Logistic Regression

Harvey Alférez, Ph.D.

Introduction

- This is an ***exciting presentation*** because this is the first presentation where we encounter **optimization algorithms**!
- If you think about it, many of the things we do in life are **optimization problems**.

Introduction (Cont.)

- Some examples of optimization from daily life are these:
  - How do we get from point A to point B in the least amount of time?
  - How do we make the most money doing the least amount of work?
  - How do we design an engine to produce the most horsepower while using the least amount of fuel?

Introduction (Cont.)

- Perhaps you've seen some data points and then someone fit a line called the best-fit line to these points; that's **regression**.
  - What happens in **logistic regression** is we have a bunch of data, and with the data we try to build an equation to do classification for us.

Introduction (Cont.)

- In statistics, logistic regression, or logit regression, or logit model is a regression model where the **dependent variable** (DV) is **categorical**.
    - A **categorical variable** is a variable that can take on one of a limited, and usually fixed, number of possible values, thus assigning each individual to a particular group or "category."

Introduction (Cont.)

- The **binary logistic model** is used to estimate the probability of a **binary response** based on one or more predictor (or independent) variables (features).

Introduction (Cont.)

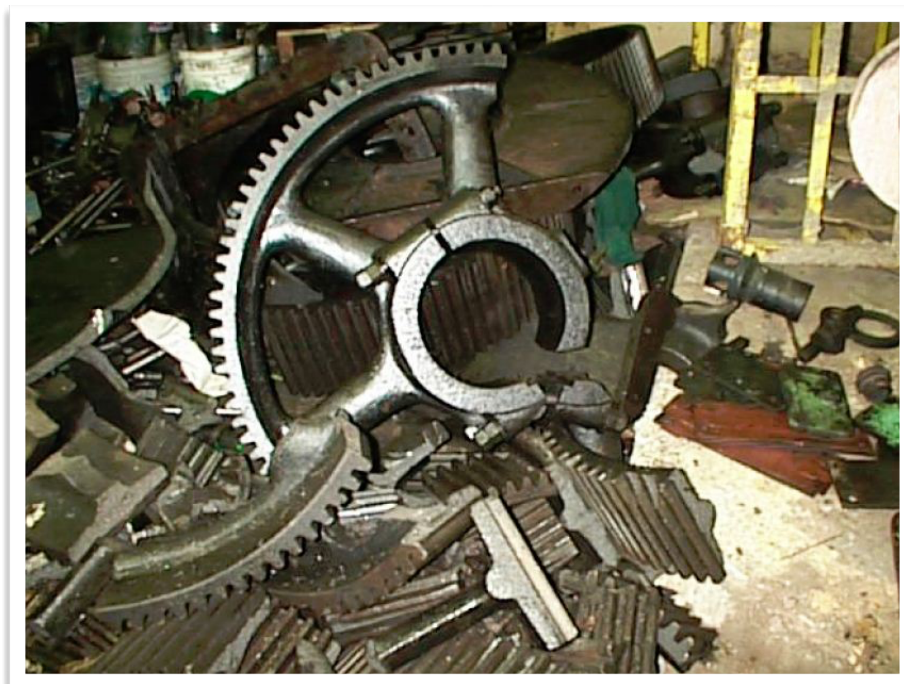
Logistic regression may be used to **predict** whether a **patient** has a given disease (e.g. diabetes; coronary heart disease), based on observed characteristics of the patient (age, sex, body mass index, results of various blood tests, etc.).

Introduction (Cont.)

Another example might be to **predict** whether an **American voter** will vote Democratic or Republican, based on age, income, sex, race, state of residence, votes in previous elections, etc.

Introduction (Cont.)

The technique can also be used in engineering, especially for **predicting** the **probability of failure** of a given process, system or product.



Classification with Logistic Regression and the Sigmoid Function

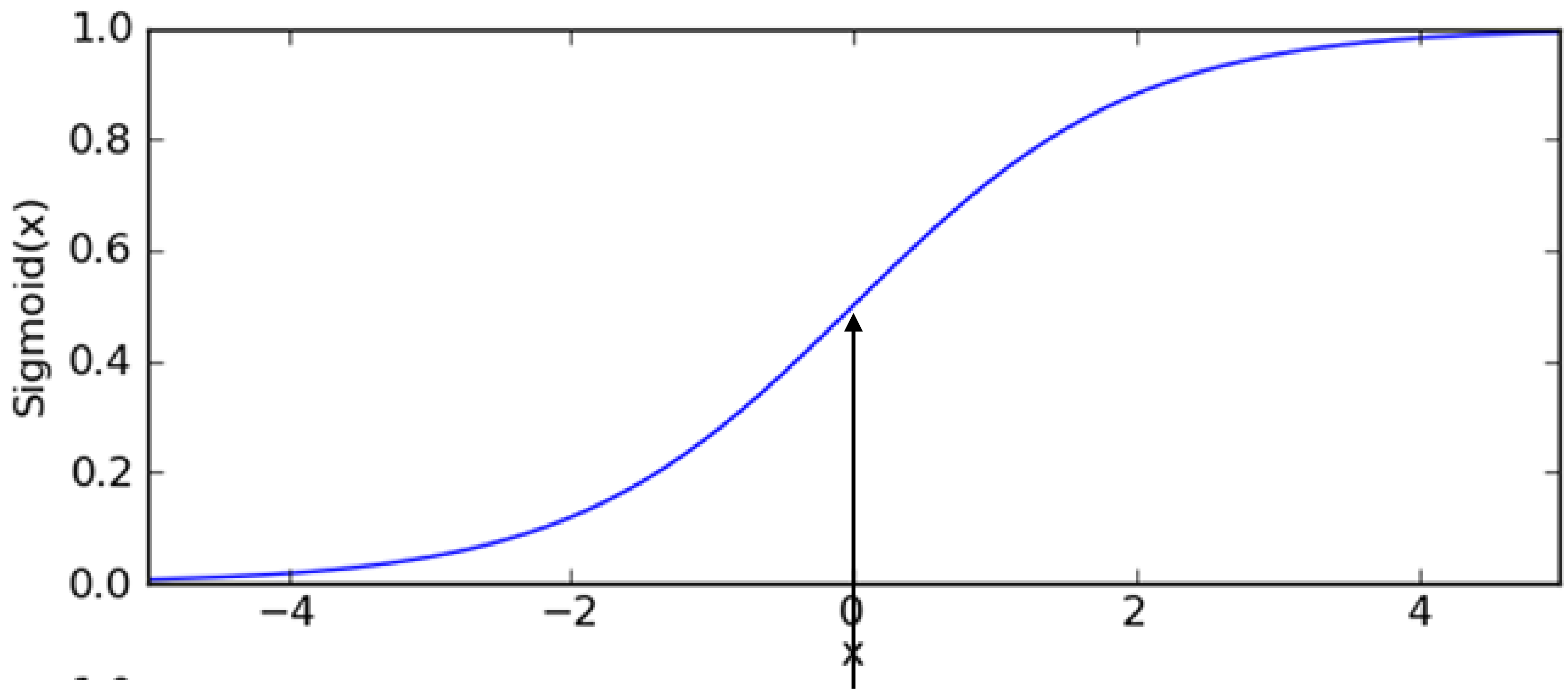
- We'd like to have an **equation** we can give all of our **features** and it will **predict the class**.

- In the two-class case, the function will spit out a **0** or a **1**.

- The **sigmoid** is given by the following equation:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Classification with Logistic Regression and the Sigmoid Function (Cont.)



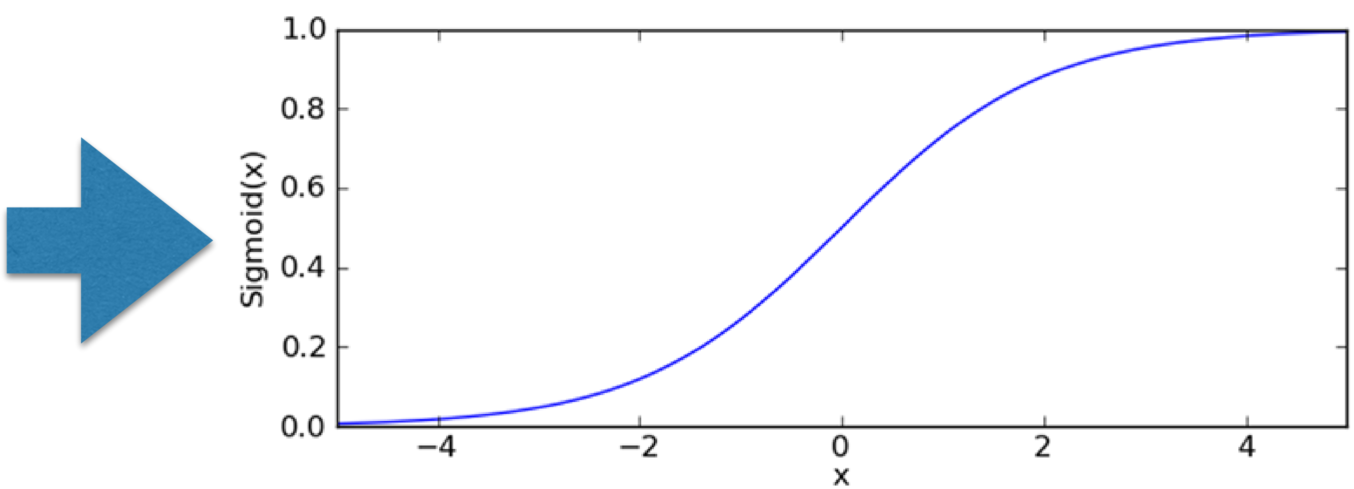
At 0 the value of the sigmoid is 0.5.

For increasing values of x, the sigmoid will approach 1, and for decreasing values of x, the sigmoid will approach 0.

Classification with Logistic Regression and the Sigmoid Function (Cont.)

- For the logistic regression classifier we'll take our **features** and **multiply** each one by a **weight** and then **add them up.**

    - This result will be put into the sigmoid, and we'll get a number between 0 and 1.



Classification with Logistic Regression and the Sigmoid Function (Cont.)

- **Anything above 0.5 we'll classify as a 1, and anything below 0.5 we'll classify as a 0.**

- The question now becomes, what are the **best weights, or regression coefficients to use, and how do we find them**?

Using Optimization to Find the Best Regression Coefficients

- The **input** to the **sigmoid function** described will be **z**, where z is given by the following:
  - $z = w_0x_0+w_1x_1+w_2x_2+\boxed{?}...+w_nx_n$
- The **vector x** is **our input data**, and we want to find **the best coefficients w**, so that this classifier will be as successful as possible.

Using Optimization to Find the Best Regression Coefficients (Cont.)

- Steps:
  - We’ll first look at **optimization** with **gradient ascent**.
  - We’ll then see how we can use this method of optimization **to find the best parameters** to model our dataset.

Using Optimization to Find the Best Regression Coefficients (Cont.)

- In mathematics and computer science, an **optimization problem** is the problem of **finding the best solution** from all feasible solutions.

Using Optimization to Find the Best Regression Coefficients (Cont.)

- **Gradient Ascent**

We'll move in the x direction by this amount

$$\nabla f(x,y) = \begin{pmatrix} \dfrac{\partial f(x,y)}{\partial x} \\ \dfrac{\partial f(x,y)}{\partial y} \end{pmatrix}$$

We'll move in the y direction by this amount

The **gradient operator** will **always point** in the **direction** of the **greatest increase**.

Using Optimization to Find the Best Regression Coefficients (Cont.)

In **vector notation** we can write the **gradient ascent** algorithm as

$$\boldsymbol{w} := \boldsymbol{w} + \alpha \nabla_{\mathbf{w}} f(w)$$

step = magnitude of movement

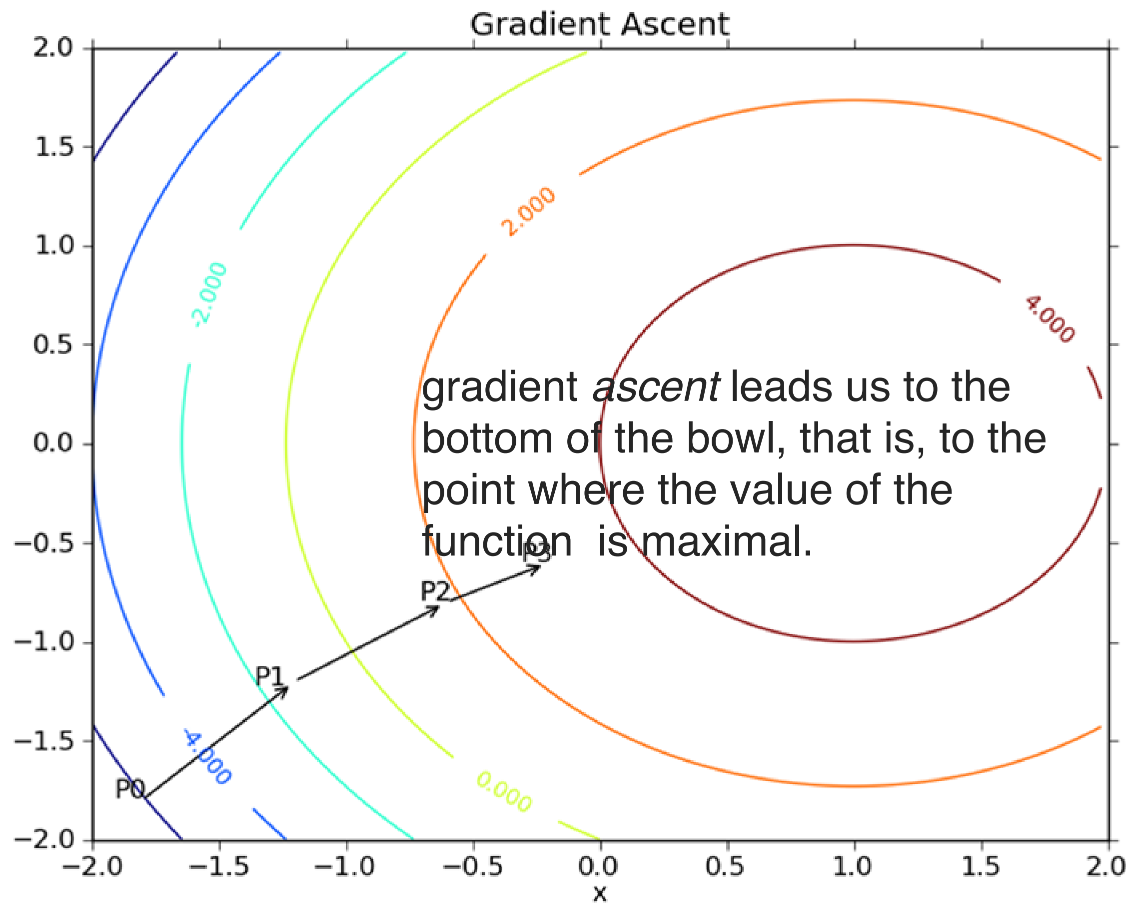


Figure 5.2 **The gradient ascent algorithm moves in the direction of the gradient evaluated at each point. Starting with point P0, the gradient is evaluated and the function moves to the next point, P1. The gradient is then reevaluated at P1, and the function moves to P2. This cycle repeats until a stopping condition is met. The gradient operator always ensures that we're moving in the best possible direction.**

Using Optimization to Find the Best Regression Coefficients (Cont.)

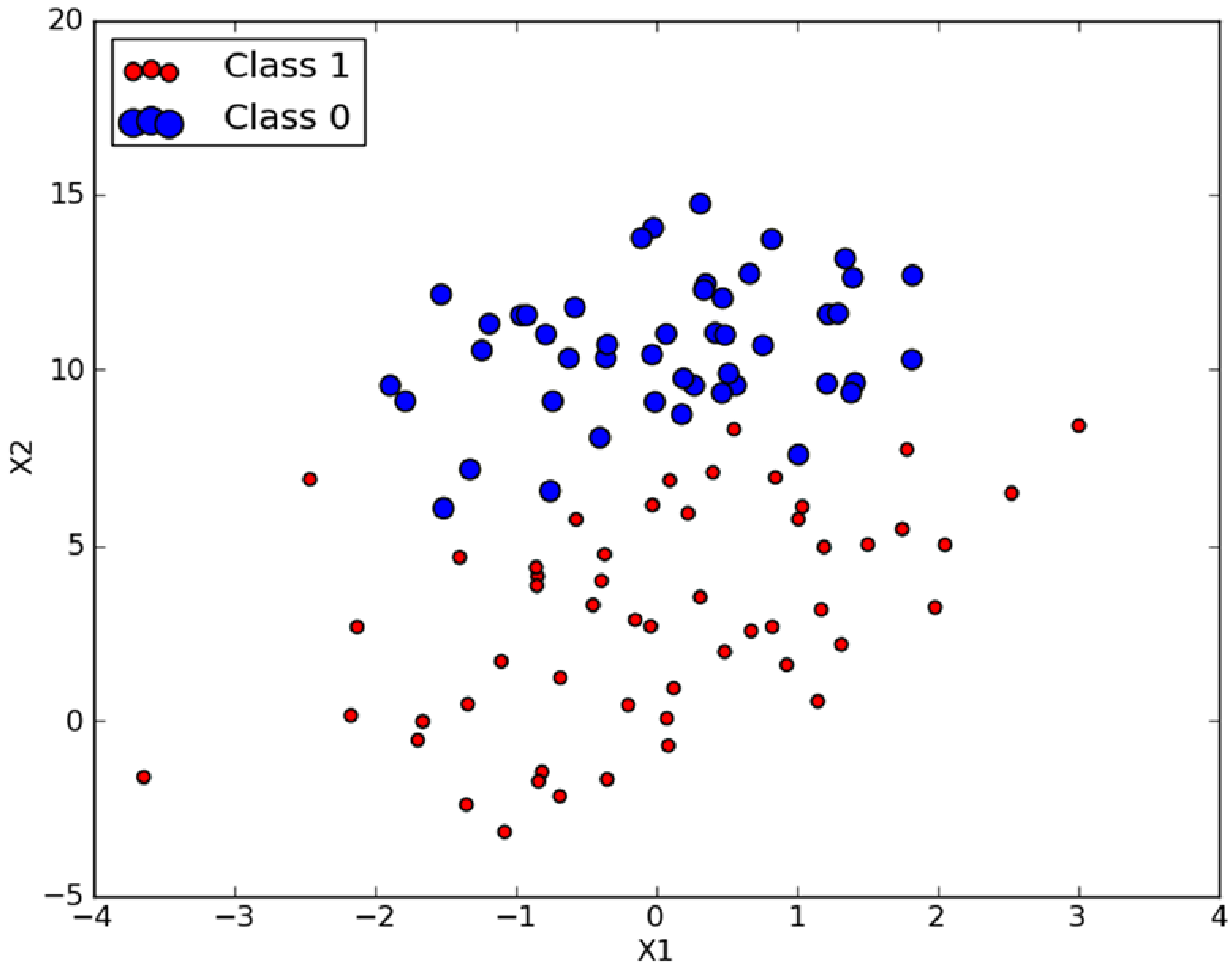
Gradient descent

Perhaps you've also heard of gradient descent. It's the same thing as gradient ascent, except the plus sign is changed to a minus sign. We can write this as

$$\boldsymbol{w} := \boldsymbol{w} - \alpha \nabla_{\mathbf{w}} f(w)$$

With gradient descent we're trying to minimize some function rather than maximize it.

Using Optimization to Find the Best Regression Coefficients (Cont.)



Using Optimization to Find the Best Regression Coefficients (Cont.)

- **Train: Using Gradient Ascent to Find the Best Parameters**
  - There are 100 data points in the figure.
  - Each point has two numeric features: X1 and X2.
  - We'll try to use gradient ascent to fit the best parameters for the logistic regression model to our data.
    - We'll do this by finding the **best weights** for this given dataset.

Using Optimization to Find the Best Regression Coefficients (Cont.)

Start with the weights all set to 1
Repeat R number of times:
    Calculate the gradient of the entire dataset
    *Update the weights vector by alpha*gradient*
    Return the weights vector

$$\boldsymbol{w} := \boldsymbol{w} + \alpha \nabla_{\mathbf{w}} f(w)$$

```
>>> import logRegres
>>> dataArr,labelMat=logRegres.loadDataSet()
>>> weights = logRegres.gradAscent(dataArr,labelMat)
matrix([[ 4.12414349],
        [ 0.48007329],
        [-0.6168482 ]])
```

Using Optimization to Find the Best Regression Coefficients (Cont.)

- Analyze: Plotting the Decision Boundary

```
>>> reload(logRegres)
<module 'logRegres' from 'logRegres.py'>
>>> logRegres.plotBestFit(weights.getA())
```

Best fit line:

$0 = w0 + w1 \; X \; x1 + w2 \; X \; x2$ *(z to 0)*

-w0 - w1 X X x1 = w2 X x2

(-w0 - w1 X x1)/w2 = x2

(-4.12414349 - (0.48007329)(-3))/-0.6168482 = x2

```
y = (-weights[0]-weights[1]*x)/weights[2]
```

Summary

- Logistic regression is finding best-fit parameters to a function called the sigmoid.
- Methods of optimization can be used to find the best-fit parameters.
- Among the optimization algorithms, one of the most common algorithms is gradient ascent.