Relazione

Home sales prices

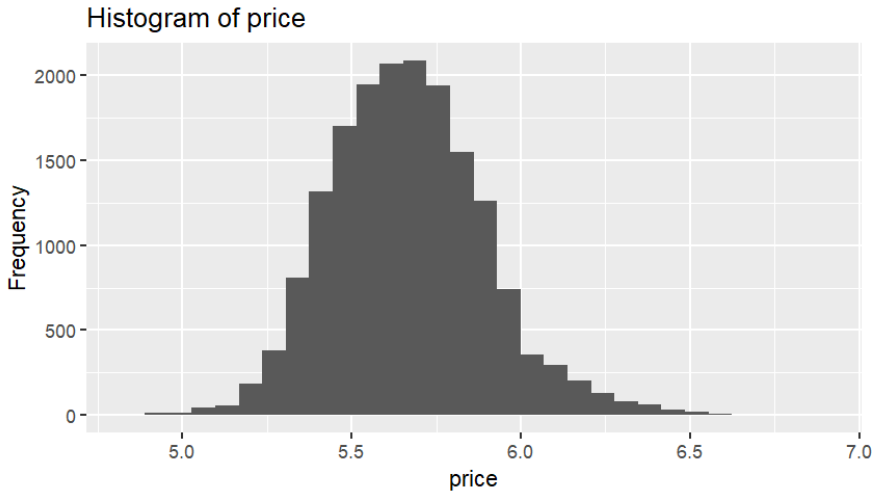
Davide Zulato, mat. 876101

Sintesi del processo di modellizzazione

1. ***Analisi esplorativa dei dati ed Exploratory Spatial Data Analysis***

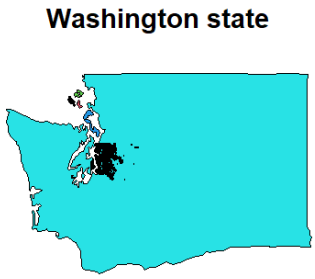
È stata considerata la distribuzione della variabile risposta, rappresentata con l'istogramma seguente (*Figura 1*), da un'analisi grafica non si discosta da un'assunzione di normalità. Il prezzo di vendita è espresso in scala logaritmica. L'obiettivo dell'analisi è la previsione del prezzo di vendita (*price*, in scala log10) delle 4320 abitazioni del test set.

Figura 1

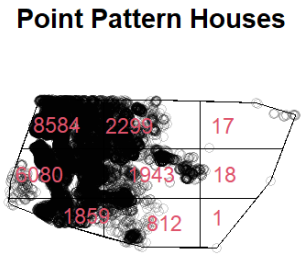


All'interno del dataset (training e test set) non sono presenti valori mancanti.

Figura 2



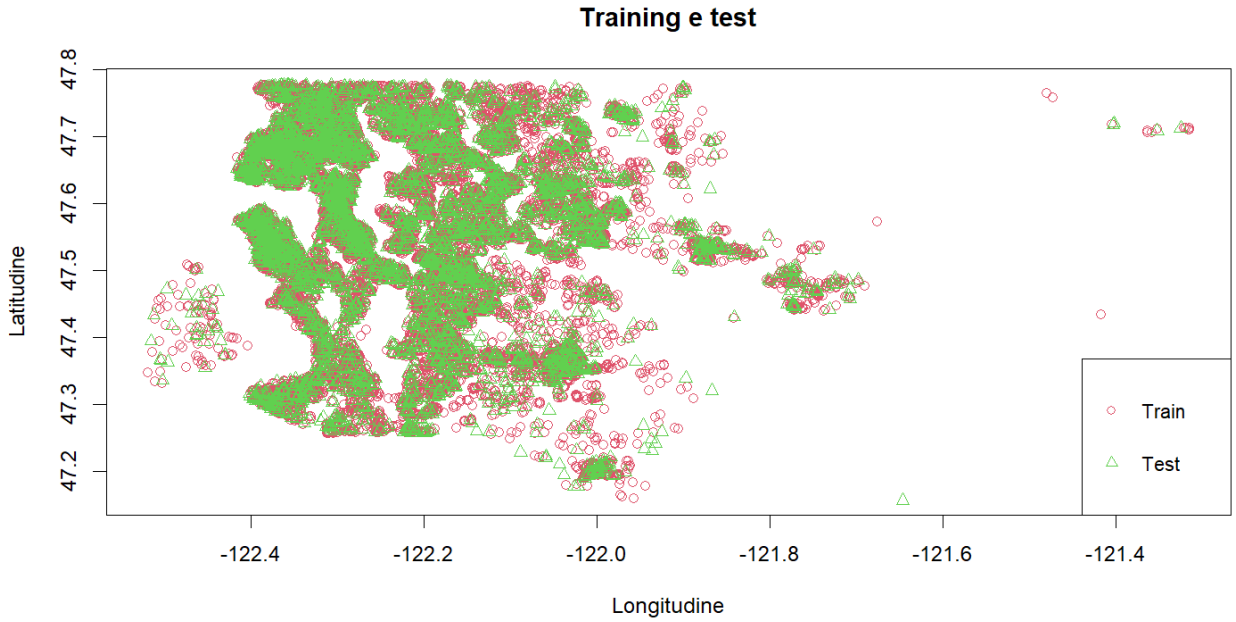
House location



Quadrat counting for a point pattern

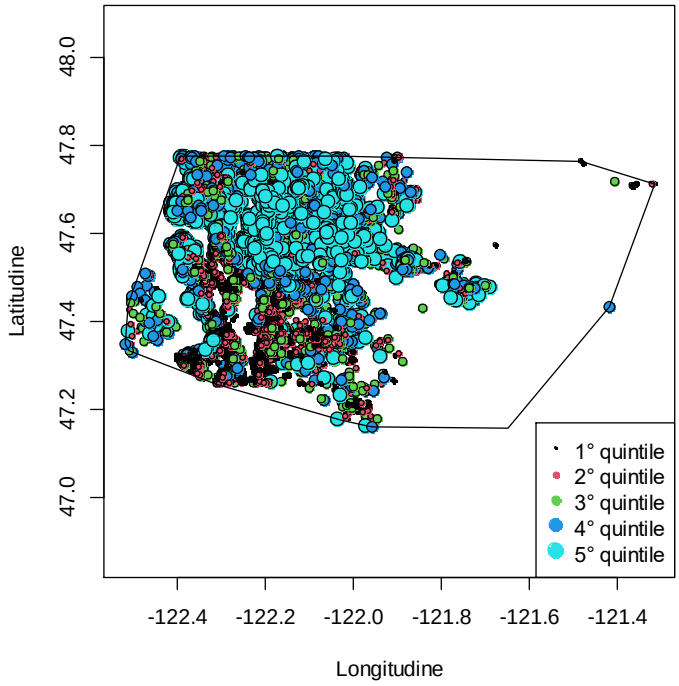
L’immagine (*figura 2*) mostra sulla sinistra la locazione delle 21613 abitazioni (training e test set) all’interno dello stato di Washington nell’area occupata dalla contea di King. Sulla destra è invece mostrata, attraverso una griglia 3x3 la locazione delle abitazioni nell’insieme convesso più piccolo in grado di contenere tutte le osservazioni. In rosso è indicata la numerosità di ciascuna cella. La densità più elevata si registra a nord-ovest della contea, in corrispondenza dell’area metropolitana di Seattle. Le celle a est presentano invece un point pattern meno denso e identificano le zone della contea di King più montuose e meno densamente popolate.

Figura 3



L’immagine precedente (*Figura 3*) mostra la distribuzione spaziale delle case in training e test, sono presenti più abitazioni sulla stessa coordinata; questo, considerando che non sono presenti duplicati, è imputabile a edifici contigui o appartamenti all’interno dello stesso stabile. Sono visibili, inoltre, delle aree vuote: da un’osservazione dell’area della contea “King” si nota che sono presenti diverse aree coperte d’acqua come Lake Washington, Lake Sammamish e lo stretto di Puget che separa la contea da Vashon-Maury Island (insieme di proprietà a sud-ovest) oppure aree industriali/aeroporti [2]. Il point pattern del training set è comparabile con quello del test set. Non sorprende notare che nell’area metropolitana è frequente osservare coordinate sovrapposte (e.g. lotti nello stesso stabile).

Figura 4



Nella figura (*Figura 4*) I punti sono divisi in 5 diverse dimensioni e colori a seconda dei quintili del logaritmo del prezzo di vendita. Si nota che a sud-ovest della contea le abitazioni tendono ad avere un prezzo di vendita che si posiziona tra il primo e il terzo quintile della distribuzione (è presente una vasta zona industriale a nord della città di Kent e il Seattle-Tacoma international Airport che potrebbero avere un effetto sul valore). La latitudine e la longitudine sono variabili importanti per la previsione del prezzo di vendita delle case come mostrano anche le misure di *variable importance* ottenute via Random Forest, (Mean Decrease Accuracy (%IncMSE) stimata con out-of-bag-CV) latitudine e longitudine sono le prime 2 variabili più importanti.

2. Pre-elaborazione dei dati e Feature engineering

La fase di modellizzazione comprende anche una pre-elaborazione dei dati. Al fine di implementare l'algoritmo Xgboost le variabili categoriali e sono state convertite in dummy. La letteratura mostra inoltre che l'utilizzo della Principal Component Analysis aiuta a migliorare le performances dell'algoritmo Xgboost [7].

Per i modelli lineari, dalla variabile *date_sold* è stato estratto il mese di vendita al fine di poter considerare attraverso variabili dummy, la stagionalità mensile delle vendite nel corso dell'anno; la prima vendita è stata effettuata il 2 maggio 2014 mentre l'ultima il 27 maggio 2015. La variabile ricodificata è quindi passata da categoriale con 12 classi a dummy. Nell'implementazione di XGBoost questa trasformazione è stata abbandonata poiché non porta a miglioramenti delle performances.

In sintesi:

Tabella 1

Operations in Preprocessing	
Operation	**columns**
Collapsing factor levels	condition
Dummy variables	condition
Centering	All numeric predictors
Scaling	All numeric predictors
PCA extraction	sqft_(above,basement,living,lot),nn_sqft_(living,lot),year_renovated,yr_built

Dall'analisi esplorativa è emersa la forte correlazione tra alcuni predittori; per non incorrere in problematiche legate alla multicollinearità alcune variabili sono state trasformate (*Tabella 1*).

Nel dataset sono presenti diverse variabili che misurano la dimensione dell'area occupata dall'abitazione e delle proprietà limitrofe (*sqft_above*, *sqft_basement*,*sqft_living*,*sqft_lot*,*nn_sqft_living*,*nn_sqft_lot*). Principal component analysis (PCA) con 5 componenti principali è utilizzata per estrarre maggiore informazione possibile dal set di predittori iniziale, potenzialmente ridondante, usando un numero inferiore di features.

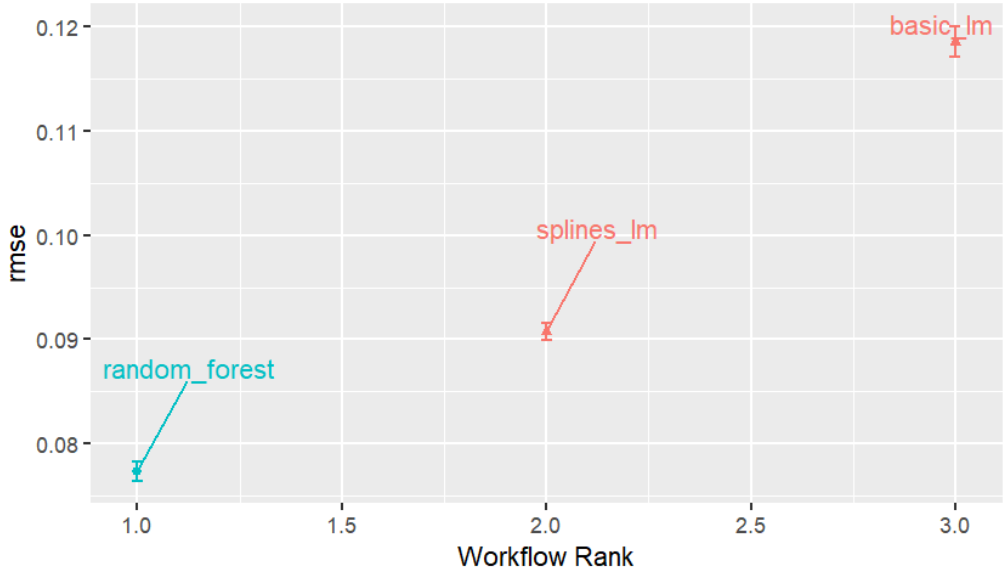
PCA è stata applicata anche alle variabili anno di costruzione e anno di ristrutturazione (*year_renovated*, *yr_built*).

Le variabili che misurano l'area delle proprietà sono espresse in *square feet* ma l'anno di rinnovo e l'anno di costruzione sono su una scala diversa: è quindi necessario centrare e riscalare le colonne prima di applicare PCA poiché è assunto che i predittori siano sulla stessa scala.

3. Confronto tra modelli

È stato effettuato il confronto tra diversi modelli, inizialmente tra alcuni modelli lineari e Random Forest e infine tra Random Forest e XGBoost. Nello specifico il primo confronto è tra modello lineare semplice (con colonne centrate e riscalate), modello lineare con splines per longitudine e latitudine e modello Random Forest (Preprocessing non ha avuto effetti significativi sulle performances).

Figura 5



Da questo plot (*Figura 5*) degli intervalli di confidenza *rmse* possiamo vedere che il metodo Random Forest presenta le performance migliori.

Si confrontano quindi il modello Random Forest e il modello XGBoost.

È stata effettuata la regolazione del modello (*model tuning*) sugli iperparametri dell'algoritmo RandomForest e XGBoost via grid-search. È stata poi effettuata la stima dell'errore di previsione attraverso il metodo della convalida incrociata (*10 folds cross-validation*). Al fine di ottimizzare i tempi computazionali stato usato *parallel processing*; poiché le diverse parti della griglia sono indipendenti [4] è stato selezionato grid = 20 in modo da scegliere 20 *grid points* automaticamente per trovare migliore combinazione di 2 iperparametri (*mtry* e *Min_n*). La tabella seguente mostra gli iperparametri del miglior modello. Le metriche utilizzate sono R^2 e *Root mean squared error* (rmse).

Tabella 2

Random Forest Model Specification (regression)	
Main arguments	**Tuned hyperparameter**
mtry	9
trees	1000
Min_n	8
Computational engine	ranger

La performance (i.e. stima dell'errore di previsione) è stimata via 10 folds cross-validation, le metriche sono presentate nella tabella seguente (*Tabella 3*).

Tabella 3

metric	**mean**	**n**	**std_err**	**config**
rmse	0.0774	10	0.000551	Preprocessor1_Model1
rsq	0.886	10	0.00169	Preprocessor1_Model1

La tabella seguente mostra l'aggiornamento degli iperparametri dell'algoritmo xgboost via grid-search (grid=20), mentre la *Tabella 5* mostra le metriche *root mse* e R^2 ottenute via cross-validazione.

Tabella 4

Boosted Tree Model Specification (regression)	
Main arguments	**Tuned hyperparameter**
loss_reduction	2.43502192814882e-10 (ca. 0.00)
learn_rate	0.0199035673943381
Tree_depth	13
trees	1000
Min_n	38
Engine-Specific Arguments	
Computational engine	xgboost
objective	reg:squarederror

Tabella 5

metric	**mean**	**n**	**std_err**	**config**
rmse	0.073	10	0.000542	Preprocessor1_Model1
rsq	0.898	10	0.00130	Preprocessor1_Model1

Le performances dell'XGBoost ottenute con una 10-fold cross-validation sono migliori di quelle dell'algoritmo Random Forest.

Scelta del modello finale

La scelta del modello è stata attuata attraverso ricampionamento (cross-validation). Il miglior modello è risultato essere XGBoost con iperparametri presentati nella *Tabella 4* e preprocessing presentato nella sezione 2 (*Tabella 1*).

Figura 6

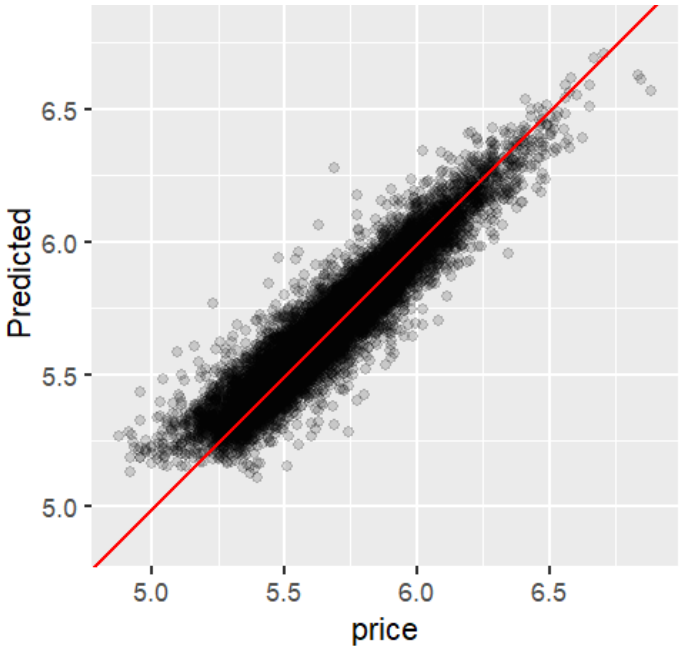


Tabella 6

metric	estimate
rmse	0.07
rsq	0.90
mae	0.05

La *Tabella 6* mostra una stima dell'errore di previsione con l'aggiunta della metrica *mae* (Mean absolute error) rispetto alla *Tabella 5*, ottenuta attraverso un hold-out set (20% del training set per la stima) mentre la figura (*Figura 6*) mostra il fit tra previsione attraverso XGBoost e prezzo di vendita nel training set via 10-fold cross-validation, migliore rispetto agli alti modelli testati.

4. Bibliografia e sitografia

[1]. Azzalini, Scarpa (2004). Analisi dei dati e data mining. Springer-Verlag Italia.
[2]. Google Earth https://www.google.it/intl/it/earth/index.html
[3]. Kuhn, Johnson (2019). Feature Engineering and Selection. Chapman and Hall/CRC.
[4]. Kuhn, Silge (2021+). Tidy Modeling with R. In progress.
[5]. Schabenberger, Gotway (2004). Statistical Methods for Spatial Data Analysis. Taylor & Francis Inc.
[6]. Wickham, Grolemund (2017). R for Data Science. O'REILLY.
[7]. XGBoost Documentation: https://xgboost.readthedocs.io/en/stable/index.html#