Machine Learning \_ HW 1

Atieh Kashani

**Question1:**

* Straight line where

**Code:**

import numpy as np

import matplotlib.pyplot as plt

x = np.arange(0.0, 10.0, 0.01) # get x values between 0 and 10 with 0.01 step

t0=30

t1=0.5

y = t0+t1\*x

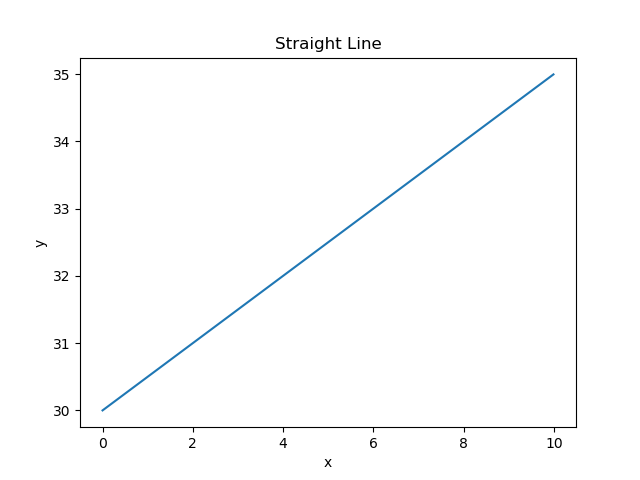
plt.plot(x, y)

plt.title("Straight Line")

plt.xlabel("x")

plt.ylabel("y")

plt.show()



* Quadratic function: 0

**Code:**

import numpy as np

import matplotlib.pyplot as plt

x = np.arange(-10.0, 10.0, 0.01) #get x values between -10 and 10 with 0.01 step

t0= 20

t1= 25

y = (x-t1)\*\*2 + t0

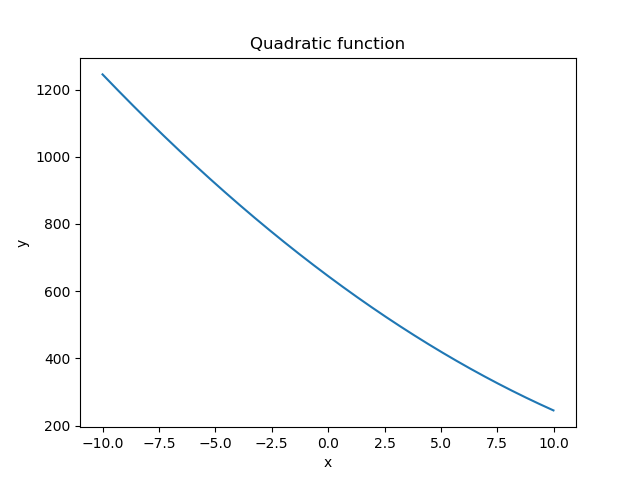
plt.plot(x, y)

plt.xlabel('x')

plt.ylabel('y')

plt.title('Quadratic function')

plt.show()



* Log function, and

**Code:**

import matplotlib.pylab as plt

import numpy as np

x = np.arange(1, 10, 0.01) #get x values between 1 and 10 with 0.01 step

f = np.log10(x)

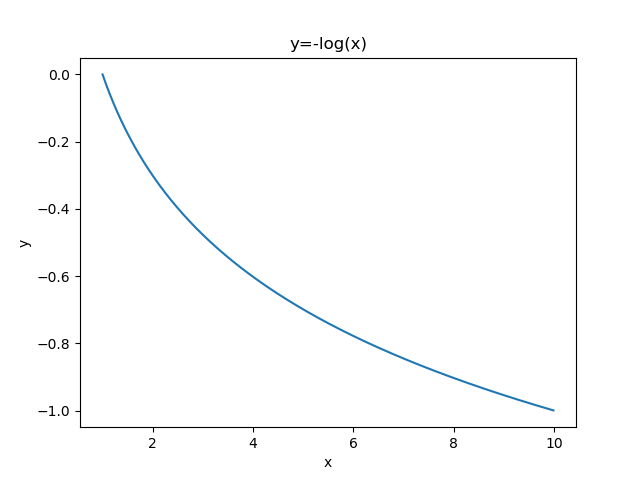
plt.plot(x, -f)

plt.xlabel('x')

plt.ylabel('y')

plt.title('y=-log(x)')

plt.show()



**Code:**

import matplotlib.pylab as plt

import numpy as np

x = np.arange(-10, 0, 0.01)

y = np.log10(1-x)

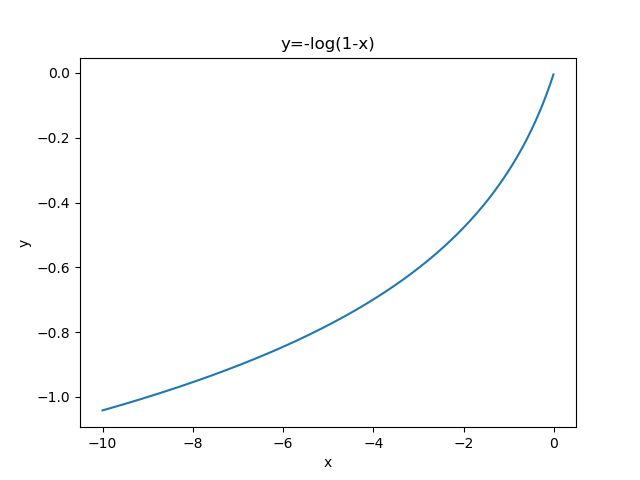
plt.plot(x, -y)

plt.title('y=-log(1-x)')

plt.xlabel('x')

plt.ylabel('y')

plt.show()



* Sigmoid function,

**Code:**

import matplotlib.pylab as plt

import numpy as np

import math

x = np.arange(-10, 10, 0.1)

y = 1 / (1 + np.exp(-x))

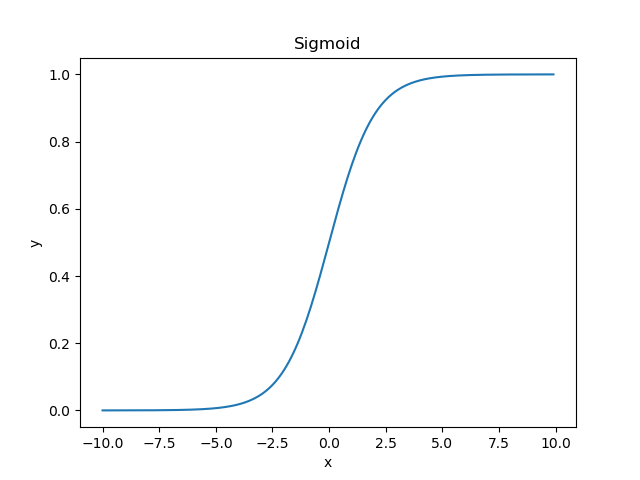
plt.plot(x, y)

plt.title('Sigmoid')

plt.xlabel('x')

plt.ylabel('y')

plt.show()



**Question 2:**

1. Classify a house to be single family or townhouse. A training set is available. Each training sample is provided with multiple features, including number of bed rooms, number of bathrooms, and house type (single family or townhouse).
2. Classify an email to be spam or not. Users already identified some emails as spam ones.
3. Human tumor Microarray data are provided as a matrix where rows correspond to genes and columns to tissue samples. The task is to cluster columns (or samples) to identify disease profiles: tissues with similar disease should yield similar expression profiles.

Answer: b) i) supervised learning with discrete predictions; ii) supervised learning with discrete predictions; iii) unsupervised learning with discrete results

i and ii can be solved with supervised learning algorithm since training data set includes features and labels. Also, the results are discrete. (Classify a house to be single family or townhouse and Classify an email to be spam or not)

iii can be solved with unsupervised learning algorithm with discrete results since training data includes only unlabeled features and the task is to cluster data into groups to identify disease profiles.

**Question 3:**

The task is determining the daily calories requirement based on the specific goal (maintaining weight, losing weight, or gaining weight). This is a regression problem.

Inputs (features): person’s age, gender, weight, height, physical activity type (i.e. running, cycling, walking....) and duration of activity (to calculate how many calories burn), and goal (maintaining weight, losing weight, gaining weight).

Outputs (labels): the amount of required daily calories.

Since the output/result (required calories) is continues (real number), this is a regression problem.

Data preparation: by using the health and nutrition science resources we know that for the specific age, gender, weight, height, and specific weight management goal how many calories are needed. We collect training data by reporting age, gender, measuring height and weight, reporting the daily physical activities, duration of each activity and weight management goal. Each physical activity burns specific number of calories that can be one of the features to help estimating the daily calories requirement. So, the training data consist of the features that are age, gender, weight, height, activity type and activity duration, and specific goal. The labels are the required daily calories.

Validation data is a part of the training data which have not been used to train the model. This data is used to check the efficiency of the trained model and adjust the architecture of the classifier.

Testing data is new input data that we don’t know the actual outputs (labels) and It is only used once a model is completely trained (using the train and validation sets). By providing the inputs data for each person the model predicts the daily calories requirement (label) for that person.

One method for splitting data set is dividing data into 60% train set, 20% validation set and 20% test set.

How to get ground-truth label for all samples: for each sample the required daily calories is calculated based on the age, gender, weight and height measurements, weight management goal, and daily physical activities type and duration.