Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Πανεπιστήμιο Πατρών

Εξόρυξη Δεδομένων και Αλγόριθμοι Μάθησης

Εργαστηριακή Άσκηση
Εαρινό Εξάμηνο 2020-21

Όνομα: Αριάδνη Μαχιά

Α.Μ.: 1059556

Έτος: 4ο

Email: up1059556@upnet.gr

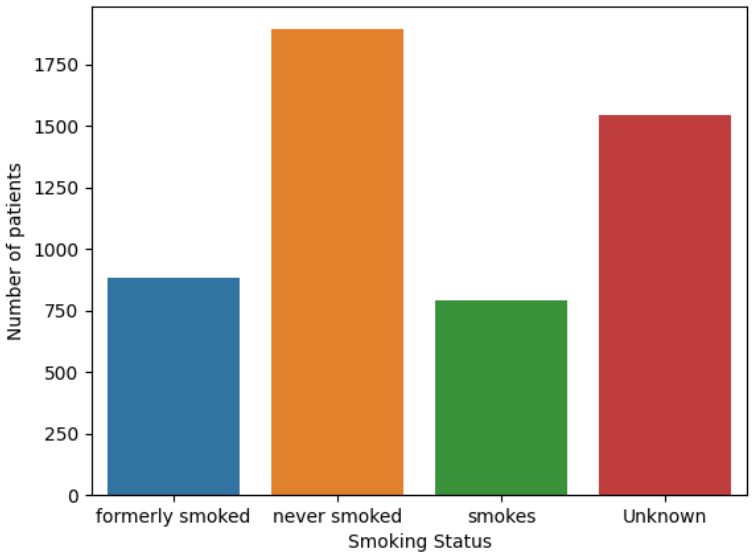
Ερώτημα 1

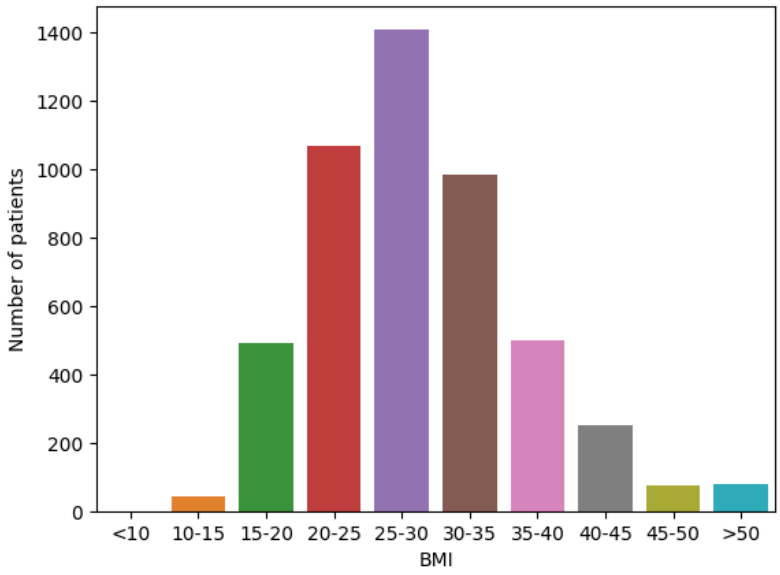
Α.

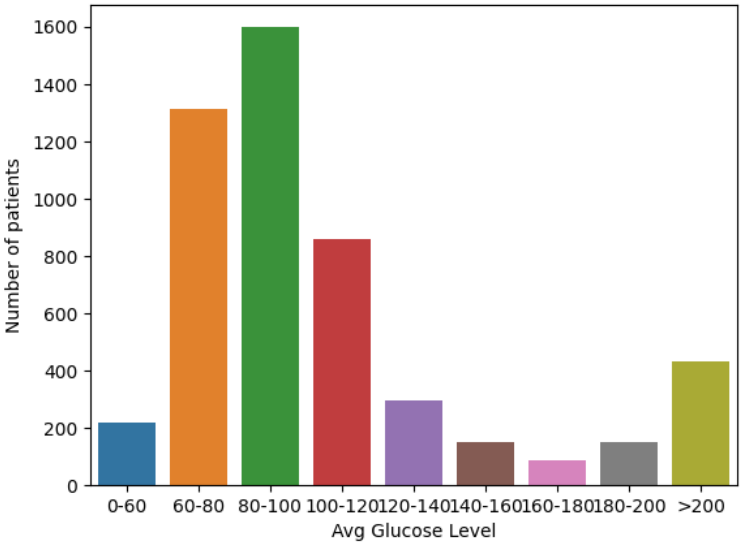
Το dataset περιέχει 5111 εγγραφές που κάθε μία έχει πληροφορίες για ένα άτομα. Πιο συγκεκριμένα, κάθε άτομα έχει ένα μοναδικό χαρακτηριστικό αριθμό (id) και πληροφορίες σχετικά με: το φύλο του (gender), την ηλικία του (age), εάν έχει υπέρταση (hypertension), εάν έχει κάποια καρδιακή πάθηση (heart_disease), εάν έχει παντρευτεί (ever_married), το είδος εργασίας του (work_type) με τις εξής κατηγορίες: εάν είναι παιδί (children), εάν δουλεύει στην κυβέρνηση (Govt_job), εάν είναι άνεργος (never_worked), εάν είναι ιδιωτικός υπάλληλος (private) ή εάν είναι αυτοαπασχολούμενος. Επιπλέον, αναγράφεται: ο τύπος κατοικίας (residence_type), όπου είναι αγροτικός ή αστικός, το μέσο επίπεδο γλυκόζης στο αίμα του (avg_glucose_level), ο δείκτης μάζας σώματος (bmi), η κατάσταση καπνίσματος του (smoking_status), όπου μπορεί να είναι: "στο παρελθόν κάπνιζε", "δεν κάπνισε ποτέ", "καπνίζει" ή "άγνωστο" και τέλος, εάν έχει πάθει εγκεφαλικό (stoke).

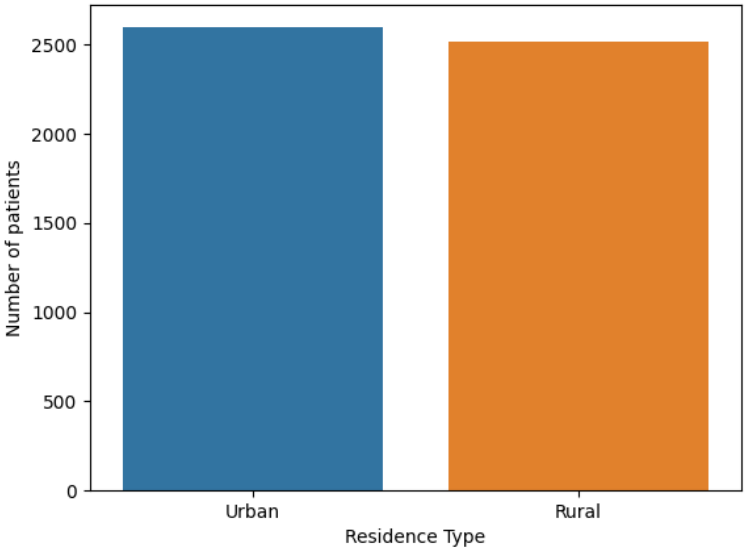
[Ο κώδικας για τα γραφήματα υπάρχει μέσα στο python αρχείο όμως μέσα σε σχόλια]

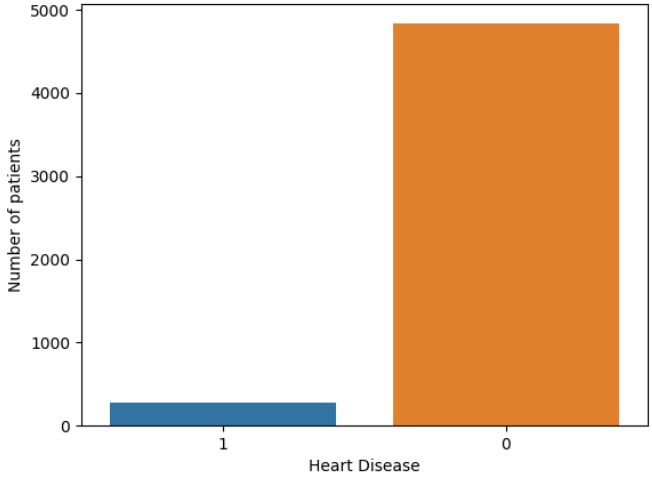
<u>Γραφήματα</u>:

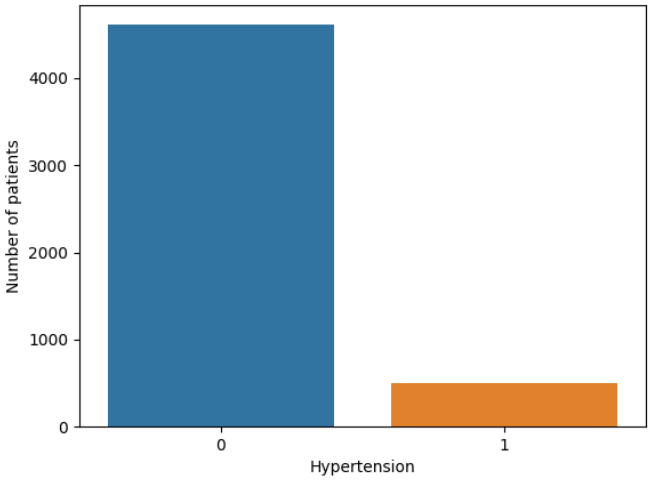


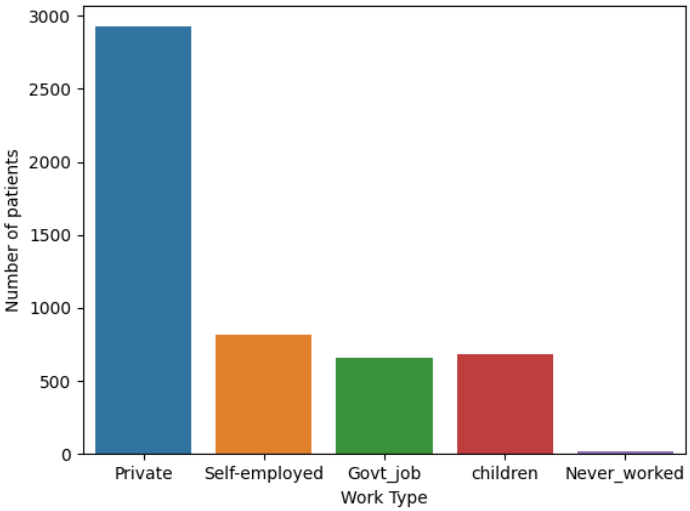


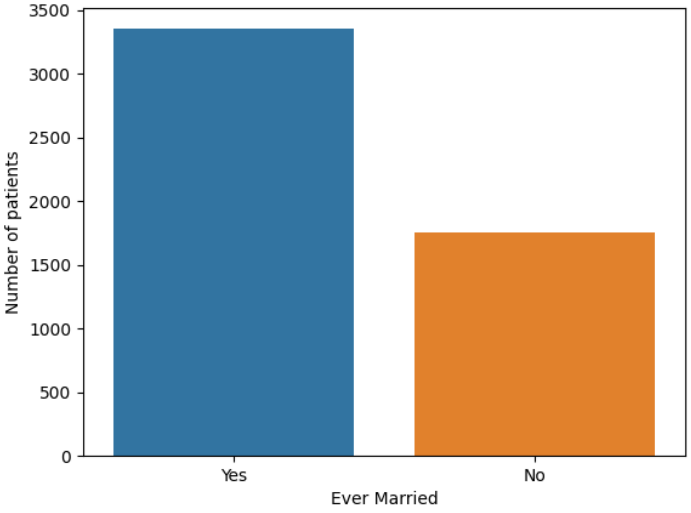

Number of patients
0
200
400
600
800
1000
1200
1400
1600
0-60
60-80
80-100
100-120
120-140
140-160
160-180
180-200
>200
Avg Glucose Level

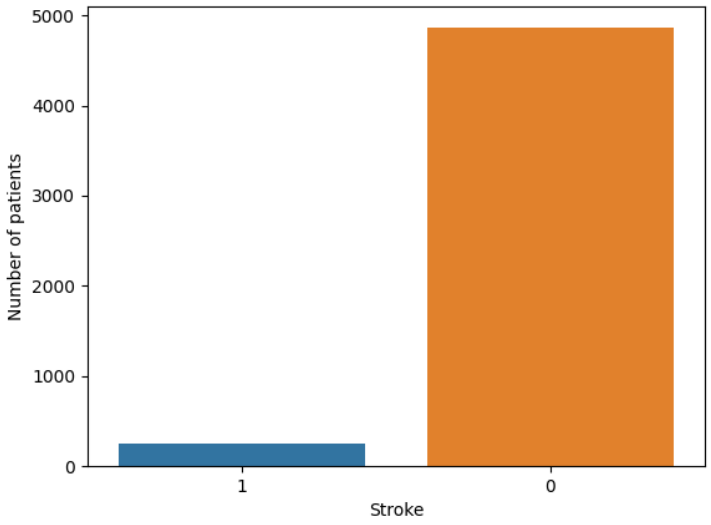

Number of patients
0
500
1000
1500
2000
2500
Urban
Rural
Residence Type

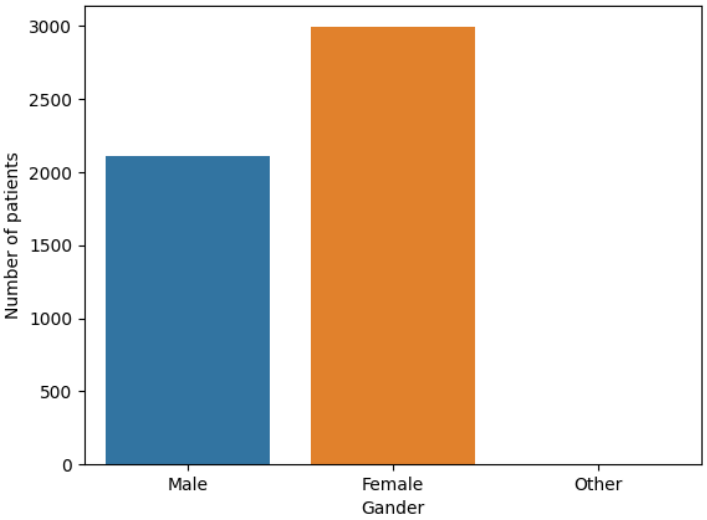

Number of patients
0
1000
2000
3000
4000
5000
1
0
Heart Disease

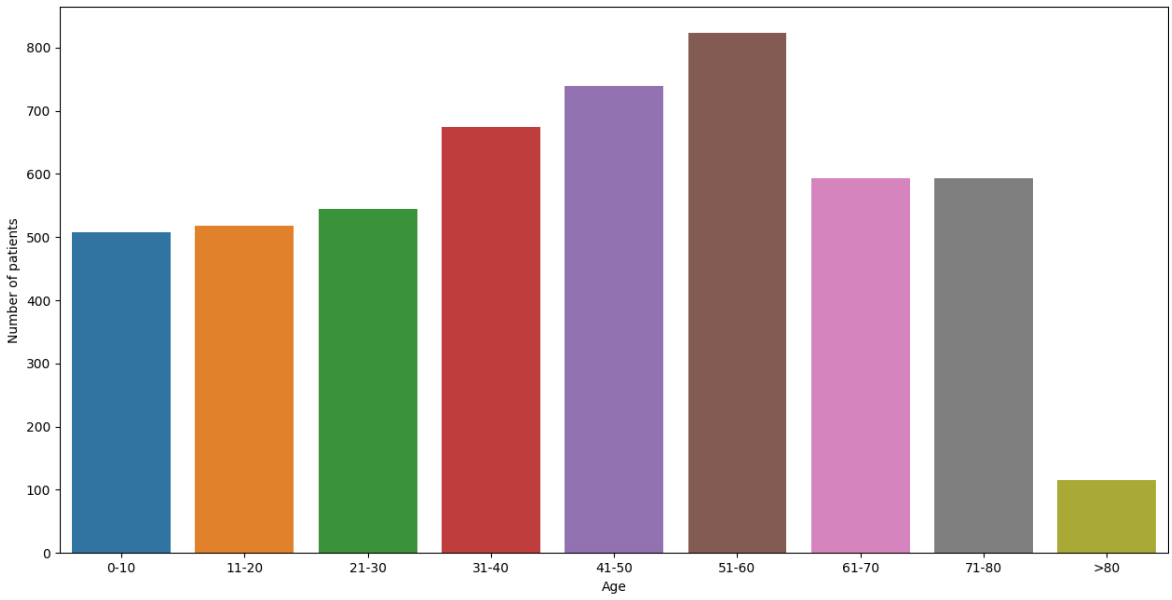

Number of patients
0
1000
2000
3000
4000
0
1
Hypertension


Number of patients
0
500
1000
1500
2000
2500
3000
Private
Self-employed
Govt_job
children
Never_worked
Work Type

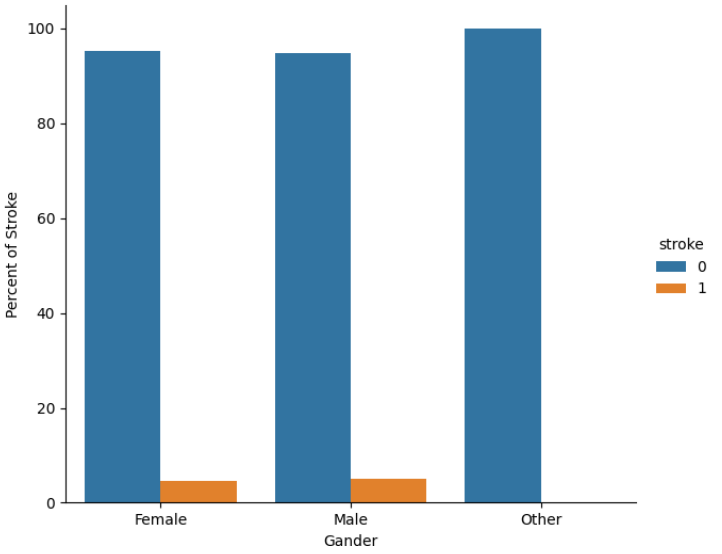

Number of patients
0
500
1000
1500
2000
2500
3000
3500
Yes
No
Ever Married

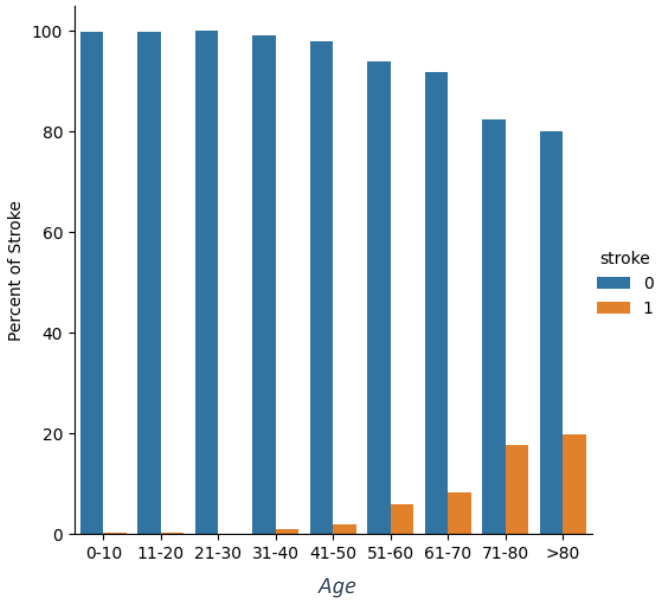

5000
4000
3000
2000
1000
0
Number of patients
1
0
Stroke

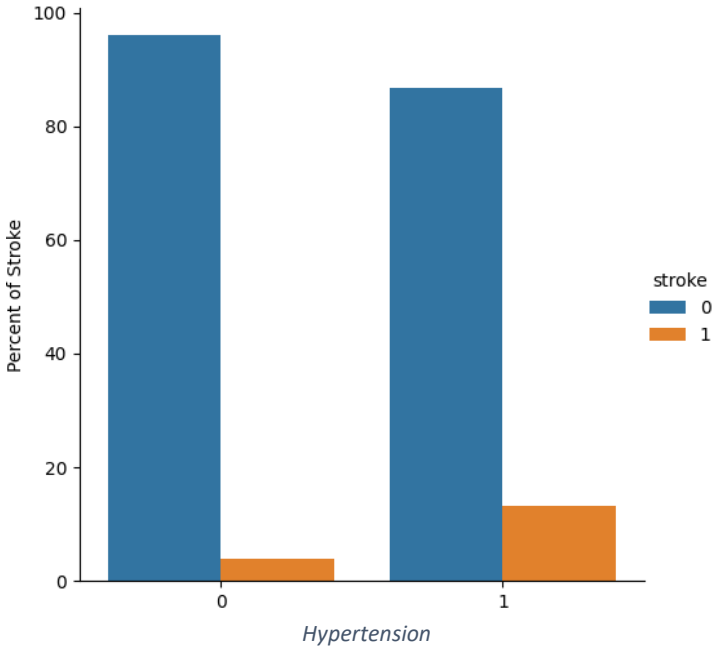

3000
2500
2000
1500
1000
500
0
Number of patients
Male
Female
Other
Gander

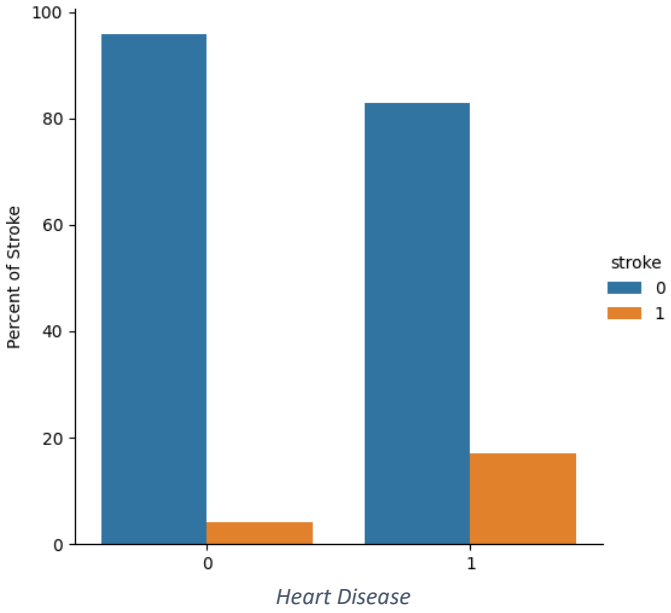

800
700
600
500
400
300
200
100
0
Number of patients
0-10
11-20
21-30
31-40
41-50
51-60
61-70
71-80
>80
Age

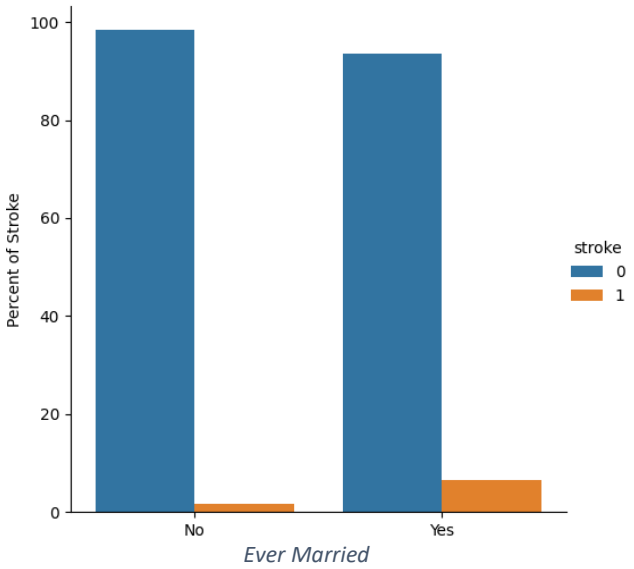
Γραφήματα σχετικά με την πιθανότητα εγκεφαλικού:



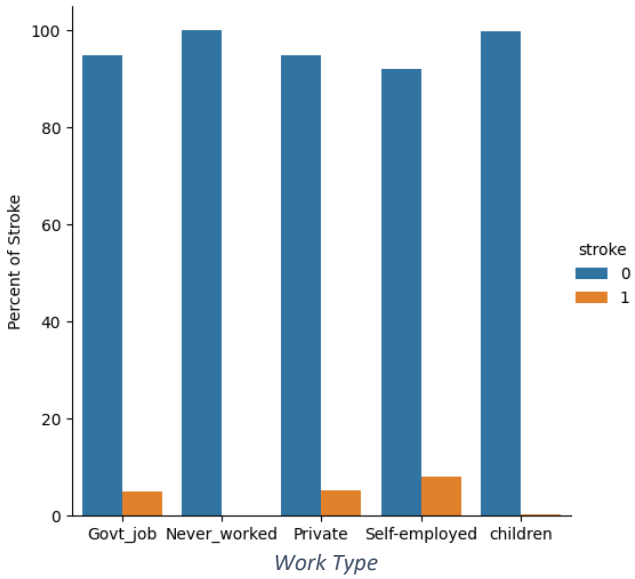




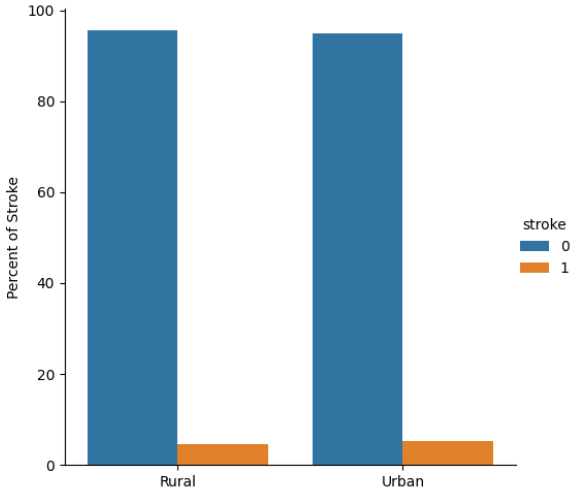




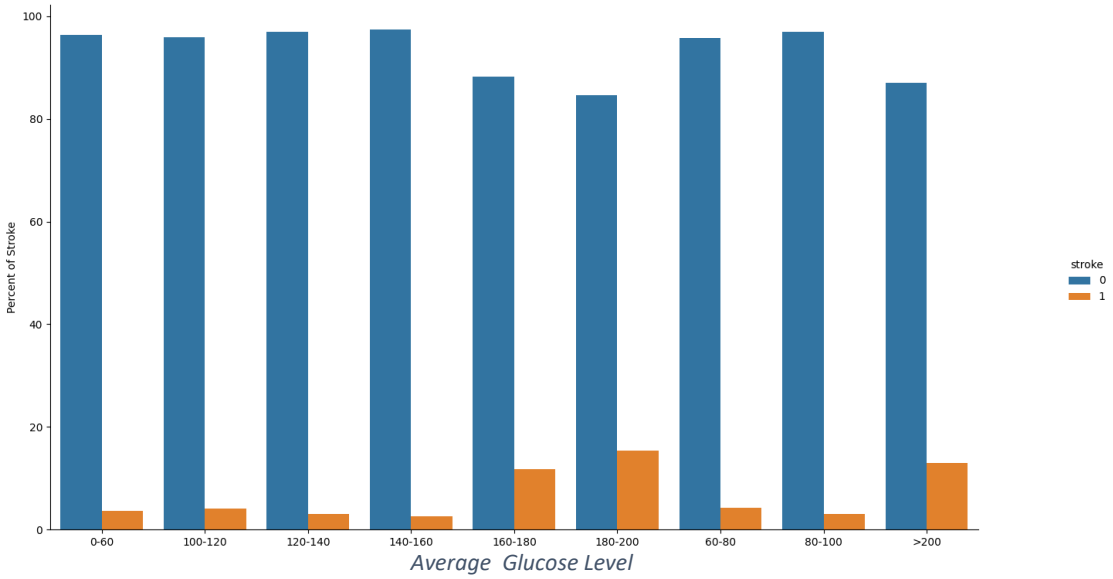
Ever Married



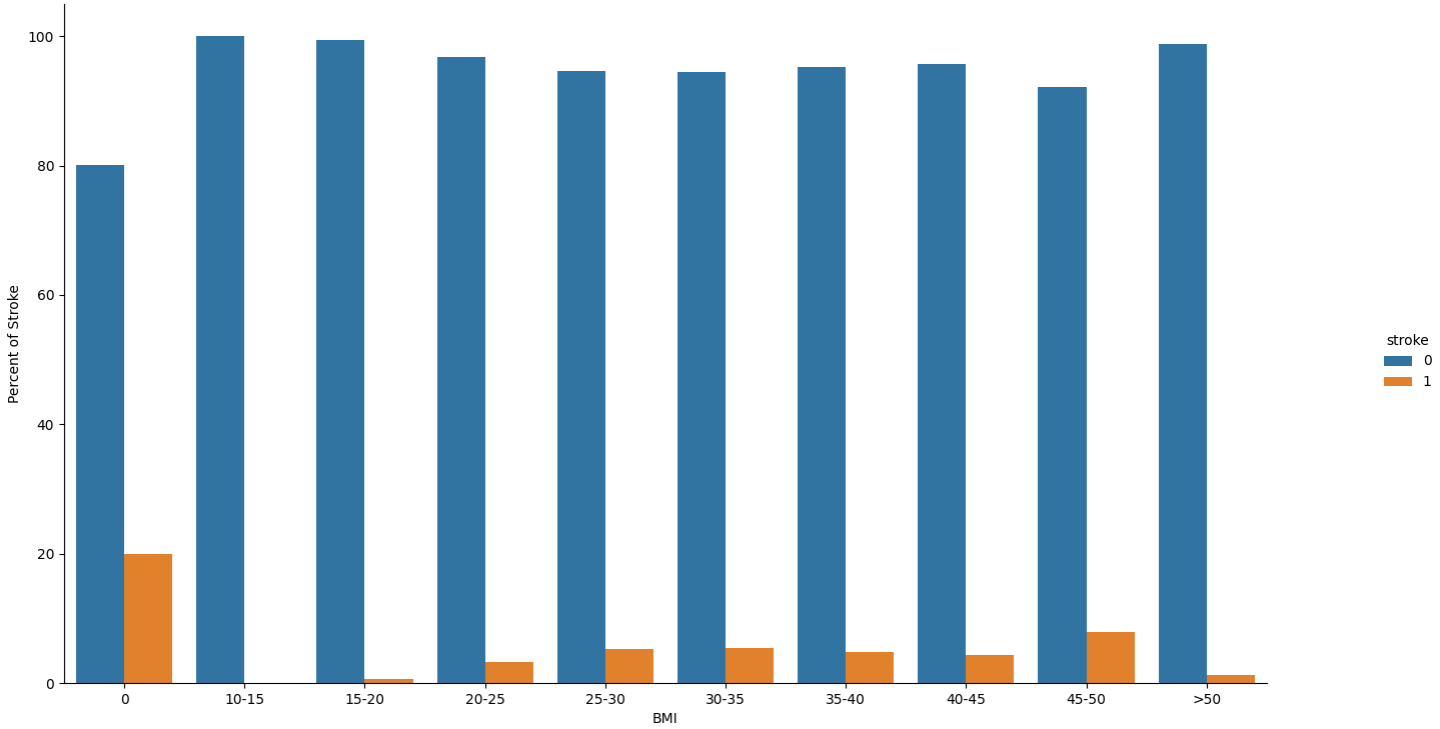
Work Type

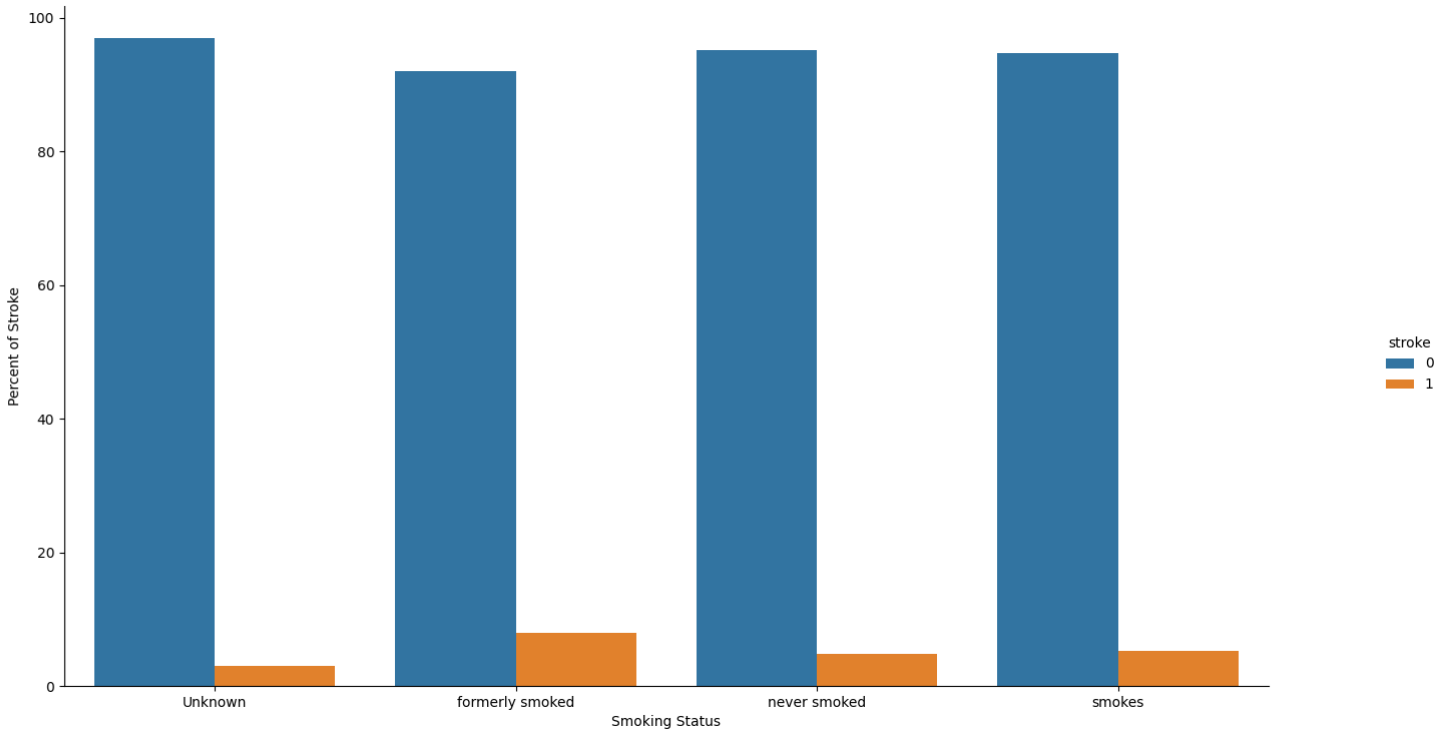


Residence Type



Average Glucose Level


Percent of Stroke
100
80
60
40
20
0
0
10-15
15-20
20-25
25-30
30-35
35-40
40-45
45-50
>50
BMI
stroke
0
1


Percent of Stroke
100
80
60
40
20
0
Unknown
formerly smoked
never smoked
smokes
Smoking Status
stroke
0
1

B.

Παίρνω ένα αντίγραφο του αρχικού dataframe, ώστε το αρχικό να μείνει άθικτο και να χρησιμοποιηθεί στα επόμενα ερωτήματα και κάνω κωδικοποίηση κατηγορικών μεταβλητών (gender, ever_married, work_type, Residence_type) για να μπορούμε να τα διαχειριστούμε. Όταν κωδικοποιούμε την μία στήλη με κατηγορικά έπειτα την διαγράφουμε, καθώς την έχουμε αντικαταστήσει με τις καινούργιες στήλες που προέκυψαν από την κωδικοποίηση, αλλά διαγράφουμε και μία από τις καινούργιες στήλες. Το τελευταίο το κάνουμε για να απόφύγουμε το πρόβλημα του “dummy variable trap”, το οποίο είναι ένα σενάριο που οι ανεξάρτητες μεταβλητές γίνονται πολυγραμμικές μετά την προσθήκη εικονικών μεταβλητών(“dummy variables”). Η πολυγραμμικότητα είναι ένα φαινόμενο στο οποίο δύο ή περισσότερες μεταβλητές συσχετίζονται σε μεγάλο βαθμό, δηλαδή η τιμή μιας μεταβλητής μπορεί να προβλεφθεί από τις τιμές άλλων μεταβλητών. Αυτό καταρρίπτει την υπόθεση της linear regression ότι οι παρατηρήσεις πρέπει να είναι ανεξάρτητες μεταξύ τους και αυτό ονομάζετε “dummy variable trap”. Αυτό έχει ως κίνδυνο την απώλεια ακρίβειας του regression model. Το καινούργιο dataframe με τις κωδικοποιήσεις dataframe το αντιγράφω σε άλλα dataframes που θα χρησιμοποιήσω στην συνέχεια για τα υπόλοιπα ερωτήματα.
Οι στήλες που περιέχουν ελλιπείς τιμές είναι οι: “bmi” και “smoking_status”.

1. Διαγράφω τις στήλες “bmi” και “smoking_status”, διότι περιέχουν ελλιπείς τιμές. Η στήλη “bmi” περιέχει κάποιες τιμές N/A, ενώ η “smoking_status” ορισμένες “Unknown”.
2. Για την στήλη bmi υπολογίζω τον μέσο όρο των όλων των τιμών της και αντικαθιστώ με αυτόν κάθε κελί της που έχει τιμή N/A. Από την άλλη για την στήλη smoking_status υπολογίζω πόσες φορές υπάρχει κάθε κατηγορία της ("formerly smoked", "never smoked", "smokes" και "Unknown") και αντικαθιστώ κάθε κελί της με την πιο συχνά εμφανιζόμενη κατηγορία από τις υπόλοιπες.
3. Για την στήλη bmi, κρατάω σε ένα dataframe όλες τις γραμμές που περιέχουν N/A τιμές στη bmi και έπειτα διαγράφω αυτές τις γραμμές από το αντίγραφο του αρχικού dataframe που χρησιμοποιώ για αυτό το ερώτημα. Στη συνέχεια κάνω linear regression και για την εκπαίδευση του μοντέλου χρησιμοποιώ ως X: όλες τις στήλες του καινούργιου dataframe εκτός της bmi και smoking_status και ως Y: την καινούργια στήλη bmi. Τέλος, κάνω την πρόβλεψη για το dataframe που έχει τις γραμμές που περιέχουν N/A τιμές και το αποτέλεσμα της πρόβλεψης το ενώνω με το αντίγραφο του αρχικού dataframe που χρησιμοποιώ για αυτό το ερώτημα. Έτσι, έχω το αρχικό dataframe, αλλά με προβλεπόμενες τιμές στα κελιά του bmi που είχαν αρχική τιμή N/A. Για την στήλη smoking_status, πρώτα την κωδικοποιώ ώστε κάθε κατηγορία που περιέχει να αντιστοιχεί σε έναν αριθμό (0 για “Unknown”, 1 για “formerly smoked”, 2 για “never smoked” και 3 για “smokes”). Έπειτα, κρατάω σε ένα dataframe όλες τις γραμμές που περιέχουν “Unknown” στη smoking_status και έπειτα διαγράφω αυτές τις γραμμές από το αντίγραφο του αρχικού dataframe που χρησιμοποιώ για αυτό το ερώτημα. Στη συνέχεια κάνω linear regression και για την εκπαίδευση του μοντέλου χρησιμοποιώ ως X: όλες τις στήλες του καινούργιου dataframe

εκτός της smoking_status και bmi, και ως Y: την καινούργια στήλη smoking_status. Τέλος, κάνω την πρόβλεψη για το dataframe που έχει τις γραμμές που περιέχουν “Unknown” και το αποτέλεσμα της πρόβλεψη το ενώνω με το αντίγραφο του αρχικού dataframe που χρησιμοποιώ για αυτό το ερώτημα. Έτσι, έχω το αρχικό dataframe, αλλά με προβλεπόμενες τιμές στα κελιά του smoking_status που είχαν αρχική κατηγορία “Unknown”.

4. Όσο αφορά την μέθοδο με τον KNN, έγραψα μια συνάρτηση(find_K) που δίνει ως όρισμα τα X και Y προς εκπαίδευση και επιστρέφει το καλύτερο k ανάμεσα από ένα διάστημα αριθμό που του δίνεις. Έτσι για την στήλη bmi φτιάχνω ένα KNN μοντέλο με k το καλύτερο αριθμό που επιστρέφει η συνάρτηση find_K. Το μοντέλο αυτό το εκπαιδεύω με τα ίδια X και Y που χρησιμοποίησα και στο υποερώτημα 3 για το bmi και ύστερα κάνω την πρόβλεψη για το dataframe που έχει τις γραμμές που περιέχουν N/A τιμές. Το αποτέλεσμα της πρόβλεψης το ενώνω με το αντίγραφο του αρχικού dataframe που χρησιμοποιώ για αυτό το ερώτημα και έτσι, έχω το αρχικό dataframe, αλλά με προβλεπόμενες τιμές στα κελιά του bmi που είχαν αρχική τιμή N/A. Για την στήλη smoking_status φτιάχνω ένα άλλο KNN μοντέλο με k το καλύτερο αριθμό που επιστρέφει η συνάρτηση find_K, με τα καινούργια ορίσματα. Το μοντέλο αυτό το εκπαιδεύω με τα ίδια X και Y που χρησιμοποίησα και στο υποερώτημα 3 για το smoking_status και ύστερα κάνω την πρόβλεψη για το dataframe που έχει τις γραμμές που περιέχουν “Unknown”. Το αποτέλεσμα της πρόβλεψη το ενώνω με το αντίγραφο του αρχικού dataframe που χρησιμοποιώ για αυτό το ερώτημα και έτσι, έχω το αρχικό dataframe, αλλά με προβλεπόμενες τιμές στα κελιά του smoking_status που είχαν αρχική κατηγορία “Unknown”.

Γ.

Για τα νέα μητρώα που προέκυψαν στο υποερώτημα Β, χρησιμοποιώντας Random Forest με dataset αναλογίας training 75%-25% test, επιχείρησα να προβλέψω εαν ένας ασθενής είναι επιρρεπής ή όχι να πάθει εγκεφαλικό. Η απόδοση του μοντέλου για κάθε dataframe του Β είναι τα παρακάτω:

Υποερώτημα του Β	F1 score	Precision	Recall
1	0.505244508409887	0.5803782505910166	0.5095784641068447
2	0.5071392863094246	0.669363707776905	0.5112479131886477
3 – BMI	0.4986215235792019	0.7217868338557993	0.5064343773091571
3 – Smoking Status	0.4990090816037633	0.5732914375490966	0.5055921170928184
4 – BMI	0.5112503470712426	0.7221350078492936	0.5128687546183142
4 – Smoking Status	0.4994014090334825	0.5983124018838304	0.5060056820225123

Precision: είναι μια μέτρηση που ποσοτικοποιεί τον αριθμό των σωστών θετικών προβλέψεων. Υπολογίζεται ως ο λόγος των σωστά προβλεπόμενων θετικών δειγμάτων δια του συνολικού αριθμού των θετικών δειγμάτων που είχαν προβλεφθεί. Η μεγιστοποίηση της precision ελαχιστοποιεί τα ψευδώς θετικά και η μεγιστοποίηση της recall ελαχιστοποιεί τα ψευδώς

αρνητικά. Το Precision απαντά στο εξής: Πόσοι από αυτούς που ονομάσαμε επιρρεπείς να πάθουν εγκεφαλικό είναι πραγματικά επιρρεπείς;

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives}$$

Recall: είναι μια μέτρηση που ποσοτικοποιεί τον αριθμό των σωστών θετικών προβλέψεων που έγιναν από όλες τις θετικές προβλέψεις που θα μπορούσαν να είχαν γίνει. Υπολογίζεται ως ο λόγος των σωστά προβλεπόμενων θετικών δειγμάτων δια του συνολικού αριθμού θετικών δειγμάτων που θα μπορούσαν να προβλεφθούν. Δεν ασχολείται με ψευδώς θετικά και ελαχιστοποιεί τα ψεύτικα αρνητικά. Το Recall απαντά στο εξής: Από όλους τους ανθρώπους που είναι επιρρεπείς να πάθουν εγκεφαλικό, πόσους από αυτούς που προβλέπουμε σωστά;

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives}$$

F1 Score: παρέχει έναν τρόπο συνδυασμού τόσο της precision όσο και της recall σε ένα μόνο μέτρο που συλλαμβάνει και τις δύο ιδιότητες. Είναι ο αρμονικός μέσος όρος των δύο κλασμάτων.

$$F1 - Score = \frac{2 \bullet Precision \bullet Recall}{Precision + Recall}$$

Στο F-score, οι δύο μετρικές είναι ισορροπημένες, δηλαδή μόνο μια καλή precision και μια καλή recall οδηγούν σε ένα καλό μέτρο F1. Με άλλα λόγια, η F1 δεν είναι τόσο υψηλή αν το ένα μέτρο βελτιωθεί σε βάρος του άλλου.

Στον παραπάνω πίνακα φαίνεται ότι και οι τρεις μετρικές είναι κοντά στο 0,5. Δηλαδή, πετύχαμε σωστά μόνο ένα 50% από αυτούς που προβλέψαμε επιρρεπείς να πάθουν εγκεφαλικό, να είναι όντως. Ένα τρόπος για να βελτιώσουμε τα αποτελέσματα θα ήταν να βρούμε ένα καλύτερο αριθμό για το πλήθος δέντρων αποφάσεων στον Random Forest. Επίσης, για το KNN ένα ακόμα καλύτερο k από αυτό που διαλέξαμε να βελτίωνε την πρόβλεψή μας και την τιμή των μετρικών.

Περιβάλλον Υλοποίησης

Τα παραπάνω έχουν πραγματοποιηθεί σε python 3.7 στο PyCharm Community Edition 2020.2.1.

Οι βιβλιοθήκες που χρησιμοποιήθηκαν είναι: pandas, matplotlib.pyplot, numpy, seaborn, sklearn.linear_model, sklearn.preprocessing, sklearn.neighbors, sklearn.model_selection, sklearn.neighbors, sklearn.model_selection, sklearn.ensemble, sklearn.metrics και sklearn.