King County Revolution

next slide →

Transforming the Future
by Vinicius Jodar & Sergio Eguakun

02

Introduction

- *This project aims to predict house sales prices in King County, Seattle, based on data from May 2014 to May 2015. The dataset includes 21,613 records with 20 property features, and the target variable is the price column.*

- *To find the most accurate approach, we trained and compared five models: **Linear Regression, Random Forest, XGBoost, Gradient Boosting, and Ridge.** Finally, we reveal which model delivers the best prediction performance.*

next slide →

About the Models

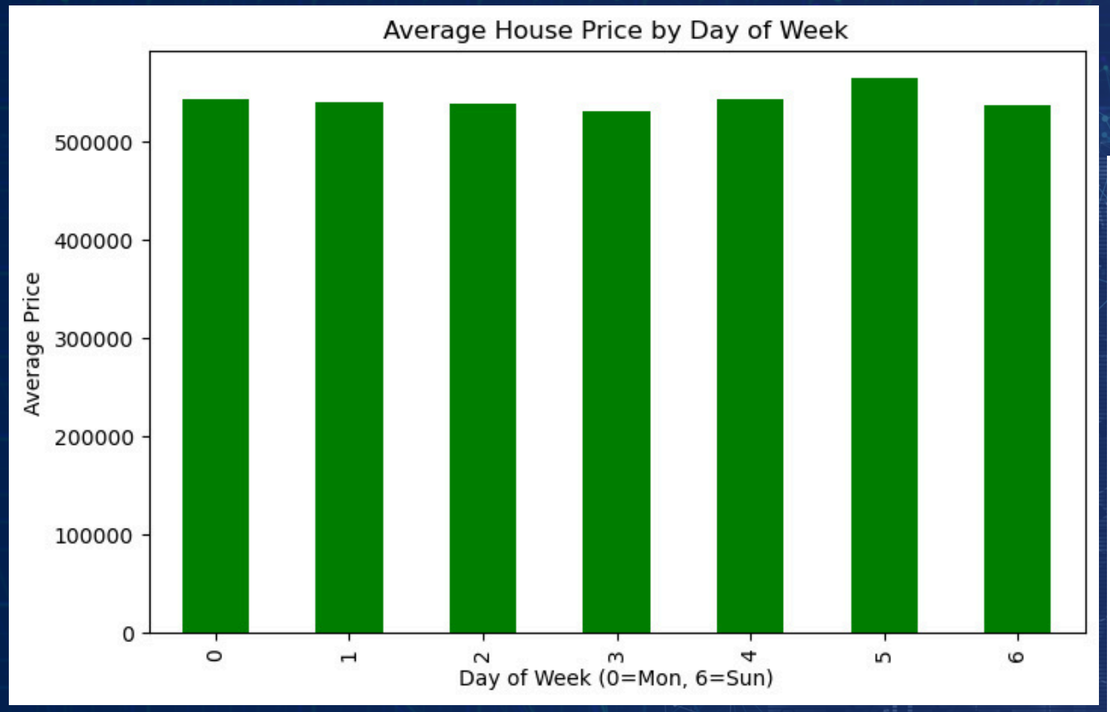
- **Linear Regression** → simple, interpretable, and a strong baseline for comparison;
- **Ridge Regression** → regularized linear model, reduces overfitting by penalizing large coefficients, good for multicollinearity;
- **Random Forest** → ensemble method, reduces variance and captures non-linear patterns;
- **Gradient Boosting** → sequential boosting, balances accuracy and interpretability;
- **XGBoost** → optimized boosting, highly efficient and often top performer in tabular data.

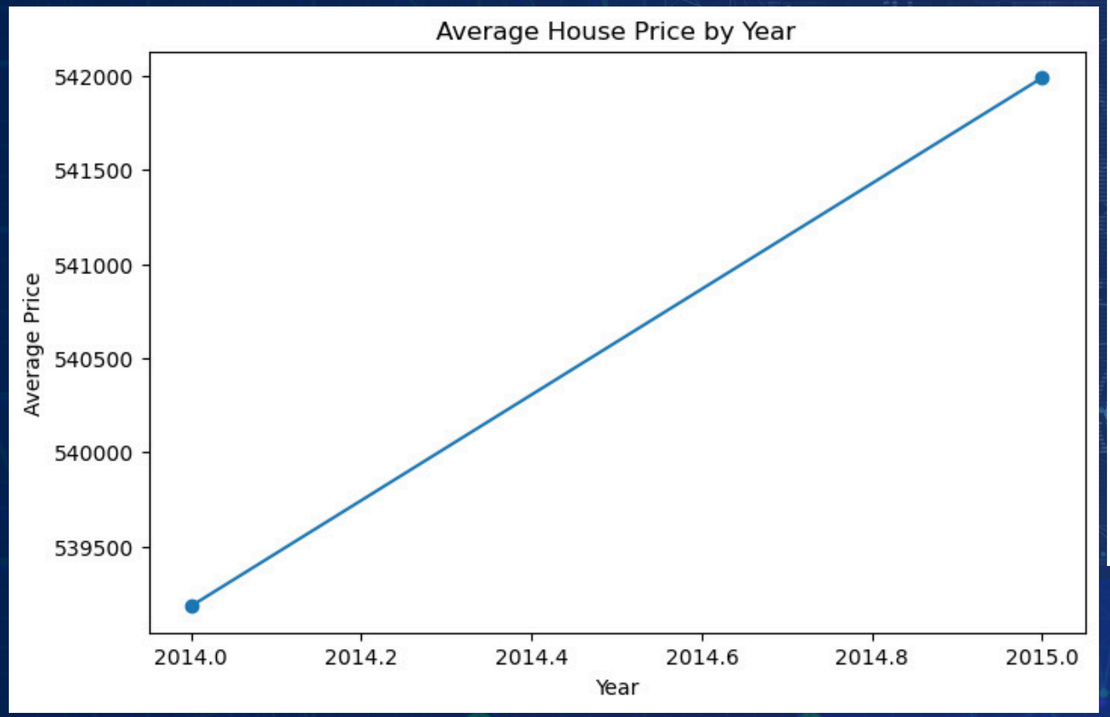
Why these models?

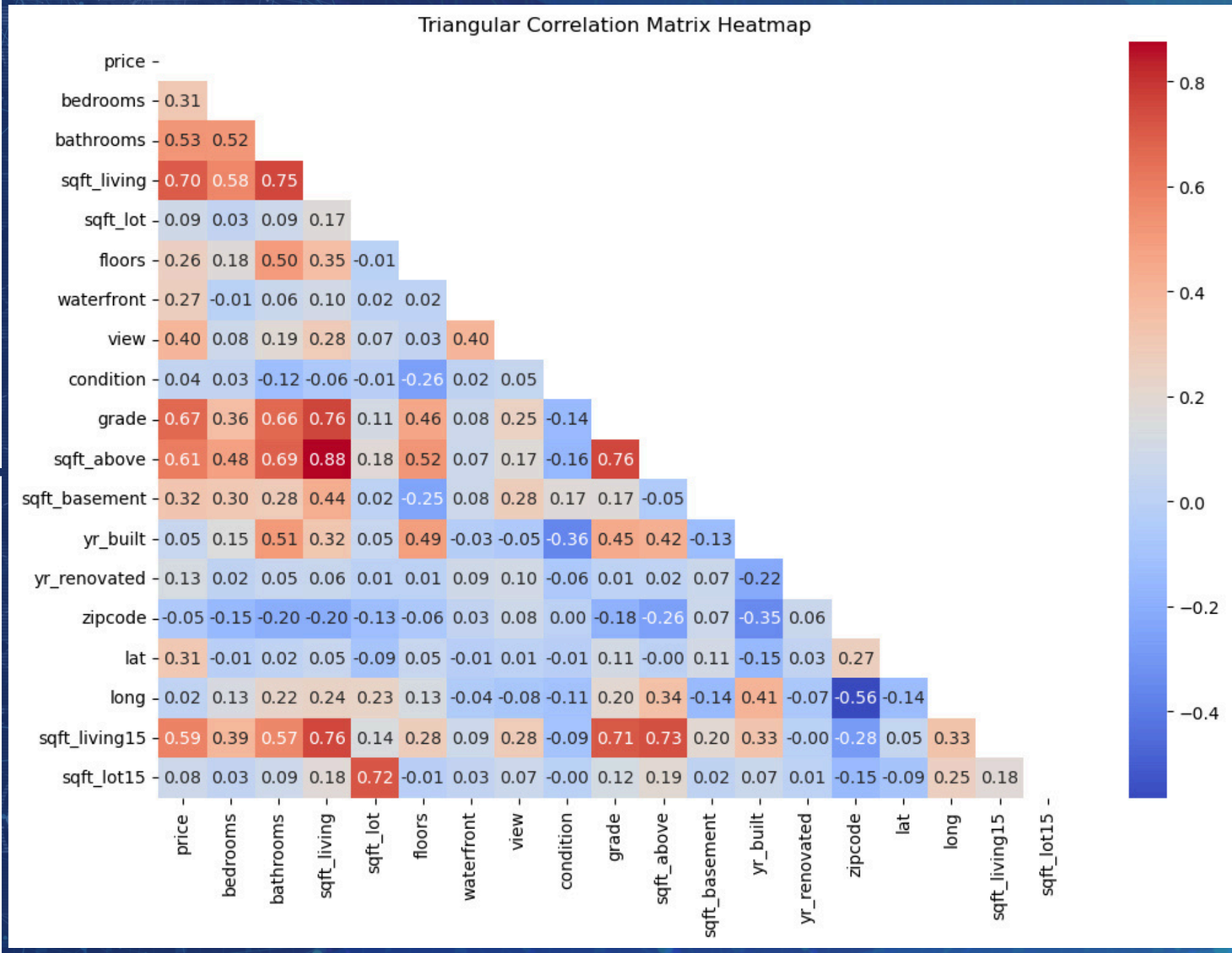
- Represent a mix of simple **vs**. complex approaches;
- Allow us to compare ensemble boosting techniques;
- Widely used and proven effective for house price prediction.

Data Preprocessing with EDA

- **Checked data quality** → handled missing values, duplicates, and outliers;
- **Explored distributions** → within specific features ;eg date
- **Correlation analysis** → heatmap to identify relationships with price;
- **Feature preparation** → selected key variables, transformed where needed;
- **Train/test split** → 80/20 for model evaluation.


Average House Price by Day of Week
Average Price
0
100000
200000
300000
400000
500000
0
1
2
3
4
5
6
Day of Week (0=Mon, 6=Sun)


Average House Price by Year
Average Price
539500
540000
540500
541000
541500
542000
2014.0
2014.2
2014.4
2014.6
2014.8
2015.0
Year


Triangular Correlation Matrix Heatmap
price
bedrooms 0.31
bathrooms 0.53 0.52
sqft_living 0.70 0.58 0.75
sqft_lot 0.09 0.03 0.09 0.17
floors 0.26 0.18 0.50 0.35 -0.01
waterfront 0.27 -0.01 0.06 0.10 0.02 0.02
view 0.40 0.08 0.19 0.28 0.07 0.03 0.40
condition 0.04 0.03 -0.12 -0.06 -0.01 -0.26 0.02 0.05
grade 0.67 0.36 0.66 0.76 0.11 0.46 0.08 0.25 -0.14
sqft_above 0.61 0.48 0.69 0.88 0.18 0.52 0.07 0.17 -0.16 0.76
sqft_basement 0.32 0.30 0.28 0.44 0.02 -0.25 0.08 0.28 0.17 0.17 -0.05
yr_built 0.05 0.15 0.51 0.32 0.05 0.49 -0.03 -0.05 -0.36 0.45 0.42 -0.13
yr_renovated 0.13 0.02 0.05 0.06 0.01 0.01 0.09 0.10 -0.06 0.01 0.02 0.07 -0.22
zipcode -0.05 -0.15 -0.20 -0.20 -0.13 -0.06 0.03 0.08 0.00 -0.18 -0.26 0.07 -0.35 0.06
lat 0.31 -0.01 0.02 0.05 -0.09 0.05 -0.01 0.01 -0.01 0.11 -0.00 0.11 -0.15 0.03 0.27
long 0.02 0.13 0.22 0.24 0.23 0.13 -0.04 -0.08 -0.11 0.20 0.34 -0.14 0.41 -0.07 -0.56 -0.14
sqft_living15 0.59 0.39 0.57 0.76 0.14 0.28 0.09 0.28 -0.09 0.71 0.73 0.20 0.33 -0.00 -0.28 0.05 0.33
sqft_lot15 0.08 0.03 0.09 0.18 0.72 -0.01 0.03 0.07 -0.00 0.12 0.19 0.02 0.07 0.01 -0.15 -0.09 0.25 0.18
price
bedrooms
bathrooms
sqft_living
sqft_lot
floors
waterfront
view
condition
grade
sqft_above
sqft_basement
yr_built
yr_renovated
zipcode
lat
long
sqft_living15
sqft_lot15
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

Models and Improvements

Linear Regression Raw Data

- *Simple and interpretable baseline model*
- *Assumes a linear relationship between features and price*

R^2 Train: *0.70* ***MSE Train:*** *4.13e+10* ***RMSE Train:*** *203268.61*
R^2 Test*: 0.69* ***MSE Test:*** *3.61e+10* ***RMSE Test:*** *190053.61*

Linear Regression Standardization

- *Simple and interpretable baseline model*
- *Assumes a linear relationship between features and price*

R^2 Train: *0.70* ***MSE Train:*** *4.13e+10* ***RMSE Train:*** *203268.61*
R^2 Test*: 0.69* ***MSE Test:*** *3.61e+10* ***RMSE Test:*** *190053.61*

next slide →

Models and Improvements

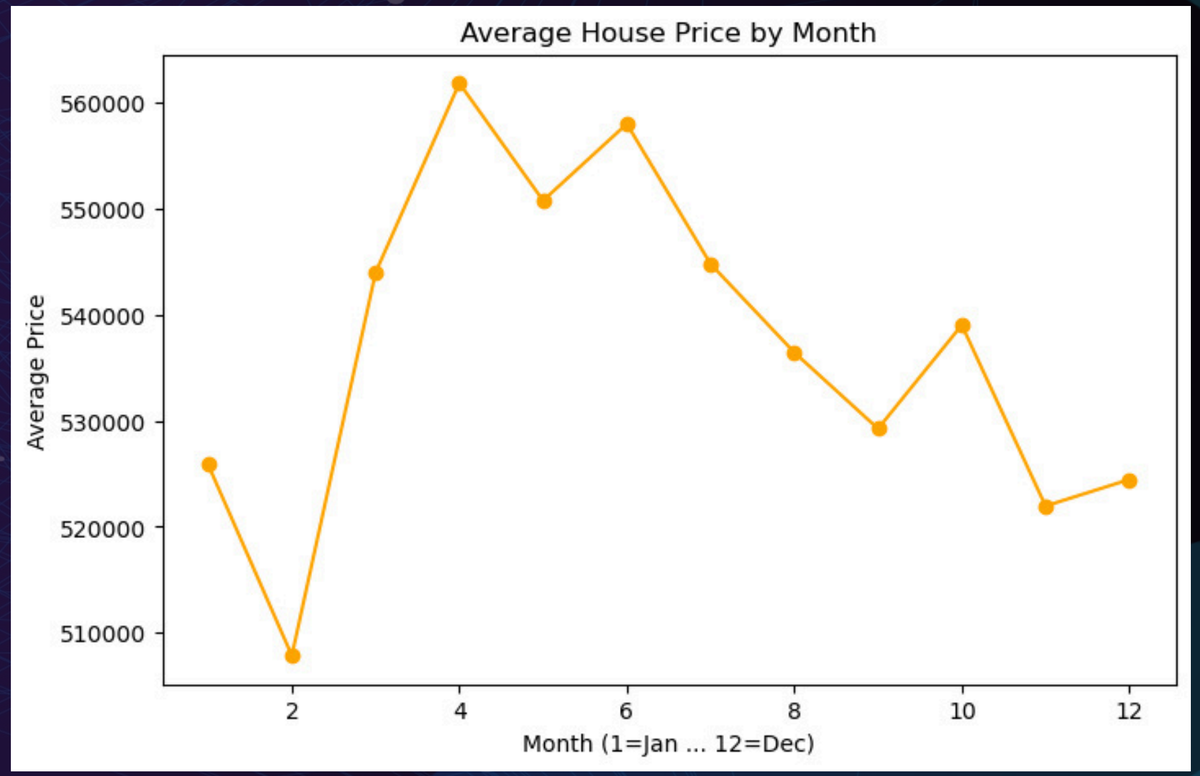
Linear Regression Dropped Columns
R^2 Train: *0.65* ***R^2 Test:*** *0.64*

Linear Regression Feature engineering
R^2 Train: *0.65* ***R^2 Test:*** *0.64*

Ridge Regression

- *Tested with L2 Regularization to see if the model would perform improve*
- *Ridge slightly improved the overall fit*

R^2 Train: *0.70* ***R^2 Test:*** *0.70*



next slide →

Models and Improvements

Random Forest Regressor

- *Ensemble of many decision trees trained on random samples*
- *Reduces variance and captures non-linear relationships*

R^2 Train: *0.98* ***MSE Train:*** *2.195 e+6* ***RMSE Train:*** *46566.51*

R^2 Test*: 0.85* ***MSE Test:*** *2.17 e+10* ***RMSE Test:*** *132394.27*

Random Forest Regressor Improved

- *Improved using randomized seach*
- *Improved overfitting*

R^2 Train: *0.93* ***MSE Train:*** *2.16* ***RMSE Train:*** *46566.51*

R^2 Test: *0.86* ***MSE Test:*** *2.15* ***RMSE Test:*** *146657.14*

next slide →

Models and Improvements

Gradient Boosting Regressor

- *Builds models sequentially, where each new tree corrects the errors of the previous ones*
- *Achieves strong predictive performance by combining many weak learners into one powerful model*

R^2 Train: *0.96* ***MSE Train:*** *4.658 e+9* ***RMSE Train:*** *68,251*

R^2 Test*: 0.88* ***MSE Test:*** *1.759 e+10* ***RMSE Test:*** *132,634*

next slide →

Models and Improvements

- ***XGBoost Regressor***
- *Optimized version of gradient boosting with regularization to prevent overfitting*
- *Highly efficient and scalable, making it one of the fastest boosting algorithms*

R^2 Train: *0.98* ***MSE Train:*** *2.92 e+9* ***RMSE Train:*** *46.858*

R^2 Test*: 0.86* ***MSE Test:*** *2.17 e+9* ***RMSE Test:*** *149.670*

- ***XGBoost Regressor Improved***

R^2 Train: *0.96* ***MSE Train:*** *5.4 e+10* ***RMSE Train:*** *73,739*

R^2 Test*: 0.90* ***MSE Test:*** *1.4 e+9* ***RMSE Test:*** *119,217*

next slide →

Key takeaways

- *Ensemble methods/algorithms provided the highest performances*
- *Ridge Regression slightly improved the linear regression model*
- *Randomizedsearch together with a 70/30 split showed the most improvement on XGBoost it the best model*

next slide →

Feature Importance

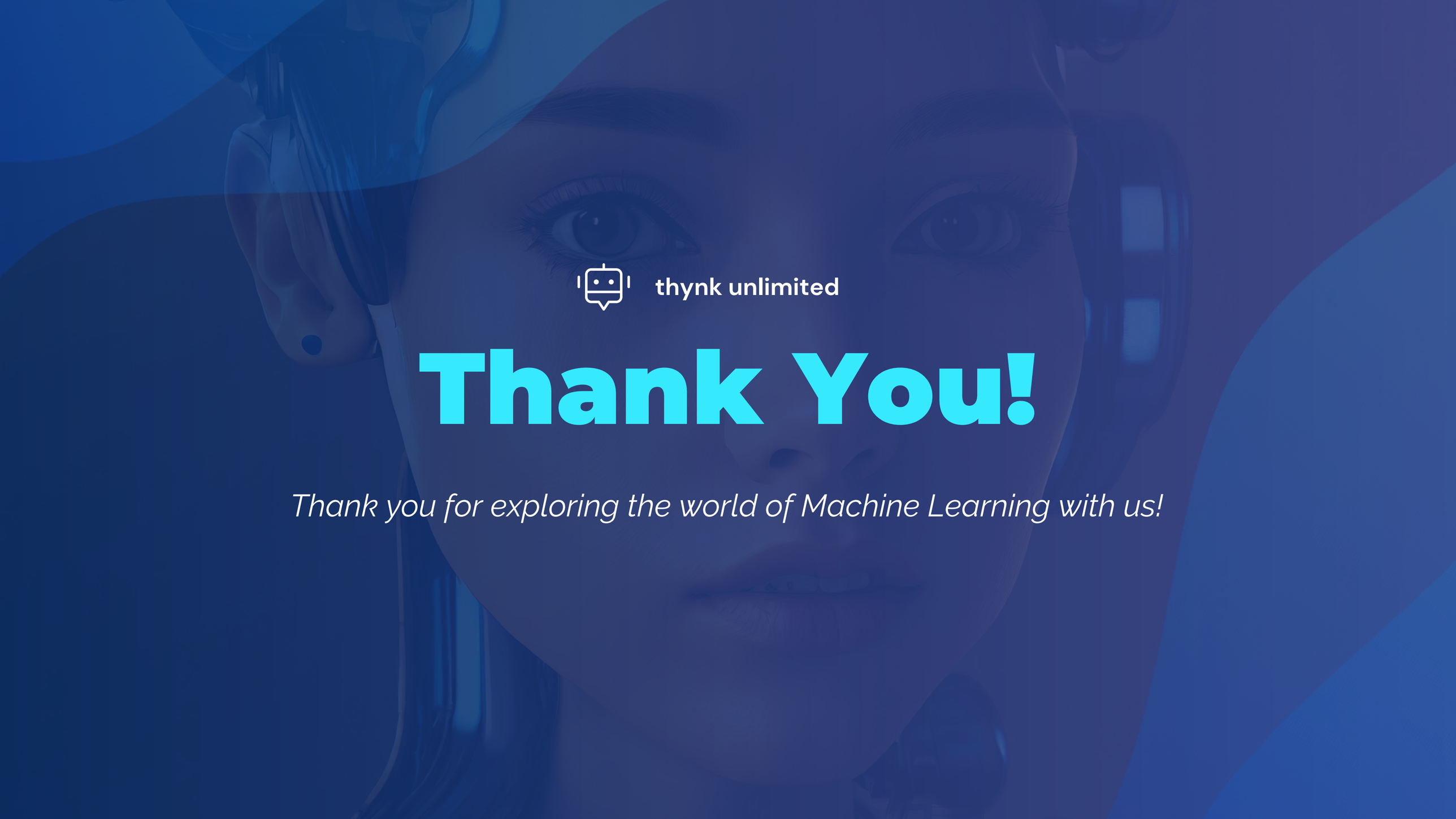
Grade: *the most influential factor — it reflects the overall quality of construction and design, which has the biggest impact on house prices.*

XGBoost *shows that grade is the strongest predictor of house prices, which makes sense since construction quality directly drives market value.*

	Feature	Importance
8	grade	0.307339
2	sqft_living	0.159512
5	waterfront	0.130566
14	lat	0.065442
6	view	0.051300
16	sqft_living15	0.042291
15	long	0.039760
9	sqft_above	0.035293
1	bathrooms	0.028870
11	yr_built	0.028468
13	zipcode	0.021579
12	yr_renovated	0.015638
10	sqft_basement	0.013069
7	condition	0.010922
17	sqft_lot15	0.010587
3	sqft_lot	0.010249
18	year	0.009819
4	floors	0.006701
0	bedrooms	0.004879
19	month	0.004220
20	dayofweek	0.003494

Model Leaderboard (sorted by R^2 Test)

Model	R2 Test	R2 Train	RMSE (Test)	MSE (Test)	RMSE (Train)	MSE (Train)
XGBoost Regressor Optimized	0.9016	0.9584	119,217	14,212,810,647	73,740	5,437,585,990
GradientBoosting Regressor	0.8836	0.9643	132,635	17,592,003,138	68,251	4,658,204,367
XGBoost Regressor	0.8564	0.9776	147,349	21,711,668,500	54,085	2,925,218,639
Random Forest regressor	0.8518	0.9832	149,670	22,401,056,284	46,858	2,195,627,863
RandomForest Regressor Optimized	0.8455	0.9334	152,823	23,354,884,549	93,310	8,706,789,190
Ridge	0.7025	0.7007	212,068	44,972,770,535	197,753	39,106,335,675
Linear Regression standardized data	0.6963	0.7021	190,054	36,120,375,752	203,269	41,318,129,690
Linear Regression raw data	0.6963	0.7021	190,054	36,120,375,752	203,269	41,318,129,690
Linear Regression dropped columns	0.6443	0.6537	205,678	42,303,392,304	219,155	48,028,706,100
Linear Regression feature engineering	0.6443	0.6537	205,678	42,303,392,304	219,155	48,028,706,100


thynk unlimited
Thank You!
Thank you for exploring the world of Machine Learning with us!