wbs
WARWICK BUSINESS SCHOOL
THE UNIVERSITY OF WARWICK

For the Change Makers

Advanced Programming for Data Science

Week 8: Data Analysis and Modeling
Information Systems and Management
Warwick Business School

Supervised Learning: Regression

Supervised Learning: Regression

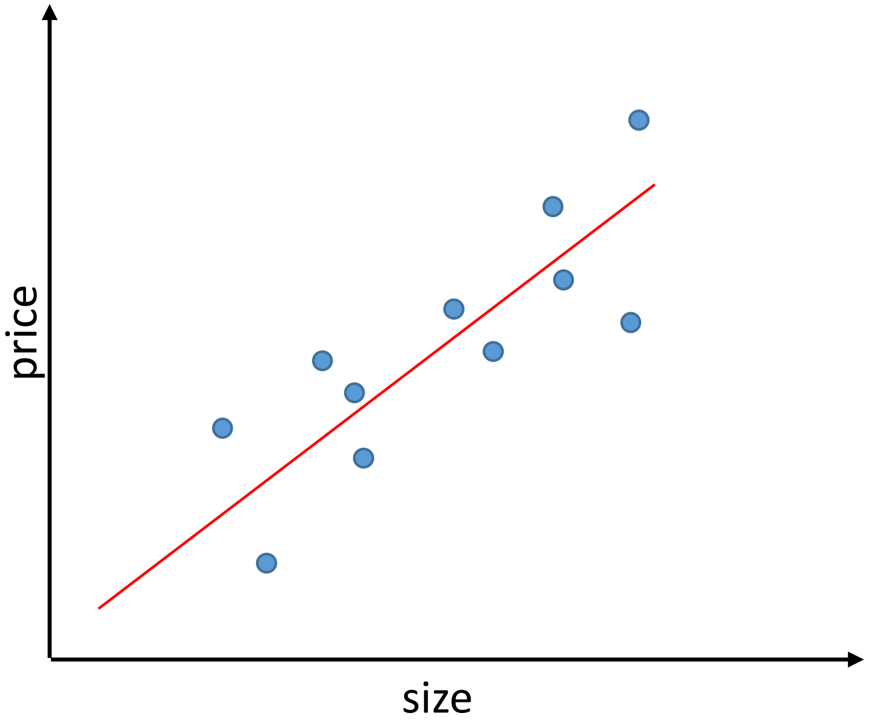
- Regression is a type of supervised learning to predict continuous value outputs, such as house price, instead of discrete categories.
  - ➢Linear regression (simple/multiple)
  - ➢Regression Tree
  - ➢Lasso
  - ➢SVM
  - ➢Etc.

Linear Regression

- Linear regression describes linear relationships between the inputs and output.

$$y = w_0 + w_1 \cdot x + \varepsilon$$

$$y = w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 + \cdots + w_3 \cdot x_3 + \varepsilon$$



Linear regression with SKLearn

```
# import linear_model from sklearn
from sklearn import linear_model
# instantizing LinearRegression learner
reg = linear_model.LinearRegression()
# training the model
reg.fit(X_train,y_train)
# making prediction
y_pred = reg.predict(X_test)
```

Understand your regressor

You can check the linear model's intercept and coefficient by the following attributes:

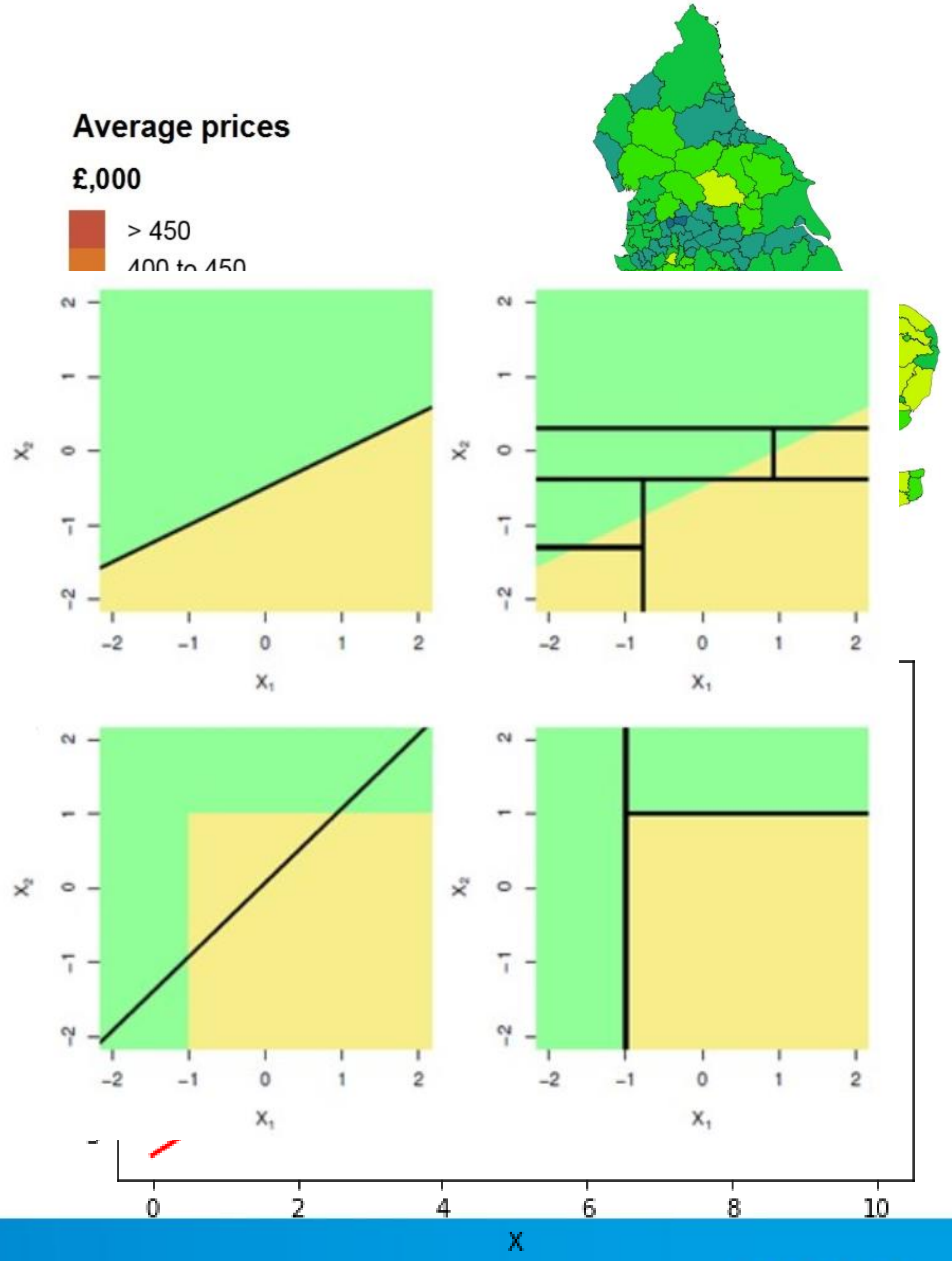
- reg.intercept_
- reg.coef_

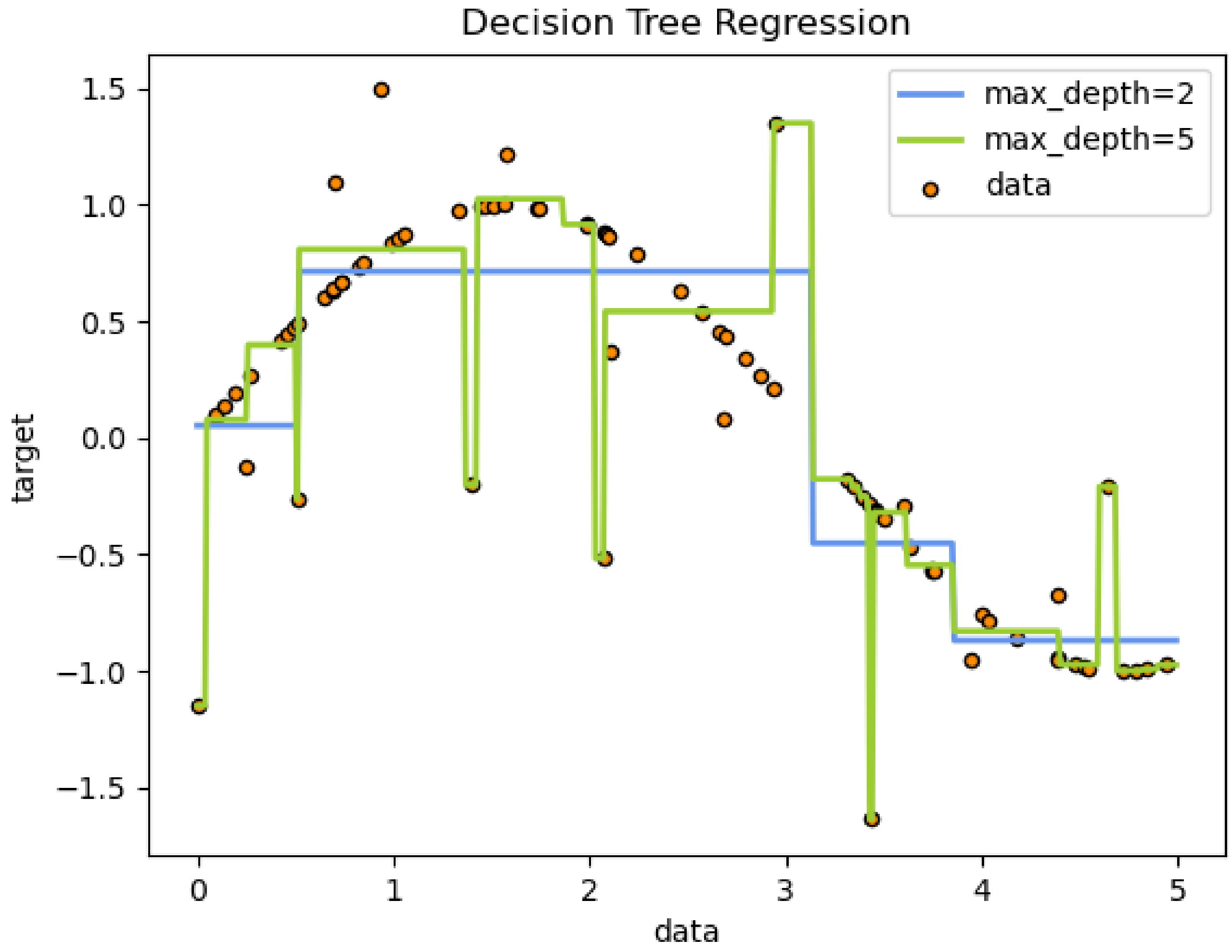
Regression Trees

- Decision tree can also be used for regression problems.
- In SKLearn, you can find corresponding regressors to classifiers, and they can be used in similar manner.
  - DecisionTreeClassifier() -> DecisionTreeRegressor()
  - RandomForestClassifier() -> RandomForestRegressor()
  - AdaBoostClassifier() -> AdaBoostClassifier()
  - GradientBoostingClassifier() -> GradientBoostingRegressor()
- Criterion is changed to gini/entropy to mse, etc.

Tree vs. Linear Model

- Approximated by a Linear relationship?
- Categorical features?
- Collinearity?




Decision Tree Regression
max_depth=2
max_depth=5
data
target
data
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
1
2
3
4
5

Model selection for regression

- Metrics
  1. MAE (Mean Absolute Error): mean of the absolute value of the errors.

$$MAE = \frac{1}{n}\sum_{j=1}^{n} |yj - \hat{y}_j|$$

  2. MSE (Mean Squared Error): mean of the squared errors.

$$MSE = \frac{1}{n}\sum_{j=1}^{n} (yj - \hat{y}_j)^2$$

  3. RMSE (Root Mean Squared Error): the square root of the mean of the squared errors.

$$RMSE = \sqrt{\frac{1}{n}\sum_{j=1}^{n} (yj - \hat{y}_j)^2}$$

Evaluating Regression with SKLearn

```
# import metrics from sklearn
from sklearn import metrics
# MAE
metrics.mean_absolute_error(y_test, y_pred)
# MSE
metrics.mean_squared_error(y_test, y_pred)
# RMSE
np.sqrt(metrics.mean_squared_error(y_test, y_pred))
```

Exercise

- Following our previous exercises on house_sale dataset.
- Import the data as df_house.
- Remove rows with more than 1 missing values from df_house.
- Remove columns with more than 33% records as missing values from you df_house.

Exercise

- Impute missing data with appropriate transformers from sklearn
- Explore your data and scale your data based on your observation.
- Create a new column called 'PriceGroup' by cutting the SalePrice into three groups, below 125000, between 125000 and 200000, and above 200000.

Exercise

- Create new dataframe called y1 using column 'PriceGroup', y2 using column 'SalePrice', and X using all other columns.
- Convert categorical variables in X with one-hot encoding, by sklearn's onehotencoder and pandas' get_dummies.

Exercise

- Build a basic decision tree model with data X and y1.
- *Build a regression model with data X and y2. You can try either linear regression or tree regression.