Machine Learning Model To Predict **California Housing Prices**

Name: Le Nguyen Ngoc Yen
Class: TC-DA32
Lecturer: Tran Thi Tham

Contents

01

About the dataset and project objective

1. About the dataset - General info

Dataset: This dataset is taken from Kaggle and is originated from the second chapter of Aurélien Géron's recent book 'Hands-On Machine learning with Scikit-Learn and TensorFlow'. This dataset has 20,640 rows and 10 columns.
Project objective: Build a model to predict housing prices in California based on some features for a real estate agency in California, the US.

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value	ocean_proximity
17400	-120.44	34.93	15.0	868.0	244.0	1133.0	253.0	2.0995	87500.0	<1H OCEAN
280	-122.18	37.80	34.0	1355.0	195.0	442.0	195.0	6.2838	318200.0	NEAR BAY
14386	-117.23	32.75	5.0	1824.0	NaN	892.0	426.0	3.4286	137500.0	NEAR OCEAN
17848	-121.86	37.42	20.0	5032.0	808.0	2695.0	801.0	6.6227	264800.0	<1H OCEAN
4592	-118.27	34.05	25.0	1316.0	836.0	2796.0	784.0	1.7866	325000.0	<1H OCEAN

1. About the dataset - Data dictionary

DATA DICTIONARY			
column	**meaning**	**value**	**role**
longitude	a measure of how far West a house is; a higher value is farther West	float	feature
latitude	a measure of how far North a house is; a higher value is farther North	float	feature
housing_median_age	median age of a house within a block; a lower number is a newer building	float	feature
total_rooms	total number of rooms within a block	int	feature
total_bedrooms	total number of bedrooms within a block	int	feature
population	total number of people residing within a block	int	feature
households	total number of households, a group of people residing within a home unit, for a block	int	feature
median_income	median income for households within a block of houses (measured in USD 10,000)	float	feature
median_house_value	median house value for households within a block (measured in USD)	float	target
ocean_proximity	location of the house wrt ocean/sea	INLAND: situated in the interior of the country, far from the coast < 1H OCEAN: 1 hour away from the ocean NEAR BAY --> a bay is a recessed, coastal body of water (often smaller than ocean) NEAR OCEAN ISLAND: situated on a piece of land surrounded by water	feature

1. About the project - Processes

Step 1

Data understanding

Step 2

Data pre-processing

Check and process null values
Check duplicates
Change false data types
Check outliers
Encode `object` data

Step 3

Create and train models

Linear Regression
Random Forest
XGBoost

Step 4

Evaluate, validate and improve models

Compare the r2_score, MAE and RMSE of three models
k-fold cross validation
Hyperparameter tuning through OPTUNA

Step 5

Select the final model

Step 6

Check feature importance

Check which features influence the model's decision the most through SHAP
→ Further refine the model

02

Data understanding and pre-processing

2.1 Data understanding

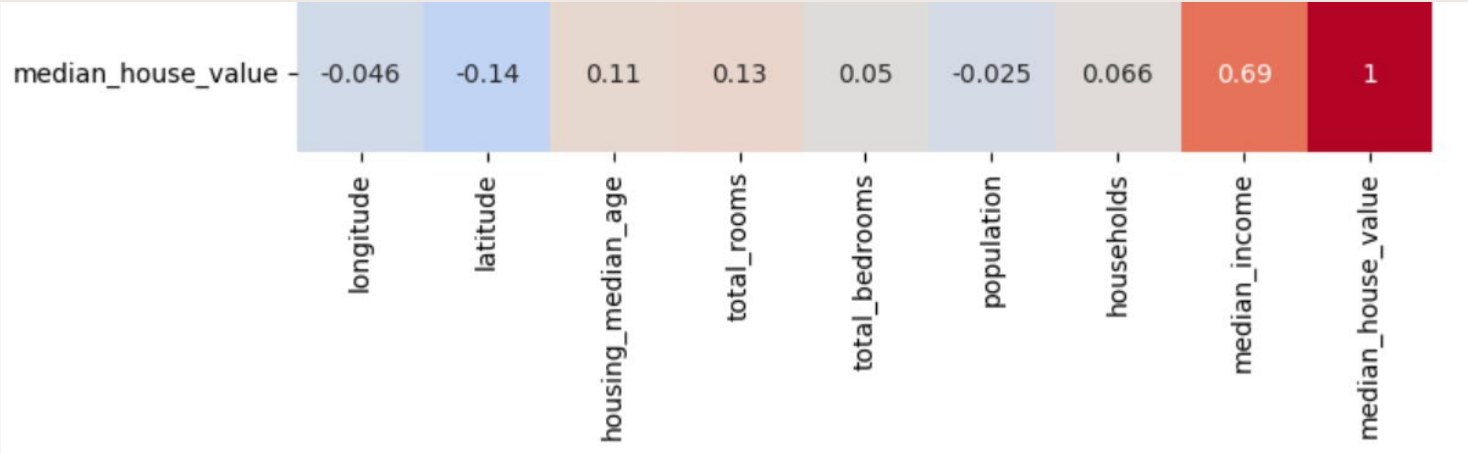
General observations:

- The average housing age of the dataset is 29
- Most houses are located 1 hour away from the ocean (9136)
- The median house value is $206,856
- `total_bedrooms` is the only column with null values

2.1 Data understanding

Corellations:

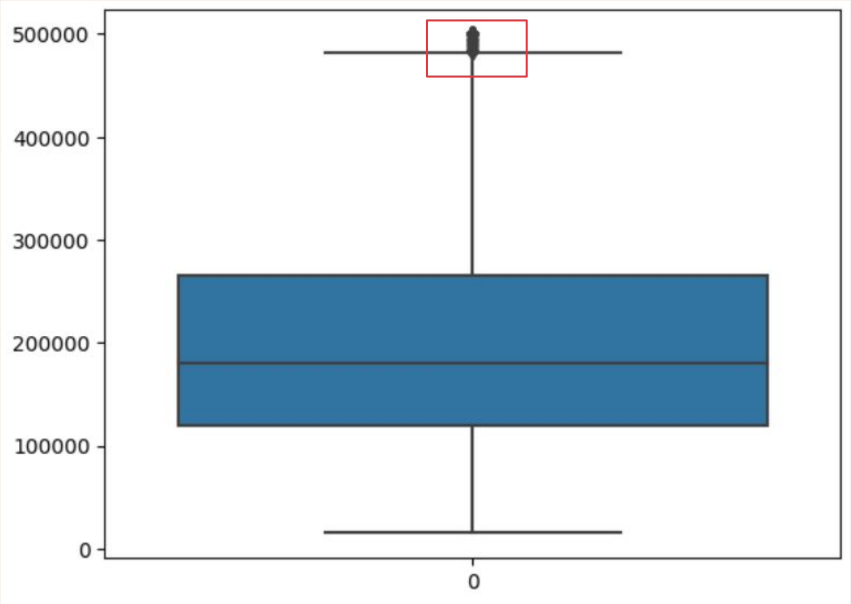
- median_income has a significantly high correlation with median_house_value
- housing_median_age and total_rooms has a moderate correlation with median_house_value

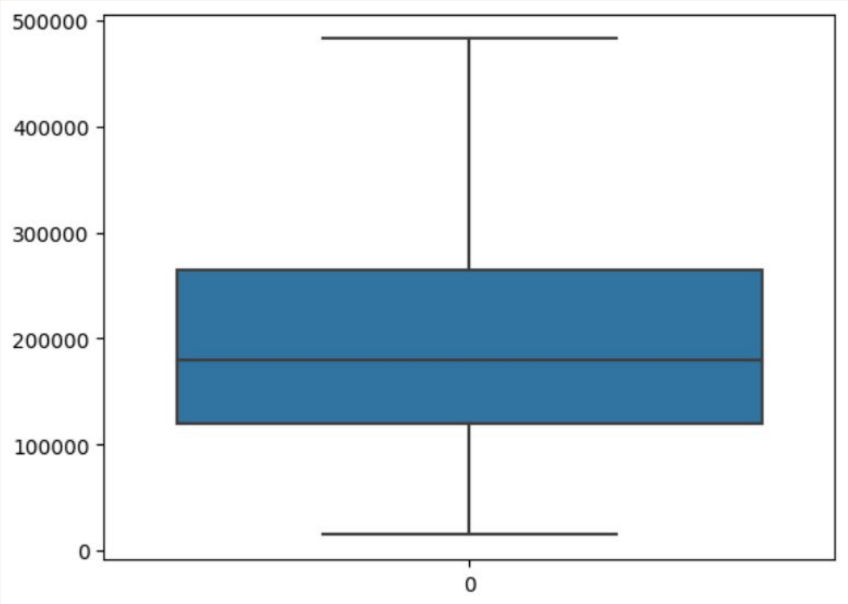


2.2 Data pre-processing

Problem	Presence	Action
Duplicates	No	NA
Null/Missing values	Yes - `total_bedrooms` has 207 null values	Fill the null values with mode of the column
False data types	Yes - 4 columns	Change from ‘float’ to ‘int’
Outliers	Yes - `median_house_value` has some outliers	Use box plot to determine the outliers and use capping method to change those values

2.2 Data pre-processing - Outliers





Before

After

2.2 Data pre-processing

Problem	Presence	Action
Duplicates	No	NA
Null/Missing values	Yes - `total_bedrooms` has 207 null values	Fill the null values with mode of the column
False data types	Yes - 4 columns	Change from ‘float’ to ‘int’
Outliers	Yes - `median_house_value` has some outliers	Use box plot to determine the outliers and use capping method to change those values
Data encoding	Yes - `ocean_proximity`	Encode manually based on the idea of ordinal encoding

2.2 Data pre-processing - Cleaned data

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value	ocean_proximity_new
4697	-118.37	34.07	52.0	1084	247	468	255	3.4286	474300.0	4
19629	-120.84	37.53	14.0	3643	706	2070	697	3.1523	141800.0	5
2125	-119.71	36.79	34.0	1891	323	966	355	3.6681	82000.0	5
15702	-122.44	37.79	52.0	3785	808	1371	799	6.4209	482412.5	3
2105	-119.76	36.75	39.0	2233	563	2031	491	1.8641	50800.0	5

03

Machine learning models selection and evaluation

3.1 Create and train models

Define features/ target

- Target: `median_house_value`
- Features: Other columns

Train-test split

The data is split into 70:30 for train-test.

Scale data

The data is then scaled with StandardScaler to improve model's interpretability and ensure the model is not influenced by the magnitude of the features.

3.2 Evaluation

Model	Train performance		Test performance		Degree of fit
	r2_score	RMSE	r2_score	RMSE	
Linear Regression	0.64	68171.03	0.63	68187.84	Slightly underfitting
Random Forest Regressor	0.97	18125.76	0.82	47641.34	Goodfitting
XGBoost Regressor	0.94	27404.41	0.83	46310.13	Goodfitting

3.3 k-fold cross validation (k=10)

Model	**r2_score**		**Average r2_score**	**Standard deviation**
	Min	Max		
Linear Regression	0.58	0.67	0.63	2.6%
Random Forest Regressor	0.78	0.84	0.81	1.9%
XGBoost Regressor	0.79	0.85	0.82	1.7%

3.4 Hyperparameter tuning (OPTUNA)

Model	Set 1		Set 2		Set 3	
	r2_score	RMSE	r2_score	RMSE	r2_score	RMSE
Random Forest Regressor	0.82	47581.83	0.82	48057.01	0.82	47638.26
XGBoost Regressor	0.85	44256.43	0.83	46493.86	0.84	45531.57

3.5 Model selection

Number of trials:

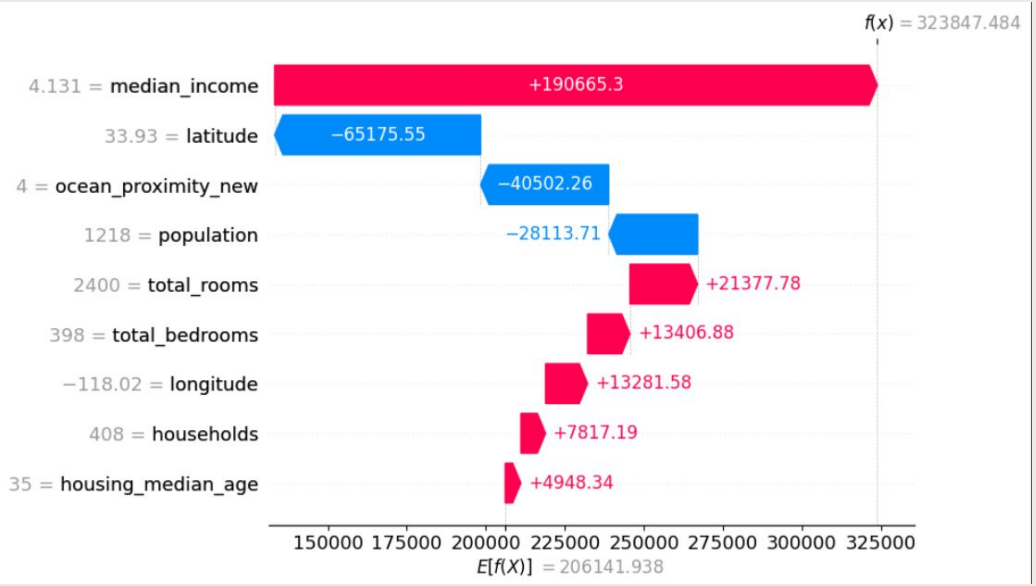
- Random Forest Regressor: 20 time (single objective) and 10 times (multi-objective)
- XGBoost Regressor: up to 100 times (for both single and multi-objective)

Consistency and improvement:

- Random Forest: the performance is not consistent; r2 score improves by a little while RMSE still stays the same or even increases
- XGBoost: the performance is more consistent; r2 score improves and RMSE reduces by about 1000 - 2000

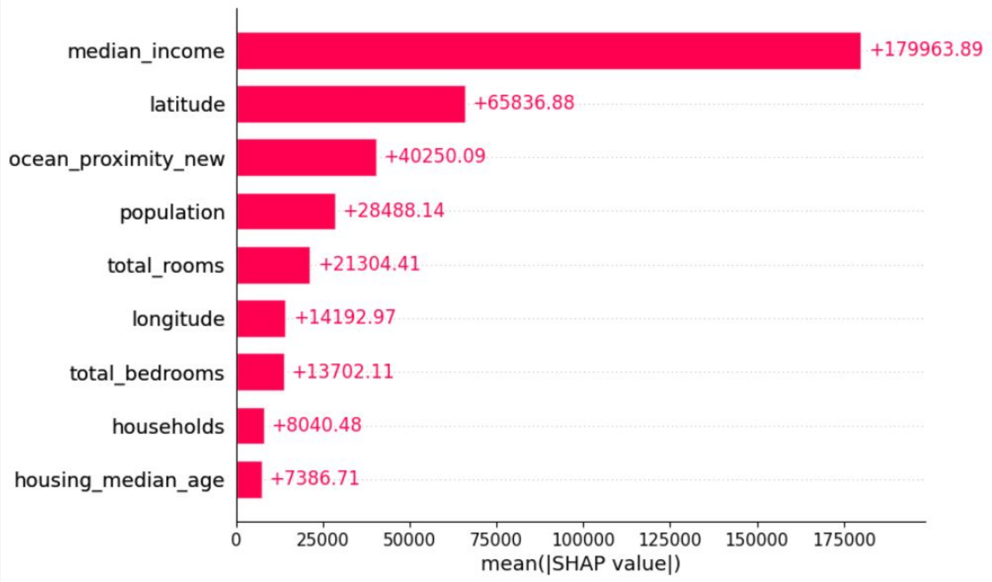
→ XGBoost Regressor model will be selected for further improvement.

3.6 Feature importance analysis (SHAP)



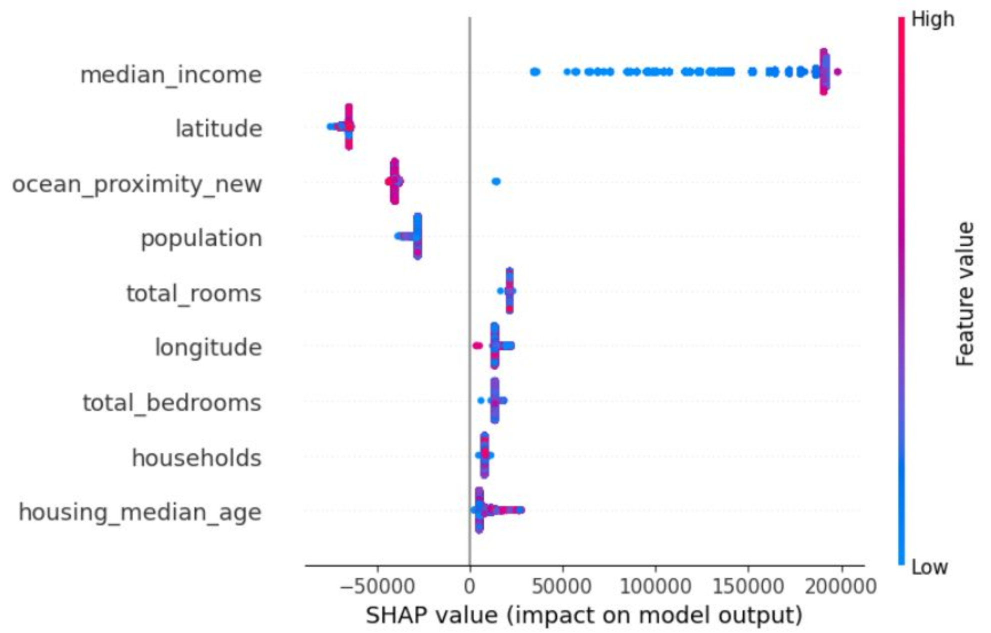
In this waterfall plot, it can be observed that `median_income` positively contributes to the prediction and increases it by a large amount. Meanwhile, `latitude` and `ocean_proximity` negatively contribute to the prediction and decrease it by a certain amount.

3.6 Feature importance analysis (SHAP)



According to the aboslute mean SHAP plot, `median_income`, `latitude` and `ocean_proximity` are the three most important features affecting the decision of this model.

3.6 Feature importance analysis (SHAP)



In this plot, only the SHAP values of `median_income` and `housing_median_age` can be clearly inferred. Both features are positively correlated with the target variable (the higher the feature values, the higher the SHAP values).

3.7 Re-train the model with fewer features

- **Model**:

The 3 least important features are dropped and XGBRegressor model is re-trained with one of the set of tuned parameters above.

- **Result**:

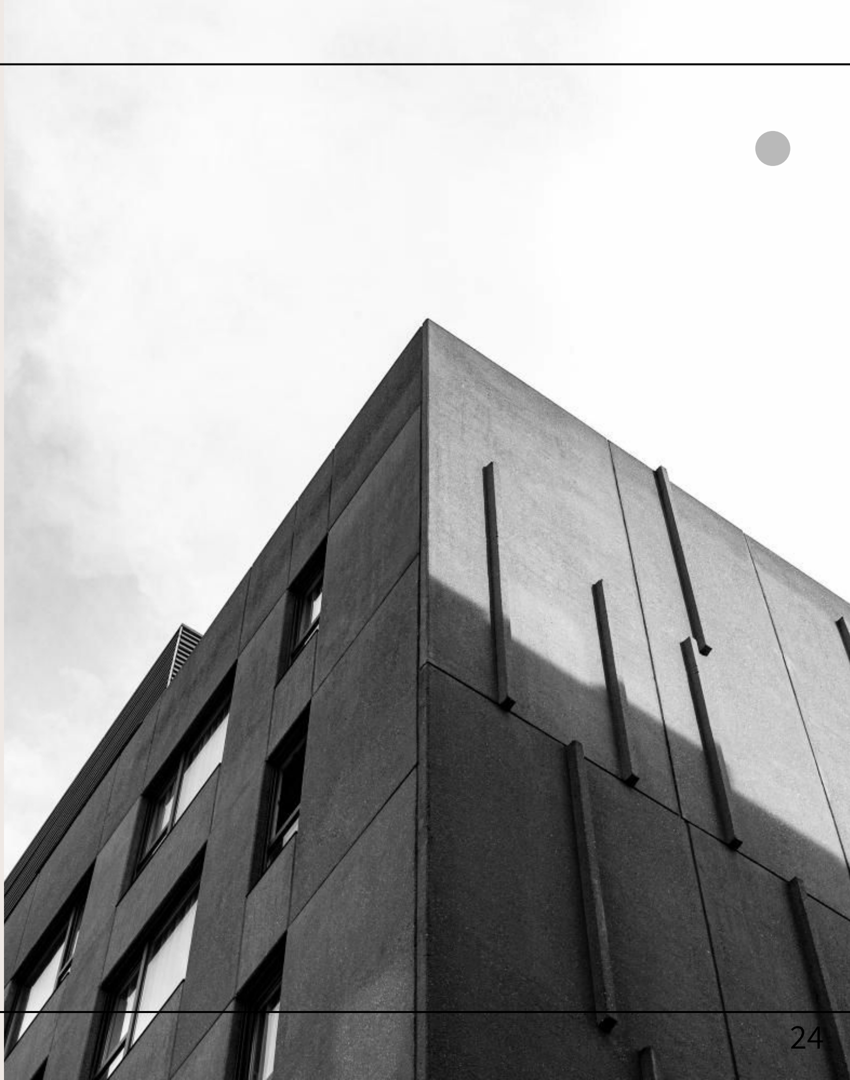
r2 score and RMSE value seem to not change much

→ dropping these features seem to have little effect on the model

Model	**With 9 features**		**With 6 features**	
	r2_score	RMSE	r2_score	RMSE
XGBoost Regressor	0.85	44256.43	0.84	44552.56

04

Conclusion and future directions



4.1 Conclusion

- XGBRegressor is the most optimal model amongst the three models trained (Linear Regression, Random Forest and XGBoost).
- Dropping features do not affect the results much → all the features should be kept for better model's decision.

4.2 Future directions

- More models can be tested to see if they produce a higher r2 score and lower RMSE value (NBGaussian, KNN and SVM).
- More trials of hyperparameter tuning through OPTUNA can be done on Random Forest model to see if there are any better sets of parameters.
- The features provided in this dataset are quite limited. Feature engineering can be applied to extract more relevant features for housing price prediction such as area of houses, proximity to public transport, amenities and crime rate.

References

- Abid, Awan A. "An Introduction to SHAP Values and Machine Learning Interpretability." Learn Data Science and AI Online | DataCamp, June 2023, www.datacamp.com/tutorial/introduction-to-shap-values-machine-learning-interpretability.
- Magaga, Alamin M. "Identifying, Cleaning and Replacing Outliers | Titanic Dataset." Medium, 12 Nov. 2021, medium.com/analytics-vidhya/identifying-cleaning-and-replacing-outliers-titanic-dataset-20182a062893.
- OPTUNA. "Multi-objective Optimization with Optuna — Optuna 3.4.0 Documentation." Optuna: A Hyperparameter Optimization Framework — Optuna 3.4.0 Documentation, optuna.readthedocs.io/en/stable/tutorial/20_recipes/002_multi_objective.html#sphx-glr-tutorial-20-recipes-002-multi-objective-py.
- Shin, Terence. "Understanding Feature Importance and How to Implement It in Python." Medium, 10 Nov. 2022, towardsdatascience.com/understanding-feature-importance-and-how-to-implement-it-in-python-ff0287b20285.

Thank you for listening!