Predicting House Prices with Machine Learning

Ερευνητική Αναφορά

Μηχανική Μάθηση

Σκουρογιάννης Κωνσταντίνος MTN2115

Παλληκάρης Ηρακλής MTN2119

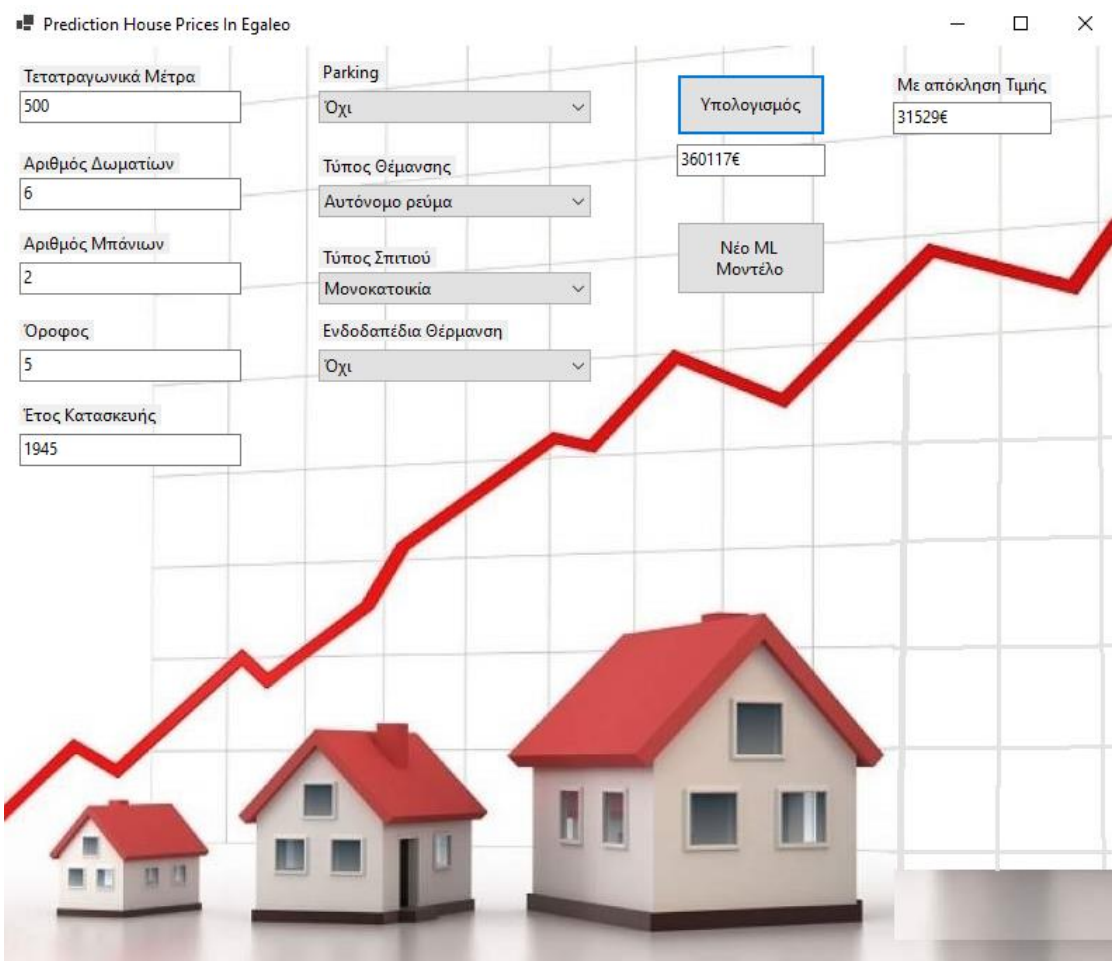
Εισαγωγή

Η εργασία

Σε αυτή την εργασία ο σκοπός μας είναι να αποκτήσουμε προσπαθήσουμε προβλέψουμε την τιμή μας κατοικίας στο Αιγάλεω, βάση των στοιχείων που θα δίνει ένας χρήστης, για την κατοικία και τα χαρακτηριστικά της. Συγκεκριμένα θα πρέπει να αποκτήσουμε δεδομένα σπιτιών, τα οποία θα τα περιεργαστούμε, ώστε να μπορέσουμε να φτιάξουμε ένα εύστοχο μοντέλο με αυτά. Το μοντέλο αυτό θα χρησιμοποιείται σε μια εφαρμογή όπου θα εισάγει τα στοιχεία ο χρήστης και θα παίρνει ως αποτέλεσμα την τιμή της κατοικία της οποίας τα στοιχεία εισήγαγε και το εύρος της απόστασης της τιμής αυτής από την πραγματικότητα.

Απαιτήσεις

Η εργασία είναι φτιαγμένη σε Python 3 αλλά για να την τρέξουμε θα χρειαστεί απλά να τρέξουμε το αρχείο ‘Ml UI.exe’. Για να τρέξει το exe η μόνη απαίτηση είναι να εγκαταστήσουμε το .NET Core τελευταίας έκδοσης. Στην περίπτωση των Linux ισχύει ακριβώς το ίδιο και μπορούμε έπειτα να τρέξουμε την εφαρμογή με το wine αφού έχουμε εγκατεστημένα τα προηγούμενα προγράμματα. Στην εφαρμογή βάζουμε μόνο νούμερα για κάθε πληροφορία που εισάγουμε, ενώ η ημερομηνία πρέπει να έχει απλά την χρονολογία όπου η κατοικία χτίστηκε (π.χ. 1998). Για να δουλέψει ο υπολογισμός πατάμε το κουμπί ‘Νέο ML Μοντέλο’ όπου θα κάνει train ένα μοντέλο βασισμένο στα δεδομένα του dataset που έρχεται με την εφαρμογή. Τέλος, μετά το συμπλήρωμα των πεδίων και τη δημιουργία μοντέλου ML, πατάμε το κουμπί ‘Υπολογισμός’ για να εκτυπωθούν οι δύο τιμές που χρειάζεται. Η τιμή στο πεδίο κάτω από το κουμπί είναι η τελική τιμή της κατοικίας με τα στοιχεία που συμπληρώσαμε, ενώ επάνω και δεξιά είναι η απόκλιση που μπορεί να έχει η τιμή αυτή από τον αρχικό υπολογισμό βάση του μέσου απόλυτου λάθους του μοντέλου.



Εικόνα 1: Παράδειγμα συμπληρωμένων πεδίων στην εφαρμογή και υπολογισμού τιμής

Τα δεδομένα

Πηγή

Τα δεδομένα για την εκπαίδευση του μοντέλου βρέθηκαν από την ιστοσελίδα spitogatos.gr, όπου εκεί χρήστες ανεβάζουν αγγελίες για κατοικίες που θέλουν να ενοικιάσουν ή να πουλήσουν. Εμείς χρησιμοποιήσαμε εκείνες που αφορούσαν τις πωλήσεις στην περιοχή του Αιγάλεου. Συνολικά οι εγγραφές ήταν 620 κατοικιών με 9 χαρακτηριστικά το κάθε ένα: Ευρώ ανά τετραγωνικό, τετραγωνικά, τύπος, θέρμανση, δωμάτια, μπάνια, όροφος, πάρκινγκ, έτος κατασκευής. Η τελική τιμή της κάθε κατοικίας έρχεται από το γινόμενο των τετραγωνικών με τα ευρώ ανά τετραγωνικά.

Teliki Timi	€/m^2	Tetragonika	typos	thermansi	domatia	mpania	orofos	parking	kataskeyi
66495	465	143	Διαμέρισμα	Αυτόνομη θέρμανση (Ρεύμα)	4	2	0	Οχι	1967
34000	493	69	Διαμέρισμα	Αυτόνομη θέρμανση (Ρεύμα)	1	1	0	Οχι	1983
34000	493	69	Διαμέρισμα	Αυτόνομη θέρμανση (Ρεύμα)	1	1	0	Οχι	1983
85000	500	170	Μονοκατοικία	Αυτόνομη θέρμανση (Ρεύμα)	4	1	0	Οχι	1964
120000	508	236	Κτίριο	Αυτόνομη θέρμανση (Ρεύμα)	4	1	0	Οχι	1963
170000	509	334	Μονοκατοικία	Αυτόνομη θέρμανση (Πετρέλαιο)	3	1	0	Οχι	1960
60000	545	110	Διαμέρισμα	Αυτόνομη θέρμανση (Ρεύμα)	1	1	0	Οχι	1980
60000	545	110	Διαμέρισμα	Αυτόνομη θέρμανση (Πετρέλαιο)	1	1	0	Οχι	1980
89000	556	160	Συγκρότημα διαμερισμάτων	Αυτόνομη θέρμανση (Ρεύμα)	4	2	0	Οχι	1958
89000	582	153	Συγκρότημα διαμερισμάτων	Αυτόνομη θέρμανση (Ρεύμα)	4	2	0	Οχι	1958

Εικόνα 2: Παράδειγμα εγγραφών που συλλέγουμε

RPA extraction

Για την εξαγωγή αληθινών δεδομένων πωλήσεων σπιτιών από την ιστοσελίδα χρησιμοποιήσαμε την τεχνολογία Robotic Process Automation (RPA). Ουσιαστικά με αυτήν την τεχνολογία μπορούμε να κάνουμε οποιαδήποτε ενέργεια κάνει ένας άνθρωπος σε μια ιστοσελίδα, καθώς το RPA αναγνωρίζει όλα elements είτε εφαρμογών είτε από web site και επιλέγει ποια θα διαβάσει ή θα αποθηκεύσει.

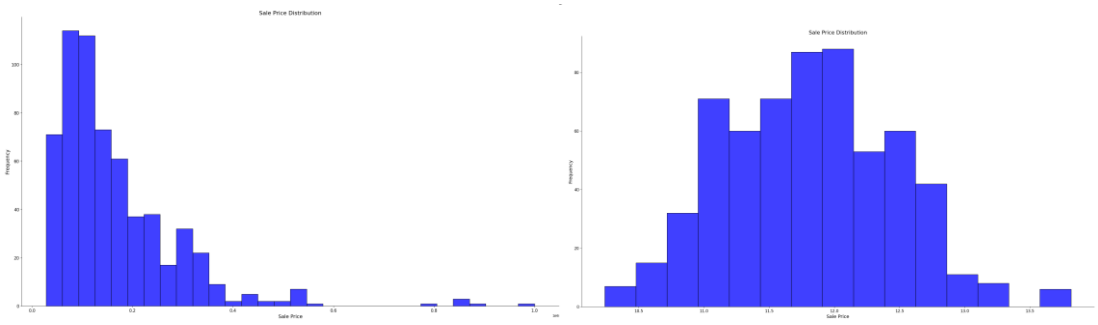
Πιο αναλυτικά βήματα του RPA για την εξαγωγή των δεδομένων:

1) Αρχικά του δώσαμε σαν Input το URL του spitogatos που έχει όλες τις αγγελίες για τις οποίες ενδιαφερόμαστε στη περιοχή Αιγάλεω.
2) Έπειτα σειριακά ελέγχει όλες τις επιλεγμένες εγγραφές, βάση φίλτρου περιοχής, ώστε να κρατήσει σε ένα collection όλα τα URL που εμφανίστηκαν από το προηγούμενο βήμα.
3) Μετά ανοίγει ένα προς ένα όλα τα URL που έχουμε από το βήμα 2. Για το κάθε URL που ανοίγει, κάνει εξαγωγή των ενδιαφερομένων τιμών για κάθε αγγελία όπου τα πεδία αναγνωρίζονται σαν web elements μέσω του HTML και άλλων αναγνωριστικών που αντιλαμβάνεται το bot για να κάνει εξαγωγή των σωστών πληροφοριών κάθε φορά. Εν τέλη έχει ένα νέο collection που έχει όλες τις πληροφορίες από όλες τις ενδιαφερόμενες αγγελίες.
4) Το τελευταίο βήμα σχετικά είναι να δημιουργήσει ένα excel και να περάσει όλες αυτές τις πληροφορίες σε αυτό όπου και θα είναι το data set του ML προγράμματος μας.

Επίσης μέσω του RPA έγινε κάθε εξεργασία του Excel/data set όπου αναφέρεται παρακάτω.

Καθαρισμός δεδομένων

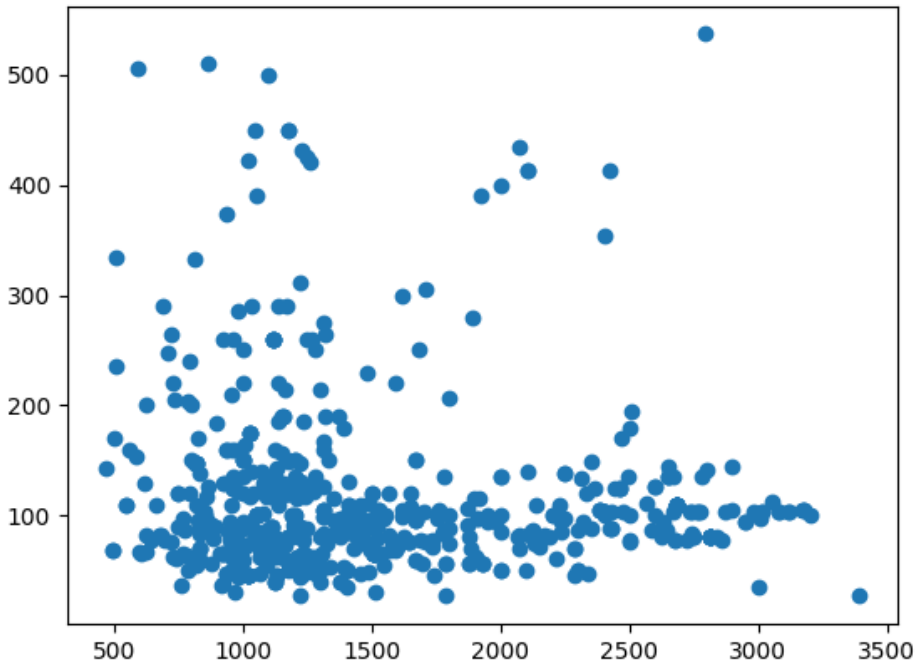
Καθώς τα δεδομένα μας προέρχονται από ιστοσελίδα στην οποία δεν γίνεται αναλυτικός έλεγχος για τις εγγραφές, σημαίνει πως θα χρειαστεί διερεύνηση για άκυρες τιμές, τιμές που είναι κενές ή εκείνες που είναι έξω από τα περιθώρια του πραγματικού (π.χ. μια κατοικία με 2000 τετραγωνικά και 2 δωμάτια).



Εικόνα 3: Ιστόγραμμα FinalPrice πριν και μετά τον μετασχηματισμό log

Αρχικά ελέγχουμε για διπλό-εγγραφές και τις σβήνουμε, ενώ μετά κοιτάμε για κενές τιμές. Ανάλογα την περίπτωση οι κενές τιμές μπορεί να σημαίνει κάτι διαφορετικό.

Για παράδειγμα αν το feature parking είναι κενό, αυτό θα σημαίνει πως δεν υπάρχει parking ενώ αν έχουμε στο κελί του ορόφου κενό σημαίνει πως δεν ξέρουμε σε ποιο όροφο βρίσκεται. Στην περίπτωση αυτή συμπληρώνουμε τις κενές τιμές του parking με όπου κενό = όχι και τις υπόλοιπες αναγκαστικά τις ακυρώνουμε καθώς δεν μπορούμε να ξέρουμε τι χαρακτηριστικό αντιστοιχεί σε εκείνη την θέση.



Εικόνα 4: Διάγραμμα των τετραγωνικών προς την τιμή ανά τετραγωνικό.

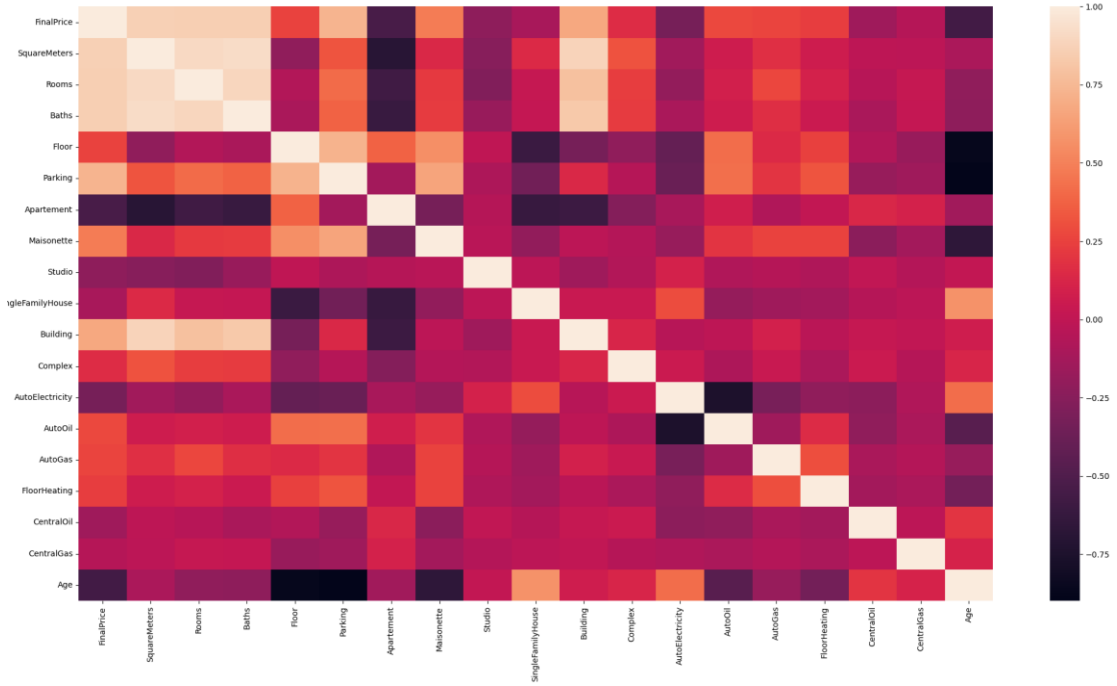
Η άλλη βασική κατηγορία είναι τα κατηγορικά δεδομένα, δηλαδή τα δεδομένα που περιγράφουν κάποια ιδιότητα με κείμενο αντί για νούμερα, όπως το είδος της θέρμανσης που μπορεί να είναι ενδοδαπέδια ή αυτόνομη θέρμανση ή και τα δύο ταυτόχρονα. Αυτά τα δεδομένα μιας και είναι μόνο κείμενο ένας τρόπος να μπορέσουμε να τα θέσουμε, ώστε να μας δίνουν την κατάλληλη πληροφορία, είναι να εντάξουμε κάθε κατηγορία ως ξεχωριστό feature που θα είναι δυαδικό δηλαδή 0 ή 1. Έτσι έχουμε την αφαίρεση του feature θέρμανση και τύπος κατοικίας αλλά στη θέση του έρχονται features όπως αυτόνομη θέρμανση, ενδοδαπέδια, διαμέρισμα, κτήριο κλπ. Έτσι τα νέα features περιέχουν κάθε κατηγορία με σύνολο να φτάνουν στα 20.

Τέλος ελέγχουμε τα features για ψεύτικες τιμές όπως μεγάλος αριθμός δωματίων και μπάνιων σε σχέση με μικρό αριθμό τετραγωνικών ή το ανάποδο. Επίσης τεράστιες χρηματικές τιμές ανά τετραγωνικό δεν βοηθούν το μοντέλο μας οπότε τις αφαιρούμε (εφόσον είναι νούμερο το οποίο αναγνωρίζουμε πως είναι ψεύτικο όπως 1.000.000 ανά τετραγωνικό). Αφού μετατρέψουμε τις χρονολογίες σε χρόνια ύπαρξής της κατοικίας (π.χ. το 1962 σε 60 αφού 2022-1962 = 60) έχουμε τελειώσει με την επεξεργασία των δεδομένων μας.

FinalPrice	Eur/m2	SquareMeters	Rooms	Baths	Floor	Parking	Apartement	Maisonette	Studio	SingleFamilyHouse	Building	Complex	AutoElectricity	AutoOil	AutoGas	FloorHeating	CentralOil	CentralGas	Age
66495	465	143	4	2	0	0	1	0	0	0	0	0	1	0	0	0	0	0	55
34000	493	69	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	39
34000	493	69	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	39
85000	500	170	4	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	58
120000	508	236	4	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	59
170000	509	334	3	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	62
60000	545	110	1	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	42
60000	545	110	1	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	42
89000	556	160	4	2	0	0	0	0	0	0	0	1	1	0	0	0	0	0	64

Εικόνα 5: Το Dataset μετά την επεξεργασία

Επίσης είναι συχνά καλό να σχεδιάζουμε έναν πίνακα συσχέτισης για να μας δώσει μια ιδέα για τις σχέσεις που υπάρχουν στα δεδομένα μας:



Εικόνα 6: Heatmap του Pearson's correlations ανάμεσα στα features

Βλέπουμε πως σε γενικές γραμμές δεν είναι συσχετισμένα τα δεδομένα με διαφορά τα δωμάτια και τα μπάνια που είναι μεταξύ τους σχετιζόμενα όπως είναι λογικό αλλά έχοντας δεδομένα κατοικιών δεν γίνεται να αποφύγουμε αυτό το πρόβλημα, ούτε γίνεται να τα σβήσουμε οπότε απλά θα δουλέψουμε με αυτά.

Έτσι μένουμε με 611 εγγραφές.

Training και Demo

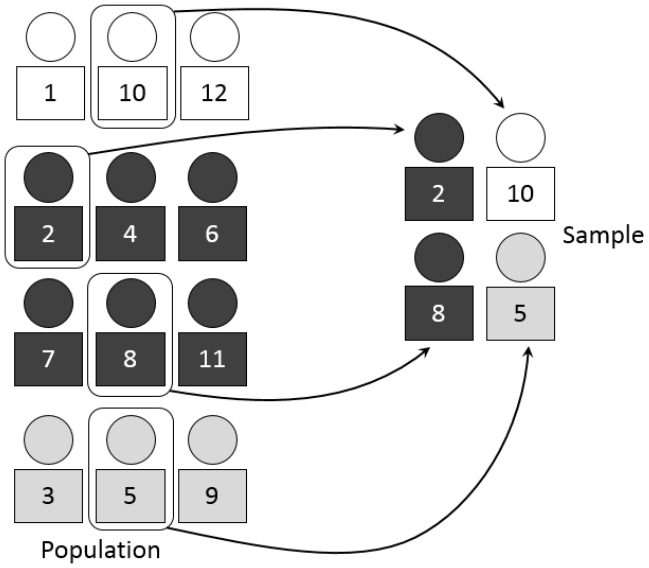
Stratified δειγματοληψία

Θέλουμε να χωρίσουμε το dataset σε 3 υπό – ομάδες. Το training, validation και test data set. Όταν παίρνουμε ένα δείγμα από έναν πληθυσμό, αυτό που θέλουμε να επιτύχουμε είναι ένα μικρότερο σύνολο δεδομένων που διατηρεί τις ίδιες στατιστικές πληροφορίες του πληθυσμού.

Ο καλύτερος τρόπος για να δημιουργήσουμε ένα αρκετά καλό δείγμα είναι να λαμβάνουμε ομοιόμορφα αρχεία πληθυσμού, αλλά αυτός ο τρόπος εργασίας δεν είναι άψογος. Στην πραγματικότητα, ενώ λειτουργεί αρκετά καλά κατά μέσο όρο,

εξακολουθεί να υπάρχει μια μικρή, πεπερασμένη πιθανότητα ένα μεμονωμένο δείγμα να είναι πολύ διαφορετικό από ολόκληρο τον πληθυσμό. Αυτή η πιθανότητα είναι πολύ μικρή, αλλά μπορεί να εισάγει μια προκατάληψη στο δείγμα μας που θα καταστρέψει την προγνωστική ισχύ οποιουδήποτε μοντέλου μηχανικής μάθησης που εκπαιδεύουμε σε αυτό.

Το πραγματικό θέμα είναι ότι δεν θέλουμε μια θεωρητικά σωστή μέθοδο που να λειτουργεί σε μεγάλους αριθμούς. θέλουμε να εξαγάγουμε ένα σωστό δείγμα με την υψηλότερη δυνατή στατιστική σημασία. Αυτό είναι το σημείο στο οποίο μια ομοιόμορφη δειγματοληψία δεν αρκεί πλέον και χρειαζόμαστε μια πιο ισχυρή προσέγγιση.



Εικόνα 7: Παράδειγμα Stratified Sampling

Έτσι με την Stratified δειγματοληψία θα χωρίσουμε τα δεδομένα μας ομοιόμορφα σε ομάδες («στράτες») και θα εξάγουμε από κάθε μια το 10% αυτών για validation, το 20% για test και το 70% για training. Αρχικά ταξινομούμαι τα δεδομένα μας με βάση τα ευρώ ανά τετραγωνικό και μετά τα χωρίζουμε σε δεκάδες από τις οποίες μια εγγραφή θα πηγαίνει στο validation set, 2 θα πηγαίνουν στο test και 7 θα πηγαίνουν στο train. Πήραμε τα δεδομένα και τα κόψαμε σε όσο δυνατόν μικρότερες ομάδες ώστε να έχουμε αντιπροσώπευση κάθε ομάδας τιμών.

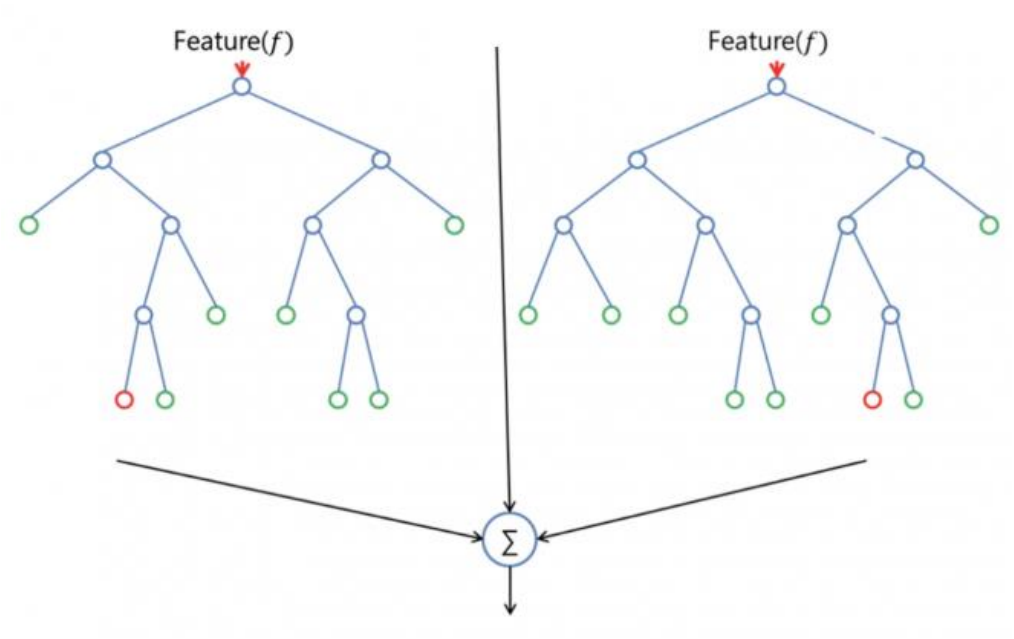
Model

Αρχικά ξεκινάμε την διαδικασία διαλέγοντας με ποιον αλγόριθμο θα φτιάξουμε το μοντέλο μας. Δοκιμάσαμε τους αλγορίθμους Linear Regression, KNN, Decision Tree και Random Forest. Λόγω του μικρού μεγέθους του dataset ο Random Forest μας έδωσε τα καλύτερα αποτελέσματα τόσο σε R2 όσο και σε RMSE και MAE.

Ο Random Forest είναι ένας εποπτευόμενος αλγόριθμος εκμάθησης. Το «δάσος» που χτίζει, είναι ένα σύνολο δέντρων απόφασης, που συνήθως εκπαιδεύονται με τη μέθοδο του «bagging». Η γενική ιδέα της μεθόδου bagging είναι ότι ένας συνδυασμός

μοντέλων εκμάθησης αυξάνει το συνολικό αποτέλεσμα. Με απλά λόγια: το τυχαίο δάσος δημιουργεί πολλαπλά δέντρα απόφασης και τα συγχωνεύει για να έχει μια πιο ακριβή και σταθερή πρόβλεψη.

Ένα μεγάλο πλεονέκτημα του τυχαίου δάσους είναι ότι μπορεί να χρησιμοποιηθεί τόσο για προβλήματα κατάταξης όσο και για προβλήματα παλινδρόμησης, τα οποία αποτελούν την πλειοψηφία των σημερινών συστημάτων μηχανικής μάθησης Παρακάτω μπορείτε να δείτε πώς θα ήταν ένα τυχαίο δάσος με δύο δέντρα:



Εικόνα 8: Random Forest με δύο δέντρα

Ο Random Forest προσθέτει επιπλέον τυχαιότητα στο μοντέλο, ενώ μεγαλώνει τα δέντρα. Αντί να αναζητά το πιο σημαντικό χαρακτηριστικό κατά τη διάσπαση ενός κόμβου, αναζητά το καλύτερο χαρακτηριστικό ανάμεσα σε ένα τυχαίο υποσύνολο χαρακτηριστικών. Αυτό έχει ως αποτέλεσμα μια μεγάλη ποικιλία που γενικά οδηγεί σε ένα καλύτερο μοντέλο. Επομένως, στο τυχαίο δάσος, μόνο ένα τυχαίο υποσύνολο χαρακτηριστικών λαμβάνεται υπόψη από τον αλγόριθμο για τη διαίρεση ενός κόμβου.

Παίρνοντας λοιπόν τις default υπέρ-παραμέτρους και αφαιρώντας το Final Price από τα features αφού αυτό θέλουμε να προβλέψουμε, ο αλγόριθμος δημιουργεί ένα μοντέλο το οποίο μετά θα βαθμολογεί. Το μοντέλο βαθμολογείται βάση του validation set και αν δεν έχει περάσει το αντίστοιχο κατώφλι στο MAE < 35000 και στο RMSE < 60000 τότε ξανά προσπαθεί έως ότου τα περνάει και τα 2 ταυτόχρονα.

Τέλος εξάγουμε δύο οντότητες από το script για να μεταφερθούν στο demo script. Οι δύο αυτές οντότητες είναι το price range όπου παίρνουμε έχοντας το μέσο απόλυτο error (MAE) και το μοντέλο μας.

```
Fitting 5 folds for each of 1 candidates, totalling 5 fits
Random Forrest
Val R2:          0.7225474194909681
Val RMSE:        74678.14956698424
Val MAE:         37919.072769971375
Fitting 5 folds for each of 1 candidates, totalling 5 fits
Random Forrest
Val R2:          0.7487744467628112
Val RMSE:        79564.99904010062
Val MAE:         39334.97499720862
Fitting 5 folds for each of 1 candidates, totalling 5 fits
Random Forrest
Val R2:          0.8642160814460307
Val RMSE:        54877.633043431626
Val MAE:         31876.842556596406
Test R2:         0.646246810039519
Test RMSE:       56636.974455298216
Test MAE:        29047.454998225814
```

Εικόνα 9: Παράδειγμα όπου τα πρώτα δύο μοντέλα δεν περνούν το κατώφλι του validation

Demo

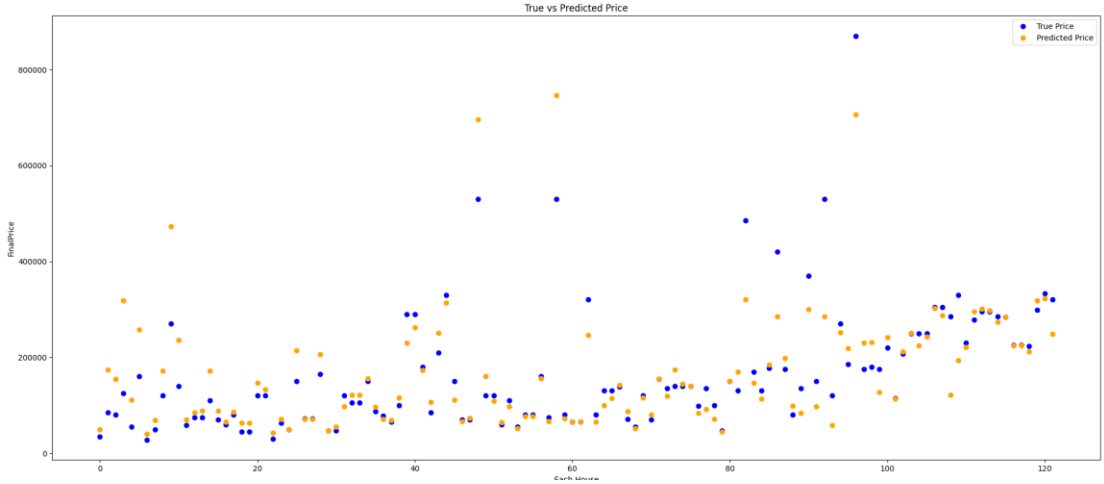
Στο script αυτό γίνεται η σύγκριση και η εξαγωγή αποτελεσμάτων. Συγκεκριμένα εισάγεται η λίστα με τις επιλογές του χρήστη και μετατρέπονται σε dataframe ώστε να συγκριθούν με τις τιμές του μοντέλου όπου και θα γίνει το prediction της τιμής. Αφού γίνει η πράξη αυτή η τιμή και το range της εξάγονται ώστε να εκτυπωθούν στην εφαρμογή.

Στο Front – End έχουμε ένα exe που όταν το τρέχουμε ανοίγει ένα παράθυρο στο οποίο θα συμπληρώσουμε τα στοιχεία του σπιτιού που θα δοκιμάσει το μοντέλο να κάνει predict. Οι τιμές αυτές μεταφέρονται στο demo.py όπου θα γίνει η παραπάνω διαδικασία και έπειτα θα εξαχθούν οι 2 τιμές που προαναφέρονται για να εκτυπωθούν στα κελιά της εφαρμογής. Εφόσον θέλουμε μπορούμε από το αντίστοιχο κουμπί να εκπαιδεύσουμε νέο μοντέλο και να βάλουμε νέα στοιχεία προς υπολογισμό.

Αποτελέσματα

Ο αλγόριθμος καταφέρνει να προβλέψει τις τιμές στα σπίτια αυτά με αρκετά καλή απόκλιση συνολικά. Όπως φαίνεται και στο σχήμα παρακάτω υπάρχει μια δυσκολία στο να προβλέψει τις τιμές που είναι πολύ μεγάλες, κάτι που είναι φυσιολογικό μιας και στο ήδη μικρό dataset που έχουμε δεν υπάρχουν πολλές τιμές τόσο μεγάλες ώστε να γίνει σωστή εξάσκηση. Επίσης στα μοντέλα που φτιάχνουμε παρατηρούμε πως το accuracy του validation set είναι μικρότερο από το accuracy του test set κάτι που είναι σημάδι για μοντέλο που δεν έχει κάνει overfit. Επίσης κατά την δοκιμή μας αν χρησιμοποιήσουμε δεδομένα από άλλα σπίτια στην περιοχή Αιγάλεω θα δούμε πως η τιμές που μας επιστρέφει το πρόγραμμα είναι εντός της προβλεπόμενης απόκλισης. Η απόκλιση είναι και αυτή στις αναμενόμενες τιμές καθώς στην ελληνική αγορά

σπιτιών βρισκόμαστε σε ένα στάδιο απότομων αυξήσεων των ακινήτων με αποτέλεσμα να είναι μεγάλη η απόκλιση τιμής ακόμα και στα σπίτια ίδιας περιοχής και ίδιων προδιαγραφών. Καθώς χρησιμοποιούμε, λοιπόν, πραγματικά δεδομένα, θα ήταν πολύ δύσκολο να μειωθεί σημαντικά η απόκλιση.



Εικόνα 10: Οι τιμές 122 σπιτιών με μπλε είναι οι πραγματικές τιμές και με πορτοκαλί είναι οι προβλεπόμενες

Random Forrest	
Val R2:	0.7731859495774637
Val RMSE:	48766.73396332703
Val MAE:	26333.448891491018
Test R2:	0.7766698917626285
Test RMSE:	59970.01623198737
Test MAE:	34201.92631993235

Εικόνα 11: Το μοντέλο του παραπάνω διαγράμματος

Σε μελλοντική ανάπτυξη τέτοιου συστήματος θα ήταν σημαντικό να συμπεριλαμβάνονται και δεδομένα από άλλες πλατφόρμες με επαλήθευση για ίδιες εγγραφές ώστε να επεκταθεί η λίστα πραγματικών δεδομένων, ενώ ταυτόχρονα θα μπορούσε να αλλάξει ο αλγόριθμος δημιουργίας μοντέλου εφόσον θα υπάρχουν περισσότερα δεδομένα για να έχουμε μικρότερες αποκλίσεις και πιο εύστοχες προβλέψεις στις ακραίες περιπτώσεις.