Zillow®

Estimating home prices using ML

Hussain Burhani

Lightning points

- Gathering, cleaning, and exploring historic data
- Data categories
- Feature selection
- Fit, predict, evaluate
- Future improvement

Linear regression process

- Load dataset
- Null and bad values
- Ordinal and categorical

- Train / test split
- Feature selection

- Instantiate
- Fit and predict
- Evaluate

Data categories

Numeric

Ordinary values

- Continuous or discrete
- Sq. feet, bedrooms
- **As-is**

Nominal

Discrete number of categories

- Numeric or non-numeric
- Zoning, type of dwelling
- **Dummify**

Ordinal

Ordering matters

- Numeric or non-numeric
- Lot Shape, overall condition
- **Ordered map**

Dummifying and ordered mapping

Nominal

```
df['Roof Style'].value_counts()
```

```
Gable      1619
Hip         397
Flat         13
Gambrel      12
Mansard       7
Shed          3
Name: Roof Style, dtype: int64
```

Dummify

	Gable	Gambrel	Hip	Mansard	Shed
0	1	0	0	0	0
1	1	0	0	0	0
2	1	0	0	0	0
3	1	0	0	0	0
4	1	0	0	0	0
5	1	0	0	0	0
6	1	0	0	0	0
7	0	0	1	0	0

Ordinal

```
df['Lot Shape'].value_counts()
```

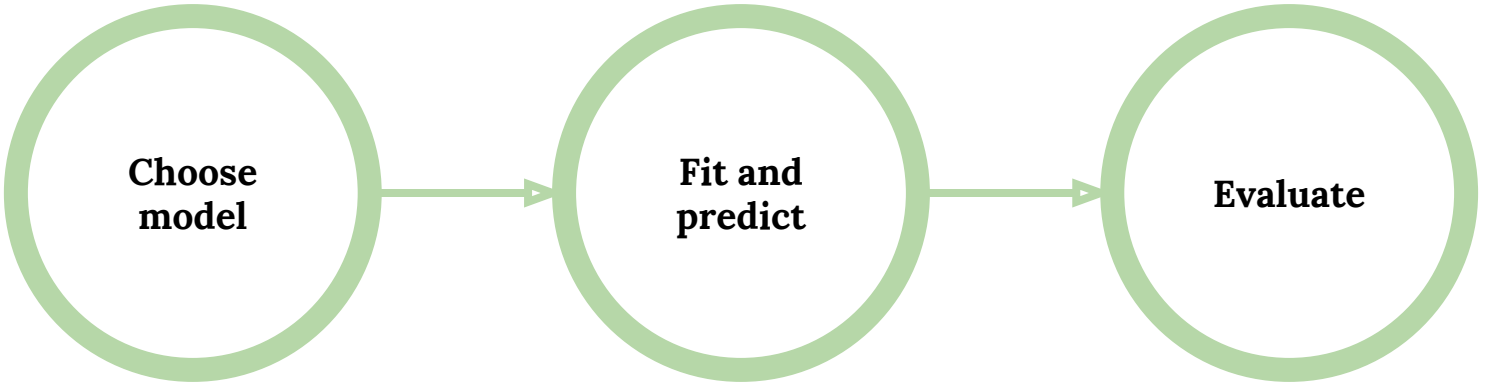
```
Reg    1295
IR1     692
IR2      55
IR3       9
Name: Lot Shape, dtype: int64
```

Ordered mapping

```
'Lot Shape': {
    np.nan: 0,
    'Reg': 1,
    'IR1': 2,
    'IR2': 3,
    'IR3': 4
    },
```

Feature selection

Model, fit, predict, evaluate



- KFold
- LassoCV
- alphas=np.logspace(0, 5, 200)

- StandardScaler
- PolynomialFeatures

- Compare train and test
- R-squared, RMSE
- Delta (train / test)

Future improvement

Tuning

- GridSearchCV

Feature Engineering

- Log transformation
- Imputing via kNN
- PCA

Modelling

- RandomForest
- XGBoost
- Neural Networks

Insights from today