Machine Learning

Regression analysis

Fernando Rodríguez Sánchez

ferjorosa@gmail.com

Universidad Politécnica de Madrid

16/10/2020

Table of contents

Table of contents

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 $f(\mathbf{x})$ that correctly predicts the value of y given $\mathbf{x}$

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

Introduction

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

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

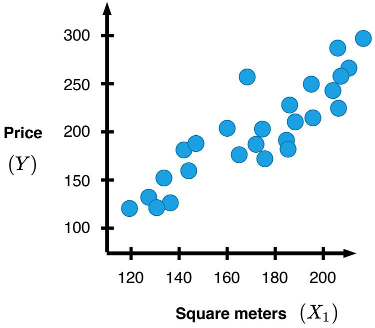


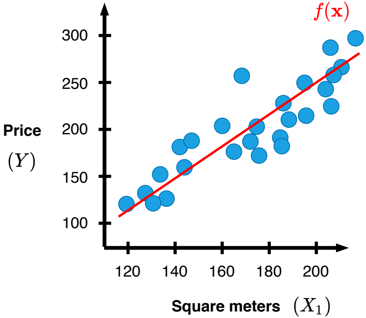
Table of contents

Linear regression (one variable)

One variable:

$f(\mathbf{x}) = \beta_0 + \beta_1 x_1$

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



Linear regression with one variable

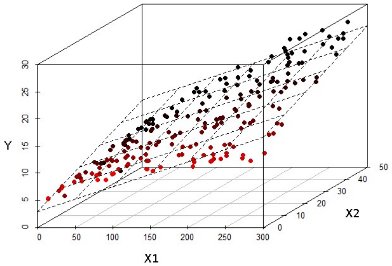
Linear regression (multiple variables)

2 variables:

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

n variables

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



Linear regression with two variables

Loss function

How can we know the goodness of $f(\mathrm{x})$?

$$\downarrow$$

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

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

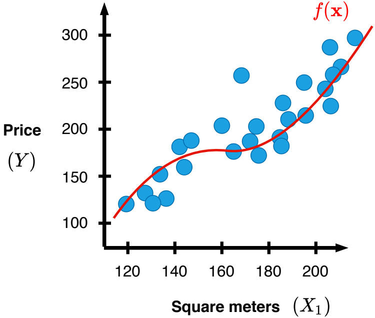
Square meters (X_1) can be transformed to allow linear regression techniques to fit much more complicated datasets

- Polynomial transformation

$$f(\mathbf{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_1^2 + \beta_3 x_1^3$$

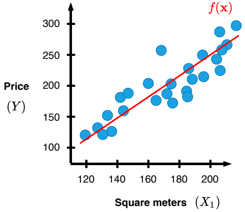
Variables generated from X_1:

- $X_2 = X_1^2$
- $X_3 = X_1^3$

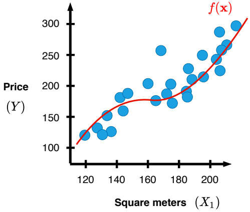


Polynomial regression with "one" variable

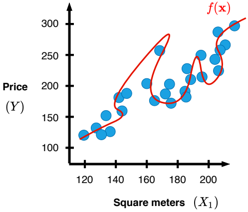
Underfitting vs Overfitting



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 ($\mathcal{L}(\hat{y}, y) \approx 0$), 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)

Strengths and weaknesses

Strengths

- Regression coefficients are **easy to understand**
- Best model when there is a linear relationship

Weaknesses

- Outliers have a big effect, especially with small data
- Doesn't work well when there is a non-linear relationship

Table of contents

Non-linear regression

Objective: To find a function $f(\mathbf{x})$ that correctly predicts the effectiveness of a drug given its dose

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

Is it appropriate to use a **linear function?**

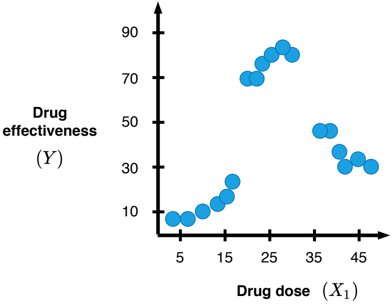
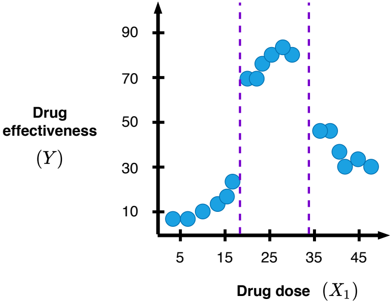


Table of contents

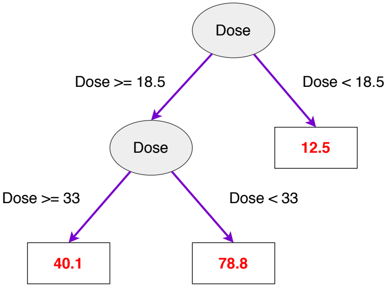
Decision trees

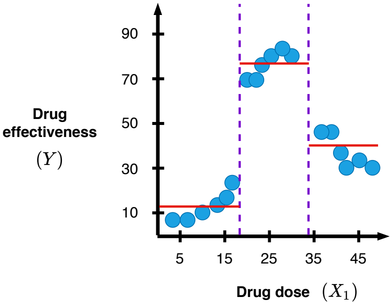
We observe three main sections of drug effectiveness

We can use a **different function** for each one of them



Decision trees

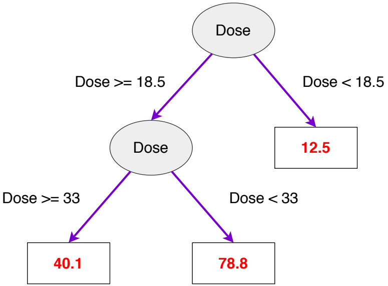


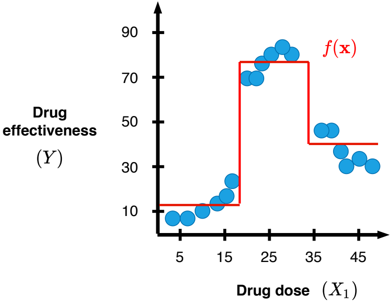


We have a numeric value (the mean) on each leaf of the tree

We could have a linear regression on each leaf $\rightarrow$ **model trees**

Decision trees





We have a numeric value (the mean) on each leaf of the tree

We could have a linear regression on each leaf → **model trees**

Strengths and weaknesses

Strengths

- Easy to understand (if-then-else rules)
- Easy to combine with other approaches (i.e., model trees)
- **Very good when done in ensembles**

Weaknesses

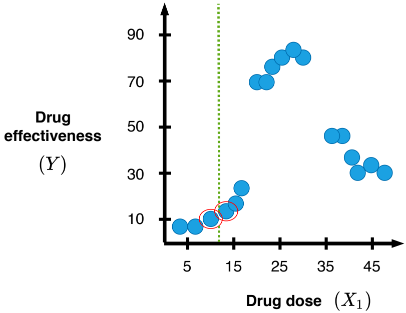
- Individual trees are prone to **overfitting**
- Pruning is usually necessary (when/how to **prune**?)

Table of contents

K-nearest neighbours

Procedure to predict a new $\mathbf{x}$:

- Measure distance to all the other instances
- Select k closest ones
- Average of the y values of those k instances

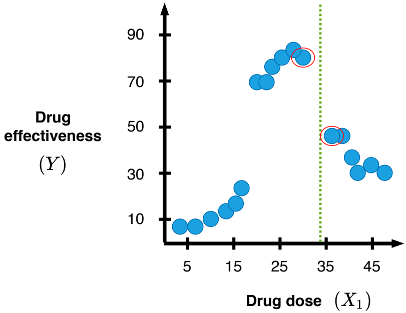


Example for $k = 2$

K-nearest neighbours

Procedure to predict a new $\mathbf{x}$:

- Measure distance to all the other instances
- Select k closest ones
- Average of the y values of those k instances

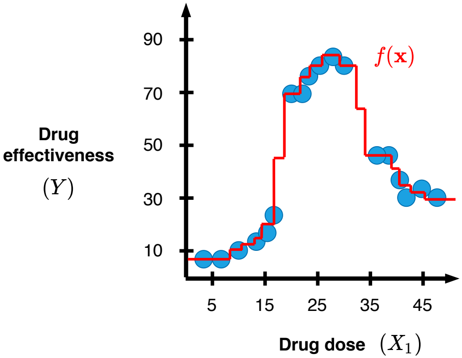


Example for $k = 2$

K-nearest neighbours

Procedure to predict a new $\mathbf{x}$:

- Measure distance to all the other instances
- Select k closest ones
- Average of the y values of those k instances



Regression function for $k = 2$

Strengths and weaknesses

Strengths

- Easy to understand
- Can represent any function with enough data

Weaknesses

- Memory intensive
- Problems on high dimensional data (distances)
- May overfit with small k

Machine Learning

Regression analysis

Fernando Rodríguez Sánchez

ferjorosa@gmail.com

Universidad Politécnica de Madrid

16/10/2020