COMP4434 Big Data Analytics

Lab 2 Gradient Descent

HUANG Xiao
xiaohuang@comp.polyu.edu.hk

Linear Regression Example

The training data contain some example measurements of the profit gained by opening an outlet in the cities with the population ranging between 30,000 and 100,000. The y-values are the profit measured in USD, and the x-values are the populations of the city. Each city population and profit tuple constitutes one training example in training dataset.

City Population (10^4) x	Profit (10^4) y
6.4862	6.5987
5.5277	9.1302
8.5186	13.662
7.0032	11.854

Linear Regression Example

```
import numpy as np

# initialize parameters
b = 0
m = 0

# set learning rate
learning_rate = 0.01

# set number of iterations
num_iterations = 5

# define dataset
x = np.array([6.4862, 5.5277, 8.5186, 7.0032])
y = np.array([6.5987, 9.1302, 13.662, 11.854])

# perform gradient descent
for i in range(num_iterations):
y_pred = m*x + b
D_m = (-1/len(x)) * sum(x * (y - y_pred))
D_b = (-1/len(x)) * sum(y - y_pred)
m = m - learning_rate * D_m
b = b - learning_rate * D_b
J = 0.5/len(x)*sum(np.power(m*x + b - y, 2))
print("Iteration: {}, m: {}, b: {}, J: {}".format(i, m, b, J))
```

scikit-learn

- scikit-learn (aka sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms, including support-vector machines, random forests, gradient boosting, and k-means, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.
- Build and train a linear regression model based on sklearn, including using a dataset called *diabetes* from sklearn.
- Compare the model trained by sklearn and the one trained by your implementation (based on the gradient descent algorithm).

Diabetes datasets

- Diabetes datasets: https://www4.stat.ncsu.edu/~boos/var.select/diabetes.html

- Change the Learning Rate