Overview

Linear Regression with One Variable

Gradient Descent

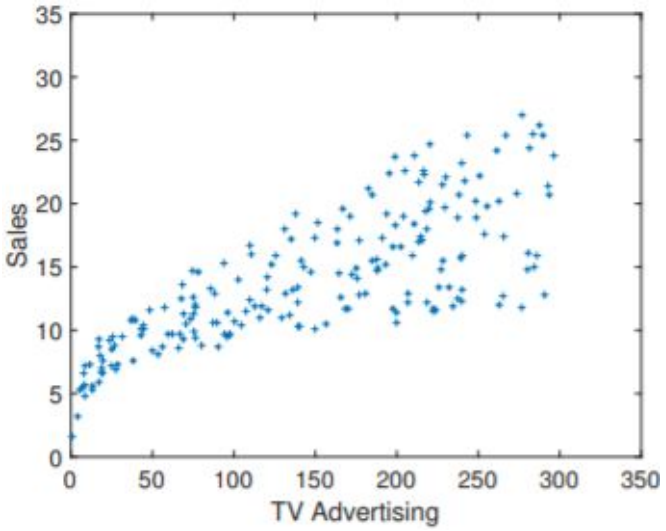
Linear Regression with Multiple Variables

Gradient Descent with Multiple Variables

Example

Data consists of the advertising budget for three media (TV, radio, and newspapers) and the overall sales in 200 different markets.

TV	Radio	Newspaper	Sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	9.3
⋮	⋮	⋮	⋮



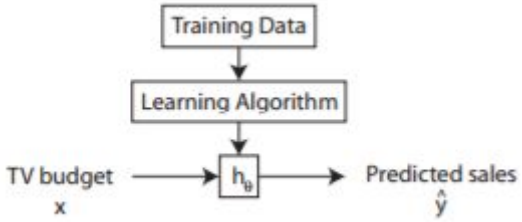
So how would we predict sales in a new area? Or sales with the TV budget increased to 350?

… Draw a line to fit the data points

Notation:

m=number of training instances
x = input variables/features
y = output variable/target
(x^i, y^i) is the ith instance

Our prediction: $\widehat{y} = h_\theta(x) = \theta_0 + \theta_1 x$. θ_0, θ_1 are unknown parameters.



Our model:

How should we choose model parameters θ?

- Idea: Choose θ_0, θ_1 so that $h_\theta(x^{(i)})$ is as close to $y^{(i)}$ for each training instance as possible
- Least squares case: select the values for θ_0, θ_1 which minimise the cost function…

$$J(\theta_0, \theta_1) = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

- Residuals are the difference between the value predicted by the fit and the observed value
    - **Do they look random or do they have some structure? Is our model satisfactory?**
    - We can estimate a confidence interval for the prediction made by our linear fit, using these residuals.
- **Cross-validation/bootstrapping could be used to estimate our confidence in the fit itself**

Gradient Descent

- A smarter minimisation approach than brute forcing it
    - **Start with some θ_0, θ_1**
    - **Repeat: Update θ_0, θ_1 with a new value which makes J(θ_0, θ_1) smaller**
    - This will eventually find the minimum if the curve is “bowl shaped” or convex
    - When a curve has several minima then we can’t be sure which we will converge to…
        - **Might converge to a local minimum, not global**
- **One option:** carry out local search of θ_0, θ_1 to find one that decreases J.
- **Another option: Gradient Descent:**

$$temp0 := \theta_0 - \alpha\frac{\partial}{\partial\theta_0}J(\theta_0, \theta_1)$$

$$temp1 := \theta_1 - \alpha\frac{\partial}{\partial\theta_1}J(\theta_0, \theta_1)$$

$$\theta_0 := temp0,\ \theta_1 := temp1$$

-

$\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1) \approx \frac{J(\theta_0+\delta,\theta_1)-J(\theta_0,\theta_1)}{\delta}$ for δ sufficiently small.

$J(\theta_0+\delta,\theta_1) \approx J(\theta_0,\theta_1)+\delta\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)$

When $\delta = -\alpha\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)$ then $J(\theta_0+\delta,\theta_1) \approx$

$J(\theta_0,\theta_1) - \alpha\left(\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)\right)^2$

- α is the step size. Too small a step size will prolong the time taken to find a minimum
  - Too large a value can lead to us overshooting the minimum
- We need to adjust the step size to converge in a reasonable amount of time

For $J(\theta_0,\theta_1) = \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$ with $h_\theta(x) = \theta_0 + \theta_1 x$:

- $\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1) = \frac{2}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})$
- $\frac{\partial}{\partial\theta_1}J(\theta_0,\theta_1) = \frac{2}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x^{(i)}$

So gradient descent algorithm is:

- repeat:

$temp0 := \theta_0 - \frac{2\alpha}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})$

$temp1 := \theta_1 - \frac{2\alpha}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x^{(i)}$

$\theta_0 := temp0,\ \theta_1 := temp1$

So back to the Advertising example…

- n=number of features (3)
- **Linear Regression with multiple variables**
  - **Hypothesis:** $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$
  - Create some dummy variable: $x_0 = 1$ to make multiplication more straight forward (same number of features as parameter weights)
  - **So we can redefine our hypothesis:** $h_\theta(x) = \theta^T x$
- **Our goals and cost functions are the same as they were with a single variable**
- Can brute force it, or…
- **Gradient descent:**
  - **Starting with some** θ

for j=0 to n $\{tempj := \theta_j - \frac{2\alpha}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}\}$

for j=0 to n $\{\theta_j := tempj\}$