Machine Learning

Regression analysis

Fernando Rodríguez Sánchez

Computational Intelligence Group

Universidad Politécnica de Madrid

27/01/2020

Introduction

	X_1	$\dots$	X_n	Y
$(\mathbf{x}^{(1)}, y^{(1)})$	$x_1^{(1)}$	$\dots$	$x_n^{(1)}$	$y^{(1)}$
$(\mathbf{x}^{(2)}, y^{(2)})$	$x_1^{(2)}$	$\dots$	$x_n^{(2)}$	$y^{(2)}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$(\mathbf{x}^{(m)}, y^{(m)})$	$x_1^{(m)}$	$\dots$	$x_n^{(m)}$	$y^{(m)}$

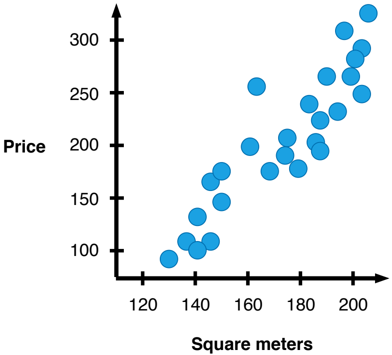
Objective: To find a function $f(x)$ that correctly predicts the value of y given x

$$f(x) \to \hat{y}^{(i)} \approx y(i)$$

Introduction

Objective: To find a function $f(x)$ that correctly predicts the Price of the house given the number of squared metres

$$f \rightarrow \hat{Price}^{(i)} \approx Price(i)$$



Linear regression (one variable)

One variable:

$f(x) = \beta_0 + \beta_1 x$

- $\beta_0 \rightarrow$ function's intercept
- $\beta_1 \rightarrow$ function's slope

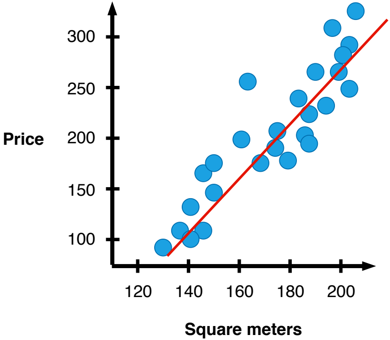


Figure 1: Linear regression with one variable

Linear regression (multiple variables)

2 variables:

$f(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$

n **variables**

$f(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$

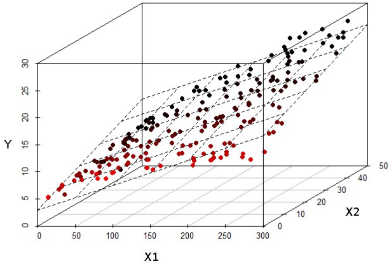


Figure 2: Linear regression with two variables

Loss function

How can we know the goodness of $f(x)$?

$$\downarrow$$

$$\mathcal{L}(\hat{y}, y) = (\hat{y} - y)^2$$

Determine the values of β_0, β_1, ..., β_n that **minimize** $\mathcal{L}(\hat{y}, y)$

Linear regression:

- Gradient descent
- Normal equations

Extensions of linear regression

X can be transformed to allow linear regression techniques to fit much more complicated datasets

- Polynomial transformation

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3^3$$

2 new columns are generated from $X \rightarrow X^2$, X^3

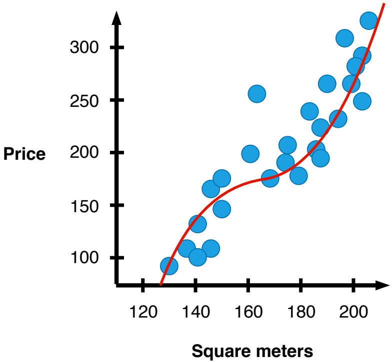
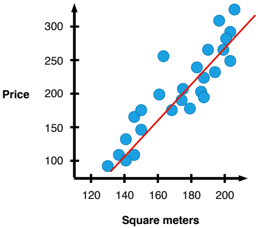
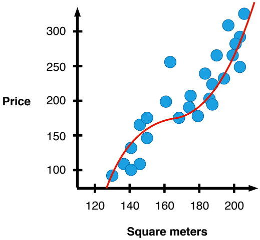


Figure 3: Polynomial regression with "one" variable

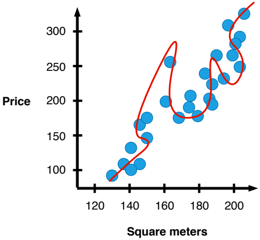
Underfitting vs Overfitting



Underfit. High bias



"Just right"



Overfit. High variance

Overfitting: If our model is too complex, the learned function may fit the training set very well ($J(\hat{y}, y) \approx 0$), but **fail to generalize** to new examples (predict prices of new houses)

Addressing overfitting

1. **Reduce the number of features**
   - Manual selection
   - With an algorithm
2. **Regularization**
   - Keep all the features, but reduce magnitude/values of β_i
   - Two types: **L1** (lasso), **L2** (ridge)

Machine Learning

Regression analysis

Fernando Rodríguez Sánchez

Computational Intelligence Group

Universidad Politécnica de Madrid

27/01/2020