Relazione Home sales prices

Lorenzo Meroni
Studente del corso di Data Mining
875319
l.meroni16@campus.unimib.it

September 5, 2022

ABSTRACT

Il proposito della relazione è la previsione della variabile prezzo di vendita (*price*, in scala log10) di m = 4320 abitazioni del test set utilizzando per prevedere la variabile risposta 18 variabili esplicative. Dopo una breve sintesi del campione effettuata per mezzo della statistica descrittiva, si passerà pertanto alla ricerca dei modelli che diano i migliori risultati in termini previsivi, in particolare si tratterà di un *modello lineare*, *Randomforest* e *Xgboost*.

1 Analisi esplorativa

Il dataset da analizzare è stato precedentemente suddiviso in una sezione di train e una di test, il dataset di train ha 17293 osservazioni mentre quello di test come anticipato ne ha 4320.
Entrambi hanno informazioni complete su 18 variabili esplicative: *datesold, bedrooms, bathrooms, sqftliving, sqftlot, floors, waterfront, view, condition, sqftabove, sqftbasement, yrbuilt, yearrenovated, zipcode, latitude, longitude, nnsqftliving, nnsqftlot*. La variabile risposta è invece *price* ovvero il prezzo di vendita della casa. E' stata considerata la distribuzione della variabile risposta, rappresentata dall'istogramma che segue.

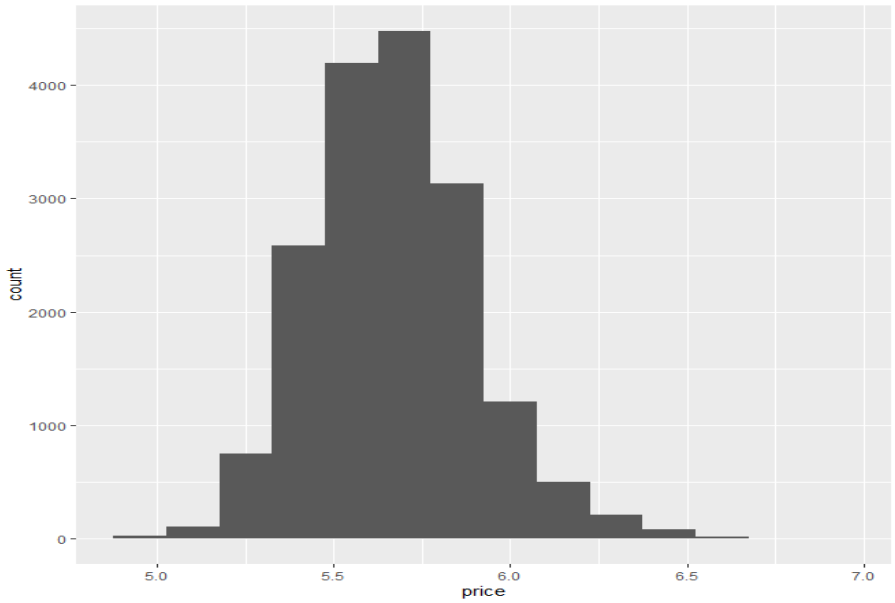
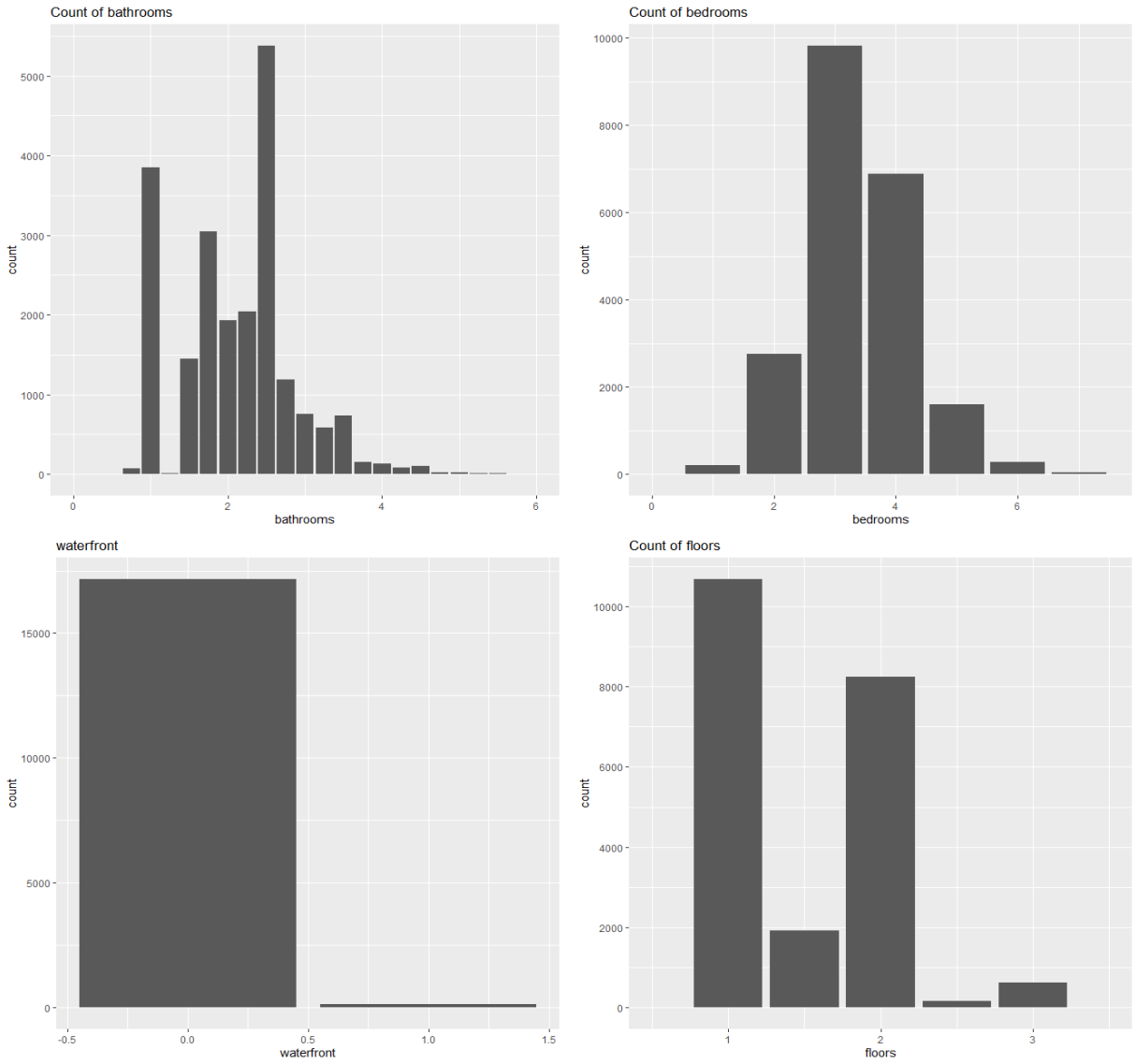


Figure 1: Distribuzione variabile risposta price

La trasformazione in *log10* della variabile risposta *price*, effettuata in un periodo antecedente alla consegna del data set ha raggiunto l'obiettivo sperato, abbiamo infatti una distribuzione approssimativamente normale e la concentrazione maggiore dei prezzi delle la ritroviamo circa tra 5.5 e 6.

1.1 Variabili discrete

Il nostro focus si sposta ora sulle variabili esplicative che ci serviranno per ottenere una previsione più vicina possibile alla realtà della variabile risposta *price*. In primo luogo la nostra analisi si focalizzerà sulla distribuzione e il range delle variabili discrete.



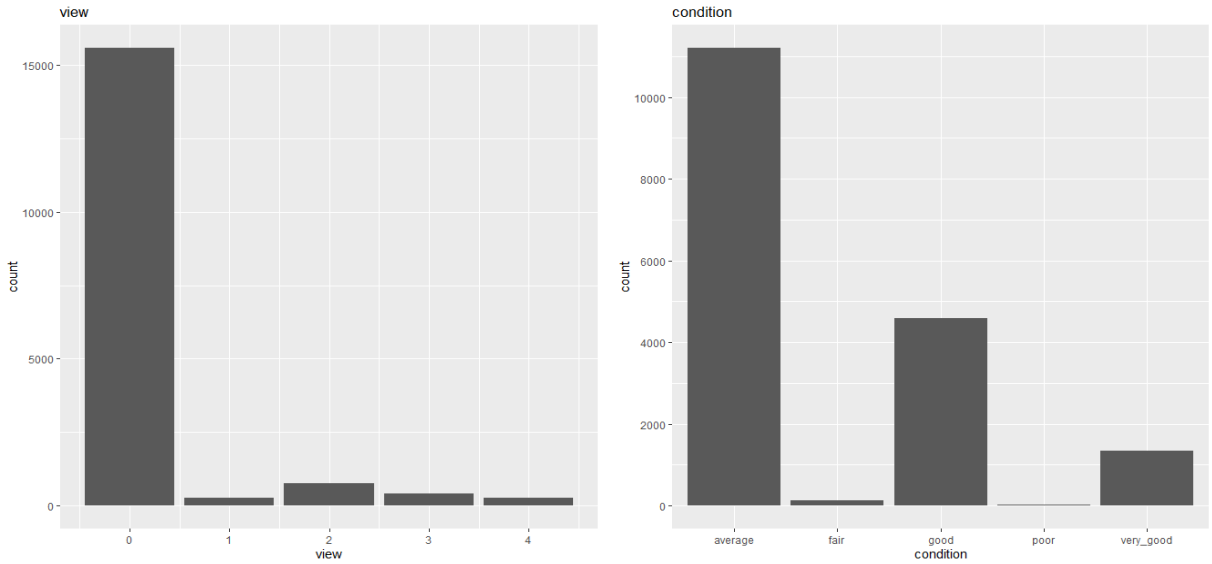


Figure 2: Distribuzione variabili esplicative discrete

Da una rapida analisi dei grafici proposti possiamo dedurre diverse informazioni.
La variabile *bedrooms* mostra una notevole asimmetria, le case hanno un numero di letti che si concentra tra i 2 e i 5 letti, così come la variabile *bedrooms* la variabile *bathrooms* mostra una notevole asimmetria verso destra, si concentra soprattutto tra i due e i tre bagni e tende poi a decrescere nonostante le case con un unico bagno siano molte. Vista la loro distribuzione le prime due variabili analizzate sembrano entrambe essere molto esplicative.
Per quanto riguarda la variabile *floors*, il numero di piani arriva sino a tre piani e per le case in esame la distribuzione premia le case ad un piano piuttosto che quelle con due o tre piani.
Analizzando la variabile *waterfront* quasi tutte le case non sono situate sulla costa quindi non fronte mare, siccome appunto la concentrazione di questa variabile non distingue molto la variabile è poco esplicativa.
Per la variabile *view*, quasi tutte le case hanno un rating della vista molto basso pari a 0 e la distribuzione pertanto si concentra esclusivamente nello 0, per la stessa motivazione spiegata per la variabile *waterfront* la variabile *view* è altrettanto poco esplicativa.
Infine soffermandoci sulla variabile *condition* notiamo che le case in condizione media sono la maggior parte delle osservazioni mentre quelle in buona sono minori e ancor minori in eccellente condizioni.

1.2 Relazione tra varibile risposta price e esplicativa yrbuilt

Ci soffermiamo temporaneamente sulla relazione tra la variabile *price* e la variabile *yrbuilt*, che per certi versi può essere molto interessante.

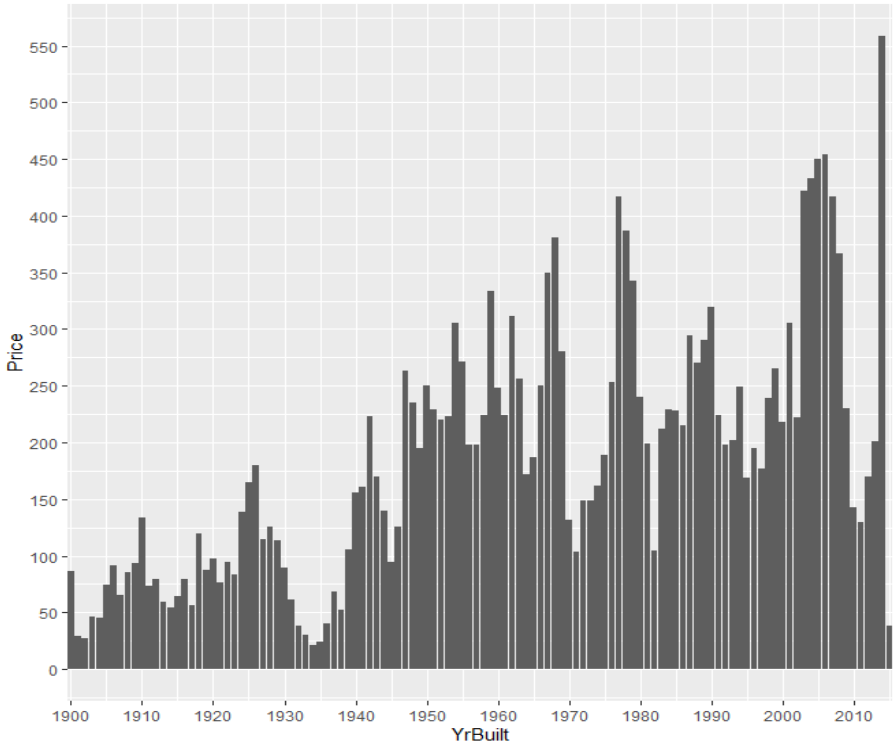
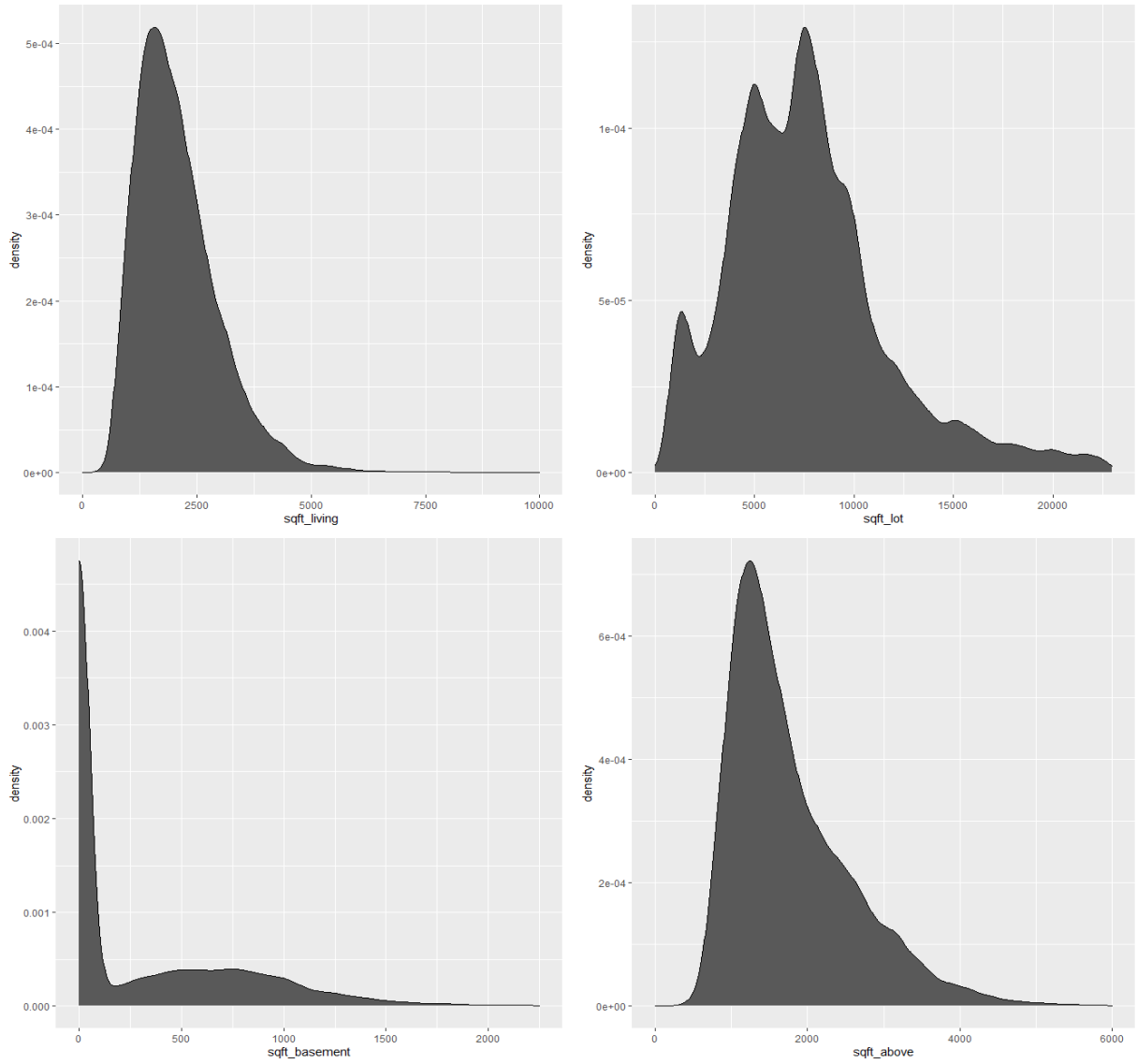


Figure 3: Distribuzione variabili esplicative discrete

Notiamo che al crescere degli anni il prezzo aumenta in maniera costante, tuttavia ci sono periodi coincidenti con la grande depressione, la crisi degli anni ’70 e quella dovuta proprio al mercato immobiliare del 2008 in cui notiamo degli shock negativi dove appunto il prezzo delle case decresce in maniera importante.

1.3 Variabili continue

L'analisi dalle variabili discrete si sposta ora alle variabili continue, dall'analisi delle distribuzioni possiamo ancora dedurre informazioni importanti:



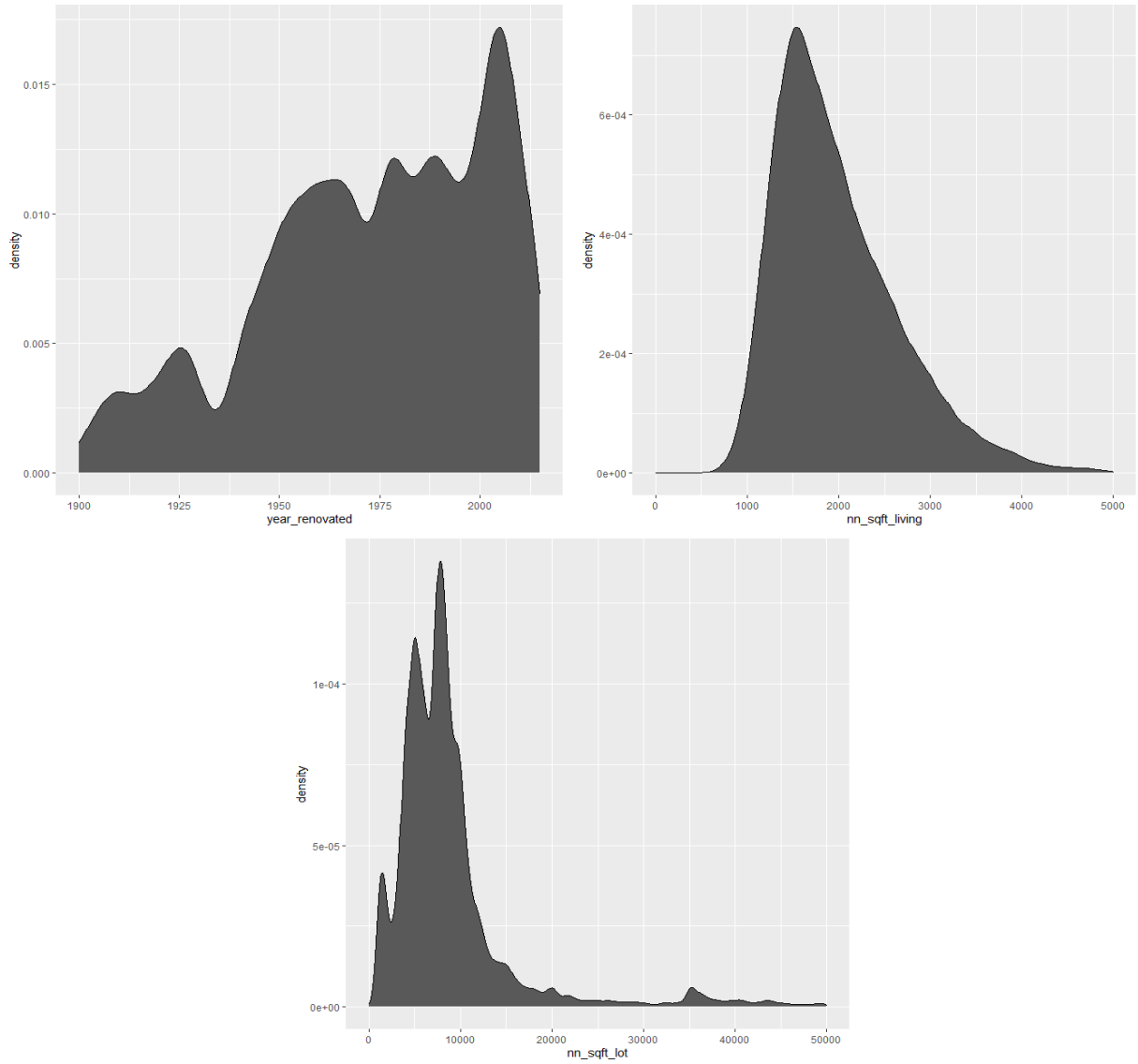
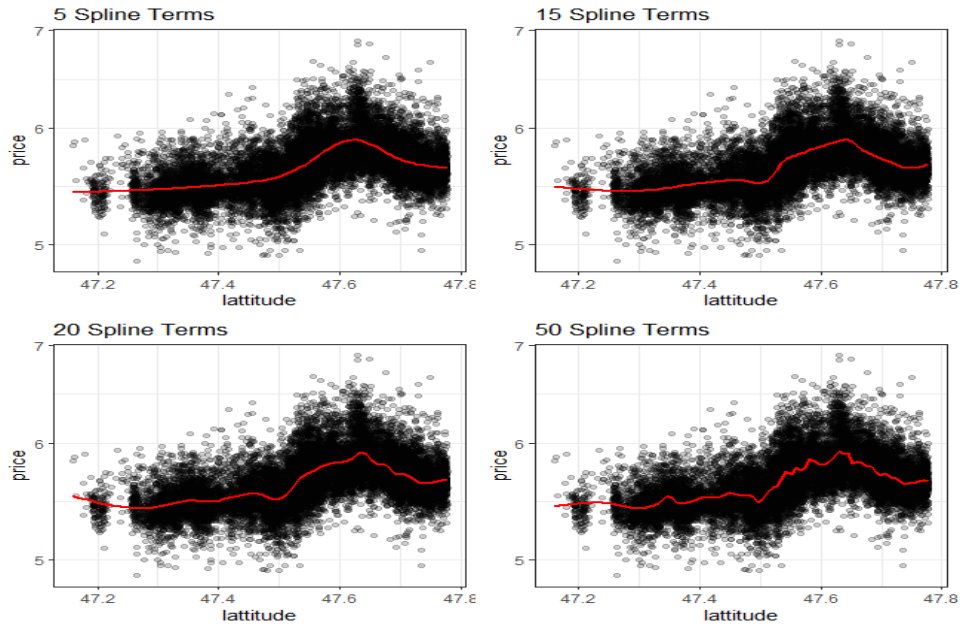


Figure 4: Distribuzione variabili esplicative continue

Dall'analisi delle distribuzioni tutte le variabili continue hanno una notevole asimmetria, spesso nel nostro caso si tratta di asimmetria positiva, tutte inoltre sembrano essere esplicative.

1.4 Relazione tra varibile risposta price e esplicative longitude e lattitude

Ci soffermiamo ora nello specifico su due variabili ovvero *longitude* e *lattitude*, si tratta infatti di soffermarsi su come queste variabili possano essere ben approssimate da delle splines di regressione, quindi abbiamo costruito una funzione che riproduce il grafico delle spline di regressione al variare del numero di gradi di libertà, in particolare come gradi di libertà abbiamo scelto 5, 15, 20 e 50, i grafici sono riportati qua sotto.



Per la variabile *lattitude* una buona approssimazione avviene con 15 gradi di libertà.

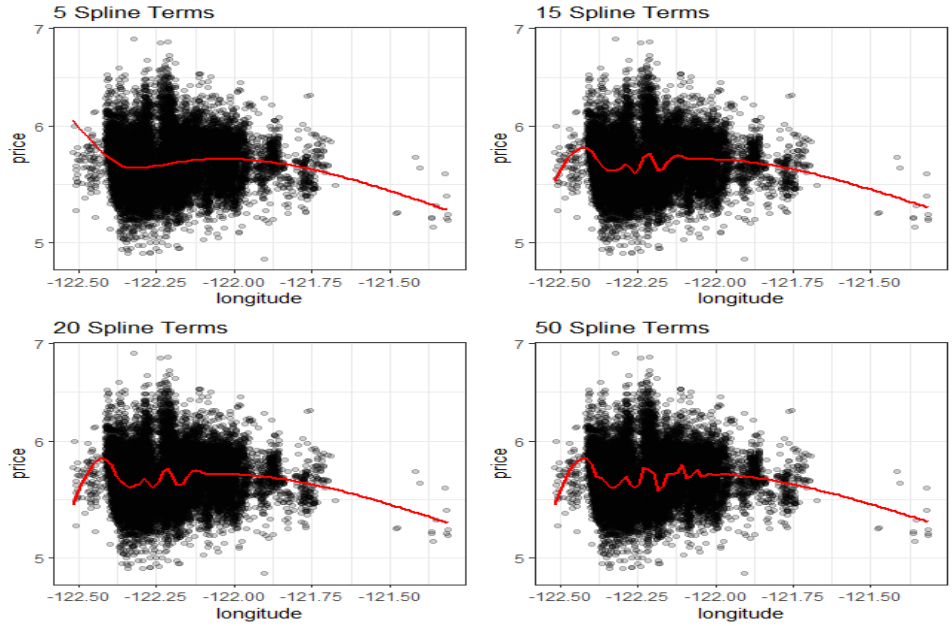


Figure 5: Splines per lattitude e longitude

Per la variabile *longitude* una buona approssimazione avviene con 20 gradi di libertà.

1.5 Analisi della correlazione tra variabili

L'analisi infine si concentra su un'altro aspetto estremamente importante ovvero sulla correlazione tra la variabile risposta *price* e le variabili esplicative che ci vengono fornite dal dataset in esame mediante l'utilizzo di un correlogramma:

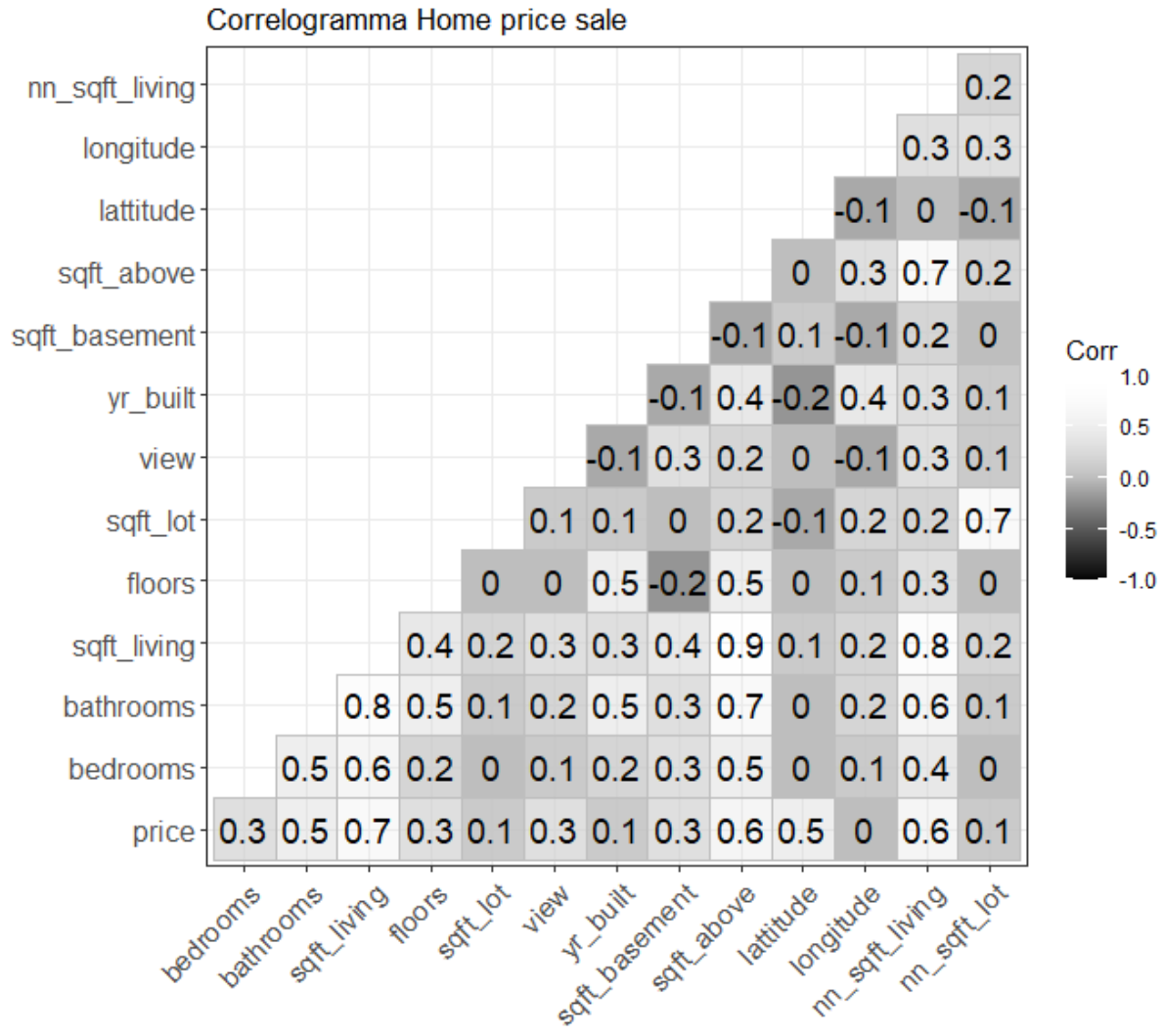


Figure 6: Correlogramma

Interprentando il correlogramma, notiamo come alcune variabili come *sqftliving* sia fortmente correlata con la variabile risposta *price*, sempre analizzando il correlogramma possiamo affermare che lo stesso discorso vale per *sqftabove*, *bathrooms*, *latitude* e *nnsqftliving*.
Altre variabli esplicative tuttavia non sono minimamente correlate con *price* come la variabile *longitude*, *yrbuilt*, *sqftlot* e *nnsqftlot* che hanno una correlazione che si attesta tra lo 0 e lo 0.1.

2 Pre elaborazione dei dati

2.1 Dati mancanti

La prima cosa da verificare è la presenza di dati mancanti, e combinando il training e test set, eliminando la variabile risposta nel training set si può verificare che non vi sia alcuna presenza di dati mancanti.

2.2 Outlier detection

Si passa poi alla ricerca di eventuali outlier o valori anomali utilizzando come strumenti i boxplot delle variabili esplicative.

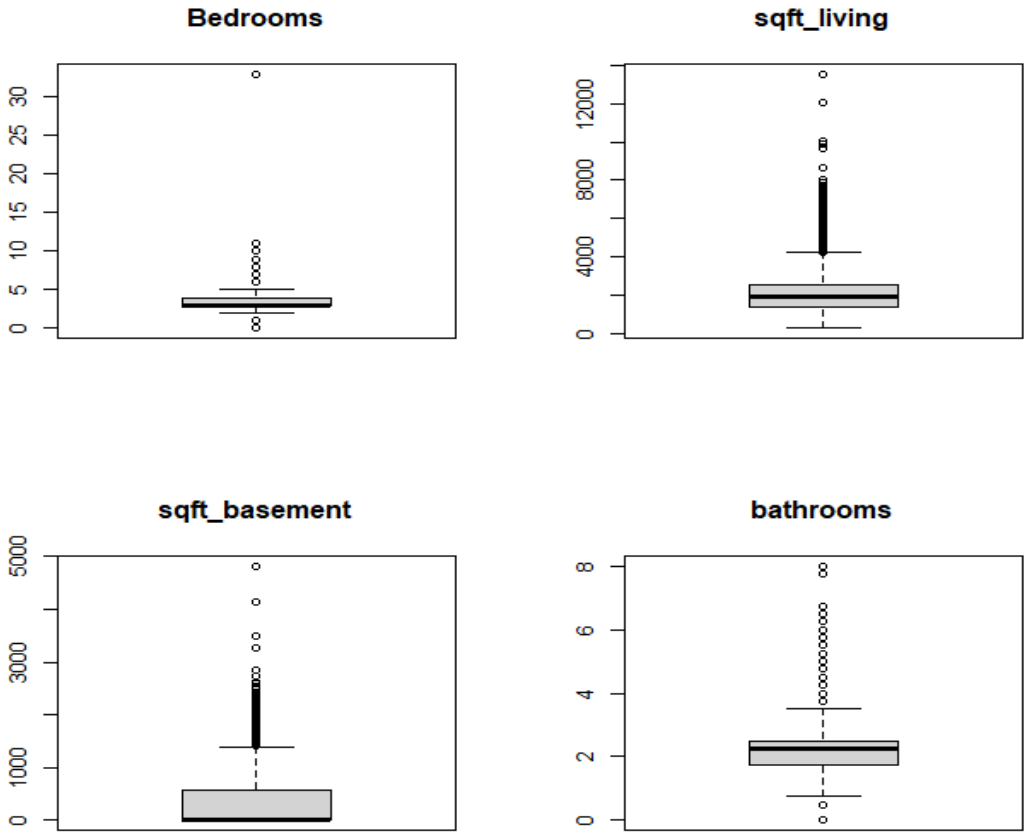


Figure 7: Boxplot outliers

Dall'analisi dei boxplot risulta particolarmente strano che ci siano case che sono state inserite nel dataset che non hanno alcuna camera da letto e dall'altro canto case che hanno esattamente 33 camere da letto e 1.75 bagni, evidentemente rappresentano dei valori anomali. Ci sono inoltre case stando al dataset che non hanno alcun bagno anche questo è molto atipico. Vi sono inoltre osservazioni che hanno valori della metratura della living area estrememante elevati e allo stesso modo per la living area sottoterra.

3 Feature engineering

A partire da alcune variabili del dataset sono state create nuove variabili più utili alla nostra analisi, le nuove variabili sono presentate nella tabella riportata qui sotto.

Variabile di partenza	Nuova variabile
datesold	daysincelastsell
yrbuilt	agebuilt
yearrenovated	agerenovated

Table 1: Riepilogo costruzione di nuove variabili

A partire dalla variabile character *datesold* è stata creata una nuova varibile chiamata *daysincelastsell*, variabile di tipo numerico che misura i giorni trascorsi dalla data di vendita dell'ultima casa.
Dalla variabile *yrbuilt* è stata invece ricavata la variabile *agebuilt* sottraendo la data corrente all'anno di costruzione dell'abitazione, la variabile ottenuta è infatti una variabile numerica che misura l'età dell'abitazione.
Dalla variabile *yearrenovated* è stata creata la variabile *agerenovated* sottraendo alla data corrente la data di ristrutturazione e in caso non ci sia stata una ristrutturazione allora la data di costruzione.
Inoltre si è deciso di sommare le due variabili *nnsqftliving e nnsqftlot* e dunque fonderle in un'unica variabile chiamata *nnsqft*.

4 Confronto tra modelli

Nella nostra analisi sono stati presi in esame 3 modelli:

- *Modello lineare*
- *Random forest*
- *XGBoost*

Sono stati poi confrontati tra di loro e mediante la stima dell'errore di previsione mediante convalida incrociata si è arrivati alla scelta di un modello finale. E' stata adottata in questo frangente come funzione di perdita il mean absolute error (mae).

4.1 Modello lineare

Il modello lineare prende in considerazione tutte le variabili esplicative eccetto *condition*, con delle opportune accortezze. Infatti per quanto riguarda le varibili esplicative *lattitude* e *longitude* è stata usata un' espansione spline più adatta alle due variabili, le espansioni sono rispettivamente di grado 15 e 20. Come motore è stato scelto *lm*.
E' stata calcolato il mae in 10 folds cross-validation che ha il compito di replicare nel modo più realistico possibile il mae sul test set e questo è risultato pari circa a 0.07, ovviamente non è stato fatto alcun model tuning in quanto non previsto per il modello lineare.

4.2 Modello randomForest

E' stato poi considerato un modello *randomForest*, si tratta dunque di un ensemble di alberi decisionali. Un modello più complesso e computazionalmente più oneroso rispetto a quello lineare, abbiamo inserito nella categoria *other* tutte le variabili nominali che avessero bassa frequenza per l'eliminazione di queste variabili, come valore di threshold abbiamo scelto un valore molto basso pari a 0.01, abbiamo inoltre eliminato tutte le variabili che hanno varianza nulla, perchè poco utili all'analisi.
Come motore per *randomForest* in questo caso abbiamo usato *ranger* perchè più efficiente rispetto al motore *randomForest*. Per il modello in esame è previsto il tuning degli iperparametri, a differenza di quello lineare, si è scelta una griglia di dimensione 25 per i parametri di tuning mtry e di minn, dove il primo rappresenta il numero di predittori che verranno campionati casualmente in ogni divisione durante la creazione di modelli, mentre il secondo rappresenta il numero minimo di data points in un nodo necessario per suddividere ulteriormente il nodo. Sono risultati 8 e 6 mentre per il numero di alberi trees abbiamo deciso di impostarlo a 500 senza effettuare alcun tuning anche di questo iperparametro. Abbiamo anche in questo caso calcolato il mae in 10 folds cross-validation per simulare il mae sul test set ed è risultato pari a 0.05512 un ottimo miglioramento rispetto al modello lineare.

4.3 Modello XGBoost

Si è infine passati a considerare il terzo modello che è un XGBoost, anche questo modello è sicuramente computazionalmente più oneroso del modello lineare, abbiamo inserito in *other* tutte le categorie delle variabili nominali che hanno bassa frequenza, threshold molto bassa pari a 0.01 e abbiamo eliminato tutte le variabili con varianza nulla.
Il motore utilizzato è *xgboost* come griglia dei parametri di tuning si è deciso di utilizzare una griglia Latin hypercube che distanzia i punti il più lontano possibile l'uno dall'altro, pertanto permette una migliore esplorazione degli iperparametri, l'ampiezza della griglia è stata fissata a 50. Si è dovuto fare il tuning di *minn, treedepth, mtry, learnrate, lossreduction* e di *samplesize*, *minn e mtry* sono stati già stati affrontati nel modello *randomForest* mentre treedepth rappresenta la profondità massima dell'albero, learnrate che corrisponde all'eta di *XGBoost*, *lossreduction* rappresenta la riduzione della funzione di perdità neccesaria per un ultriore frazionamento e *samplesize* la dimensione dei dati utilizzato per la modellazione all'interno di un'iterazione dell'algoritmo.
Un numero elevato iperparametri che al termine del processo di tuning ha portato ad un mae in 10 folds cross-validation di 0.0511555, anche qui si ha un miglioramento rispetto ai due modelli precedenti.

5 Scelta del modello finale

Dopo aver esplorato singolarmente i 3 modelli presentati precedentemente, come modello finale si è deciso di considerare un ensemble dei 3. La forza dell'ensemble è che rende le stime più consistenti oltre al fatto che gli ensemble generalmente sono i modelli che portano i risultati migliori.
E' stata costruita una griglia di dimensione 50 per il tuning degli iperparametri di *XGBoost* e *randomForest*, abbiamo ordinato i migliori modelli secondo la metrica del mae in 10 folds cross-validation e i risultati dei 5 migliori modelli ottenuti sono riportati nella tabella mostrata di seguito.

Modello	mae	rank
boost-tree	0.0518	1
boost-tree	0.0517	2
boost-tree	0.0519	3
boost-tree	0.0522	4
boost-tree	0.0524	5

Table 2: Riepilogo 5 migliori modelli dell'ensemble secondo metrica mae

Per l'ensemble erano presenti 99 candidati, tuttavia ne sono stati considerati 12 per la costruzione del modello finale, ognuno con opportuno peso. Qui sotto viene riportata la tabella riassuntiva dei modelli scelti con il relativo peso e un grafico esemplificativo.

Membro	tipo	peso
xgboost-1-24	boost-tree	0.334
xgboost-1-50	boost-tree	0.284
xgboost-1-47	boost-tree	0.186
xgboost-1-04	boost-tree	0.0953
xgboost-1-14	boost-tree	0.0461
Rf-1-03	rand-forest	0.0106
Rf-1-05	rand-forest	0.00344
Rf-1-04	rand-forest	0.000564
xgboost-1-20	boost-tree	0.000106
xgboost-1-46	boost-tree	0.0000529

Table 3: Riepilogo primi 10 modelli scelti dall'ensemble con relativo peso

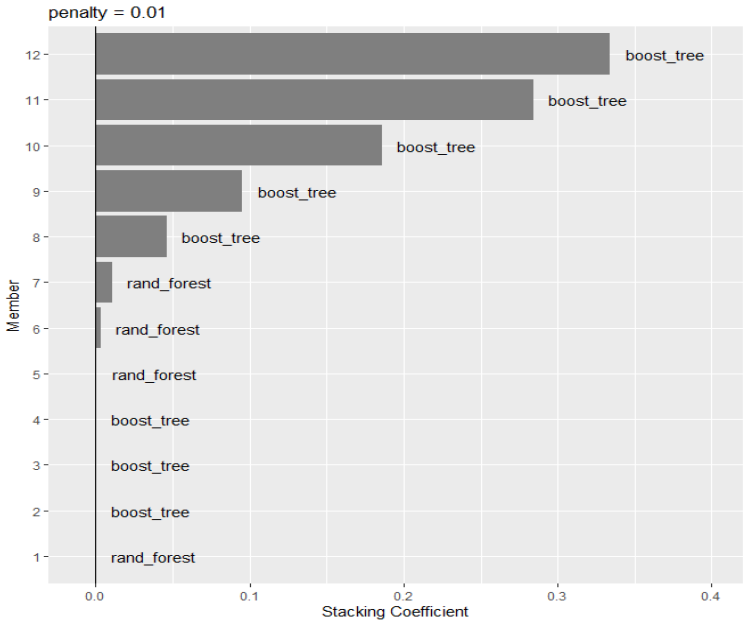


Figure 8: Modelli con relativo peso nell'ensemble

Un certo numero di candidati come abbiamo appena osservato ha coefficienti di stacking diversi da zero. Tali candidati sono indicati come *member*, notiamo subito la predominanza dei modelli *xgboost* per il semplice motivo che sono i modelli che ottengono in cross-validazione un valore più basso di *mae*.
Dobbiamo ora unire i *member* nell'ensemble. Una volta fatto ciò possiamo effettuare le previsioni finali sul test set, abbiamo così ottenuto le prevsioni del nostro modello finale.