ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ-ΒΑΘΙΑ ΜΑΘΗΣΗ

Έκθεση αποτελεσμάτων για την 1η εργασία

Η εργασία μου αφορά την πρόβλεψη των επιπέδων του καρκίνου του πνεύμονα βάσει συνόλου χαρακτηριστικών υγείας και τρόπου ζωής.

1.ΕΠΙΣΚΟΠΗΣΗ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ

Το σύνολο δεδομένων περιέχει 1000 εγγραφές ασθενών με διάφορα χαρακτηριστικά όπως ηλικία(Age), φύλο(Gender), Ατμοσφαιρική Ρύπανση(air pollution exposure),alcohol use, Αλλεργία στη σκόνη(dust allergy), occupational hazards, Γενετικός Κίνδυνος(genetic risk),Χρόνια Πνευμονική Πάθηση(chronic lung disease),Ισορροπημένη διατροφή(balanced diet),Παχυσαρκία(obesity), Κάπνισμα(smoking),Παθητικός Καπνιστής(passive smoker) , Πόνος στήθους(chest pain), Βήχας με αίμα(coughing of blood), Κόπωση(fatigue),Απώλεια βάρους(weight loss), Δύσπνοια(shortness of breath), Άσθμα(wheezing), Κατάποση(swallowing difficulty), clubbing of finger nails, Συχνό κρυολόγημα(frequent cold), Ξηρός Βήχας(dry cough), Ρόγχος(snoring)

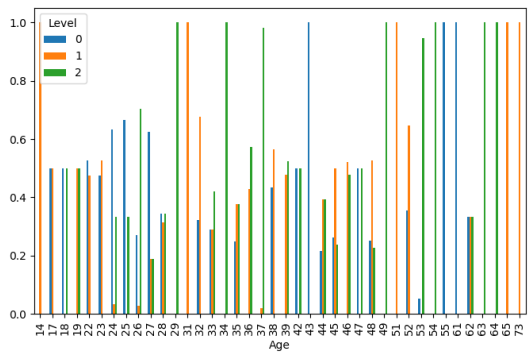
DEMOGRAPHIC FEATURES: Age, Gender

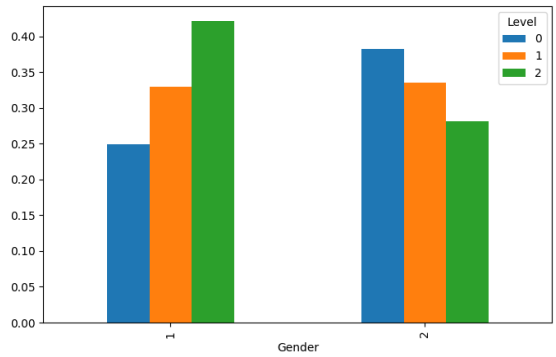
ENVIRONMENTAL FEATURES: AirPollution, DustAllergy, OccupationalHazards

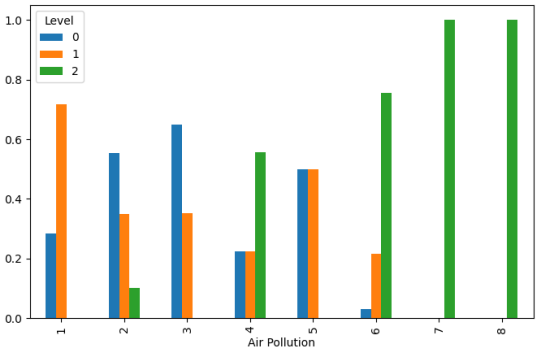
HEALTHRISKS: GeneticRisk, ChronicLungDisease, Balanced Diet, Obesity, Smoking, Passive Smoker

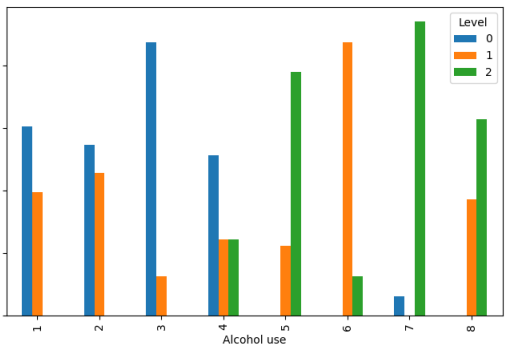
SYMPTOMS: ChestPain, CoughingOfBlood, Fatigue, WeightLoss, ShortnessOfBreath, Wheezing, Swallowing, ClubbingOfFingerNails, FrequentCold, DryCough, Snoring

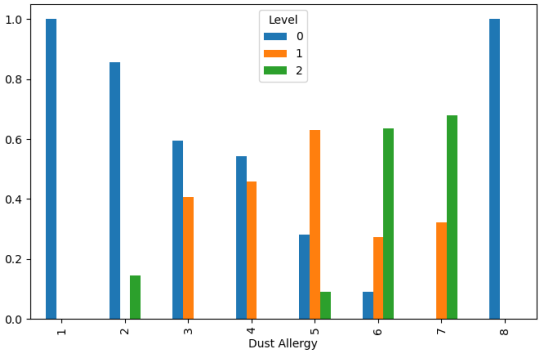
Α.ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ ΣΕ ΓΡΑΦΟ:

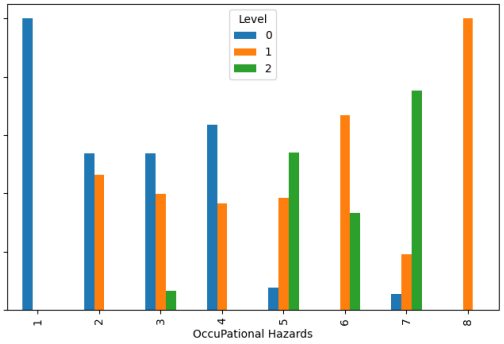


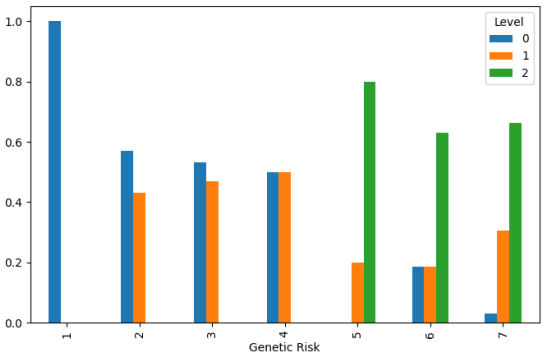


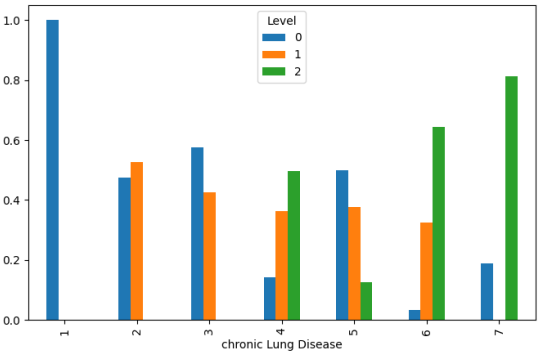

Level
0
1
2
Air Pollution

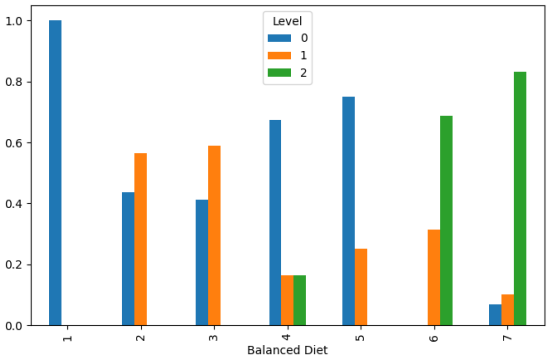

Level
0
1
2
Alcohol use

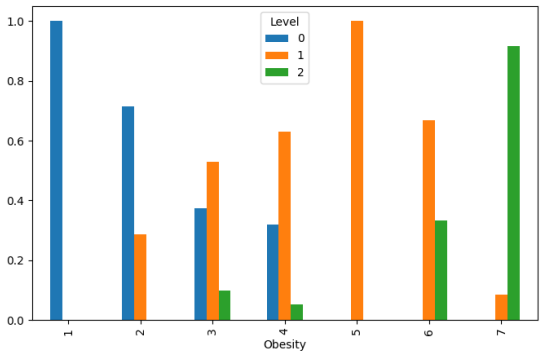

Level
0
1
2
Dust Allergy

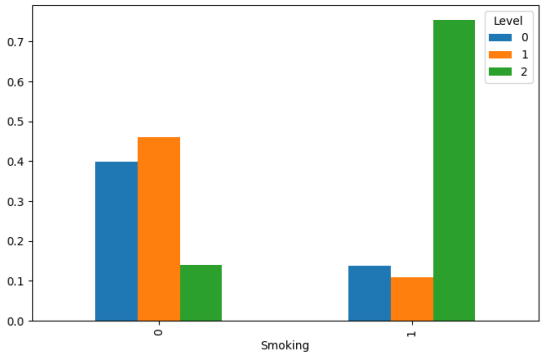

Level
0
1
2
OccuPational Hazards

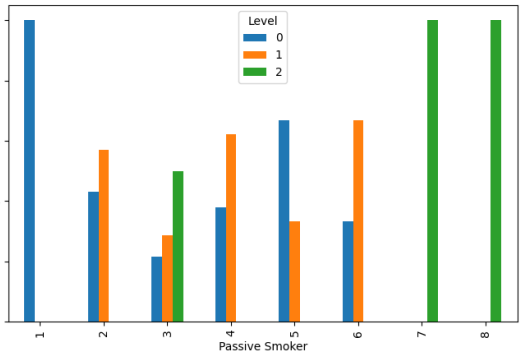

Level
0
1
2
Genetic Risk

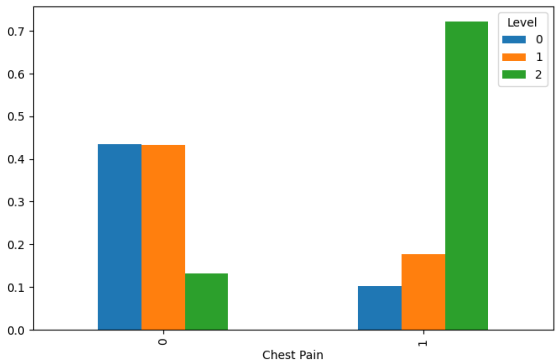

Level
0
1
2
chronic Lung Disease

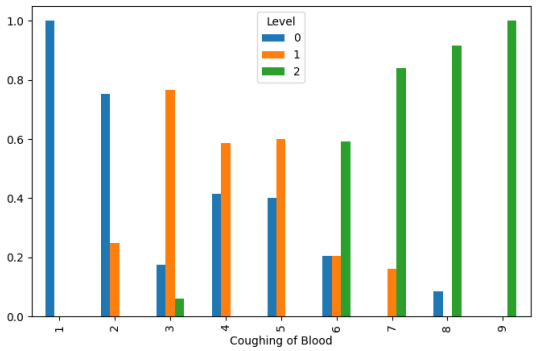

Level
0
1
2
Balanced Diet

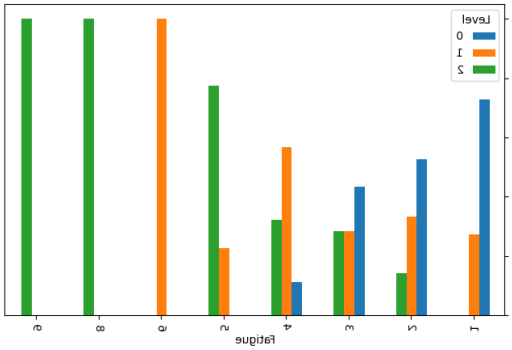

Level
0
1
2
Obesity

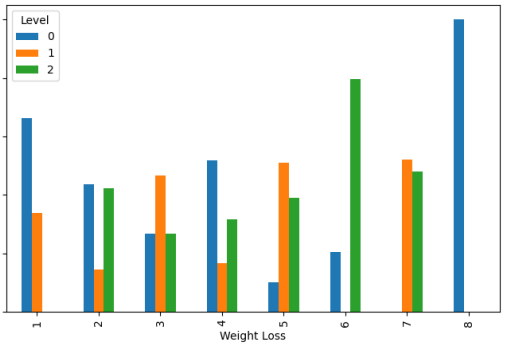

Level
0
1
2
Smoking

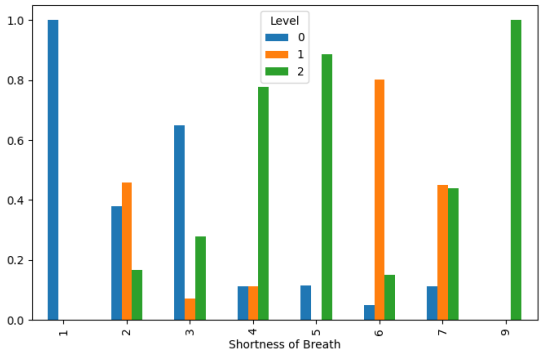

Level
0
1
2
Passive Smoker

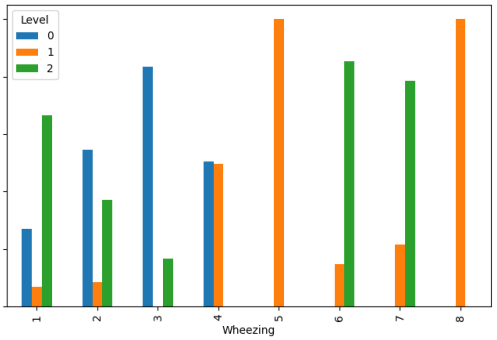

Level
0
1
2
Chest Pain

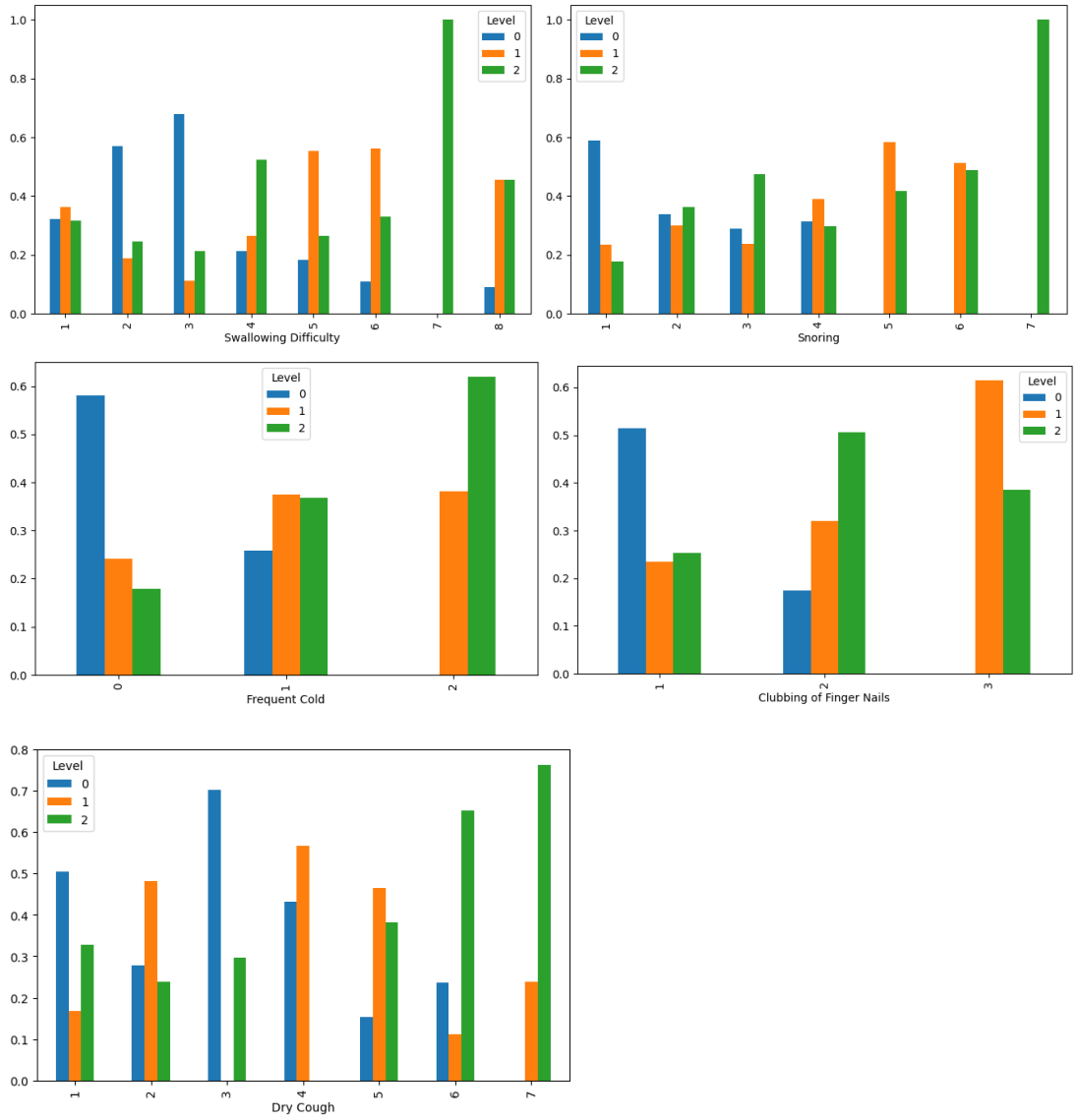

Level
0
1
2
Coughing of Blood




Level
0
1
2
Weight Loss


Level
0
1
2
Shortness of Breath


Level
0
1
2
Wheezing



ΕΡΜΗΝΕΙΑ:

1.AGE: Η ηλικία από μόνη της δεν αποτελεί ισχυρό προγνωστικό παράγοντα σοβαρότητας κάτι που υποδηλώνει ότι η αλληλεπίδραση της ηλικίας με άλλα χαρακτηριστικά είναι πιο καθοριστική.

2.GENDER: Οι διαφορές στα επίπεδα σοβαρότητας μεταξύ των φύλων δεν είναι έντονες.

3.AIR POLLUTION: Η ατμοσφαιρική ρύπανση φαίνεται να αποτελεί σημαντικό περιβαλλοντικό παράγοντα που επηρεάζει την σοβαρότητα του καρκίνου πιθανότατα λόγω μακροχρόνιας βλάβης στο αναπνευστικό σύστημα.

4.ALCOHOL USE: Τα υψηλότερα επίπεδα κατανάλωσης αλκοόλ(επίπεδα 6-8) σχετίζονται με αυξημένη σοβαρότητα έτσι μπορούμε να συμπεράνουμε ότι η υπερβολική κατανάλωση αλκοόλ μπορεί να συμβάλλει στη σοβαρότητα του

καρκίνου του πνεύμονα πιθανόν λόγω της επίδρασης στο αναπνευστικό σύστημα και γενικά στην υγεία.

5.DUST ALLERGY: Τα υψηλότερα επίπεδα αλλεργίας στην σκόνη (επίπεδα 6-8) σχετίζονται με αυξημένη σοβαρότητα έτσι έχουμε σαν συμπέρασμα ότι η ευαισθησία στη σκόνη μπορεί να διαδραματίζει σημαντικό ρόλο στην εξέλιξη του καρκίνου το πνεύμονα.

6.OCCUPATIONAL HAZARDS: Τα υψηλά επίπεδα επαγγελματικών κινδύνων (επίπεδα 7 και 8) συσχετίζονται με αυξημένη σοβαρότητα καρκίνου του πνεύμονα**,** έτσι επαγγελματική έκθεση μπορεί να αποτελεί σημαντικό περιβαλλοντικό παράγοντα κινδύνου που συμβάλλει στη σοβαρότητα του καρκίνου του πνεύμονα

7.GENETIC RISK: Άτομα με υψηλό γενετικό κίνδυνο (επίπεδα 5 έως 7) παρουσιάζουν σαφή συσχέτιση με μεγαλύτερη σοβαρότητα καρκίνου. Τα χαμηλά επίπεδα γενετικού κινδύνου δείχνουν ποικιλία σε πιο χαμηλά επίπεδα σοβαρότητας συνεπώς ο γενετικός κίνδυνος είναι ένας σημαντικός παράγοντας, καθώς τα υψηλά επίπεδα γενετικής προδιάθεσης συσχετίζονται με αυξημένη σοβαρότητα καρκίνου του πνεύμονα.

8.CHRONIC LUNG DISEASE: Η χρόνια πνευμονοπάθεια φαίνεται να αποτελεί σημαντικό παράγοντα κινδύνου για τα υψηλά επίπεδα σοβαρότητας καρκίνου του πνεύμονα.

9.BALANCED DIET: Η έλλειψη ισορροπημένης διατροφής συνδέεται με την αυξημένη σοβαρότητα του καρκίνου.

10.OBESITY: Επίσης η παχυσαρκία φαίνεται να αποτελεί σημαντικό παράγοντα κινδύνου για αυξημένη σοβαρότητα του καρκίνου του πνεύμονα

11.SMOKING: Το κάπνισμα σχετίζεται έντονα με τη μεγαλύτερη σοβαρότητα δηλαδή επίπεδο 2 ενώ οι μη καπνιστές δείχνουν περισσότερα περιστατικά με χαμηλή η με μέτρια σοβαρότητα.

12.PASSIVE SMOKER: Τα υψηλά επίπεδα έκθεσης στο παθητικό κάπνισμα(επίπεδο 7,8) συσχετίζονται έντονα με υψηλή σοβαρότητα ενώ τα χαμηλότερα επίπεδα έκθεσης δείχνουν περισσότερα περιστατικά με χαμηλή σοβαρότητα.

13.CHEST PAIN: Ο πόνος στο στήθος σχετίζεται έντονα με υψηλή σοβαρότητα επιπέδου του καρκίνου ενώ η απουσία πόνου στο στήθος δείχνει ευρεία κατανομή στις σοβαρότητα του καρκίνου.

14.COUGHING OF BLOOD: Ο βήχας με αίμα αποτελεί σοβαρό σύμπτωμα που υποδεικνύει προχωρημένο επίπεδο της ασθένειας.

15.FATIGUE: Τα υψηλά επίπεδα κόπωσης (επίπεδα 6+) συσχετίζονται έντονα με υψηλή σοβαρότητα ενώ τα χαμηλότερα επίπεδα κόπωσης συνδέονται με χαμηλότερη σοβαρότητα.

16.WEIGHT LOSS: Η σημαντική απώλεια βάρους μπορεί να αποτελεί σύμπτωμα προχωρημένου καρκίνου.

17.SHORTNESS OF BREATH: Τα υψηλότερα επίπεδα δύσπνοιας (επίπεδα 5+) συνδέονται με αυξημένη σοβαρότητα ενώ τα χαμηλότερα επίπεδα συνδέονται με χαμηλότερη σοβαρότητα.

18.WHEEZING: Υψηλά επίπεδα συριγμού (επίπεδα 5+συνδέονται σχεδόν αποκλειστικά με υψηλότερη σοβαρότητα Τα χαμηλά επίπεδα συριγμού παρουσιάζουν μεγαλύτερη ποικιλία στα επίπεδα σοβαρότητας κυρίως όμως χαμηλά επίπεδα.

19.SWALLOWING DIFFICULTY: Η αυξημένη δυσκολία στην κατάποση (επίπεδα 6-8) συνδέεται έντονα με υψηλότερη σοβαρότητα ενώ τα χαμηλότερα επίπεδα δείχνουν πιο ευρεία κατανομή μεταξύ των επιπέδων σοβαρότητας.

20.CLUBBING OF FINGER NAILS: Τα υψηλά επίπεδα δακτυλογραφίας (επίπεδο 3) συσχετίζονται περισσότερο με υψηλή σοβαρότητα ενώ χαμηλά ή ανύπαρκτα επίπεδα δακτυλογραφίας σχετίζονται με χαμηλότερη σοβαρότητα**.**

21.FREQUENT COLD: Τα συχνά αναπνευστικά προβλήματα μπορεί να συμβάλλουν σε μεγαλύτερη σοβαρότητα καρκίνου του πνεύμονα, ενδεχομένως υποδεικνύοντας μια αποδυναμωμένη ανοσολογική απόκριση στις πιο σοβαρές περιπτώσεις.

22.SNORING: Yψηλότερα επίπεδα ροχαλητού (επίπεδο 7) σχετίζονται κυρίως με τη μεγαλύτερη σοβαρότητα του καρκίνου ενώ χαμηλότερα επίπεδα ροχαλητού δείχνουν μεγαλύτερη κατανομή σε χαμηλότερα επίπεδα σοβαρότητας.

23.DRY COUGH: Τα υψηλά επίπεδα του ξηρού βήχα σχετίζονται με το υψηλό επίπεδο του καρκίνου του πνεύμονα ενώ τα χαμηλότερα επίπεδα δείχνουν αντίστοιχα το χαμηλό επίπεδο του καρκίνου.

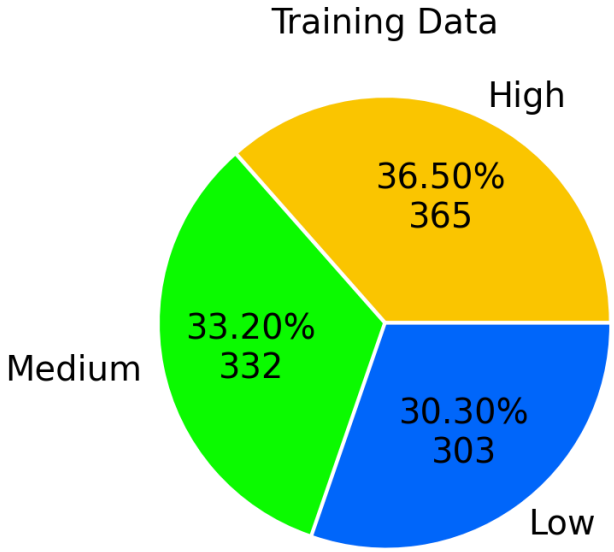
Πρόβλεψη:

Θα αναλύσουμε και θα αναπτύξουμε μια μέθοδο πρόβλεψης για το επίπεδο σοβαρότητας του καρκίνου του πνεύμονα που χωρίζεται σε τρία επίπεδα, χαμηλό, μεσαίο και υψηλό(low,medium,high). Αναλύοντας αυτά τα δεδομένα θα αποκτήσουμε μια εικόνα για το τι προκαλεί τον καρκίνο του πνεύμονα και πώς να τον αποτρέψουμε καλύτερα.

Άρα υπάρχουν 23 στήλες χαρακτηριστικών και 1 στήλη στόχος(επίπεδο σοβαρότητας καρκίνου του πνεύμονα) άρα στο σύνολο 24 στήλες. Επίσης υπάρχουν και 1000 δείγματα χαρακτηριστικών των ανθρώπων άρα στο σύνολο έχουμε διάσταση 1000x24

2. TRAINING TEST SPLITTING:

TRAINING DATA IN A PIE:



Splitting the data size into training set size(600 samples) and test set size(400 samples) 60%-40%

3. ΠΡΟΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ

Εφαρμόστηκαν διάφορες τεχνικές έτσι ώστε να γίνει η σωστή προεπεξεργασία του κάθε χαρακτηριστικού και κατ' επέκταση των δεδομένων έτσι ώστε να υπάρχει καλύτερη ακρίβεια των ταξινομητών(knn classifier with 1 neighbor,with 3 neighbors and nearest centroid)

ΕΝΕΡΓΕΙΑ 1: Αφαίρεση μη σχετικών στηλών

Αφαιρέθηκαν από το dataset οι στήλες index , patient id καθώς δεν συμβάλλουν στην ακρίβεια πρόβλεψης. Με την απόρριψη άσχετων στηλών, το μοντέλο επικεντρώνεται αποκλειστικά σε προγνωστικά χαρακτηριστικά, μειώνοντας το θόρυβο και βελτιώνοντας την ταχύτητα επεξεργασίας. Αυτό ελαχιστοποίησε επίσης τον κίνδυνο υπερβολικής προσαρμογής σε μη ενημερωτικά δεδομένα.

{data=data.drop(columns=["index","Patient Id"])}

ΕΝΕΡΓΕΙΑ 2: Μετασχηματισμός ορισμένων χαρακτηριστικών από κλίμακα σε δυαδική μορφή

Μετασχηματίστηκαν οι στήλες Smoking, Chest Pain σε δυαδική μορφή διότι κάθε τιμή αντιπροσωπεύει μια απλή παρουσία ή απουσία ενός χαρακτηριστικού και χρησιμοποιήθηκε ένα threshold που το έθεσα ίσο με 5 έτσι ώστε να είναι το mid point της κλίμακας για πιο ακριβή αποτελέσματα(Do you smoke/have chest pain YES:1 NO:0)

Αυτή η ενέργεια βοηθά το μοντέλο να μην υποθέσει σχέσεις που δεν έχουν κάποιο ιδιαίτερο νόημα και το απλοποιεί ενισχύοντας έτσι την ερμηνευτικότητα του μοντέλου.

ΕΝΕΡΓΕΙΑ 3: Ορισμένα χαρακτηριστικά κατηγοριοποιήθηκαν σε 4 επίπεδα για μεγαλύτερη συνέπεια(0,1,2,3)

Κατηγοριοποίηση στηλών Frequent Cold , Clubbing Finger Nails σε επίπεδα. Το χαρακτηριστικό Frequent Cold μετατράπηκε σε κατηγορίες για να δείξει τη συχνότητα(σπάνια=0 έως πολύ συχνά=3) ενώ αντίστοιχα το άλλο χαρακτηριστικό για να δείξει την σοβαρότητα της πάθησης(απουσία=0 έως και σοβαρό=3). Με αυτό το τρόπο μετατρέπονται σε τακτικές κατηγορίες συχνότητας και σοβαρότητας χωρίς να μειώνονται σε δυαδική μορφή.

ΕΝΕΡΓΕΙΑ 4: Κωδικοποιήση Μεταβλητής στόχου

Η μεταβλητή Level κωδικοποιήθηκε σε αριθμητικές τιμές (0, 1, 2), που αντιπροσωπεύουν τα επίπεδα κινδύνου Χαμηλό, Μεσαίο και Υψηλό(Low,Medium,High) έτσι ώστε να είναι συμβατό με τα μοντέλα.

ΕΝΕΡΓΕΙΑ 5: Κλιμάκωση Χαρακτηριστικών

Για την κλιμάκωση των χαρακτηριστικών(Age,Genetic Risk,Air Pollution,Dust Allergy,Weight Loss) χρησιμοποιήθηκε η τεχνική Min-Max Scaling για την βελτίωση της ευαισθησίας που έχει ο knn. Αυτή η κλιμάκωση βελτίωσε την ακρίβεια και τη σταθερότητα του μοντέλου κανονικοποιώντας τα εύρη των πιο πάνω χαρακτηριστικών, διασφαλίζοντας με αυτό τον τρόπο ότι κάθε χαρακτηριστικό συνέβαλε σημαντικά στις αποφάσεις του μοντέλου. Ο Nearest Centroid, επωφελήθηκε από αυτήν την κλιμάκωση, καθώς επέτρεψε ακριβέστερους υπολογισμούς κεντροειδών με βάση την απόσταση, με αποτέλεσμα πιο σαφή διαφοροποίηση της κατηγορίας και υψηλότερη ακρίβεια ταξινόμησης.

ΕΝΕΡΓΕΙΑ 6 : Ελέγχω ένα υπάρχουν μηδενικές τιμές

```
print(data.isnull().any())
```

4.ΤΑΞΙΝΟΜΗΤΕΣ(CLASSIFIERS):

Α. ΕΠΙΛΟΓΗ ΤΑΞΙΝΟΜΗΤΩΝ:

K-Nearest Neighbors (KNN):

KNN με 1 Γείτονα (k=1): Μοντέλο όπου ΚΑΘΕ δείγμα δοκιμής ταξινομείται βάσει του πλησιέστερου σημείου δεδομένων.

KNN με 3 Γείτονες (k=3): Μοντέλο που χρησιμοποιεί τα τρία πλησιέστερα σημεία για τη λήψη αποφάσεων ταξινόμησης μέσω της πλειοψηφικής ψήφου.

Πλησιέστερο Κέντρο(Nearest Centroid): Ο ταξινομητής αυτός αναθέτει ετικέτες βάσει του πλησιέστερουκέντρου κάθε τάξης σε κάθε δείγμα δοκιμής. Είναι λιγότερο ευαίσθητος σε θόρυβο αλλά μπορεί να υστερεί σε διάκριση σε σύνθετα σύνολα δεδομένων.

Β.ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΑΞΙΝΟΜΗΤΩΝ:

K-Nearest Neighbors (KNN): Και οι δύο διαμορφώσεις πέτυχαν ακρίβεια 100%, γεγονός που υποδεικνύει ότι το KNN καταφέρνει να καταλάβει τη δομή και τα όρια του συνόλου δεδομένων αποτελεσματικά. Αυτό το αποτέλεσμα υποδηλώνει ότι τα δεδομένα έχουν καθαρούς και σωστούς διαχωρισμούς μεταξύ των χαρακτηριστικών, καθιστώντας το KNN(κ=1,κ=3) εξαιρετικά αποδοτικό για αυτήν την ταξινόμηση.

Πλησιέστερο Κέντρο(Nearest centroid): Με ακρίβεια 89%, το Πλησιέστερο Κέντρο απέδωσε καλά αλλά δεν έφτασε την τελειότητα που επιτεύχθηκε με το KNN. Αυτή η διαφορά στην απόδοση υποδεικνύει ότι, ενώ το Πλησιέστερο Κέντρο επωφελείται από τα χαρακτηριστικά που έχουν κλιμακωθεί, ενδέχεται να μην καταλάβει τόσο αποτελεσματικά τα σύνθετα όρια των κατηγοριών όσο το KNN. Ωστόσο, η υψηλή ακρίβεια υποδηλώνει ότι τα βήματα προεπεξεργασίας, ιδίως η κλιμάκωση χαρακτηριστικών και η επιλεκτική δυαδική μετατροπή, έκαναν το σύνολο δεδομένων πιο κατάλληλο για ταξινόμηση βάσει αποστάσεων.

ΤΑΞΙΝΟΜΗΤΗΣ	**ΑΚΡΙΒΕΙΑ(%)**
KNN ΜΕ 1 ΓΕΙΤΟΝΑ	100%
KNN ΜΕ 3 ΓΕΙΤΟΝΕΣ	100%
ΠΛΗΣΙΕΣΤΕΡΟ ΚΕΝΤΡΟ	89.5%

1η ΕΡΓΑΣΙΑ:

Ζητούμενο εργασίας:

Να γραφεί πρόγραμμα σε οποιαδήποτε γλώσσα προγραμματισμού το οποίο να υλοποιεί ένα νευρωνικό δίκτυο εμπρόσθιας τροφοδότησης (feedforward NN) (το δίκτυο μπορεί να είναι πλήρως συνδεδεμένο (MLP) ή συνελικτικό (CNN) ή συνδυασμός) που θα εκπαιδεύεται με τον αλγόριθμο back-propagation. Το NN αυτό θα εκπαιδευτεί για να επιλύει οποιοδήποτε πρόβλημα κατηγοριοποίησης πολλών κλάσεων ΕΚΤΟΣ MNIST με επιβλεπόμενη μάθηση ή αυτό-επιβλεπόμενη μάθηση

Πριν υλοποιήσουμε ένα νευρωνικό δίκτυο υπάρχει μια διαδικασία η οποία πρέπει να γίνει έτσι ώστε τα δεδομένα μας να εκπαιδευτούν σωστά για να πάρουμε καλά αποτελέσματα. Πρέπει να γίνει μια προεπεξεργασία των δεδομένων. Στη πιο πάνω

αναφορά κάναμε μια προεπεξεργασία η οποία ήταν κυρίως για την υλοποίηση των ταξινομητών knn και nearest centroid. Σε αυτή την 1η εργασία έκανα drop out τις στήλες index&PatientID οι οποίες δεν είναι σημαντικές για την εξαγωγή αποτελεσμάτων και την εκπαίδευση των δεδομένων και χρησιμοποίησα των StandardScaler για να εφαρμόσω scaling σε όλες τις στήλες εκτός του target.

ΕΠΕΞΗΓΗΣΗ: Στην 1η ενδιάμεση εργασία έκανα περισσότερη προεπεξεργασία στα δεδομένα όμως τώρα επειδή θέλω όλα τα χαρακτηριστικά να είναι περίπου όμοια στα αριθμητικά χαρακτηριστικά τους θα χρησιμοποιήσω μόνο StandardScaler για να κάνω όλα τα χαρακτηριστικά (εκτός το target) scaled. Οι ταξινομητές knn & nearest centroid μειώθηκε η ακρίβεια τους και έχουμε τώρα σαν αποτελέσματα

Accuracy of 1-Nearest Neighbor Classifier: 99.50%

Accuracy of 3-Nearest Neighbor Classifier: 99.50%

Accuracy of Nearest Centroid Classifier: 73.50%

OUTLIER:

OUTLIER: Οι ακραίες τιμές (outliers) είναι δεδομένα σε ένα σύνολο που είναι σημαντικά διαφορετικά από τις άλλες τιμες . Είναι είτε μεγαλύτερες είτε μικρότερες από τις υπόλοιπες τιμές του συνόλου και μπορεί να υποδηλώνουν σφάλμα. Άρα τα outlier , ένα νευρωνικό δίκτυο δεν θα ήθελε να υπάρχει στο dataset στο οποίο θα εκπαιδεύσει.

Κάνω μια υλοποίηση z-score όπου ελέγχω για κάθε χαρακτηριστικό αν είναι outlier.

ΠΩΣ? Αρχικά υπολογίζω τα z-scores κάθε χαρακτηριστικού και ορίζω μια τιμή-όριο(threshold) όπου αν το z-score κάποιου χαρακτηριστικού είναι μεγαλύτερη από αυτή την τιμή τότε θα το χαρακτηρίσω ως outlier. Στο τέλος εκτυπώνει κάθε χαρακτηριστικό και αν υπάρχει ή όχι Outlier(TRUE/FALSE)

if (z-score>threshold) →OUTLIER

RESULT: In my dataset there is not an outlier feature

FEATURE SELECTION:

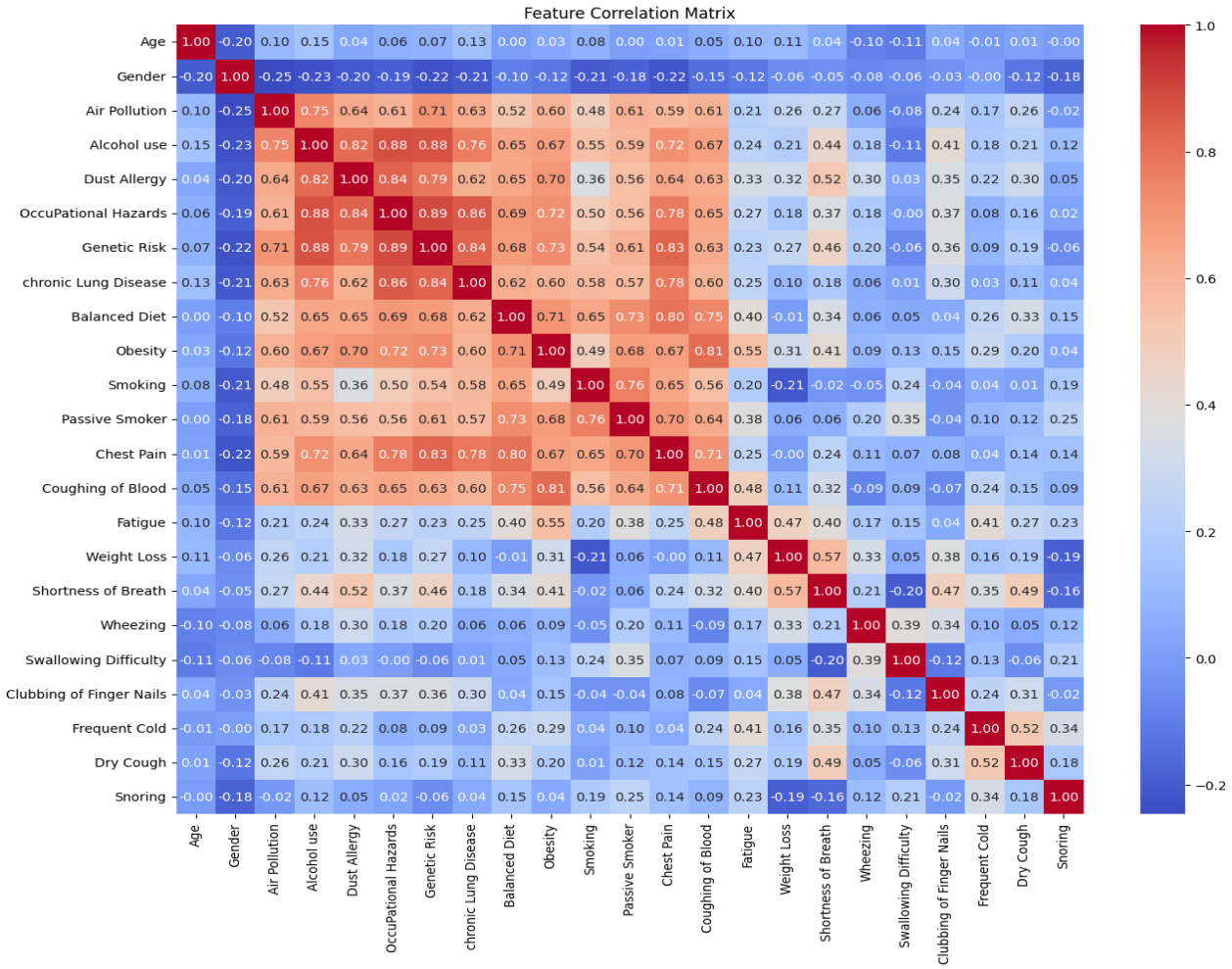
USE CORRELATION MATRIX:

Είναι ένας πίνακας συσχέτισης που δείχνει τις σχέσεις μεταξύ των χαρακτηριστικών του συνόλου δεδομένων . Μετρά τον βαθμό συσχέτισης ανάμεσα σε κάθε ζευγάρι χαρακτηριστικών , για παράδειγμα το βαθμό συσχέτισης Age-Gender , Age-ChronicLungDisease και οι τιμές κυμαίνονται από -1 εως 1. Όπου 1 ισχυρός βαθμός συσχέτισης και -1 αδύναμος βαθμός συσχέτισης.

Άρα υλοποιώ correlationmatrix μαζί στα χαρακτηριστικά είναι και το target variable και έπειτα όταν κάνω plot τον πίνακα με τα ζευγάρια ελέγχω έαν υπάρχουν ζευγάρια τα οποία έχουν υψηλό βαθμό συσχέτισης(>=0.9) έτσι ώστε να τα πάρω σαν

αποτελέσματα και να τα συνενώσω ή να αφαιρέσω το ένα από τα δυο χαρακτηριστικά από το dataset.

RESULT: Στο σύνολο δεδομένων δεν υπάρχουν no highly correlated pairs που θα μπορούσα να συνενώσω χρησιμοποιώντας correlation matrix όλα τα ζευγάρια είναι μικρότερα της τιμής του 0.9. Αυτό έχει επίσης σαν αποτέλεσμα να μην υλοποιήσω PCA γιατί έκανα correlation analysis και είδα ότι δεν έχω ζευγάρια. Όλα μου τα αποτελέσματα κυμαίνονται μεταξύ 0.3-0.7 όπου σημαίνει ότι τα χαρακτηριστικά έχουν ιδιαίτερη σημασία με τον στόχο.



Υπάρχουν κάποια ζευγάρια τα οποία κυμαίνονται από 0.7-0.8 και θεωρούνται λίγο ως υψηλοί βαθμοί συσχέτισης αλλά θα το αγνοήσουμε για τώρα.

CLASS-WEIGHT:

Ελέγχω πόσα samples καταλήγουν στο LEVEL 0,1,2 του target variable(Level).

Βλέπω ότι High:365 Medium:332 Low:303 άρα υπάρχει μικρή ανισορροπία έτσι θα χρησιμοποιήσω την μέθοδο class weight όπου χειρίζεται την ανισορροπία μεταξύ των κλάσεων και βοηθάει στην εκπαίδευση των δεδομένων.

Weight(i)=total samples/(number of classes*frequencyofclass(i)

RESULT:

Class 0(LOW)= έχει πιο υψηλό βάρος διότι έχει λίγα λιγότερα δείγματα έτσι το μοντέλο που θα εκπαιδεύσουμε θα δώσει λίγη περισσότερη αξία σε αυτή την κλάση.

Class 1(MEDIUM)= είναι πιο κοντά στην ισορροπία αρά δεν αλλάζει σχεδόν καθόλου.

Class 2(HIGH)= έχει λιγότερο βάρος γιατί έχει τα πιο πολλά δείγματα άρα το μοντέλο που θα εκπαιδεύσουμε θα του δώσει λιγότερη αξία για να προβλέψει σωστά την κλάση.

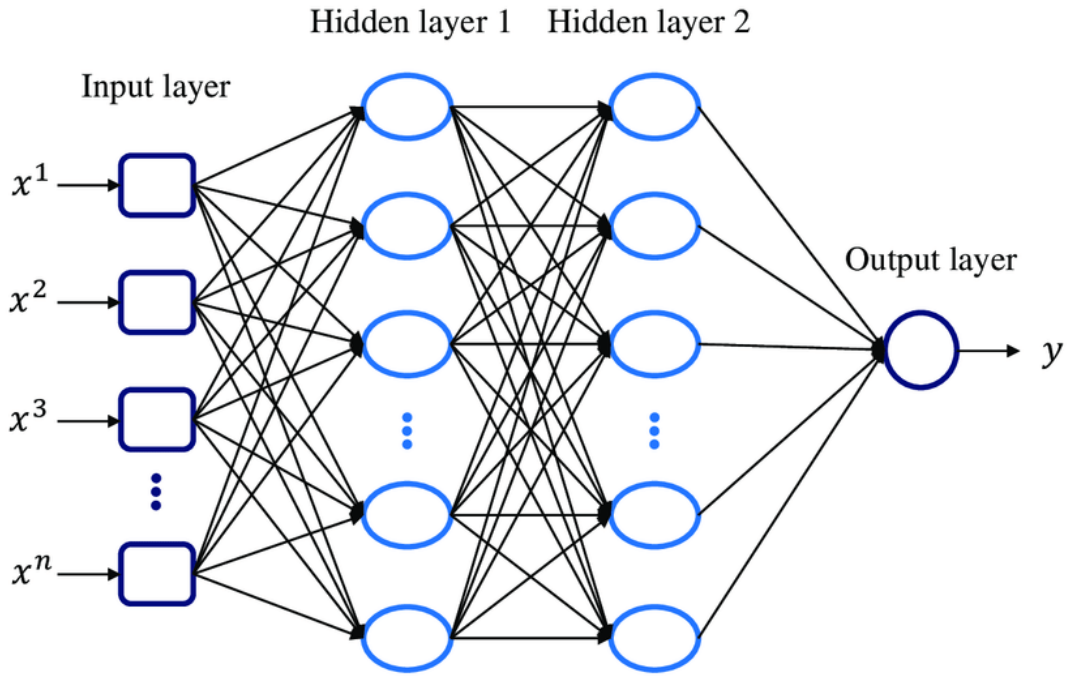
{0: 1.098901098901099, 1: 1.0050251256281406, 2: 0.91324200913242}

ΓΙΑΤΙ ΧΡΗΣΙΜΟΠΟΙΩ MLP NN ARCHITECTURE ΚΑΙ ΟΧΙ CNN?

Το MLP είναι κατάλληλο για δεδομένα πίνακα, πιο συγκεκριμένα το σύνολο δεδομένων μας αποτελείται από αριθμητικά χαρακτηριστικά και το MLP έχει σχεδιαστεί για τη διαχείριση δομημένων δεδομένων όπου το κάθε χαρακτηριστικό του πίνακα είναι ανεξάρτητο.

MLP ARCHITECTURE:

.



INPUT LAYER:

Κάθε νευρώνας είναι ένα χαρακτηριστικό ή διάσταση της εισόδου δεδομένων. Ο αριθμός των νευρώνων στο input layer είναι ο αριθμός των χαρακτηριστικών άρα στη δική μας περίπτωση(23) .

Οι νευρώνες στο input layer δεν κάνουν υπολογισμούς και απλά προωθούν τα δεδομένα των νευρώνων στο πρώτο κρυφό επίπεδο.

HIDDEN LAYER:

Βρίσκεται ανάμεσα στο input layer και στο output layer και μπορεί να υπάρχουν περισσότερα από ένα hidden layer. Κάθε νευρώνας στο hidden layer παίρνει ως είσοδο όλους τους νευρώνες του προηγούμενου layer και παράγει εξόδους οι οποίοι θα εισαχθούν στο επόμενο επίπεδο. Ο αριθμός των hidden layers και του αριθμού των νευρώνων που υπάρχει σε κάθε επίπεδο εξαρτάται από τις υπερπαραμέτρους που πρέπει να αποφασιστούν κατά τη διάρκεια της δημιουργίας του μοντέλου.

INPUTS are multiplied by corresponding weights.

Weighted sum=∑(Wi*Xi)+b(bias)

OUTPUT LAYER:

Αποτελείται από νευρώνες που παράγουν την τελική έξοδο του δικτύου. Ο αριθμός των νευρώνων είναι αναλόγως με τον αριθμό των κλάσεων . Κάθε νευρώνας παίρνει ως είσοδο το output κάθε νευρώνα του προηγούμενου επιπέδου και εφαρμόζει την συνάρτηση ενεργοποίησής

ACTIVATION FUNCTION(ΣΥΝΑΡΤΗΣΗ ΕΝΕΡΓΟΠΟΙΗΣΗΣ):

Κάθε νευρώνας στο κρυφό επίπεδο και στο επίπεδο εξόδου εφαρμόζει μια συνάρτηση ενεργοποίησής στο weighted sum. Κάποιες συνήθεις συναρτήσεις ενεργοποίησής είναι sigmoid,tanh,RELU

ΠΕΙΡΑΜΑΤΙΣΜΟΣ:

1.BASELINE MODEL:

Έπειτα θα κάνω κάποιους πειραματισμούς στο dataset για να ελέγξουμε ποιο μοντέλο είναι πιο κατάλληλο για να το χρησιμοποιήσουμε καθώς και ποιες είναι οι κατάλληλες τιμές των υπερπαραμέτρων.

Αρχικά, προτίμησα να υλοποιήσω ένα μοντέλο χρησιμοποιώντας μέθοδο Λογιστικής Παλινδρόμησης το οποίο λειτουργεί ως μια αρχή για να λύσουμε αυτό το πρόβλημα και μας βοηθάει να συγκρίνουμε την απόδοση με το μοντέλο MLP. Αρχικοποιούμε το μοντέλο και έπειτα του κάνουμε εκπαίδευση με τη χρήση της μεθόδου **fit** χρησιμοποιώντας τα δεδομένα X_train,y_train , αμέσως μετά χρησιμοποιώντας την μέθοδο **predict** παίρνουμε το εκπαιδευόμενο μοντέλο για να κάνει προβλέψεις στα δεδομένα ελέγχου(X_test). Συμπερασματικά αξιολογήσαμε την απόδοση του μοντέλου στα δεδομένα ελέγχου χρησιμοποιώντας την ακρίβεια και το classification report

<u>Logistic Regression Performance:</u>

	precision	recall	f1-score	support
0	0.97	0.93	0.95	121
1	0.94	0.98	0.96	133
2	1.00	1.00	1.00	146

Baseline Accuracy: 0.97

Η ακρίβεια είναι υψηλή άρα τα δεδομένα ενδέχεται να είναι γραμμικά διαχωρίσιμα

2.EXPERIMENT NEURAL NETWORK WITH ONE HIDDEN LAYER:

Στόχος είναι να πειραματιστώ με διάφορες MLP αρχιτεκτονικές δηλαδή διαφορετικούς αριθμούς κρυφών επιπέδων, διαφορετικοί αριθμοί νευρώνων σε κάθε επίπεδο καθώς και διαφορετικές συναρτήσεις ενεργοποίησης και Προσθήκη/Διαγραφή επιπέδων(Dropout) έτσι ώστε να εντοπιστούν οι πιο κατάλληλες παράμετροι.

Στο πρώτο μας παράδειγμα χρησιμοποιούμε InputLayer, (1) single HiddenLayer, OutputLayer.

INPUT LAYER : Χρήση Sequential→ορίζει ένα sequential νευρωνικό δίκτυο όπου κάθε επίπεδο είναι το ένα πίσω από το άλλο.

HIDDEN LAYER : Χρήση Dense→ 64(δηλαδή ένα πλήρες συνδεδεμένο δίκτυο με 64 νευρώνες) και με συνάρτηση ενεργοποίησης relu & input_dim είναι ο αριθμός των χαρακτηριστικών των δεδομένων εκπαίδευσης

OUTPUT LAYER : Αυτό το επίπεδο έχει 3 νευρώνες που αντιστοιχούν στις 3 κλάσεις και έχει συνάρτηση ενεργοποίησης softmax.

Κάνω διάφορους πειραματισμούς διαφοροποιώντας τις υπερπαραμέτρους για να βρω ποιες δίνουν την καλύτερη ακρίβεια στο νευρωνικό δίκτυο που μοντελοποιώ και σε κάθε επανάληψη που επιστρέφει την χρονική διάρκεια ms/step,accuracy,loss,validation_accuracy,validation_loss

1.TRAINING PARAMETERS:

Number of neurons per layer:
16(acc=0.87),32(acc=0.92),64(acc=0.98),128(acc=0.997),**256(acc=1.0)**

Optimizer: **Adam**

Loss Function : **Sparse categorical cross entropy**

Epochs:**50**

Batch size:**32**

Validation split:**20%**

SHALLOW NN ACCURACY: 1.0

2.TRAINING PARAMETERS:

Number of neurons per layer:

16(acc=0.87),32(acc=0.92),64(acc=0.93),128(acc=0.90),**256(acc=0.98)**

Optimizer: **sgd**

Loss Function : **Sparse categorical cross entropy**

Epochs:**50**

Batch size: **32**

Validation split:**20%**

SHALLOW NN ACCURACY: 0.98

3.TRAINING PARAMETERS:

Number of neurons per layer:

16(acc=0.87),32(acc=0.92),64(acc=0.96),128(acc=0.93),**256(acc=0.992)**

Optimizer: **sgd**

Loss Function :**Sparse categorical cross entropy**

Epochs:**100**

Batch size: **64**

Validation split:**20%**

SHALLOW NN ACCURACY: 0.992

4.TRAINING PARAMETERS:

16(acc=0.97),32(acc=1.0),64(acc=1.0),128(acc=1.0),256(acc=1.0)

Optimizer: **adam**

Loss Function: **Sparse categorical cross entropy**

Epochs:**100**

Batch size: **16**

Validation split:**20%**

SHALLOW NN ACCURACY: 1.0

3.EXPERIMENT WITH DEEPER NEURAL NETWORK:

Σε αυτό το πείραμα υλοποιούμε ένα deeper neural network με πρόσθετα επίπεδα και χρησιμοποιούμε dropout regularization.

Στόχος αυτού του πειράματος είναι να αξιολογήσουμε αν βελτιώνεται η απόδοση σε σχέση με το shallow neural network & baseline model.

Εδώ χρησιμοποιούμε 2 κρυφά επίπεδα στο πρώτο κρυφό επίπεδο έχουμε πλήρως συνδεδεμένο επίπεδο με 128 νευρώνες και συνάρτηση ενεργοποιήσης relu ενώ στο δεύτερο επίπεδο είναι πάλι πλήρως συνδεδεμένο αλλά με 64 νευρώνες και συνάρτηση ενεργοποιήσης relu και σε αυτό το παράδειγμα χρησιμοποιούμε και dropout layer όπου τυχαία απενεργοποιούνται το 30% των νευρώνων κατά τη διάρκεια της εκπαίδευσης έτσι ώστε να αποφύγουμε το πρόβλημα το overfitting. Τέλος έχουμε το ouput layer με 3 νευρώνες που αντιστοιχούν στις 3 κλάσεις και γίνεται χρήση της συνάρτησης ενεργοποιήσης softmax όπου μετατρέπει τις αρχικές εξόδους σε πιθανότητες έτσι ώστε το άθροισμα να είναι ίσο με 1.

1.TRAINING PARAMETERS:

Number of neurons per layer1:256(0.95),128(acc=0.93),**64(acc=0.98)**

Number of neurons per layer2:128(0.95),64(acc=0.93),**32(acc=0.98)**

Optimizer: sgd

Loss Function:Sparse categorical cross entropy

Epochs:50

Batch size:64

Validation split:20%

Deeper NN Accuracy: 0.98

2.TRAINING PARAMETERS:

Number of neurons per layer1:**256(acc=1.0),**128(acc=0.97),64(acc=0.98)

Number of neurons per layer2:**128(acc=1.0),**64(acc=0.97).32(acc=0.98)

Optimizer: sgd

Loss Function:Sparse categorical cross entropy

Epochs: 100

Batch size:16

Validation split:20%

Deeper NN Accuracy: 1.0

3.TRAINING PARAMETERS:

Number of neurons per layer1:**256(acc=1.0),**128(acc=1.0),64(acc=0.98)

Number of neurons per layer2:**128(acc=1.0),**64(acc=1.0),32(acc=0.98)

Optimizer: adam

Loss Function:Sparse categorical cross entropy

Epochs:100

Batch size:32

Validation split:20%

Deeper NN Accuracy: 1.0

4.PARAMETERS TUNING:

Μετά από τη δημιουργία 3 μοντέλων(baseline model, shallow model with one hidden layer, deeper model with two hidden layers) όπου για το κάθε ένα υπήρχαν διαφορετικοί υπερπαραμέτροι τώρα θα δούμε με τη χρήση του KerasClassifier και του GridSearchCV.

Αρχικά, δημιουργούμε το νευρωνικό δίκτυο χρησιμοποιώντας το KerasClassifier όπου έχει InputLayer 128 νευρώνες και συνάρτηση ενεργοποίησης Relu και γίνεται Dropout 30% των νευρώνων έπειτα προχωράμε στο hiddenlayer με 64 layers και γίνεται πάλι DropOut και τέλος έχουμε το επίπεδο εξόδου με 3 νευρώνες που αντιπροσωπεύουν τις 3 κλάσεις. Το μοντέλο αυτό έχει loss function: sparse_categorical_crossentropy metrics:accuracy όπου χρησιμοποιείται για την αξιολόγηση της απόδοσης κάθε συνδυασμού των παραμέτρων και optimizer: adam.

Το KerasClassifier είναι ένας προσαρμογέας ο οποίος επιτρέπει την χρήση μοντέλων συνδυαστικά με το GridSearchCV.

Το GridSearchCV είναι ένα εργαλείο που εκτελεί εξαντλητική αναζήτηση και δοκιμάζει κάθε συνδυασμό των υπερπαραμέτρων. Ορίζει πλέγμα υπερπαραμέτρων που θα δοκιμαστούν κατά τη διάρκεια της αναζήτησης και δοκιμάζει κάθε συνδυασμό τους χρησιμοποιώντας 3-fold cross validation.

Έπειτα η μέθοδος fit εκπαιδεύει το μοντέλο για κάθε συνδυασμό υπερπαραμέτρων στο σύνολο των δεδομένων εκπαίδευσης(X_train, y_train) και υπολογίζει στο τέλος το σκορ επικύρωσης για τον κάθε συνδυασμό ξεχωριστά.

param_grid = { 'batch_size': [16, 32, 64],

'epochs': [50, 100],

'model__optimizer': ['adam', 'sgd'] }

Τι είναι όμως το κάθε ένα?

Batch_size: είναι ο αριθμός των δειγμάτων που θέτονται σε επεξεργασία πριν από την ενημέρωση των βαρών του μοντέλου και είναι συνήθως δύναμη του 2.

Epochs: είναι ο αριθμός των επαναλήψεων εκπαίδευσης πάνω από σύνολο δεδομένων(50 or 100)

Model_optimizer: είναι ένας βελτιστοποιητής που χρησιμοποιείται για εκπαίδευση είτε είναι adam/sgd

RESULT: ΕΠΙΣΤΡΕΦΕΙ ΤΙΣ ΚΑΛΥΤΕΡΕΣ ΠΑΡΑΜΕΤΡΟΥΣ ΠΟΥ ΔΙΝΟΥΝ ΤΗΝ ΠΙΟ ΨΗΛΗ ΑΠΟΔΟΣΗ ΣΥΝΔΥΑΣΤΙΚΑ

BEST PARAMETERS:

Number of neurons per layer:

LAYER1:128 / LAYER2:64

BATCHSIZE	16
EPOCHS	50
MODEL OPTIMIZER	ADAM

ΑΝΑΛΥΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ:

Batch_size=16→small batch size means the model updates its weights more frequently during training

Epochs=50→ we use smaller size to avoid overfitting

Model_optimizer=adam → adjusts learning rate and have better performance

BATCH SIZE	EPOCHS	MODEL OPTIMIZER	VALIDATION ACCURACY
16	50	ADAM	1.0
32	100	ADAM	1.0
16	100	ADAM	0.998333
32	100	SGD	0.993333
64	100	ADAM	0.986667
16	100	SGD	0.985000
32	50	ADAM	0.980000
16	50	SGD	0.973333
64	100	SGD	0.973333
64	50	ADAM	0.955000
32	50	ADAM	0.925000
64	50	SGD	0.916667

Για να γίνουν όλες αυτές οι δοκιμές του model function χρειάστηκε το μοντέλο 3 λεπτά για να ολοκληρώσει την εκπαίδευση και τις διεργασίες έτσι ώστε να εξάγει τα αποτελέσματα των υπερπαραμέτρων

NUMBER OF NEURONS PER LAYER:

LAYER 1:256 / LAYER 2:128

BATCHSIZE	16
EPOCHS	50
MODEL OPTIMIZER	ADAM

ΑΝΑΛΥΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ:

Batch_size=16→small batch size means the model updates its weights more frequently during training

Epochs=50→ we use smaller size to avoid overfitting

Model_optimizer=adam → adjusts learning rate and have better performance

BATCH SIZE	EPOCHS	MODEL OPTIMIZER	VALIDATION ACCURACY
16	50	ADAM	0.99833
32	100	ADAM	0.99833
16	100	ADAM	0.998333
32	100	SGD	0.991667
64	100	ADAM	0.99833

16	100	SGD	0.99833
32	50	ADAM	0.99833
16	50	SGD	0.971667
64	100	SGD	0.973333
64	50	ADAM	0.98667
32	50	ADAM	0.99833
64	50	SGD	0.91333

Για να γίνουν όλες αυτές οι δοκιμές του model function χρειάστηκε το μοντέλο 4 λεπτά για να ολοκληρώσει την εκπαίδευση και τις διεργασίες έτσι ώστε να εξάγει τα αποτελέσματα των υπερπαραμέτρων

NUMBER OF NEURONS PER LAYER:

LAYER 1:64 / LAYER 2:32

BATCHSIZE	16
EPOCHS	100
MODEL OPTIMIZER	ADAM

ΑΝΑΛΥΣΗ ΤΩΝ ΑΠΟΤΕΛΕΣΜΑΤΩΝ:

Batch_size=16→small batch size means the model updates its weights more frequently during training

Epochs=50→ we use smaller size to avoid overfitting

Model_optimizer=adam → adjusts learning rate and have better performance

BATCH SIZE	EPOCHS	MODEL OPTIMIZER	VALIDATION ACCURACY
16	50	ADAM	0.98000
32	100	ADAM	0.98833
16	100	ADAM	0.99833
32	100	SGD	0.96833
64	100	ADAM	0.96667
16	100	SGD	0.98166
32	50	ADAM	0.92333
16	50	SGD	0.95500
64	100	SGD	0.91500
64	50	ADAM	0.87333
32	50	ADAM	0.92333
64	50	SGD	0.88166

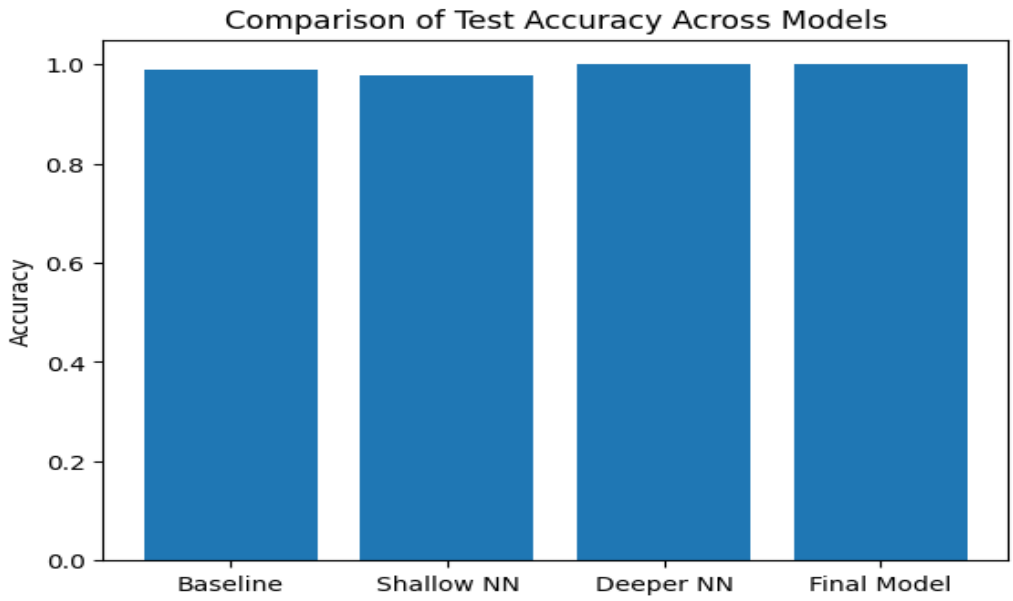
Για να γίνουν όλες αυτές οι δοκιμές του model function χρειάστηκε το μοντέλο 4λεπτά για να ολοκληρώσει την εκπαίδευση και τις διεργασίες έτσι ώστε να εξάγει τα αποτελέσματα των υπερπαραμέτρων

FINAL MODEL:

Our final model based on the above experiment has 2 hidden layers with number of neurons on layer 1 is 128 and on layer 2 is 64.

Training Parameters:

Epochs=50/Batch Size=16/Optimizer:Adam



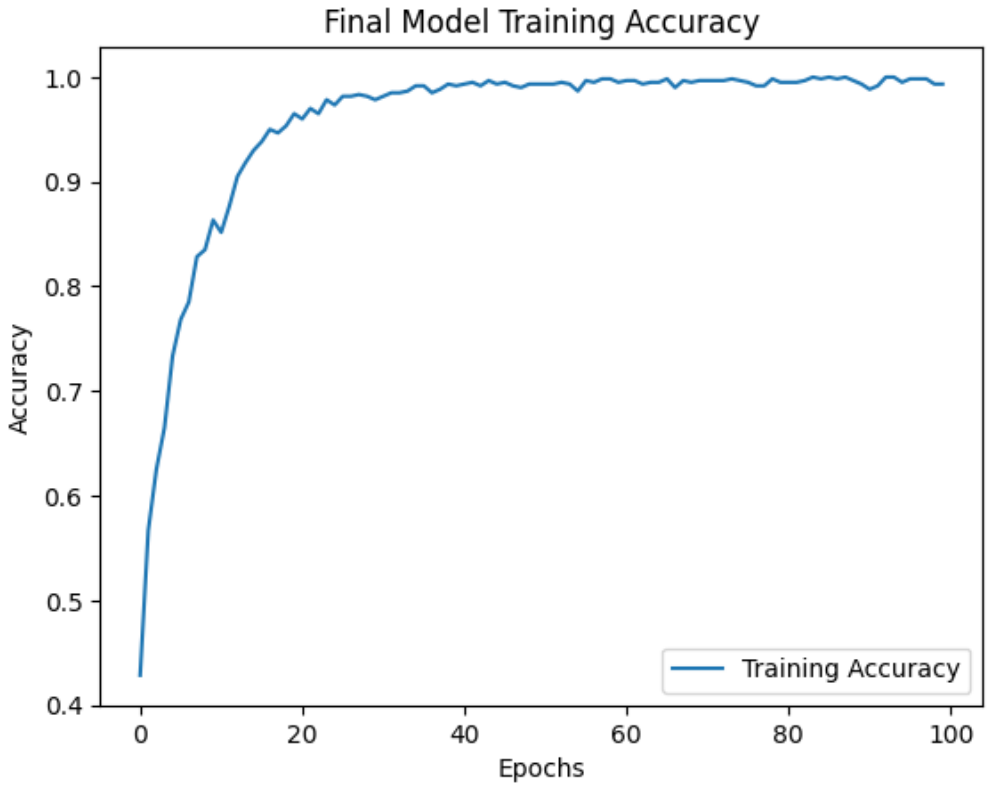
Σύγκριση ακρίβειας των διαφορετικών μοντέλων στα δεδομένα ελέγχου(test set):

Το baseline μοντέλο (λογιστική παλινδρόμηση-logistic regression) έχει υψηλή ακριβεια όμως τα βαθύτερα νευρωνικά δίκτυα (Deeper NN, Final Model) παρουσιάζουν μια μικρή αλλά σημαντική βελτίωση.

Η ακρίβεια είναι σχεδόν ίδια για το shallow neural network with one hidden layer και το deeper neural network with two hidden layers κάτι που δείχνει ότι η αύξηση της πολυπλοκότητας του μοντέλου δεν οδήγησε σε κάποια μεγάλη και συνάμα σημαντική αύξηση της απόδοσης.

Όλα τα μοντέλα που υλοποιούμε έχουν μικρή διαφορά στην ακρίβεια άρα συμπερασματικά υποδηλώνει ότι τα δεδομένα είναι πιθανώς πιο εύκολα για

ταξινόμηση.

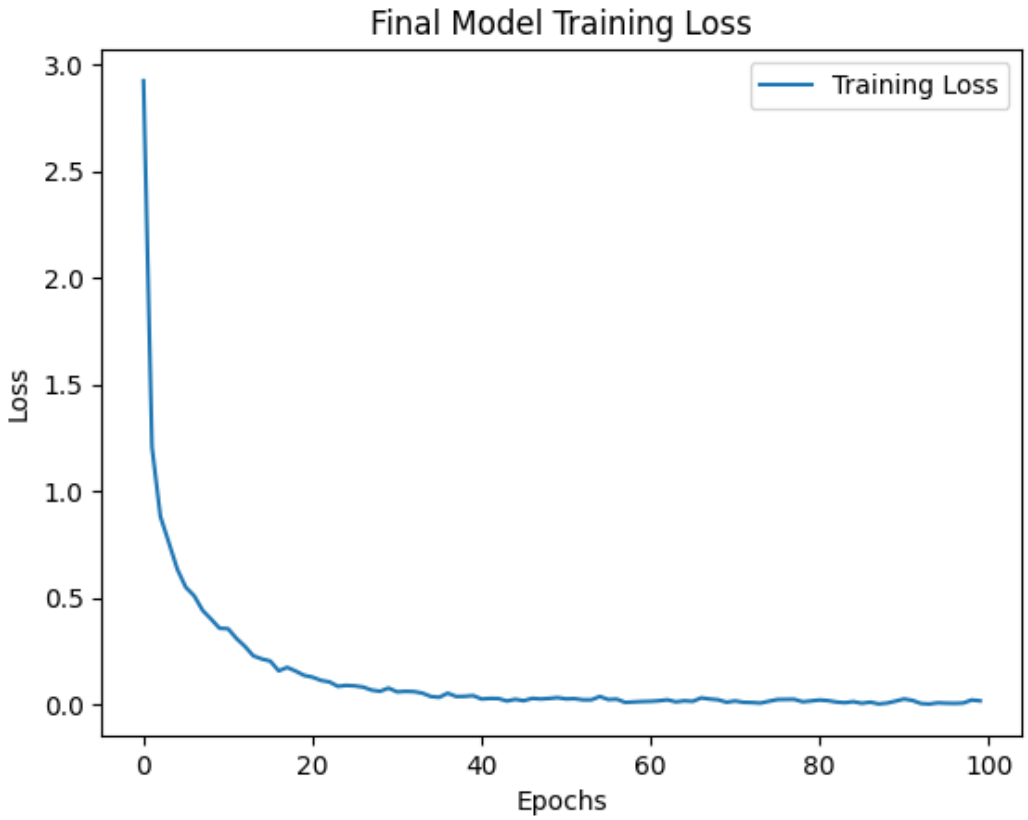


Η ακρίβεια (accuracy) κατά τη διάρκεια της εκπαίδευσης:

Στα πρώτα 20 epochs, το accuracy αυξάνεται γρήγορα ξεκινώντας από περίπου 40/100 και φτάνοντας σχεδόν στο 90/100

Μετά από περίπου 40-50 epochs accuracy δεν αυξάνεται ραγδαία και σταθεροποιείται κοντά στο 95-100/100 δείχνοντας έτσι ότι το μοντέλο προβλέπει σχεδόν όλα τα δείγματα σωστά στα δεδομένα εκπαίδευσης.

Άρα έχουμε σαν συμπέρασμα ότι το μοντέλο εκπαιδεύεται πολύ καλά, επιτυγχάνοντας πολύ υψηλή ακρίβεια. Παρόλο που η ακρίβεια στα εκπαίδευση δεδομένα είναι υψηλή θα αξιολογήσουμε και την απόδοση του τα δεδομένα ελέγχου έτσι ώστε να διασφαλιστεί ότι δεν υπάρχει κάποια υπερπροσαρμογή.



Η απώλεια (loss) κατά τη διάρκεια της εκπαίδευσης.

Στα πρώτα epochs η απώλεια μειώνεται γρήγορα γεγονός που δείχνει ότι το μοντέλο εκπαιδεύει γρήγορα τα δεδομένα.

Μετά από περίπου 40-50 epochs η καμπύλη αυτή σταθεροποιείται και έτσι το μοντέλο φτάνει κοντα στην maximum απόδοση και η απώλεια μένει κοντά στο 0 και έτσι το μοντέλο έχει ελάχιστο σφάλμα στις προβλέψεις για τα δεδομένα εκπαίδευσης.

TEST LOSS/TEST ACCURACY:

Για να κάνουμε plot το test loss+accuracy υλοποιούμε ένα προσαρμοσμένο callback TensorFlow/Keras όπου παρακολουθεί συνεχώς και καταγράφει την απώλεια και την ακρίβεια του τελικού μας μοντέλου στο test set μετά από κάθε epoch.

CALLBACK το ενσωματώνουμε στο training loop μέσα από το model.fit() επιτρέποντας έτσι την απόδοση του μοντέλου στο test dataset

test_data: Το test dataset (X_test και y_test) χρησιμοποιείται για την αξιολόγηση μετά από κάθε epoch.

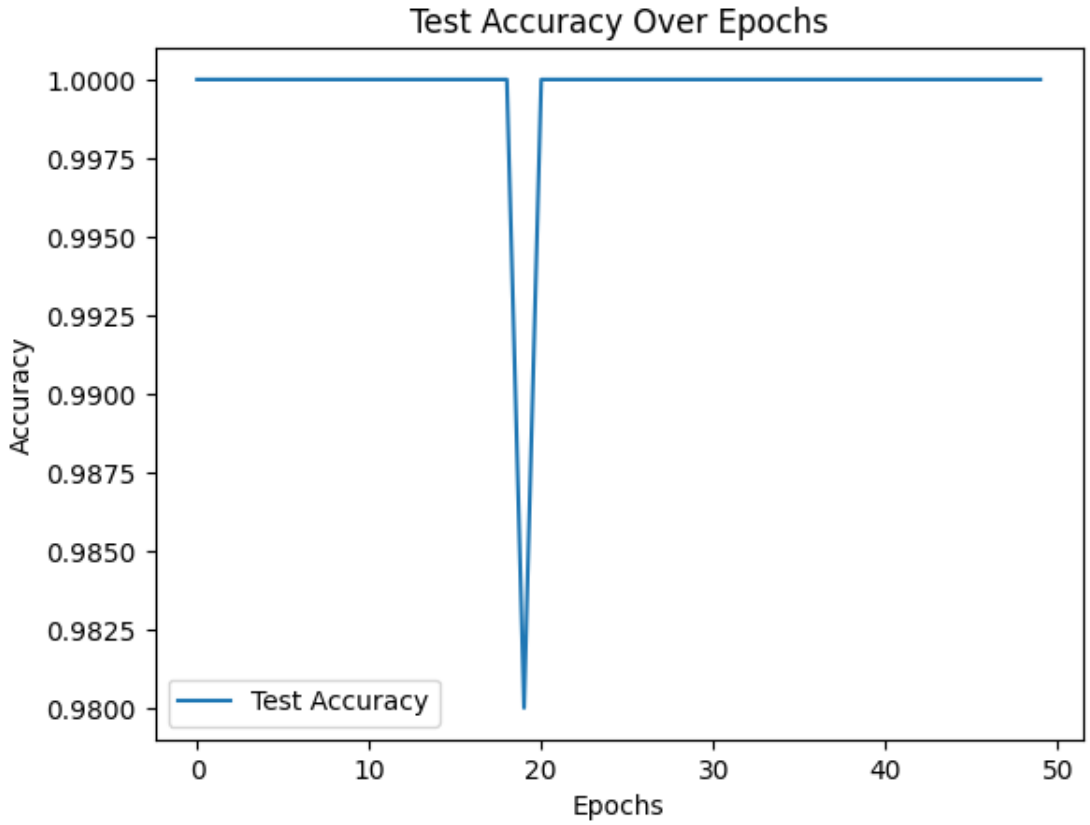
test_loss: λίστα που αποθηκεύει loss του test set για κάθε epoch.

test_accuracy: Μια λίστα που αποθηκεύει accuracy του test set για κάθε epoch

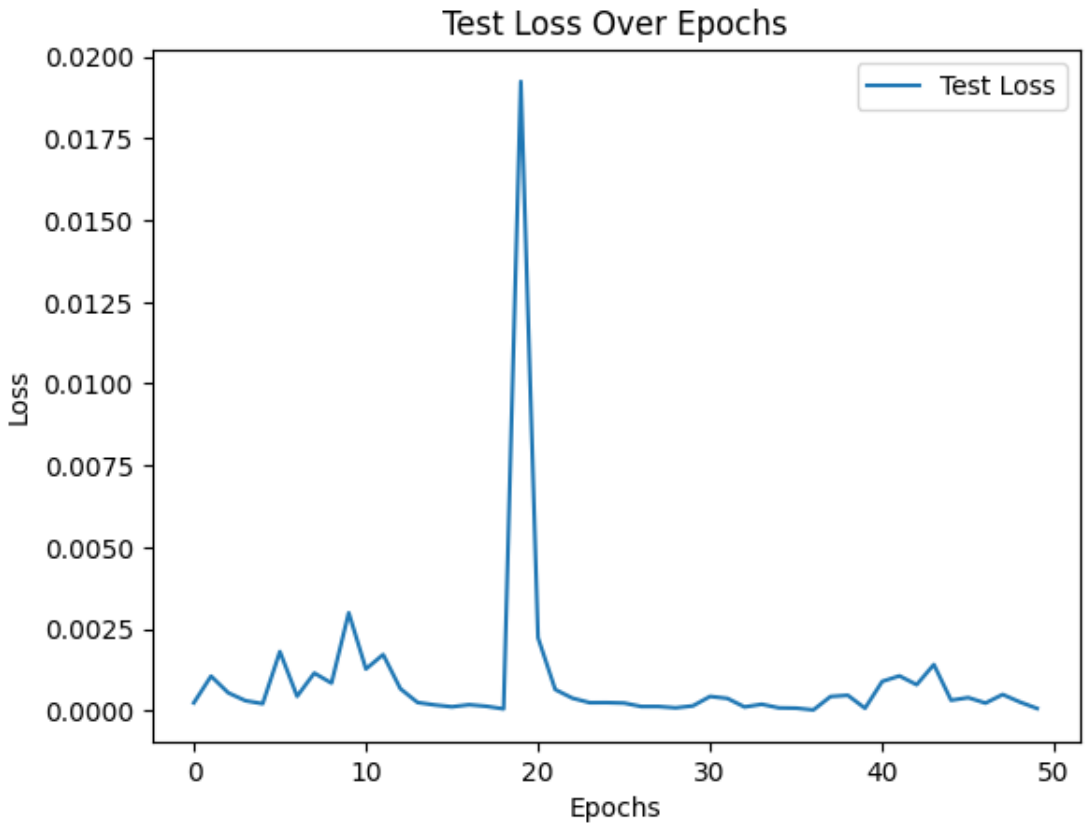
__init(self, test_data): αρχικοποίηση του callback=test dataset και συνάμα καλεί την μέθοδο super για πιο κατάλληλη αρχικοποίηση της βασικής μας κλάσης

on_epoch_end(self, epoch, logs=None): εκτελείται κάθε φορά που τελειώνει ένα epoch και χρησιμοποιεί τη self.model για να αξιολογήσει το test dataset και καταγράφει την απώλεια και την ακρίβεια

Άρα κατά την εκπαίδευση του μοντέλου, στο τέλος κάθε epoch καταγράφεται η απώλεια και η ακρίβεια και τις εμφανίζει στο terminal έτσι ώστε να μας την παρέχει σε πραγματικό χρόνο την απόδοση του test set και στο τέλος της εκπαίδευσης παίρνουμε τις μετρήσεις από τις λίστες και μπορούμε να δημιουργήσουμε τα γραφήματα



Η ακρίβεια στο test set ξεκινά σχεδόν στο 1 και παραμένει σταθερή και στα epoch=20 παρατηρείται μια πτώση στην ακρίβεια που δείχνει ότι υπάρχει αστάθεια στο μοντέλο στην αξιολόγηση στο συγκεκριμένο διάστημα για τα δεδομένα, έπειτα μετά την πτώση σιγά σιγά η ακρίβεια επανέρχεται στο 1 για τα υπόλοιπα, έτσι έχουμε σαν συμπέρασμα ότι η σχεδόν σταθερά ψηλή ακρίβεια δείχνει ότι το μοντέλο έχει καλή απόδοση στα συγκεκριμένα δεδομένα στο test set



Η απώλεια ξεκινά από χαμηλές τιμές και σε κάποιες φάσεις πριν τα 20 epochs παρουσιάζει μικρές διακυμάνσεις. Έπειτα στην epoch=20 παρατηρείται μια αύξηση απώλειας η οποία πιθανόν προκύπτει και από την πτώση της ακρίβειας που αναφέρθηκε στο πιο πάνω και αυτή η απότομη αύξηση πιθανά προκύπτει από κάποιες ανωμαλίες στο test set ή κάποια αστάθεια στα βάρη.

Μετά την απότομη αύξηση η απώλεια σταθεροποιείται χαμηλά και αποδεικνύει ότι το μοντέλο έμαθε αρκετά καλά την κατανομή των δεδομένων test set

ΠΑΡΑΔΕΙΓΜΑΤΑ ΟΡΘΗΣ ΚΑΙ ΕΣΦΑΛΜΕΝΗΣ ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗΣ:

1.ΟΡΘΗ ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗ:

Correct indices: περιέχει τα παραδείγματα που τα true labels(y_test) match y_pred.

Υπολογίζονται με τη χρήση np.where(y_test == y_pred)[0].

ΕΣΦΑΛΜΕΝΗ ΚΑΤΗΓΟΡΙΟΠΟΙΗΣΗ:

Incorrect indices: περιέχει παραδείγματα όπου τα true labels(y_test) dont match y_pred.

Υπολογίζονται με τη χρήση np.where(y_test != y_pred)[0]

Βλέπουμε ότι το μοντέλο απέτυχε σε κάποιες περιπτώσεις:

Αυτά μπορεί να προκύπτουν με 1) **Επικαλυπτόμενα χαρακτηριστικά**: Οι κατηγορίες που υπάρχουν μπορεί να έχουν παρόμοιες τιμές για ορισμένα χαρακτηριστικά έτσι μπορεί να υπάρχουν λανθασμένες κατηγοριοποιήσεις.

2)**Ανεπαρκή δεδομένα**: Δεν έγινε αρκετή εκπαίδευση του μοντέλου σε κάποιες αλλά πολύ λίγες κατηγορίες.

3)**Θόρυβο στα δεδομένα**: Υπάρχουν ίσως μη αντιπροσωπευτικά δειγμάτα στο dataset