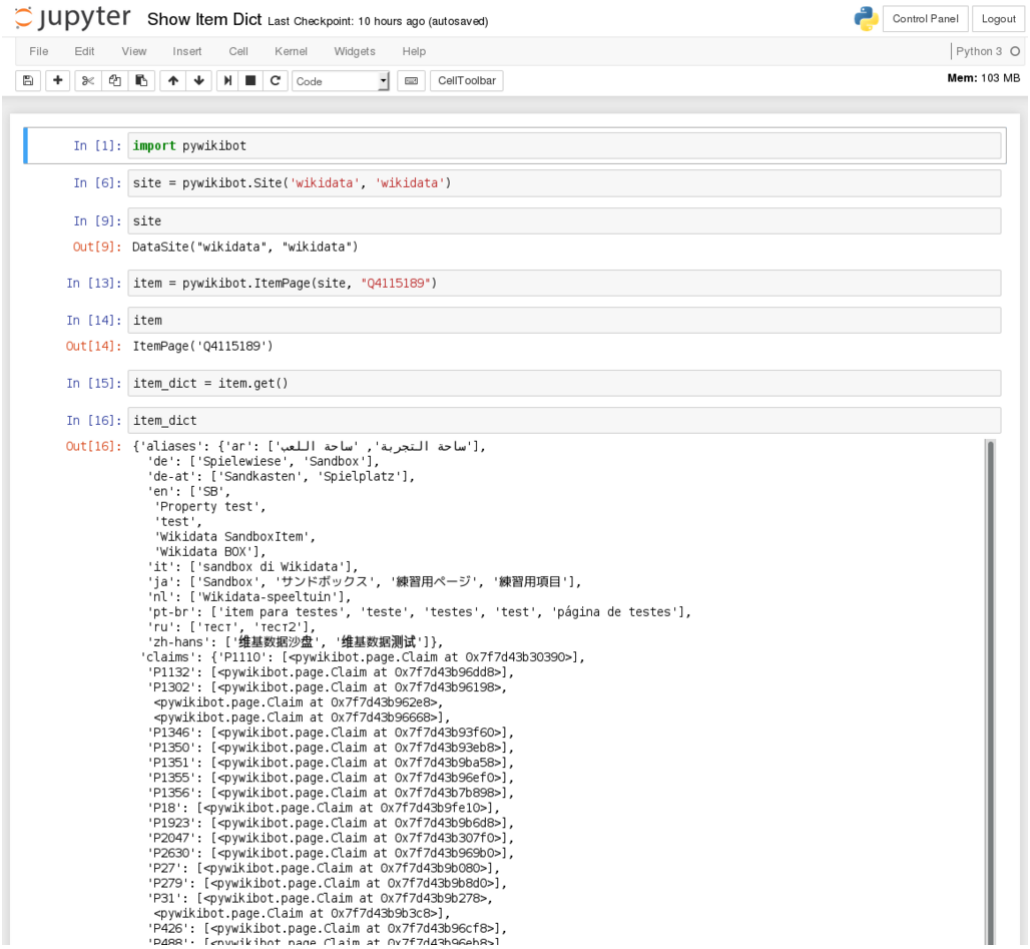
Εικόνα 2.14. Το logo του Tensorflow (Πηγή: https://en.wikipedia.org/wiki/TensorFlow#/media/File:TensorFlow_logo.svg)

Ωστόσο, η σύγκριση του TensorFlow και του Keras δεν είναι ο καλύτερος τρόπος κατανόησης του καλύτερου πλαισίου, καθώς το Keras λειτουργεί ως περιτύλιγμα στο πλαίσιο του TensorFlow. Έτσι, θα μπορούσε να οριστεί ένα μοντέλο με τη διεπαφή του Keras, το οποίο είναι πιο εύκολο στη χρήση και, στη συνέχεια, να γίνει χρήση του TensorFlow όταν χρειάζεται να χρησιμοποιηθεί μια δυνατότητα που δεν διαθέτει το Keras ή μια συγκεκριμένη λειτουργικότητα του TensorFlow. Με αυτόν τον τρόπο, είναι δυνατή η απευθείας τοποθέτηση του κώδικα TensorFlow στο μοντέλο εκπαίδευσης Keras.

2.3 Jupyter Notebook

Το Jupyter, γνωστό και ως υπολογιστικό σημειωματάριο (computational notebook), είναι μια δωρεάν, διαδραστική εφαρμογή ιστού - περιβάλλοντος, η οποία επιτρέπει σε ερευνητές να συνδυάσουν κώδικα, αποτελέσματα, επεξηγηματικό κείμενο και πολυμέσα σε ένα ενιαίο έγγραφο. Το υπολογιστικό σημειωματάριο, ενώ είναι υπαρκτό εδώ και δεκαετίες, γνώρισε μεγάλη δημοτικότητα τα τελευταία χρόνια, λόγω της ανόδου του κλάδου της επιστήμης

δεδομένων (data science). Έχει υποστήριξη από μεγάλο κοινό προγραμματιστών-χρηστών, οι οποίοι και βοήθησαν στον επανασχεδιασμό της αρχιτεκτονικής του. Αυτό είχε ως αποτέλεσμα να υποστηρίζει πλέον δεκάδες γλώσσες προγραμματισμού, γεγονός που αντανακλάται στο όνομά του, το οποίο είναι εμπνευσμένο από τις γλώσσες προγραμματισμού Julia(Ju), Python(py) και R.



Εικόνα 2.15. Διεπαφή Jupyter (Project Jupyter – Wikipedia, 2022)

2.4 Ημι-δομημένα δεδομένα

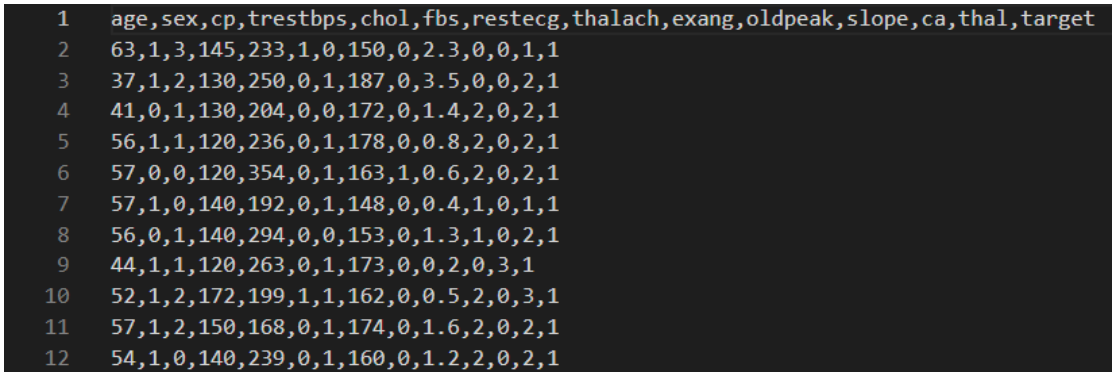
Τα ημι-δομημένα δεδομένα (semi-structured data) είναι τα δεδομένα, τα οποία δεν έχουν συλληφθεί ή μορφοποιηθεί με συμβατικούς τρόπους. Τα ημι-δομημένα δεδομένα δεν ακολουθούν τη μορφή ενός πίνακα ή των σχεσιακών Βάσεων Δεδομένων, επειδή δεν διαθέτουν σταθερή μορφή. Ωστόσο, τα δεδομένα δεν βρίσκονται σε εντελώς ακατέργαστη (raw data) ή μη-δομημένη μορφή

(unstructured data) και περιέχουν ορισμένα δομικά στοιχεία, όπως ετικέτες και οργανωτικά μεταδεδομένα που διευκολύνουν την ανάλυση. Τα πλεονεκτήματα των ημι-δομημένων δεδομένων είναι ότι είναι πιο ευέλικτα και απλούστερα, σε σύγκριση με τα δομημένα δεδομένα. Μερικά παραδείγματα ημι-δομημένων δεδομένων είναι τα email, οι ιστοσελίδες (web pages), οι NoSQL Βάσεις Δεδομένων και τα CSV αρχεία.

2.5 CSV

Τα CSV (Comma Separated Values) είναι ημι-δομημένα αρχεία. Κάθε γραμμή ενός CSV αρχείου είναι μια εγγραφή και τα πεδία μιας εγγραφής χωρίζονται με κόμμα. Σε ένα CSV αρχείο αποθηκεύονται συνήθως δεδομένα ενός πίνακα, οπότε κάθε γραμμή-εγγραφή θα έχει τον ίδιο αριθμό πεδίων.

Η μορφή ενός αρχείου CSV δεν είναι πλήρως τυποποιημένη. Ο διαχωρισμός των πεδίων με κόμματα είναι το θεμέλιο, αλλά τα κόμματα σε δεδομένα (π.χ. 23,37) ή οι διακοπές γραμμών (line breaks) πρέπει να αντιμετωπίζονται ειδικά. Ορισμένες υλοποιήσεις απαγορεύουν την χρήση αυτών, ενώ άλλες υλοποιήσεις περιβάλλουν το πεδίο με εισαγωγικά (π.χ. “23,37”), που δημιουργεί αντίστοιχο πρόβλημα για τα εισαγωγικά. Επιπλέον, συχνά πολλά CSV αρχεία χρησιμοποιούν άλλα σύμβολα ως διαχωριστικά των πεδίων (το ερωτηματικό, το tab, το κενό κλπ.), που μπορεί να προκαλέσει πρόβλημα στην ανταλλαγή δεδομένων.



```
1   age,sex,cp,trestbps,chol,fbs,restecg,thalach,exang,oldpeak,slope,ca,thal,target
2   63,1,3,145,233,1,0,150,0,2.3,0,0,1,1
3   37,1,2,130,250,0,1,187,0,3.5,0,0,2,1
4   41,0,1,130,204,0,0,172,0,1.4,2,0,2,1
5   56,1,1,120,236,0,1,178,0,0.8,2,0,2,1
6   57,0,0,120,354,0,1,163,1,0.6,2,0,2,1
7   57,1,0,140,192,0,1,148,0,0.4,1,0,1,1
8   56,0,1,140,294,0,0,153,0,1.3,1,0,2,1
9   44,1,1,120,263,0,1,173,0,0,2,0,3,1
10  52,1,2,172,199,1,1,162,0,0.5,2,0,3,1
11  57,1,2,150,168,0,1,174,0,1.6,2,0,2,1
12  54,1,0,140,239,0,1,160,0,1.2,2,0,2,1
```

Εικόνα 2.16. Μορφή ενός CSV αρχείου

3 Δεδομένα

3.1 Χρονοσειρές

Οι χρονοσειρές (time series) πρόκειται για ακολουθίες σημείων δεδομένων που λαμβάνονται σε τακτά ίσα χρονικά διαστήματα. Μια χρονοσειρά μπορεί να ληφθεί σε οποιαδήποτε μεταβλητή που αλλάζει με την πάροδο του χρόνου. Τα δεδομένα χρονοσειρών μπορούν να παρακολουθούν αλλαγές τόσο βραχυπρόθεσμα όσο και μακροπρόθεσμα.

Αυτό που διαφοροποιεί τα δεδομένα χρονοσειρών από άλλα δεδομένα είναι ότι η ανάλυση τους μπορεί να δείξει το πως οι μεταβλητές αλλάζουν με την πάροδο του χρόνου. Ουσιαστικά ο χρόνος είναι μια καθοριστική μεταβλητή διότι δείχνει το πως τα δεδομένα προσαρμόζονται κατά την διάρκεια των σημείων δεδομένων καθώς και τα τελικά αποτελέσματα.

Η ανάλυση χρονοσειρών απαιτεί συνήθως έναν τεράστιο αριθμό σημείων δεδομένων με σκοπό την εξασφάλιση της συνέπειας και τις αξιοπιστίας των συμπερασμάτων. Ένα μεγάλο σύνολο δεδομένων εξασφαλίζει την ύπαρξη ενός αντιπροσωπευτικού μεγέθους δειγμάτων και εξασφαλίζει επίσης τον περιορισμό του θορύβου στα δεδομένα. Επιπροσθέτως, εξασφαλίζει ότι τυχόν τάσεις και μοτίβα που δεν είναι ακραίες και μπορούν να ευθύνονται για την εποχιακή διακύμανση. Τέλος τα δεδομένα χρονοσειρών μπορούν να χρησιμοποιηθούν για την πρόβλεψη δεδομένων.

Η πρόβλεψη χρονοσειρών είναι μια μέθοδος που χρησιμοποιείται για την πρόβλεψη μελλοντικών γεγονότων εξετάζοντας μοτίβα και δεδομένα του παρελθόντος. Παραδείγματα χρονοσειρών υπάρχουν σχεδόν σε όλους τους τομείς των επιστημών όπως για παράδειγμα:

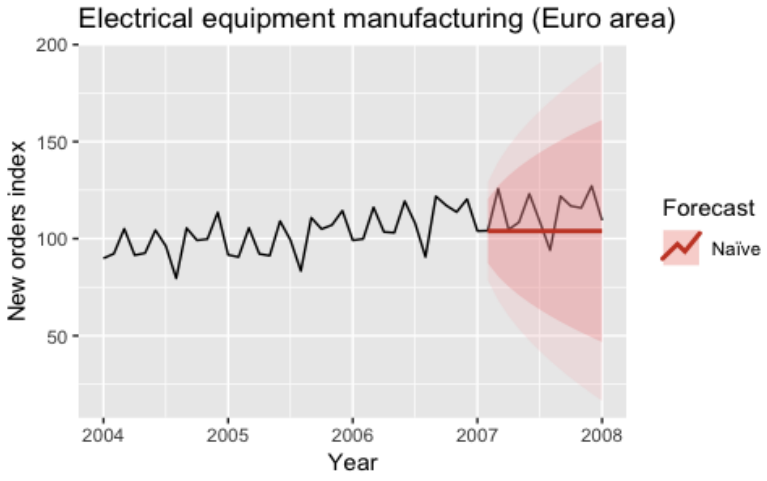
- Στον τομέα της οικονομίας για το ετήσιο ΑΕΠ, ο μηνιαίος πληθωρισμός, η τιμή κλεισίματος μια μετοχής.
- Στην μετεωρολογία με τις ημερήσιες θερμοκρασίες και την μηνιαία βροχόπτωση.
- Στην Δημογραφία με τα ετήσια στοιχεία που αφορούν τις γεννήσεις και τους θανάτους.
- Στις επιχειρήσεις με τα έξοδα και τις πωλήσεις.

Υπάρχουν πολλές μέθοδοι για την μελέτη των χρονοσειρών. Ενδεικτικά κάποιες από τις μεθόδους είναι:

- Naïve, Snaive
- Arima, Sarima
- Seasonal Decomposition
- Αναδρομικά νευρωνικά δίκτυα

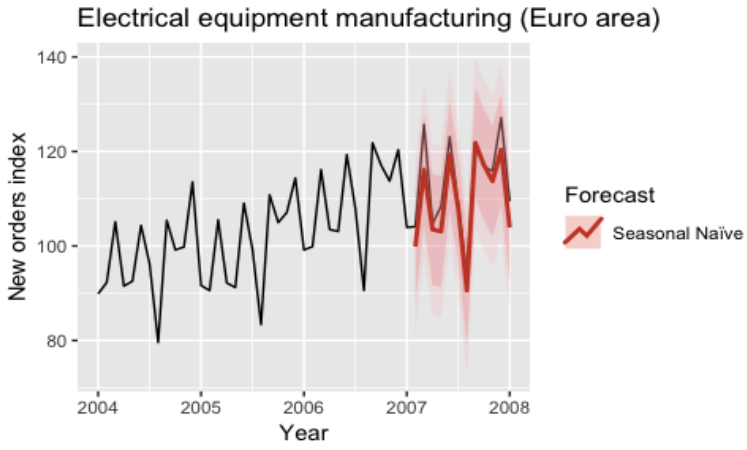
3.1.1 Naïve-SNaive

Πρόκειται για ένα μοντέλο που χρησιμοποιεί ελάχιστες ποσότητες δεδομένων και πρόβλεψη. Στο μοντέλο Naïve οι προβλέψεις αντιστοιχούν στην τελευταία παρατηρούμενη τιμή. Το SNaive είναι μια επέκταση του μοντέλου Naïve και οι προβλέψεις για τα ακόλουθα χρονικά διαστήματα X είναι ίσα με τα προηγούμενα χρονικά βήματα X.



Εικόνα 3.1. Παράδειγμα τεχνικής Naïve.

(Πηγή: https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb)

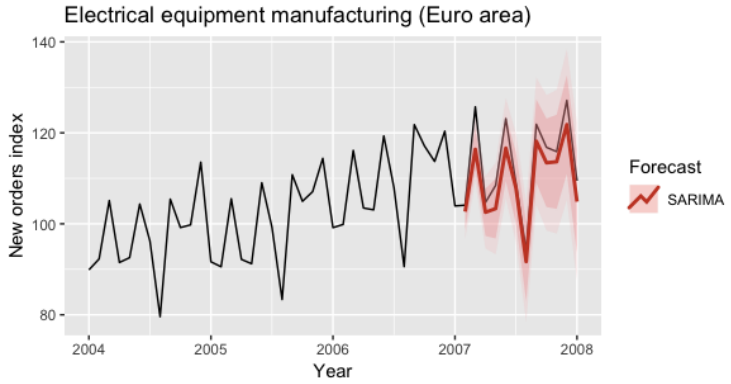


Εικόνα 3.2.. Παράδειγμα τεχνικής Naïve.

(Πηγή: https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb)

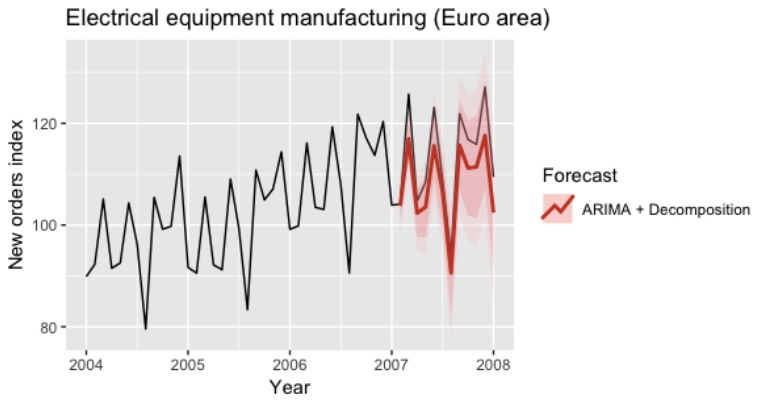
3.1.2 Arima-Sarima

Η μέθοδος Arima (Autoregressive Integrated Moving Average) είναι ένα μοντέλο που οι προβλέψεις αντιστοιχούν σε έναν γραμμικό συνδυασμό προηγούμενων τιμών της μεταβλητής. Στο μοντέλο αυτό οι προβλέψεις αντιστοιχούν σε έναν γραμμικό συνδυασμό προηγούμενων σφαλμάτων πρόβλεψης. Η μέθοδος Sarima επεκτείνει την ARIMA προσθέτοντας έναν γραμμικό συνδυασμό εποχικών προηγούμενων τιμών ή/και σφαλμάτων πρόβλεψης.



Εικόνα 3.2. Παράδειγμα τεχνικής SARIMA.

(Πηγή: https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb)

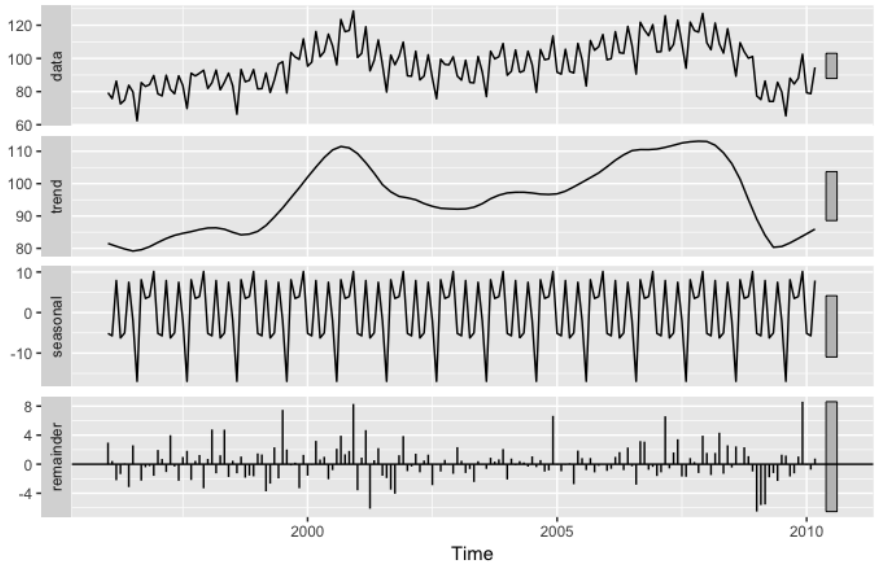


Εικόνα 3.3. Παράδειγμα τεχνικής ARIMA.

(Πηγή: https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb)

3.1.3 Seasonal Decomposition

Το Seasonal Decomposition είναι μια μέθοδος που χρησιμοποιείται για την αναπαράσταση χρονοσειρών ως άθροισμα ή γινόμενο τριών στοιχείων, την γραμμική τάση, την εποχιακή συνιστώσα και τα τυχαία κατάλοιπα. Αυτή η μέθοδο είναι χρήσιμη στην ανάλυση χρονοσειρών που επηρεάζονται από παράγοντες που αλλάζουν στον χρόνο με περιοδικό τρόπο.



Εικόνα 3.4. Εξαγωγή πληροφοριών με χρήση της μεθόδου Seasonal Decompose.

(Πηγή: https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb)

3.1.4 Αναδρομικά νευρωνικά δίτκυα

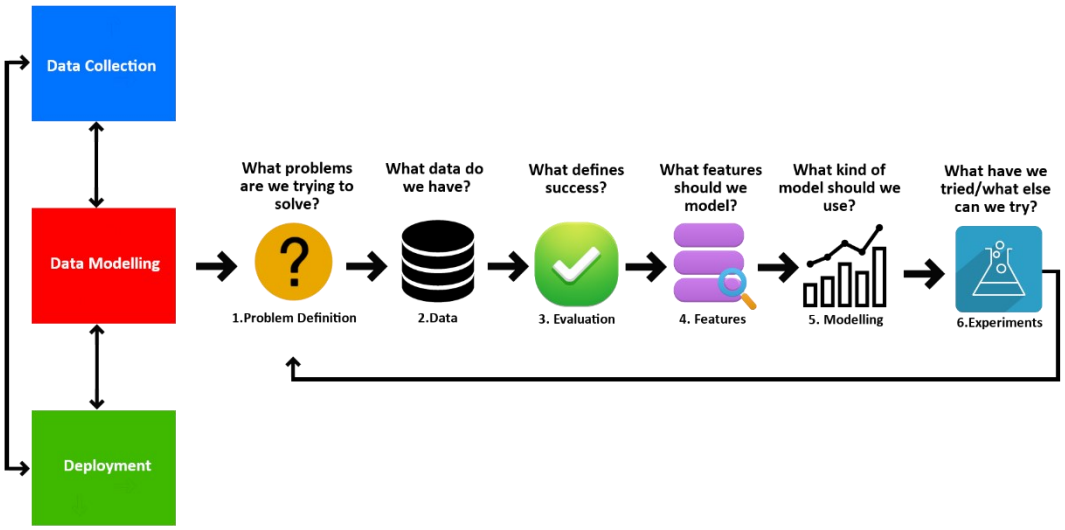
Τα αναδρομικά νευρωνικά δίτκυα (RNN) θα αναλυθούν παρακάτω, καθώς χρησιμοποιήθηκε μια τεχνική που βασίζεται στην αρχιτεκτονική του (LSTM).

3.2 Διαδικασία Εξόρυξης Δεδομένων

Η διαδικασία Εξόρυξης Δεδομένων είναι μια επαναληπτική διαδικασία που αποτελείται από τα παρακάτω βήματα:

1. Συλλογή Δεδομένων
2. Μοντελοποίηση Δεδομένων
3. Εφαρμογή λογισμικού

Στην Εικόνα 3.1 αποτυπώνεται η διαδικασία ΕΔ που θα εφαρμοστεί στο έργο. Ο ορισμός της διαδικασίας αποτελεί κρίσιμο κομμάτι της ΕΔ (και κυρίως στο κομμάτι μοντελοποίησης των δεδομένων), καθώς σε αυτό το κομμάτι απαιτούνται σημαντικά ερωτήματα, που θα βοηθήσουν στην λύση του προβλήματος.



Εικόνα 3.5. Διαδικασία της Εξόρυξης Δεδομένων

Ξεκινώντας από την συλλογή των δεδομένων, προχωράμε στην μοντελοποίηση των δεδομένων. Σε αυτό το στάδιο πρέπει να ορίσουμε το πρόβλημα που μας απασχολεί, δηλαδή τί πρόβλημα προσπαθούμε να λύσουμε. Έπειτα, πρέπει να εξετάσουμε τα δεδομένα που έχουμε και να ορίσουμε τί θεωρούμε επιτυχία για το πρόβλημα, θέτοντας ένα ελάχιστο κατώφλι ακρίβειας (π.χ. το μοντέλο πρόβλεψης να βγάζει σωστά αποτελέσματα στο 99% των περιπτώσεων). Στη συνέχεια, γίνεται εξαγωγή των πιο σχετικών χαρακτηριστικών για το πρόβλημα που θέλουμε να επιλύσουμε. Τέλος, γίνεται εφαρμογή διαφόρων αλγορίθμων και γίνονται δοκιμές τροποποίησης των παραμέτρων μέχρι να βγει το καλύτερο δυνατό αποτέλεσμα.

3.2.1 Συλλογή των δεδομένων

Η συλλογή των δεδομένων πραγματοποιήθηκε με κώδικα Python από μια ιστοσελίδα που παρέχει μετεωρολογικά δεδομένα για χώρες της Ευρώπης και της Αφρικής[1]. Ο κώδικας πραγματοποιεί Http αιτήματα τύπου GET στην σελίδα ανά τακτά χρονικά διαστήματα, συλλέγοντας έτσι δεδομένα, τα οποία και αποθηκεύει σε ένα CSV αρχείο. Για τον κώδικα χρησιμοποιήθηκε το πακέτο requests.

```python
if __name__ == "__main__":

    file_name = "dates.csv"
    genId = "1407"  # location
    specId = "9484"  # feature
    file_to_write = 'KompotiSolarRad'
    dates = load_dates_file(file_name)

    dates = convert_dates(dates, 379)

    for list_of_dates in dates:
        for date in list_of_dates:
            try:
                req = requests.get(
                    f"http://openmeteo.org/api/stations/{genId}/timeseries/{specId}/data/?fmt=json&exact_datetime=true&start_date={date}%2000:00&
                    end_date={date}%2023:50")

                url_content = req.content

                csv_file = open(f'{file_to_write}.csv', 'ab')

                csv_file.write(url_content)

            except HTTPError as http_err:
                print(f"HTTP error occurred: {http_err}")

            except Exception as err:
                print(f"Other error occurred: {err}")
    time.sleep(30)

csv_file.close()
```

Εικόνα 3.6. Υλοποίηση κώδικα για την συλλογή δεδομένων.

[1] https://openmeteo.org/

3.2.2 Το σύνολο δεδομένων

Συνολικά συλλέχθηκαν τέσσερα σύνολα δεδομένων, από τέσσερις διαφορετικούς σταθμούς, εκ των οποίων τρεις σταθμοί είναι σχετικά κοντά (εντός Νομού Άρτας) και έναν σταθμό πιο απομακρυσμένο (Νομός Ιωαννίνων). Σκοπός της συλλογής πολλαπλών συνόλων δεδομένων είναι η εξέταση γενίκευσης του μοντέλου, αφού έχει εκπαιδευτεί, αρχικά σε διαφορετικές κοντινές περιοχές, και στη συνέχεια σε πιο απομακρυσμένες περιοχές. Τα σύνολα δεδομένων αποτελούνται από πέντε στήλες:

- Ημερομηνία
- Θερμοκρασία αέρα
- Ηλιακή ακτινοβολία
- Ταχύτητα ανέμου
- Υγρασία.

Οι σταθμοί ενημερώνονται κάθε δέκα λεπτά, επομένως οι μετρήσεις που έχουν αποθηκευτεί στα αρχεία είναι ανά δεκάλεπτο. Ο αριθμός γραμμών του κάθε συνόλου δεδομένων είναι 382001.

```
dateTime,temperature,windSpeed,humidity,solarRad,rain
2015-02-25 00:00,7.0,0.0,100.0,1.0,0.0
2015-02-25 00:10,6.9,0.1,100.0,1.0,0.0
2015-02-25 00:20,7.1,0.0,100.0,1.0,0.0
2015-02-25 00:30,6.9,0.1,100.0,1.0,0.0
2015-02-25 00:40,6.9,0.2,100.0,1.0,0.0
2015-02-25 00:50,7.2,0.5,98.0,1.0,0.0
2015-02-25 01:00,6.9,0.3,99.0,1.0,0.0
2015-02-25 01:10,7.6,0.2,97.0,1.0,0.0
2015-02-25 01:20,7.3,0.0,97.0,1.0,0.0
2015-02-25 01:30,7.1,0.1,97.0,1.0,0.0
2015-02-25 01:40,6.8,0.2,97.0,1.0,0.0
2015-02-25 01:50,6.6,0.1,98.0,1.0,0.0
2015-02-25 02:00,6.6,0.2,97.0,1.0,0.0
2015-02-25 02:10,7.1,0.2,95.0,1.0,0.0
2015-02-25 02:20,6.9,0.3,97.0,1.0,0.0
2015-02-25 02:30,7.1,0.1,98.0,1.0,0.0
2015-02-25 02:40,7.6,0.2,94.0,1.0,0.0
2015-02-25 02:50,7.9,0.6,93.0,1.0,0.0
2015-02-25 03:00,8.3,0.7,91.0,1.0,0.0
2015-02-25 03:10,8.9,1.3,90.0,1.0,0.0
2015-02-25 03:20,9.2,1.3,88.0,1.0,0.0
2015-02-25 03:30,9.2,1.3,89.0,1.0,0.0
2015-02-25 03:40,9.1,1.0,90.0,1.0,0.0
2015-02-25 03:50,9.5,1.1,86.0,1.0,0.0
2015-02-25 04:00,9.8,1.0,85.0,1.0,0.0
2015-02-25 04:10,10.0,1.4,83.0,1.0,0.0
2015-02-25 04:20,10.1,2.9,86.0,1.0,0.2
2015-02-25 04:30,9.7,1.5,89.0,1.0,0.4
2015-02-25 04:40,10.0,3.6,89.0,1.0,0.6
```

Εικόνα 3.7. Μορφή συνόλου δεδομένων

3.2.2.1 Ημερομηνία

Τα σύνολα δεδομένων έχουν χρονικό διάστημα από 25 Φεβρουαρίου του 2015 και ώρα 00 : 00 έως και 31 Μαΐου του 2022 23 : 50 και βρίσκεται σε μορφή «έτος/μήνας/ημέρα Ώρα: λεπτά: δευτερόλεπτα».

3.2.2.2 Θερμοκρασία του αέρα

Η θερμοκρασία έχει μετρηθεί σε βαθμούς κελσίου (°C).

3.2.2.3 Ηλιακή ακτινοβολία

Είναι το ποσό της ηλιακής ακτινοβολίας που προσπίπτει κάθετα στη μμονάδα επιφάνειας που βρίσκεται στη μέση απόσταση Γης-Ήλιου, ανά μονάδα χρόνου (κατά μέσο όρο). Η ηλιακή ακτινοβολία έχει μονάδα μέτρησης W/m^2.

3.2.2.4 Ταχύτητα ανέμου

Η ταχύτητα ανέμου, ή αλλιώς ένταση ανέμου, μετριέται σε m/s και είναι ανάλογη προς τη διαφορά της πίεσης μεταξύ του τόπου γέννησης του ανέμου και του τόπου άφιξης.

3.2.2.5 Σχετική Υγρασία

Η σχετική υγρασία αντιπροσωπεύει το ποσοστό της μέγιστης πιθανής ποσότητας υδρατμών στον αέρα με αναφορά την θερμοκρασία την στιγμή της μέτρησης. Η σχετική υγρασία μετριέται σε ποσοστό (%).

3.2.2.6 Βροχή

Η βροχή είναι , ως γνωστόν, βασική μετεωρολογική παράμετρος. Εκείνο που ενδιαφέρει ιδιαίτερα είναι η ποσότητα του νερού που πέφτει σε μια επιφάνεια. Αυτή εκφράζεται με το "ύψος βροχής". Αυτό ορίζεται ως εκείνο το ύψος στο οποίο θα έφτανε η στάθμη του νερού της βροχής αν έπεφτε πάνω σε μια οριζόντια επιφάνεια αποκλείοντας τους παράγοντας διαρροή, απορρόφηση και εξάτμιση. Μονάδα μέτρησης του ύψους της βροχής είναι το "**mm** βροχής". Στην πράξη λέγοντας βροχή ύψους 1mm εννοούμε τη βροχή εκείνη που απέδωσε ποσότητα νερού ίση με $1Kgr/m^2$ ή $1\ m^3$ νερού / στρέμμα.

3.2.3 Ορισμός προβλήματος και επιλογή χαρακτηριστικών

Σκοπός του μοντέλου είναι η πρόβλεψη της θερμοκρασίας της επόμενης μέρας. Η πρόβλεψη θερμοκρασίας μπορεί να θεωρηθεί ως πρόβλεψη πραγματικής τιμής. Για την αξιολόγηση των αποτελεσμάτων πρόβλεψης πραγματικών τιμών, χρησιμοποιούνται μέτρα όπως το μέσο τετραγωνικό σφάλμα (MSE – mean squared error) και η ρίζα του τετραγωνικού σφάλματος (RMSE – root mean squared error) κ.ά. Το μέτρο που θα χρησιμοποιήσουμε είναι το μέσο τετραγωνικό σφάλμα.

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \hat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

Εικόνα 3.8. Τύπος μέσου τετραγωνικού σφάλματος

Επιπλέον, θα γίνει κατηγοριοποίηση των αποτελεσμάτων σε:

- Κρύο - Cold, για θερμοκρασίες < 20°.
- Κανονική - Normal, για θερμοκρασίες 20° - 30°.
- Ζέστη - Hot, για θερμοκρασίες > 30°.

Αυτό θα βοηθήσει στην χρήση μετρικής ακρίβειας για την εύρεση κατάλληλου μοντέλου βελτιστοποίησης του μοντέλου και τον καλύτερο διαχωρισμό συνόλου εκπαίδευσης και δοκιμής.

Ως ακρίβεια (accuracy) ορίζεται ο αριθμός των σωστά ταξινομημένων στοιχείων προς όλα τα στοιχεία.

$$\text{Accuracy} = \frac{1}{N}\sum_{i}^{N} 1(y_i = \hat{y}_i)$$

Where y is a tensor of target values, and $\hat{y}$ is a tensor of predictions.

Εικόνα 3.9. Μετρική ακρίβειας.

Στην πρόβλεψη θερμοκρασίας συνηθίζεται η χρήση χρονοσειρών. Στις χρονοσειρές, όπως έχει αναφερθεί πιο πάνω, ο χρόνος αποτελεί καθοριστική μεταβλητή. Επιπλέον, πρόκειται για πρόβλεψη της θερμοκρασίας, επομένως τα δύο χαρακτηριστικά που είναι απαραίτητα για την επίλυση του προβλήματος είναι η ημερομηνία και η θερμοκρασία, τα οποία και θα επιλεχθούν.

3.2.4 Προεπεξεργασία των δεδομένων

Για να μπορέσει να πραγματοποιηθεί προεπεξεργασία στα δεδομένα, πρέπει να γίνει πρώτα η ανάλυσή τους. Έπειτα από την επιλογή των χαρακτηριστικών, έμειναν δύο χαρακτηριστικά: η ημερομηνία και η θερμοκρασία, το οποίο κάνει εύκολη την οπτικοποίηση, αφού πρόκειται για μόνο δύο διαστάσεις. Για την ανάλυση θα χρησιμοποιηθούν τα πακέτα pandas και matplotlib. Το πακέτο pandas θα χρησιμοποιηθεί για την φόρτωση των δεδομένων, κάνοντας χρήση της κλάσης Dataframe, ενώ το πακέτο matplotlib θα χρησιμοποιηθεί για την οπτικοποίηση των δεδομένων.

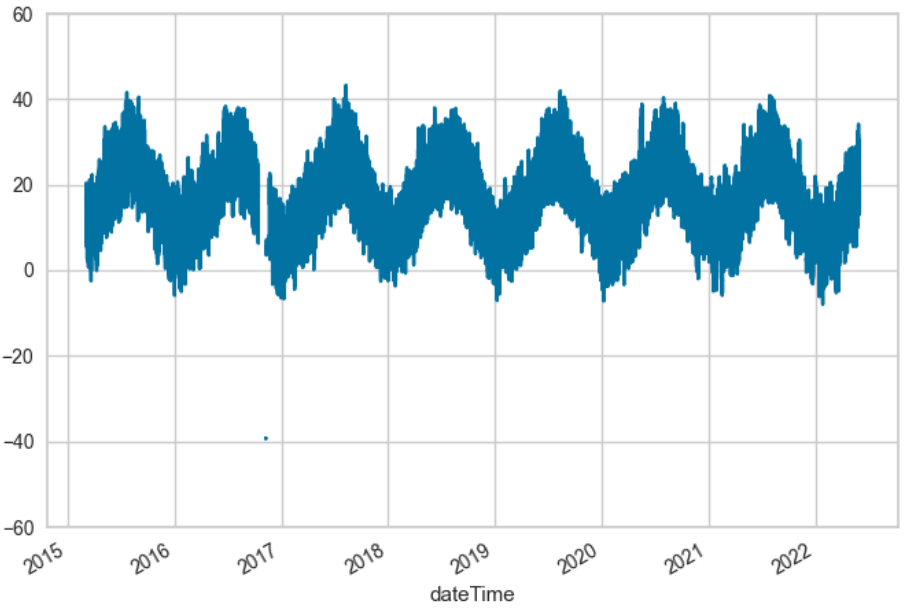
```python
import pandas as pd


def load_dataset(file_path: str) -> pd.DataFrame:
    """
    Input: path of file -> string
    Output: dataset -> Dataset
    """
    try:
        dataset = pd.read_csv(
            file_path)
        dataset['dateTime'] = pd.to_datetime(dataset['dateTime'])
        return dataset
    except FileNotFoundError:
        print("File not found.")
    except pd.errors.EmptyDataError:
        print("No data")
    except pd.errors.ParserError:
        print("Parse error")
    except Exception:
        print("Some other exception")
```

Εικόνα 3.10. Κώδικας για την φόρτωση του συνόλου δεδομένων στην Python

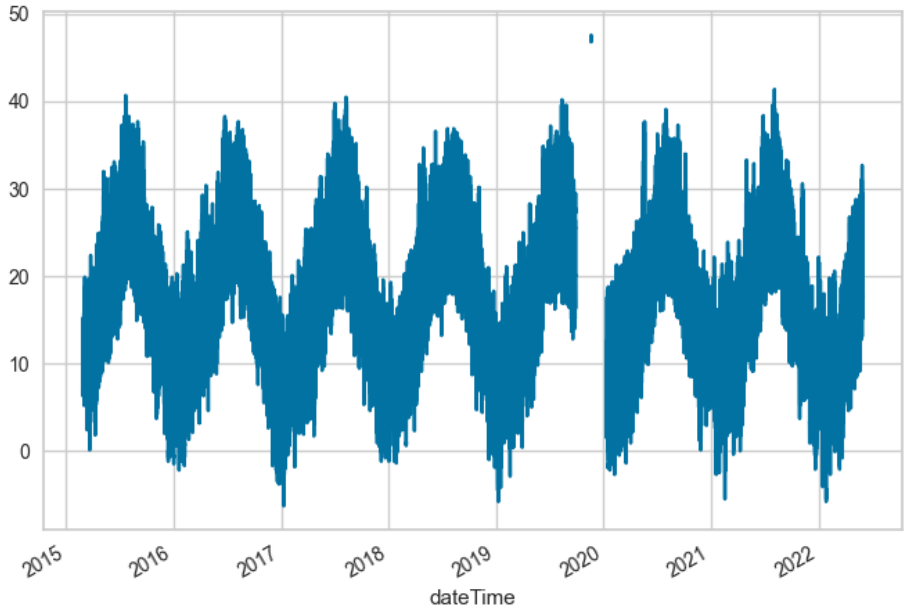
dateTime	temperature
2015-03-03 17:30:00	17.8
2015-03-03 18:00:00	16.4
2015-03-03 18:30:00	15.3
2015-03-03 19:00:00	15.1
2015-03-03 19:30:00	15.2
...	...
2022-05-31 23:10:00	15.2
2022-05-31 23:20:00	15.2
2022-05-31 23:30:00	15.4
2022-05-31 23:40:00	15.0
2022-05-31 23:50:00	15.5

350899 rows × 1 columns

Εικόνα 3.11. Το σύνολο δεδομένων σε pandas Dataframe



Εικόνα 3.12. Διάγραμμα χρονοσειράς 1



Εικόνα 3.13. Διάγραμμα χρονοσειράς 2

Παρατηρούμε ότι στις χρονοσειρές υπάρχουν ελλιπείς τιμές και ακραίες τιμές, πιθανώς λανθασμένες μετρήσεις. Θα πρέπει να αντιμετωπιστούν και τα δύο προβλήματα για την ομαλή λειτουργία του μοντέλου πρόβλεψης στη συνέχεια. Επιπλέον, η μεταβολή θερμοκρασίας ανά δεκάλεπτο είναι ελάχιστη έως και ανύπαρκτη, επομένως θα διατηρηθούν οι μετρήσεις ανά ώρα, μειώνοντας το σύνολο δεδομένων σε 63511 γραμμές, το οποίο ταυτόχρονα επιλύει και σε έναν βαθμό το πρόβλημα των ελλιπών τιμών. Για όσες επιπλέον τιμές απομένουν, υπάρχουν δύο περιπτώσεις:

- Οι τιμές να βρίσκονται ανάμεσα σε μη ελλιπείς τιμές
- Να υπάρχει μεγάλο κενό τιμών

Επομένως, αρχικά εφαρμόζεται μια διαδικασία στα δεδομένα, όπου στα δεδομένα που βρίσκονται ανάμεσα σε μη ελλιπείς τιμές εκχωρείται ο μέσος όρος των δύο τιμών, και στη συνέχεια διαγράφονται οι υπόλοιπες, για τις οποίες δεν υπάρχει αρκετή πληροφορία.

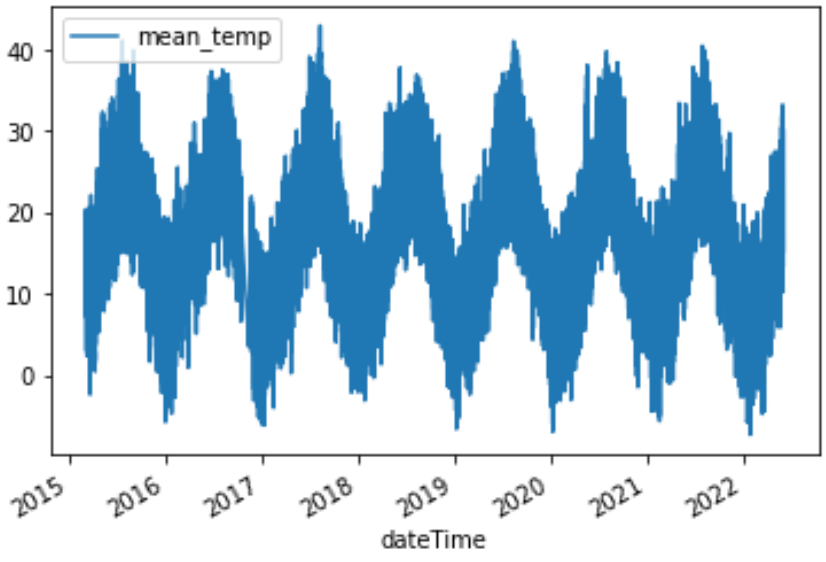
```
#preprocess of data
dataset_hourly = pd.DataFrame()
dataset_hourly['mean_temp'] = dataset.resample('H').mean()

dataset_hourly.info()
dataset_hourly = dataset_hourly.dropna()
```

Εικόνα 3.14. Κώδικας για την προεπεξεργασία δεδομένων.

dateTime	mean_temp
2015-03-03 17:00:00	17.800000
2015-03-03 18:00:00	15.850000
2015-03-03 19:00:00	15.150000
2015-03-03 20:00:00	15.050000
2015-03-03 21:00:00	14.500000
...	...
2022-05-31 19:00:00	22.633333
2022-05-31 20:00:00	19.233333
2022-05-31 21:00:00	17.016667
2022-05-31 22:00:00	15.733333
2022-05-31 23:00:00	15.283333

59585 rows × 1 columns

Εικόνα 3.15. Το σύνολο δεδομένων μετά την προεπεξεργασία.



Εικόνα 3.16. Διάγραμμα του συνόλου δεδομένων μετά την προεπεξεργασία.

3.2.4.1 Κλιμάκωση δεδομένων

Τέλος, θα πραγματοποιηθεί κλιμάκωση στα δεδομένα, καθώς μεγαλύτερες διαφορές μεταξύ των σημείων δεδομένων των μεταβλητών εισόδου αυξάνουν την αβεβαιότητα στα αποτελέσματα του μοντέλου. Μοντέλα μηχανική μάθησης παρέχουν βάρη στις μεταβλητές εισόδου και τα συμπεράσματα για την έξοδο-πρόβλεψη. Αν η διαφορά μεταξύ των σημείων δεδομένων είναι αρκετά υψηλή, το μοντέλο θα πρέπει να παρέχει μεγαλύτερα βάρη-συντελεστές στα σημεία και τα αποτελέσματα, που δεν είναι επιθυμητό, αφού μοντέλα με μεγάλες τιμές βάρους είναι συχνά ασταθή. Αυτό σημαίνει ότι το μοντέλο μπορεί να παράγει κακά αποτελέσματα ή μπορεί να αργεί πολύ στο στάδιο της εκπαίδευσης και πρόβλεψης.

Η κλιμάκωση επιτυγχάνεται κάνοντας χρήση μιας συνάρτησης του πακέτου scikit-learn, η οποία κλιμακώνει το χαρακτηριστικό της θερμοκρασίας στο εύρος [0,1].

```
#Feature Scaling
from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range=(0,1))
training_set_scaled = sc.fit_transform(dataset_hourly)
```

Εικόνα 3.17. Κλιμάκωση των δεδομένων

Το σύνολο δεδομένων έχει δεχτεί την απαραίτητη προεπεξεργασία, ώστε να χρησιμοποιηθεί για την εκπαίδευση του μοντέλου. Η δημιουργία του μοντέλου πρόβλεψης αναλύεται στο επόμενο κεφάλαιο.

4 Υλοποίηση

4.1 Αναδρομικά Νευρωνικά Δίκτυα

Το Αναδρομικό Νευρωνικό Δίτκυο (RNN - Recurrent Neural Network) πρόκειται για έναν ειδικό τύπο τεχνητού νευρωνικού δικτύου προσαρμοσμένο να λειτουργεί για δεδομένα χρονοσειρών ή δεδομένα που περιλαμβάνουν ακολουθίες. Τα συνηθισμένα νευρωνικά δίκτυα τροφοδοσίας προορίζονται μόνο για σημεία δεδομένων, τα οποία είναι ανεξάρτητα μεταξύ τους. Ωστόσο, εάν έχουμε δεδομένα σε μια ακολουθία τέτοια ώστε ένα σημεία δεδομένων να εξαρτάται από το προηγούμενο σημείο δεδομένων, πρέπει να τροποποιήσουμε το νευρωνικό δίκτυο ώστε να ενσωματωθούν οι εξαρτήσεις μεταξύ αυτών των σημείων δεδομένων. Τα RNN έχουν την έννοια της μνήμης που τους βοηθά να αποθηκεύουν τις καταστάσεις ή τις πληροφορίες προηγούμενων εισόδων για να δημιουργήσουν την επόμενη έξοδο της ακολουθίας.

Υπάρχουν διάφορα είδη RNNs με διάφορες αρχιτεκτονικές, κάποια από αυτά είναι:

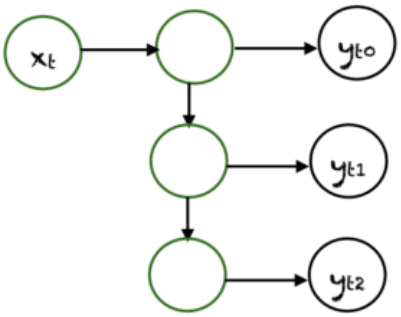
- **One to One**

Εικόνα 4.1. Αρχιτεκτονική RNN ένα προς ένα (Πηγή: https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/)

Τα παραδοσιακά νευρωνικά δίκτυα χρησιμοποιούν αρχιτεκτονική ένα προς ένα.

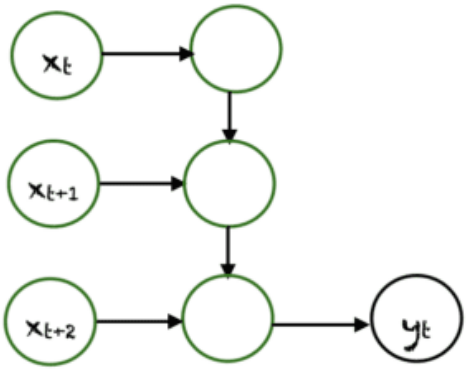
- **One to Many**

Στα δίκτυα ένα προς πολλά, μια είσοδο μπορεί να έχει πολλαπλές εξόδους.



Εικόνα 4.2. Αρχιτεκτονική RNN ένα προς πολλά (Πηγή: https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/)

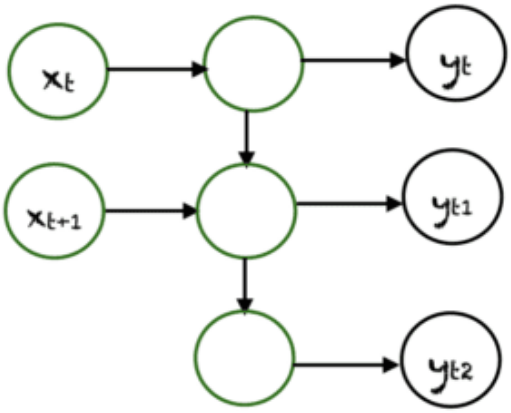
- **Many to One**



Εικόνα 4.3. Αρχιτεκτονική RNN πολλά προς ένα

Στην περίπτωση αυτή, πολλές είσοδοι από διαφορετικά χρονικά βήματα παράγουν μια μόνο έξοδο.

- **Many to Many**



Εικόνα 4.4. Αρχιτεκτονική RNN πολλά προς πολλά

Τα δίκτυα πολλά προς πολλά έχουν πολλές δυνατότητες. Στο παραπάνω σχήμα υπάρχουν 2 είσοδοι που παράγουν 3 εξόδους.

Πλεονεκτήματα και μειονεκτήματα RNN

Τα RNN έχουν διάφορα πλεονεκτήματα όπως:

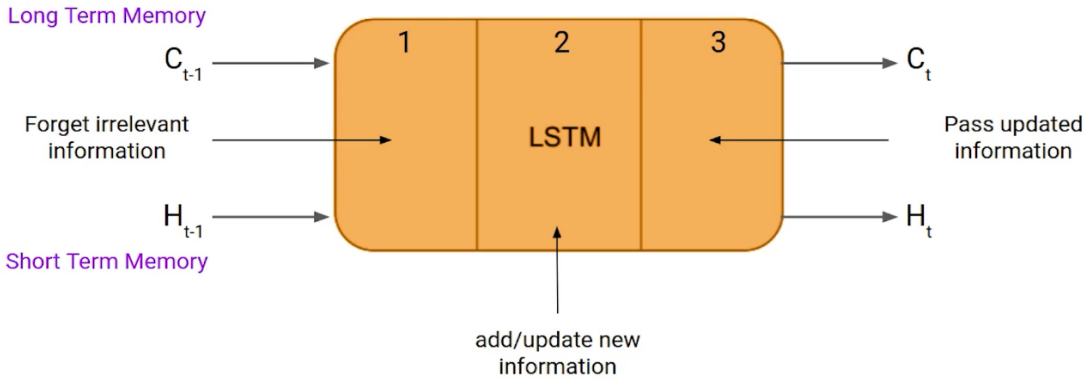
- Δυνατότητα χειρισμού δεδομένων ακολουθίας.
- Δυνατότητα χειρισμού εισόδων διαφορετικού μήκους.
- Δυνατότητα αποθήκευσης ή απομνημόνευσης ιστορικών πληροφοριών
- Τα μειονεκτήματα είναι:
- Η αργή ταχύτητα υπολογισμού
- Το RNN δεν λαμβάνει υπόψη μελλοντικές εισόδους για την λήψη αποφάσεων
- Εξαφανιζόμενο πρόβλημα κλίσης, όπου οι διαβαθμίσεις που χρησιμοποιούνται για τον υπολογισμό της ενημέρωσης βάρους μπορεί να είναι πολύ κοντά στο μηδέν, αυτό σημαίνει πως το δίκτυο εμποδίζεται να μάθει νέα βάρη. Όσο πιο βαθύ είναι το δίκτυο, τόσο πιο έντονο είναι αυτό το πρόβλημα.

4.1.1 LSTM

Το LSTM (Long Short Term Memory) πρόκειται για ένα προηγμένο διαδοχικό δίκτυο RNN που επιτρέπει στις πληροφορίες να διατηρούνται και χρησιμοποιείται για μόνιμη μνήμη. Αντίθετα με το RNN το LSTM είναι σε θέση να χειρίζεται το πρόβλημα της εξαφανιζόμενης διαβάθμισης. Επιπροσθέτως, ενώ τα RNN δίκτυα δεν μπορούν να θυμηθούν μακροπρόθεσμες εξαρτήσεις λόγω της εξαφανιζόμενης διαβάθμισης τα LSTM έχουν σχεδιαστεί ρητά με σκοπό την αποφυγή μακροπρόθεσμων προβλημάτων εξάρτησης.

4.1.2 Αρχιτεκτονική LSTM

Το LSTM λειτουργεί όπως ένα κελί RNN. Το κελί αποτελείται από 3 μέρη όπως φαίνεται στην εικόνα παρακάτω, και κάθε μέρος εκτελεί μια μεμονωμένη λειτουργία.



Εικόνα 4.5. Κελί ενός LSTM

Τα τρία αυτά μέρη είναι γνωστά και ως πύλες. Το πρώτο κομμάτι αποκαλείται Forget Gate, το δεύτερο Input Gate και το τρίτο Output gate. Στην Forget Gate γίνεται η επιλογή εάν οι πληροφορίες που προέρχονται από την προηγούμενη χρονική σήμανση είναι αξιομνημόνευτες ή άσχετες και μπορούν να ξεχαστούν. Στην Input Gate το κελί προσπαθεί να μάθει νέες πληροφορίες από την είσοδο του σε αυτό. Τέλος, στο Output Gate το κελί μεταβιβάζει τις ενημερωμένες πληροφορίες από την τρέχουσα χρονική σήμανση στην επόμενη χρονική σήμανση.

Όπως τα RNN, έτσι και τα LSTM έχουν μια κρυφή κατάσταση όπου το H(t-1) αντιπροσωπεύει την κρυφή κατάσταση της προηγούμενης χρονικής σήμανσης και το H(t) είναι η κρυφή κατάσταση της τρέχουσας χρονικής σήμανσης. Εκτός από αυτό το LSTM έχει επίσης μια κατάσταση κελιού που αντιπροσωπεύεται από C(t-1) και C(t) για την προηγούμενη και την τρέχουσα χρονική σήμανση αντίστοιχα.

Η κρυφή κατάσταση είναι γνωστή ως βραχυπρόθεσμη μνήμη και η κατάσταση κελιού είναι γνωστή ως μακροπρόθεσμη μνήμη.

4.2 Εκπαίδευση

4.2.1 Δημιουργία συνόλου εκπαίδευσης

Για την εκπαίδευση θα χωριστεί ένα τμήμα του συνόλου δεδομένων σε τιμές παρελθόντος και μέλλοντος. Το μοντέλο θα παίρνει ως είσοδο τρεις ημέρες και θα προβλέπει την τέταρτη μέρα. Επιπλέον, θα γίνει μετατροπή των δεδομένων από Dataframe σε πίνακα του πακέτου numpy.

```
x_train = []
y_train = []
n_future = 24 # next day temperature forecast
n_past = 72 # Past 3 days

for i in range(0,len(training_set_scaled)-n_past-n_future+1):
    x_train.append(training_set_scaled[i : i + n_past , 0])
    y_train.append(training_set_scaled[i + n_past : i + n_past + n_future , 0 ])
x_train , y_train = np.array(x_train), np.array(y_train)
x_train = np.reshape(x_train, (x_train.shape[0] , x_train.shape[1], 1) )
```

Εικόνα 4.6. Κώδικας για την δημιουργία συνόλου εκπαίδευσης.

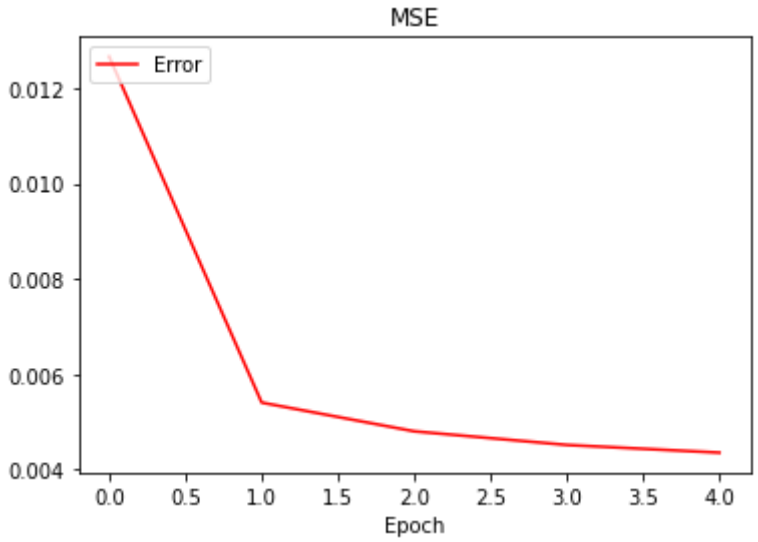
4.2.2 Μοντέλο

Για το μοντέλο θα χρησιμοποιηθεί η βιβλιοθήκη Keras. Το μοντέλο θα είναι αρχιτεκτονικής RNN και συγκεκριμένα LSTM, που έχει αναπτυχθεί πιο πάνω. Θα αποτελείται από ένα επίπεδο εισόδου, δύο κρυφά επίπεδα και ένα επίπεδο εξόδου. Το μέσο τετραγωνικό σφάλμα θα αποτελέσει την συνάρτηση απωλειών (loss function) του νευρωνικού δικτύου. Το επίπεδο εισόδου και τα κρυφά επίπεδα θα χρησιμοποιήσουν ως συνάρτηση ενεργοποίησης την υπερβολική εφαπτομένη (tanh), ενώ το επίπεδο εξόδου την γραμμική συνάρτηση. Ο αριθμός εποχών για την εκπαίδευση θα είναι 100.

Σκοπός είναι να ελαχιστοποιηθεί η συνάρτηση απωλειών με την πάροδο του χρόνου, ουσιαστικά η διαφορά μεταξύ της πρόβλεψης θερμοκρασίας και της πραγματικής θερμοκρασίας να είναι όσο πιο μικρή γίνεται.

```
from keras.models import Sequential
from keras.layers import LSTM,Dense ,Dropout, Bidirectional

regressor = Sequential()
regressor.add(Bidirectional(LSTM(units=30, return_sequences=True, input_shape = (x_train.
shape[1],1) ) ))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units= 30 , return_sequences=True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units= 30 , return_sequences=True))
regressor.add(Dropout(0.2))
regressor.add(LSTM(units= 30))
regressor.add(Dropout(0.2))
regressor.add(Dense(units = n_future,activation='linear'))
regressor.compile(optimizer='adam', loss='mean_squared_error',metrics=['mse'])
history = regressor.fit(x_train, y_train, epochs=100,batch_size=32, verbose=1 )
```

Εικόνα 4.7. Κώδικας υλοποίησης και εκπαίδευσης του μοντέλου



Εικόνα 4.8. Συνάρτηση απωλειών με την πάροδο των εποχών.

4.2.3 Επιλογή αλγορίθμου βελτιστοποίησης

Θα γίνει σύγκριση των δύο αλγορίθμων βελτιστοποίησης για να βρεθεί ο πιο κατάλληλος για την πρόβλεψη θερμοκρασίας. Οι δύο αλγόριθμοι είναι ο SGD (Stochastic Gradient Descent) και ο Adam (Adaptive Moment Estimation). Για την ακρίβεια χρησιμοποιούνται οι κατηγορίες που αναφέρθηκαν στο τρίτο κεφάλαιο (3.2.3). Η σύγκριση γίνεται με εκατό εποχές.

Μοντέλο Βελτιστοποίησης	Σύνολο εκπαίδευσης	
	Απώλεια	Ακρίβεια(%)
SGD	0.0046	87.28
Adam	0.0024	91.92

Πίνακας 4.1. Σύγκριση αλγορίθμων βελτιστοποίησης.

Το μοντέλο Adam φαίνεται να αποδίδει καλύτερα, επομένως και θα επιλεχθεί. Η καλύτερη απόδοση οφείλεται στο ότι ο Adam ενημερώνει τα βάρη αναδρομικά, ενώ ο SGD επιλέγει τυχαία ένα σύνολο δεδομένων για την ενημέρωση των βαρών.

4.3 Αποθήκευση και φόρτωση του μοντέλου

Έπειτα από την εκπαίδευση του επιθυμητού μοντέλου, συνηθίζεται η αποθήκευση του μοντέλου σε ένα αρχείο, για την επαναχρησιμοποίηση του. Το Keras προσφέρει την δυνατότητα αποθήκευσης και φόρτωσης του μοντέλου.

```
regressor.save("rnn_model" + '.h5')
```

Εικόνα 4.9. Εντολή αποθήκευσης μοντέλου.

```
from keras.models import load_model

model = load_model("rnn_model.h5")
```

Εικόνα 4.10. Εντολή φόρτωσης μοντέλου.

4.4 Πρόβλεψη

Για την πρόβλεψη θα δημιουργηθεί ένα σύνολο επαλήθευσης και θα προβλέπει την επόμενη μέρα, η οποία θα συγκριθεί με τις πραγματικές θερμοκρασίες του αντίστοιχου σταθμού. Έπειτα, το σύνολο θα μετατραπεί σε μορφή κατάλληλη για να τροφοδοτηθεί στο μοντέλο και θα πραγματοποιηθεί η πρόβλεψη. Τέλος, θα γίνει αντιστροφή μετατροπής της πρόβλεψης από το εύρος [0,1] στην πραγματική θερμοκρασία.

```
testing = sc.transform(validation_set)
testing = np.array(testing)
testing = np.reshape(testing,(testing.shape[1],testing.shape[0],1))
```

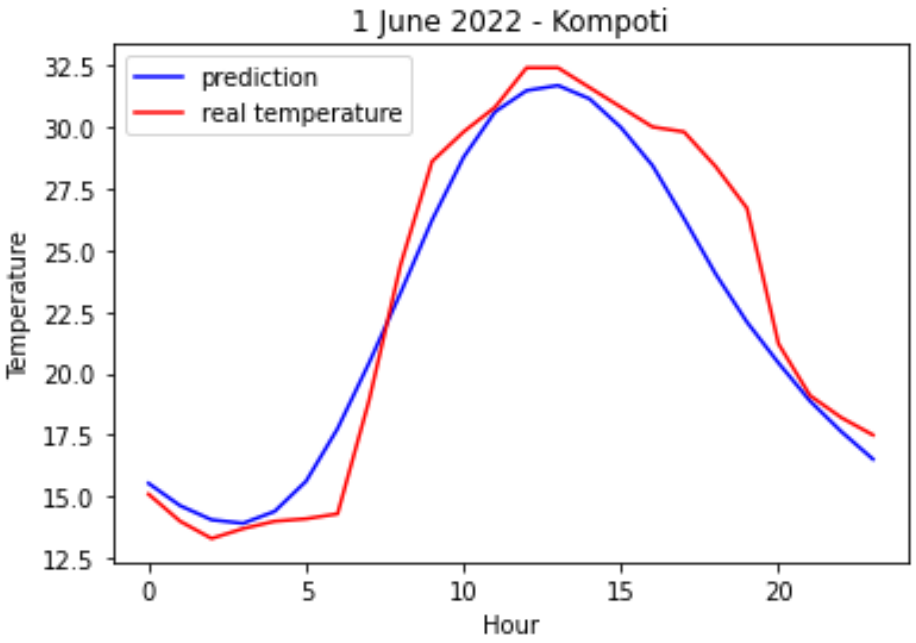
Εικόνα 4.11. Μετατροπή του συνόλου επαλήθευσης κατάλληλη για το μοντέλο

```
predicted_temperature = regressor.predict(testing)
predicted_temperature = sc.inverse_transform(predicted_temperature)
predicted_temperature = np.reshape(predicted_temperature,(predicted_temperature.shape[1],
predicted_temperature.shape[0]))
```

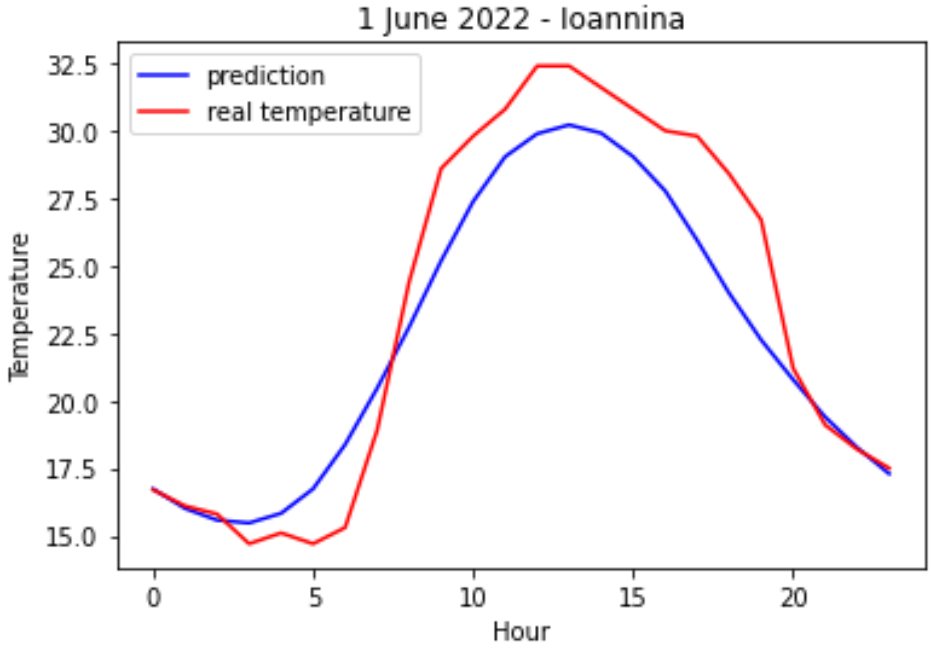
Εικόνα 4.12. Πρόβλεψη θερμοκρασίας και μετατροπή πρόβλεψης σε αναγνώσιμη μορφή.

Αφού πραγματοποιηθούν οι προβλέψεις, πρέπει να γίνει σύγκριση των αποτελεσμάτων με τις πραγματικές θερμοκρασίες, για τον έλεγχο της απόδοσης του μοντέλου.

4.5 Αποτελέσματα



Εικόνα 4.13. Παράδειγμα από σύνολο δεδομένων εντός Άρτας.



Εικόνα 4.14. Παράδειγμα πρόβλεψης εκτός Άρτας.

Στον παρακάτω πίνακα μετρήθηκε το μέσο τετραγωνικό σφάλμα για 10 ημέρες, κάνοντας χρήση των πραγματικών τιμών θερμοκρασίας των 3 προηγούμενων ημερών για την πρόβλεψη της ημέρας:

Περιοχή	**MSE**
Κομπότι	3.52
Καμπή	3.67
Κομμένο	3.63
Ιωάννινα	4.88

Πίνακας 4.2. Τιμές μέσων τετραγωνικών σφαλμάτων της πρόβλεψης θερμοκρασιών για 10 ημέρες.

Στις περιοχές εντός του δήμου Άρτας τα σφάλματα είναι κοντά μεταξύ τους, γεγονός αναμενόμενο καθώς η διαφορά θερμοκρασίας αλλάζει με παρόμοιο ρυθμό σε αυτούς τους σταθμούς. Ωστόσο, είναι φανερό ότι το μοντέλο είναι λιγότερο αποτελεσματικό για τα Ιωάννινα. Σε αυτό μπορεί να οφείλεται η υψομετρική διαφορά των δύο περιοχών, που ευθύνεται για πιο απότομη αλλαγή στις θερμοκρασίες, που παρατηρείται στο διάγραμμα παραπάνω. Λύση σε αυτό το πρόβλημα είναι η επανεκπαίδευση του μοντέλου στο σύνολο δεδομένων των Ιωαννίνων.

Τέλος, αφού το μοντέλο βγάζει επιθυμητά αποτελέσματα μετά από την εκπαίδευση, το μοντέλο αποθηκεύεται και φορτώνεται στην διαδικτυακή υπηρεσία. Η διαδικτυακή υπηρεσία θα τρέχει μια φορά την ημέρα, και θα εμφανίζει στον χρήστη τις θερμοκρασίες των επόμενων ημερών. Εάν υπάρξουν ακραίες θερμοκρασίες, θα ειδοποιείται ο χρήστης. Συγκεκριμένα, ο χρήστης θα ειδοποιείται όταν το μοντέλο θα προβλέπει θερμοκρασίες στους 0° C και κάτω και σε θερμοκρασίες άνω των 38° C.

Βιβλιογραφία

ActiveState Logo. 2022. *What Is Numpy Used For In Python?*. [online] Available at: <https://www.activestate.com/resources/quick-reads/what-is-numpy-used-for-in-python/> [Accessed 5 August 2022].

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., Vanderplas, J., Joly, A., Holt, B. and Varoquaux, G., 2022. *API design for machine learning software: experiences from the scikit-learn project*. [online] arXiv.org. Available at: <https://arxiv.org/abs/1309.0238> [Accessed 3 June 2022].

En.wikipedia.org. 2022. *Project Jupyter - Wikipedia*. [online] Available at: <https://en.wikipedia.org/wiki/Project_Jupyter> [Accessed 9 September 2022].

Kumar, B., 2022. *What Is Matplotlib And How To Use It In Python - Python Guides*. [online] Python Guides. Available at: <https://pythonguides.com/what-is-matplotlib/> [Accessed 8 August 2022].

Matplotlib.org. 2022. *Matplotlib — Visualization with Python*. [online] Available at: <https://matplotlib.org/> [Accessed 1 June 2022].

Mucherino, A., Papajorgji, P. and Pardalos, P., 2009. *Data mining in agriculture*. Dordrecht: Springer, pp.19-20.

Murugesan, A., 2022. *What Is Pandas? | How It Works | Skills And Advantages | Role & Structure*. [online] EDUCBA. Available at: <https://www.educba.com/what-is-pandas/> [Accessed 5 August 2022].

Numpy.org. 2022. *NumPy: the absolute basics for beginners — NumPy v1.24.dev0 Manual*. [online] Available at: <https://numpy.org/devdocs/user/absolute_beginners.html> [Accessed 3 June 2022].

Numpy.org. 2022. *What is NumPy? — NumPy v1.22 Manual*. [online] Available at: <https://numpy.org/doc/stable/user/whatisnumpy.html> [Accessed 1 June 2022].

Pandas.pydata.org. 2022. *pandas - Python Data Analysis Library*. [online] Available at: <https://pandas.pydata.org/> [Accessed 1 June 2022].

Pip.pypa.io. 2022. *pip install - pip documentation v22.1.2*. [online] Available at: <https://pip.pypa.io/en/stable/cli/pip_install/> [Accessed 6 June 2022].

PyPI. 2022. [online] Available at: <https://pypi.org/> [Accessed 1 June 2022].

Ronquillo, A., 2022. *Python'S Requests Library (Guide) – Real Python*. [online] Realpython.com. Available at: <https://realpython.com/python-requests/> [Accessed 8 August 2022].

Saeed, M., 2021. An introduction to recurrent neural networks and the math that powers them. Machine Learning Mastery. Available at: https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/ [Accessed September 12, 2022].

scikit-learn. 2022. *Choosing the right estimator*. [online] Available at: <https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html> [Accessed 6 June 2022].

Saxena, S., 2021. LSTM: Introduction to LSTM: Long short term memory. Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/ [Accessed September 12, 2022].

Scikitlearn.org. 2022. *Loading...*. [online] Available at: <http://scikitlearn.org/stable/)> [Accessed 1 June 2022].

Seaborn.pydata.org. 2022. *seaborn: statistical data visualization — seaborn 0.11.2 documentation*. [online] Available at: <https://seaborn.pydata.org/> [Accessed 1 June 2022].

Tan, P., Steinbach, M., Karpatne, A. and Kumar, V., 2006. *Introduction to data mining*. pp.2,3.

Tutorialspoint.com. 2022. *Python - Overview*. [online] Available at: <https://www.tutorialspoint.com/python/python_overview.htm> [Accessed 8 August 2022].