Introduction to Machine Learning
NPFL 054

http://ufal.mff.cuni.cz/course/npfl054

Barbora Hladká
hladka@ufal.mff.cuni.cz

Martin Holub
holub@ufal.mff.cuni.cz

Charles University,
Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics

Outline

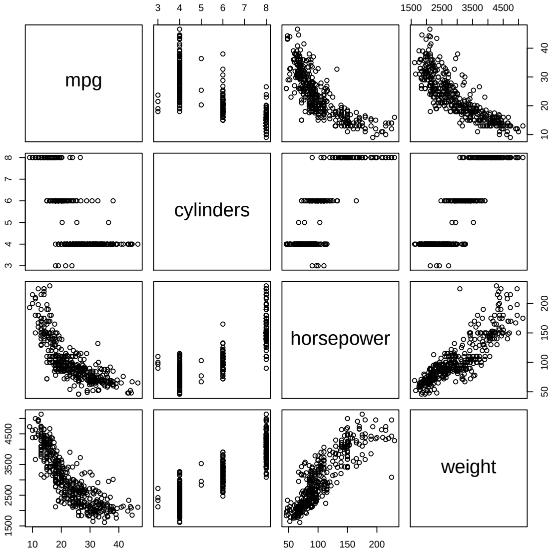
- **Linear regression**
  - `Auto` data set

Dataset `Auto` from the `ISLR` package

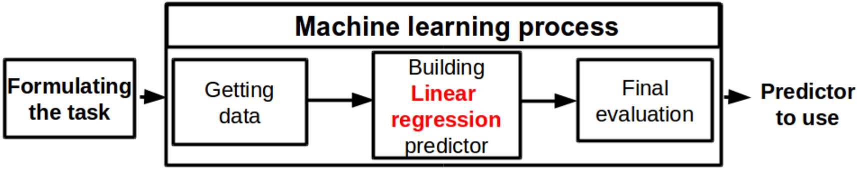
392 instances on the following 9 features

mpg	Miles per gallon
cylinders	Number of cylinders between 4 and 8
displacement	Engine displacement (cu. inches)
horsepower	Engine horsepower
weight	Vehicle weight (lbs.)
acceleration	Time to accelerate from 0 to 60 mph (sec.)
year	Model year (modulo 100)
origin	Origin of car (1. American, 2. European, 3. Japanese)
name	Vehicle name

Dataset Auto from the ISLR package



Linear regression



Linear regression

Linear regression is a class of regression algorithms assuming that there is at least a linear dependence between a target attribute and features.

A target hypothesis f has a form of **linear function**

$$f(\mathbf{x}; \Theta) = \theta_0 + \theta_1 x_1 + \cdots + \theta_m x_m \tag{1}$$

– $\theta_0, \ldots, \theta_m$ are regression parameters
– **simple linear regression** if $m = 1$

Linear regression

Notation

$$\mathbf{y} = \begin{pmatrix} y_1 \\ \dots \\ y_n \end{pmatrix}$$

$$\mathbf{x}_i = \langle 1, x_{i1}, \dots, x_{im} \rangle$$

$$\Theta^\top = \begin{pmatrix} \theta_0 \\ \dots \\ \theta_m \end{pmatrix}, \mathbf{X} = \begin{pmatrix} 1 & x_{11} & \dots & x_{1m} \\ 1 & x_{21} & \dots & x_{2m} \\ \dots & \dots & \dots & \dots \\ 1 & x_{n1} & \dots & x_{nm} \end{pmatrix}$$

Now we can write $\mathbf{y} = \mathbf{X}\Theta^\top$, $f(\mathbf{x}) = \Theta^\top \mathbf{x}$

Parameter interpretation

Numerical feature

θ_i is the average change in y for a unit change in A_i holding all other features fixed

Parameter interpretation

Categorical feature with k values
Replace the feature with $k-1$ dummy numerical features $\mathrm{DA}^1, \ldots, \mathrm{DA}^{k-1}$

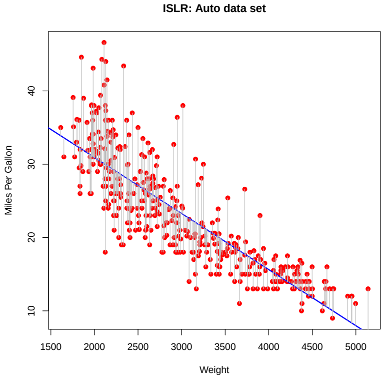
Example: run simple linear regression `mpg` $\sim$ `origin`

	DA^1	DA^2
American	0	0
European	1	0
Japanase	0	1

- $y = \theta_0 + \theta_1 \mathrm{DA}^1 + \theta_2 \mathrm{DA}^2$
- $y = \theta_0 + \theta_1$ if the car is European
- $y = \theta_0 + \theta_2$ if the car is Japanese
- $y = \theta_0$ if the car is American
- θ_0 as the average `mpg` for American cars
- θ_1 as the average difference in `mpg` between European and American cars
- θ_2 as the average difference in `mpg` between Japanese and American cars

Parameter estimates
Least Square Method

- residual $y_i - \hat{y}_i$, where $\hat{y}_i = \hat{f}(\mathbf{x}_i) = \hat{\Theta}^\top \mathbf{x}_i$
- **Loss function** Residual Sum of Squares $\mathrm{RSS}(\hat{\Theta}) = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$



Parameter estimates
Least Square Method

Optimization problem

$$\Theta^{\star} = \mathrm{argmin}_{\Theta} \mathrm{RSS}(\Theta)$$

The `argmin` operator will give Θ for which $\mathrm{RSS}(\Theta)$ is minimal.

Parameter estimates
Least Square Method

Solving the optimization problem analytically

Normal Equations Calculus

Theorem

$\Theta^\star$ is a least square solution to $\mathbf{y} = \mathbf{X}\Theta^\top \Leftrightarrow \Theta^\star$ is a solution to the Normal equation $\mathbf{X}^\top\mathbf{X}\Theta = \mathbf{X}^\top\mathbf{y}$.

$\Theta^\star = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y}$

Computational complexity of a $(m+1) \times (m+1)$ matrix inversion is $O(m+1)^3$:-(

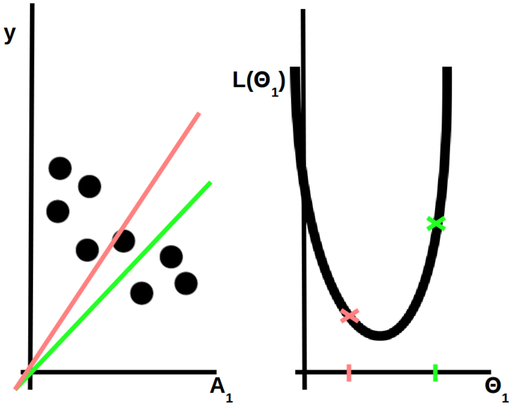
Parameter estimates
Least Square Method

Solving the optimization problem numerically

Gradient Descent Algorithm

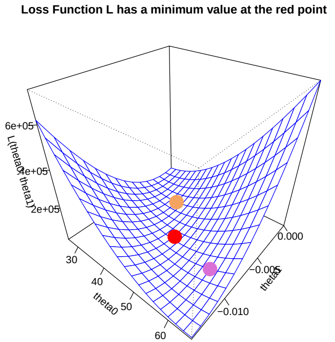
Gradient Descent Algorithm

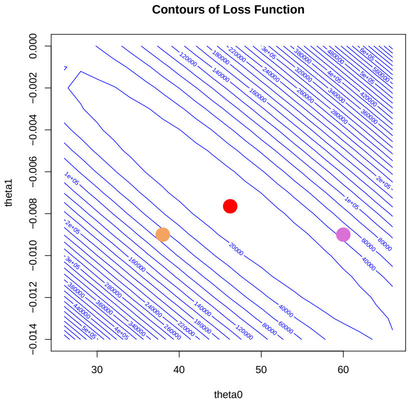
Assume: simple regression, $\theta_0 = 0$, $\theta_1 \neq 0$



Gradient Descent Algorithm

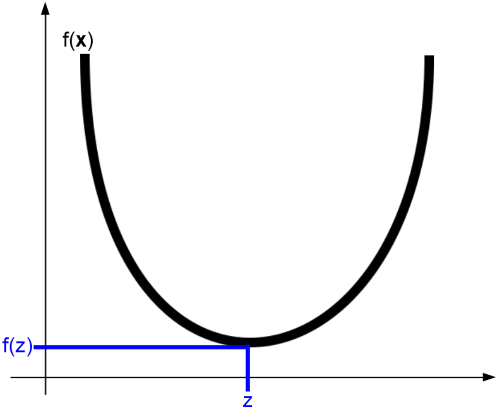
Assume: simple regression, $\theta_0 \neq 0$, $\theta_1 \neq 0$





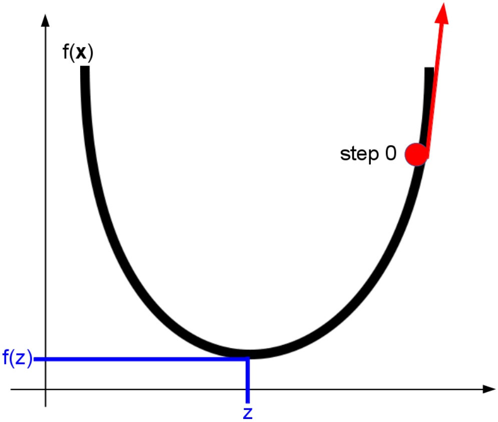
Gradient Descent Algorithm

Gradient descent algorithm is an optimization algorithm to find a local minimum of a function f.



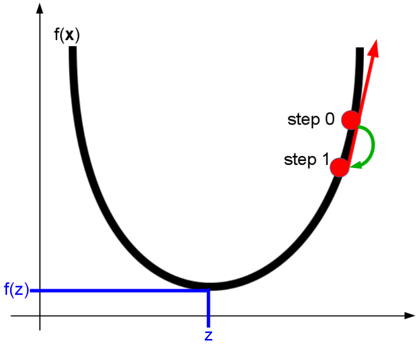
Gradient Descent Algorithm

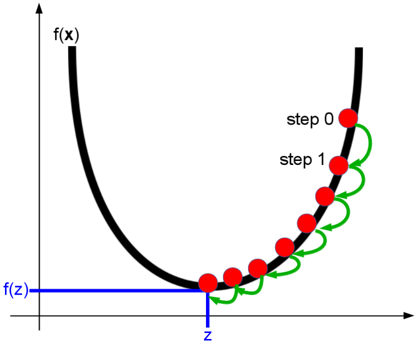
1. Start with some $\mathbf{x}_0$.



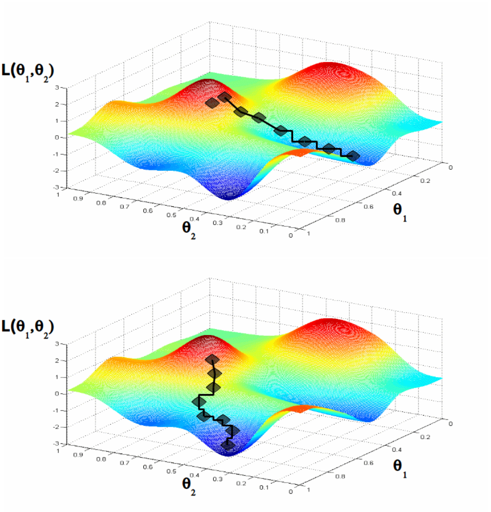
Gradient Descent Algorithm

2. Keep changing $\mathbf{x}_i$ to reduce $f(\mathbf{x}_i)$
Which direction to go? How big step to do?





Gradient Descent Algorithm



Credits: Andrew Ng

Gradient Descent Algorithm

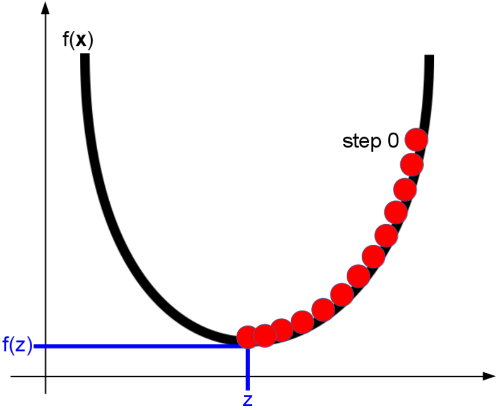
- We are seeking the solution to the minimum of a function $f(\mathbf{x})$. Given some initial value $\mathbf{x}_0$, we can change its value in many directions.
- What is the best direction to minimize f? We take the **gradient** ∇f of f

$$\nabla f(x_1, x_2, \ldots, x_m) = \langle \frac{\partial f(x_1, x_2, \ldots, x_m)}{\partial x_1}, \ldots, \frac{\partial f(x_1, x_2, \ldots, x_m)}{\partial x_m} \rangle$$

- Intuitively, the gradient of f at any point tells which direction is the steepest from that point and how steep it is. So we change $\mathbf{x}$ in the opposite direction to lower the function value.

Gradient Descent Algorithm

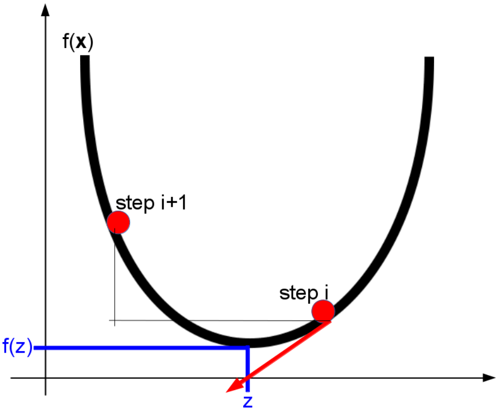
Choice of the step: assume constant value



If the step is too small, GDA can be slow.

Gradient Descent Algorithm

Choice of the step



If the step is too large, GDA can overshoot the minimum.
It may fail to converge, or even diverge.

Gradient Descent Algorithm

repeat until convergence $\{$

$$\Theta^{K+1} := \Theta^{K} - \alpha \nabla f(\Theta^{K})$$

$\}$

– α is a positive step-size hyperparameter
(another option is to choose a different step size α_k at each iteration)

I.e. simultaneously update θ_j, $j = 1, \ldots, m$

Linear regression
Gradient Descent Algorithm

For linear regression $f = RSS$

$$\theta_j^{K+1} := \theta_j^K - \alpha \frac{1}{n} \sum_{i=1}^{n} ((\Theta^K)^\top \mathbf{x} - y_i) x_{ij}$$

RSS is a convex function, so there is no local optimum, just global minimum.

Polynomial regression

Polynomial regression is an extension of linear regression where the relationship between features and target value is modelled as a d-th order polynomial.

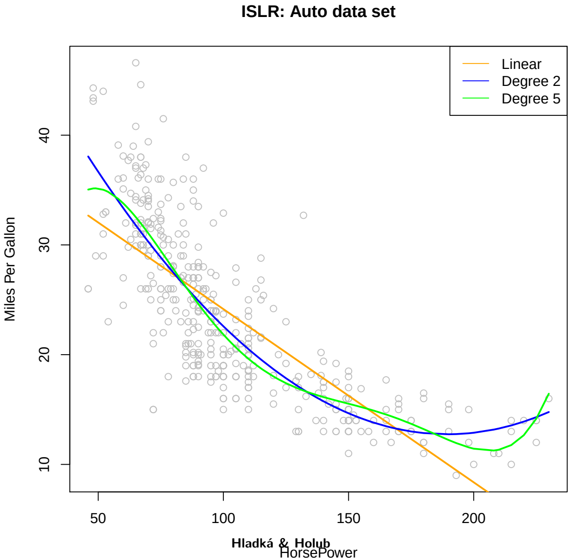
Simple regression
$y = \theta_0 + \theta_1 x_1$

Polynomial regression
$y = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \dots \theta_d x_1^d$

It is still a linear model with features $A_1, A_1^2, \dots, A_1^d$.

The *linear* in linear model refers to the hypothesis parameters, not to the features. Thus, the parameters $\theta_0, \theta_1, \dots, \theta_d$ can be easily estimated using least squares linear regression.

Polynomial regression
Auto data set



Assessing the accuracy of the model

- **Coefficient of determination** R^2 measures the proportion of variation in a target value that is reduced by taking into account **x**

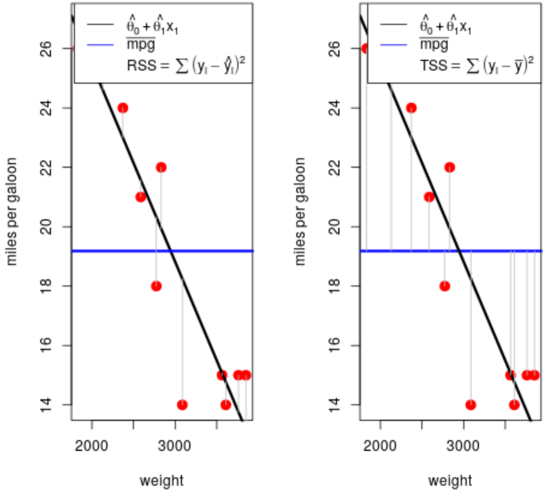
$$\mathrm{R}^2 = \frac{\mathrm{TSS} - \mathrm{RSS}}{\mathrm{TSS}} = 1 - \frac{\mathrm{RSS}}{\mathrm{TSS}}$$

  where Total Sum of Squares $\mathrm{TSS} = \sum_{i=1}^{n}(y_i - \overline{y})^2$; $R^2 \in \langle 0, 1 \rangle$

- **Mean Squared Error** MSE

$$\mathrm{MSE} = \frac{1}{n} \cdot \mathrm{RSS}$$

R^2



$\hat{\theta}_0 + \hat{\theta}_1 x_1$
$\overline{mpg}$
RSS = $\sum (y_i - \hat{y}_i)^2$
miles per galoon
weight
$\hat{\theta}_0 + \hat{\theta}_1 x_1$
$\overline{mpg}$
TSS = $\sum (y_i - \bar{y})^2$
miles per galoon
weight

Population regression line vs. Least squares line

- Population regression line: $\theta_0, \ldots, \theta_m$
- Least squares line: $\hat{\theta_0}, \ldots, \hat{\theta_m}$

- Assume random variable Y, sample $D = \{y_1, \ldots, y_n\}$
- Estimate population mean μ: $\hat{\mu}$, e.g., $\hat{\mu} = \overline{y} = \sum_{i=1}^{n} y_i$
- Standard Error of $\hat{\mu}$: $SE(\hat{\mu})^2 = \frac{\sigma^2}{n}$

Population regression line vs. Least squares line

How accurate is $\hat{\theta}_i$ as an estimate of θ_i?

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  14.8120     0.7164   20.68   <2e-16 ***
origin        5.4765     0.4048   13.53   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.447 on 390 degrees of freedom
Multiple R-squared:  0.3195,    Adjusted R-squared:  0.3177
F-statistic: 183.1 on 1 and 390 DF,  p-value: < 2.2e-16
```

- Statistical hypothesis testing (details will be provided later on): H_0 (null hypothesis): $\theta_i = 0$; H_1 (alternate hypothesis): $\theta_i <> 0$, i.e. there exists a relationship between the target attribute and the feature $\mathrm{A_i}$; t-test, p value, significance level α (the more stars, the more significant feature), we reject H_0 if $p <= \alpha$
- Adjusted R-squared $=$ R^2 adjusted for the number of features used in the model

Summary of Examination Requirements

- Linear regression, simple linear regression, polynomial regression
- Parameter interpretation
- Least Square Method
- Gradient Descent Algorithm
- Coefficient of Determination, Mean Squared Error