HOUSE PRICES IN AMES - PROJECT REPORT

Marco Bresciani - Mt. 876347

DATASET EXAMINATION

Il dataset è stato partizionato in parte di test per il 25% e in parte di train per il 75%.
Tutte le analisi sono state effettuate sulla sola parte di train, per evitare la contaminazione dei modelli, e l'influenza sul processo decisionale lungo il workflow dello sviluppo.

Sono state individuate le features e categorizzate a seconda della tipologia(**Numeriche** - **Categoriali**). I type delle colonne sono stati uniformati.
Il dataset fornito è risultato fortunatamente essere già pronto all'uso.
A seguito di controlli più approfonditi non si presentano :

- MISSING VALUES,
- valori anomali,
- inoltre i valori assunti delle features categoriali risultano tutti omogenei nella scrittura.

ANALISI CORRELAZIONE

Confermato trattarsi di un task di REGRESSIONE, si è proceduto a verificare il numero e la tipologia di features messe a disposizione con relativi valori, seguita da un'analisi della matrice di correlazione tra features numeriche, per verificare correlazioni significative con la variabile target, presenza di multicollinearità e facilitare l'individuazione delle coppie di feature.

NUMERICAL CORRELATION MATRIX

Si è proceduto poi con un' ulteriore analisi di correlazione tra features numeriche e categoriali ed estratte le coppie di features più correlate, utili per ridurre la dimensionalità del dataset nella fase di **FEATURE ENGINEERING**.

COMPLETE CORRELATION MATRIX

A seguito delle precedenti analisi pare che le features più influenti sul prezzo di vendita durante una prima fase esplorativa siano :

- **Neighborhood**
- **Overall Qual**
- **Exter Qual**
- **Bsmt Qual**
- **Gr Liv Area**
- **Kitchen Qual**
- **Garage Cars**
- **Garage Area**

Verrà poi eseguita un'ulteriore analisi a seguito della fase di feature engineering ed encoding delle variabili categoriali.

DATA VISUALIZATION - FEATURE ENGINEERING

Da una prima analisi la variabile di interesse Sale Price risulta essere asimmetrica, con una lunga coda verso destra. Non pare essere distribuita normalmente.
Si nota che il prezzo medio di vendita nella città di Ames è $178.985, in un range di prezzi che va da $12.789 come MIN a $755.000 come MAX.

Una trasformazione in scala logaritmica risulta capace di normalizzare la variabile risposta. La trasformazione verrà presa in considerazione durante la fase di utilizzo dei modelli per verificare la presenza di differenze significative.

DISTRIBUZIONE SALE PRICE - TRASFORMAZIONE LOGARITMICA

Per questioni di interpretabilità nella fase di visualizzazione viene mantenuta la scala originale della variabile risposta.

FEATURES DERIVATE BINARIE

Vengono estratte feature binarie attraverso quelle che offrono questa possibilità, visualizzato l'impatto del nuovo predittore e verificato se questo può essere interessante per il nostro modello.
Variabili estratte:

- Has_Pool : presenza di una piscina o meno
- Has_Garage : presenza di un garage
- Has_Alley : presenza di un vialetto
- Has_Basement : presenza di un seminterrato
- Has_Fireplace : presenza di un camino
- Has_Fence : presenza di una recinzione
- Is_Remodeled : indica se la casa è stata ristrutturata

FEATURES DERIVATE

In base all'analisi delle feature presenti e i valori che esse assumono, sono state derivate le seguenti nuove feature :

- Total_Bathroom : sommate le feature numeriche riguardanti la quantità di bagni
- Total_SF : feature per indicare i metri quadri totali dell'abitazione
- **close_to_park**, **close_to_street**, **close_to_railroad** : estratte caratteristiche dell'abitazione riguardo la vicinanza a parchi, strade o ferrovie,
- House_Age : calcolata l'età della casa al momento della vendita
- **Regular, Irregular - Lot Shape** : i livelli di irregolarità sono stati raggruppati in un unica categoria "Irregular Lot Shape"
- **Residential, Commercial, Industrial, Agriculture - Zoning** : tutti i livelli di densità residenziali sono stati raggruppati in una sola feature e viene creata una feature per ogni zona rimanente.

A seguito della visualizzazione dell'influenza dei predittori numerici e categoriali, su Sale_Price, apparentemente non significativa, e analisi della multicollinearità vengono **eliminate 34 features**.

Si rende disponibile la visualizzazione dei predittori analizzati QUI.

ENCODING

Le feature categoriche sono state suddivise a seconda della tipologia di encoding da applicare.
E' stato applicato **Label Encoding** a tutte le features che esprimono una scala **ORDINALE**, per tutte quelle di tipo **NOMINALE** è stato applicato **One Hot Encoding**.

- PROBLEMA RISCONTRATO
  Per una questione di praticità è stata assunta l'ipotesi di essere già a conoscenza di tutti i valori che una feature può assumere per un' abitazione, cosicché il dataset di test non introduca nuovi valori non a conoscenza dai modelli.

Questo perchè nello splitting progettuale risultava che istanze con un determinato valore nella feature "Quartiere", fossero presenti solo nella parte di test portando i dataset ad avere un numero di colonne differente a seguito del one-hot-encoding.

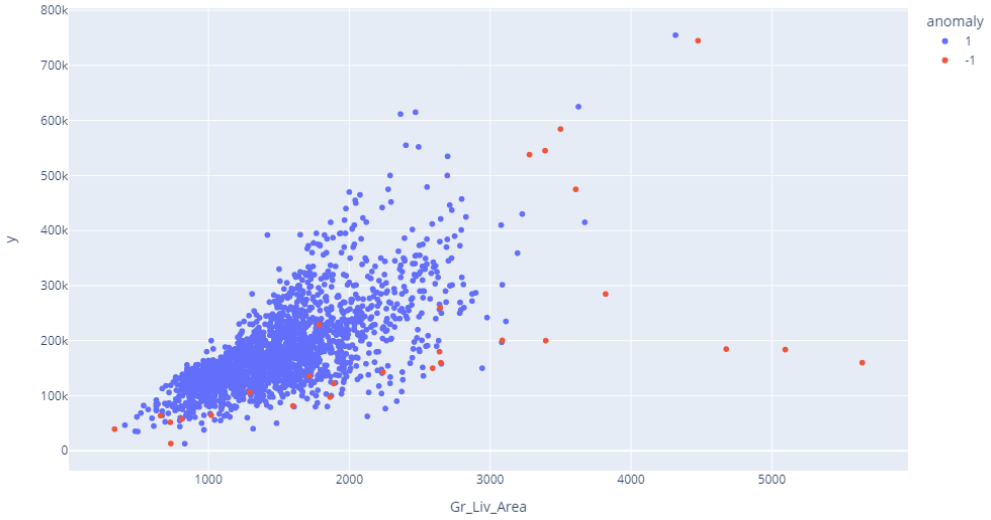
- SOLUZIONE APPLICATA
  Per uniformare il numero delle colonne post encoding :
  - fatto un merge temporaneo dei dataset di test e train,
  - applicato l'encoding,
  - ristabiliti i dataset con le medesime istanze alla situazione precedente al merge.

OUTLIERS

E' stato deciso di procedere all'individuazione delle istanze potenzialmente identificabili come outliers, per confrontare modelli allenati su dataset con outliers inclusi ed esclusi, verificando la presenza di differenze significative in termini di errore sul dataset di test.

ISOLATION FOREST

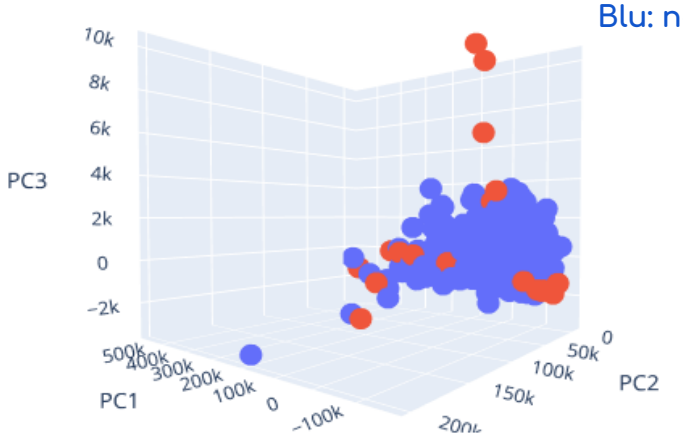
Per l'individuazione , è stata utilizzata una Isolation Forest basata su alberi decisionali.
A seguito del tuning del modello tramite GridSearchCV sono stati individuati **28 outliers.**



Gli outliers si posizionano in un grafico tridimensionale a seguito di una dimensional reduction del dataset attraverso **PCA** nel seguente modo:

Total Explained Variance: 99.98%

Rosso: outlier
Blu: non outlier



MODELLI

Si segnala che per tuning dei parametri di ogni modello è stato utilizzato GridSearchCV con 5-Cross Validation.
I modelli sono stati valutati attraverso le metriche: **RMSE(Root Mean Square Error) - R2 - MAE(Mean Absolute Error)**

Model Name : Simple RandomForest
- **RMSE** : 23036.14844556519
- **R2** : 0.9252139155062807
- **MAE** : 15442.10546371684

Model Name : Tuned RandomForest
- **Optimum parameters** : {'bootstrap': True, 'max_depth': None, 'max_features': 0.3, 'min_samples_split': 2, 'n_estimators': 5000}
- **RMSE** : 23305.545642306537
- **R2** : 0.9234545096337934
- **MAE** : 15246.369139768196

I risultati dei modelli di **RANDOM FOREST** con **outliers** e con variabile risposta in **scala logaritmica**, non risultano significativamente diversi. Random forest si dimostra robusto ad outliers e stabile a prescindere dalla trasformazione della variabile.

Model Name : Simple XGBRegressor
- **RMSE** : 23584.30236698373
- **R2** : 0.9216124434203581
- **MAE** : 15396.101312031036

Model Name : Tuned XGBRegressor
- **Optimum parameters** {'colsample_bytree': 0.3, 'learning_rate': 0.01, 'max_depth': 3, 'n_estimators': 5000}
- **RMSE** : 20375.812575003853
- **R2** : 0.9414898874889327
- **MAE** : 13504.83685943895

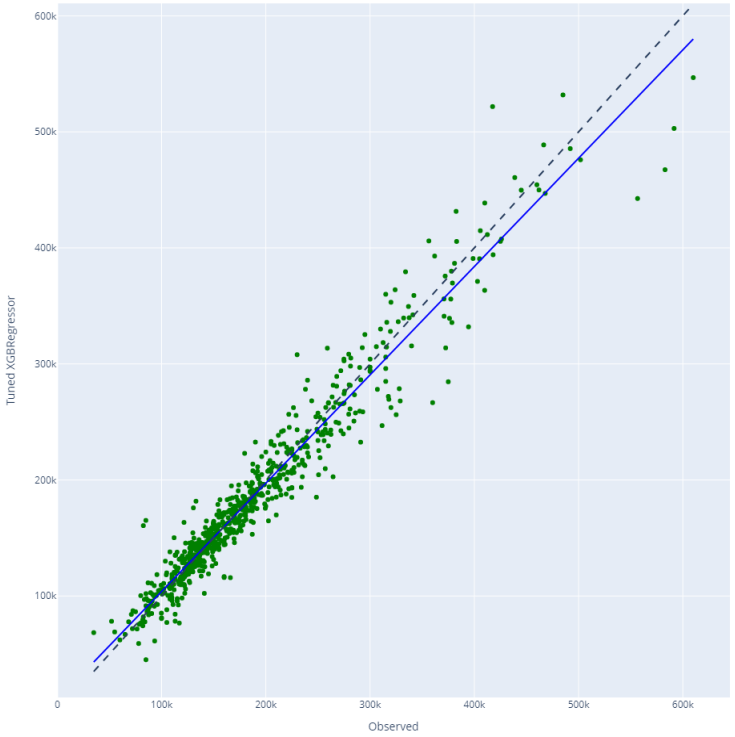
Il tuning del modello XG Boost è stato fatto utilizzando 2 strategie per valutare le performance del modello: **“r2” e "neg_root_mean_squared_error",** questo perchè dopo un primo tuning solo utilizzando **neg_root_mean_squared_error** sulle abitazioni di maggior costo, il prezzo veniva appiattito.

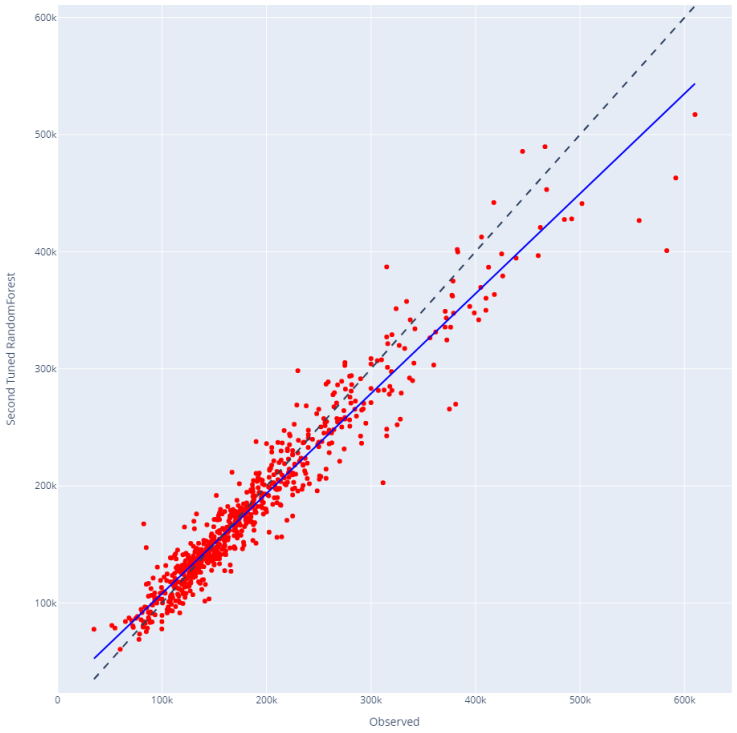
Model Name : KNN Regressor
- **Optimum parameters** {'metric': 'manhattan', 'n_neighbors': 12, 'weights': 'distance'}
- **RMSE** : 42378.86315055835
- **R2** : 0.7468956184877495
- **MAE** : 27200.587571726584

XGboost pare essere il modello migliore differisce in positivo rispetto a Random forest di:
- circa 3000$ rispetto al **RMSE**
- circa 1700$ rispetto al **MSE**
- circa 0.02 punti rispetto al **R2**

CONCLUSIONI

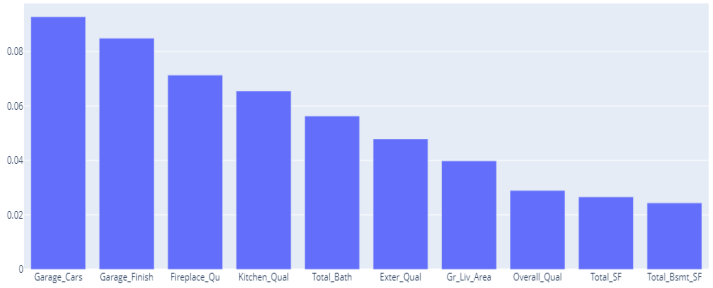
Fitted vs Observed - **XGBoost** vs **Random Forest**

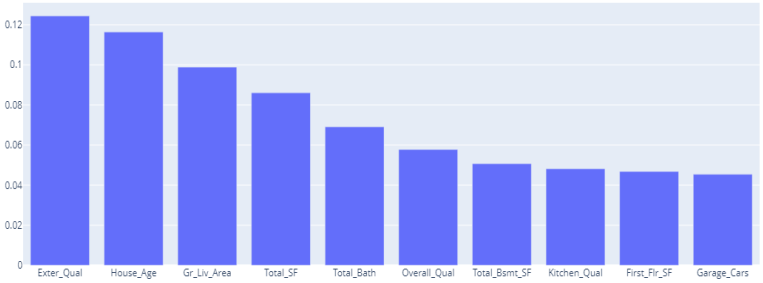




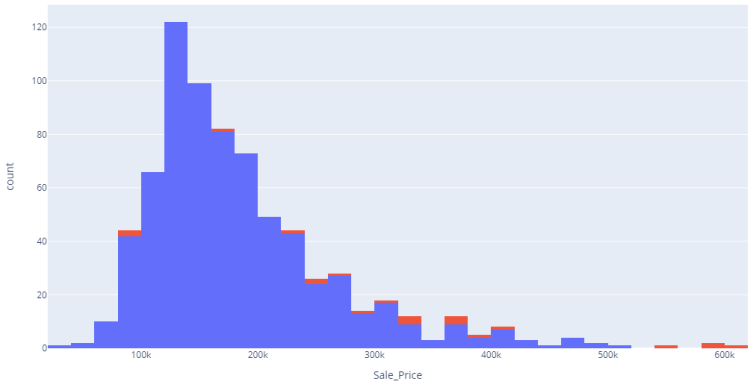
Si nota come **XGBoost** sia migliorato in maniera abbastanza significativa nella predizione delle case con prezzi **distanti dalla media** del dataset rispetto a **Random Forest**, un leggero miglioramento lo notiamo anche nelle case con il prezzo più basso. XGBoost pare avere prestazioni generalmente migliori.

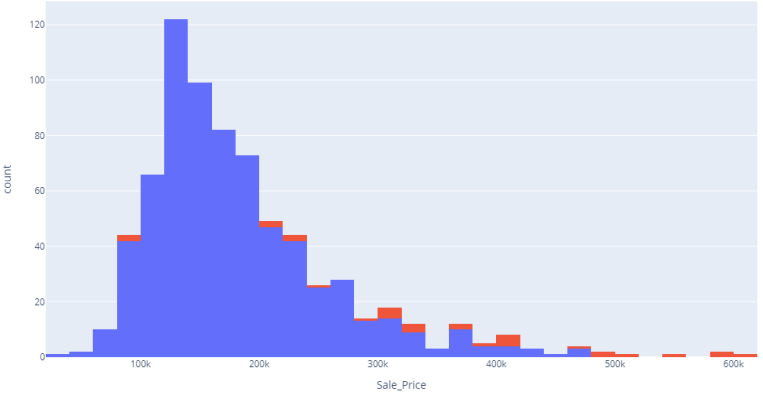
Feature Importance - **XGBoost** vs **Random Forest**





Peggiori Predizioni - **XGBoost** vs **Random Forest**





Si osserva come **XGboost** riduca il numero di predizioni molto distanti lungo tutta la coda destra a partire dai 200K, si segnala che le predizioni risultate distanti anche in XGBoost, risultano essere più vicine al valore osservato rispetto a Random Forest.