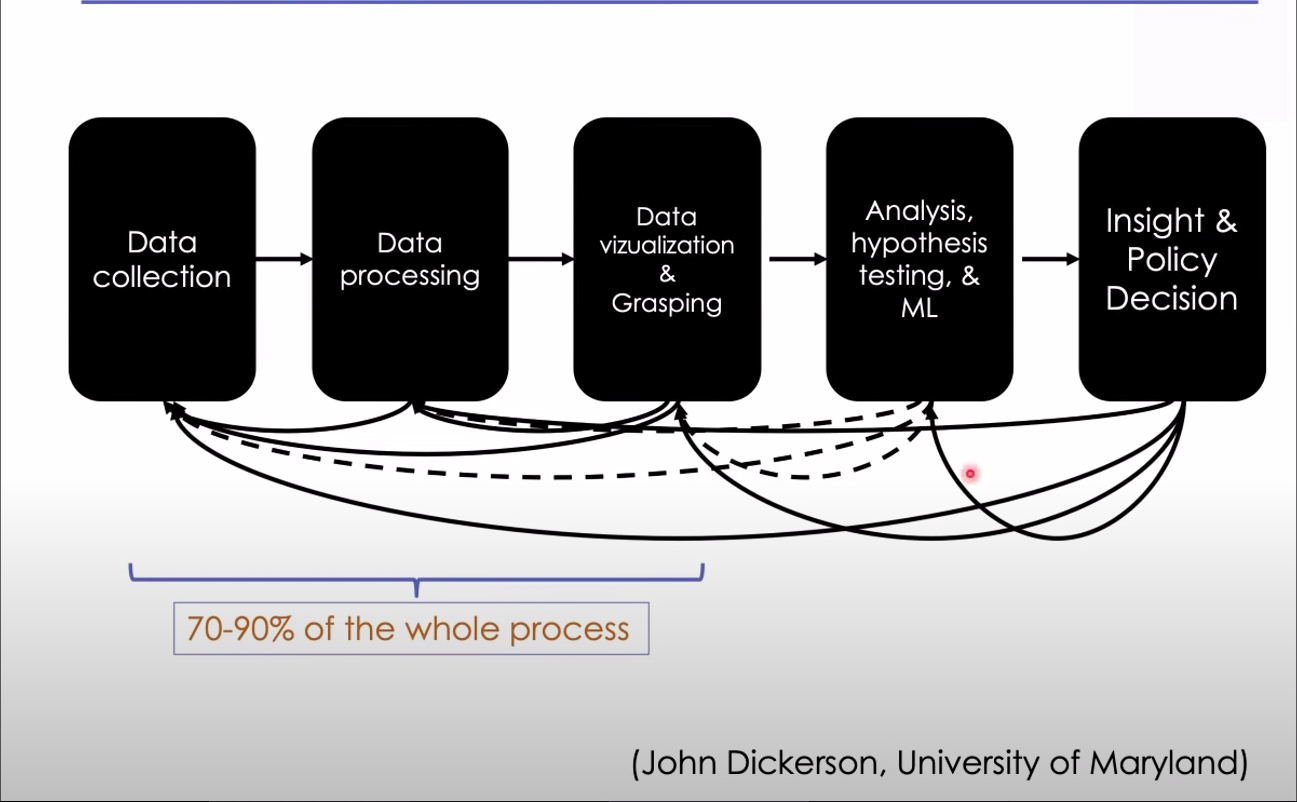
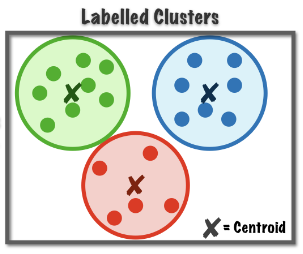
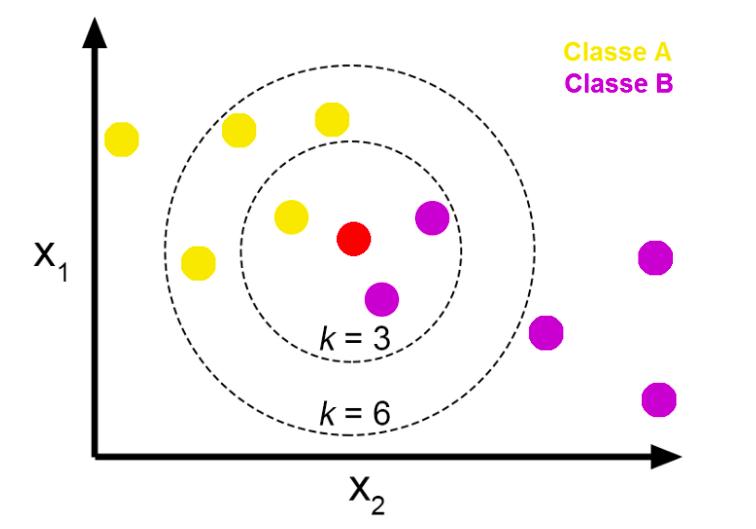
1. Tại sao phải chuẩn hóa dữ liệu ?

Chuẩn hóa là dữ liệu mà 1 bước tiền xử lý quan trọng trong Data, vì nó ảnh hưởng trực tiếp đến độ chính xác kết quả bài toán sau này.

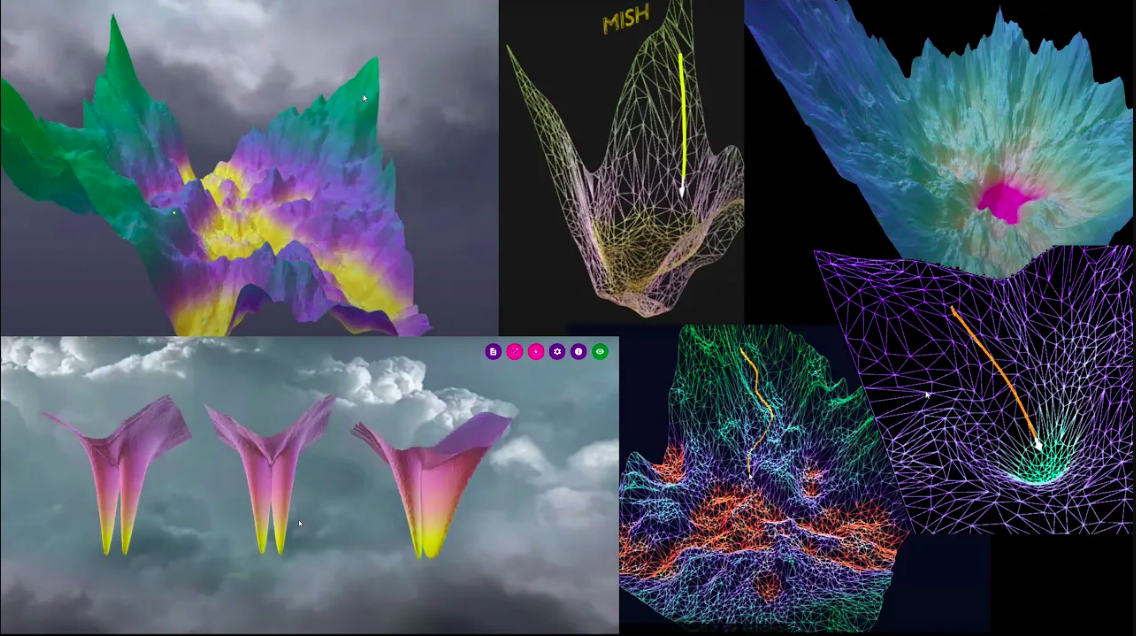


Khi có một thuộc tính trong dữ liệu (hay phần tử trong vector) lớn hơn các thuộc tính khác rất nhiều (ví dụ thay vì đo bằng mm thì một kết quả lại tính bằng cm), khoảng cách giữa các điểm sẽ phụ thuộc vào thuộc tính này rất nhiều. Để có được kết quả chính xác hơn, một kỹ thuật thường được dùng là Data Normalization (chuẩn hóa dữ liệu) để đưa các thuộc tính có đơn vị đo khác nhau về cùng một khoảng giá trị, thường là từ [0,1], [-1,1].

Trong KNN và K-means, để có thể xem 1 đối tượng thuộc lớp nào, ta cần phải tính khoảng cách, với KNN dựa vào khoảng cách để phân loại, với KMean để phân cụm, như vậy, khi các thuộc tính không cùng 1 chuẩn, rất khó để 2 thuật toán trên hoạt động chính xác vì dữ liệu bị nhiễu ảnh hưởng rất nhiều kết quả thuật toán, không phân bố đồng đều.



1 ví dụ khác: trong mạng noron nhân tạo, nói tóm lại chúntg ta cần tìm 1 cực tiểu của 1 hàm lồi (convex function) – hàm ở đây là hàm lỗi (cost function) (lỗi càng ít thì khả năng tổng quát => khả năng dự đoán của mạng càng cao.



Quá trình học (learning) của mạng là tìm bộ tham số w cho các thuộc tính x sao khi thay vào hàm active function (ví dụ hàm sigmoid trong dự đoán 1 người có khả năng trả tiền khi vay ngân hàng hay không) thì kết quả dự đoán và kết quả thật khác nhau ít nhất có thể.

Ví dụ ta cần tìm tham số w sao cho dự đoán được:

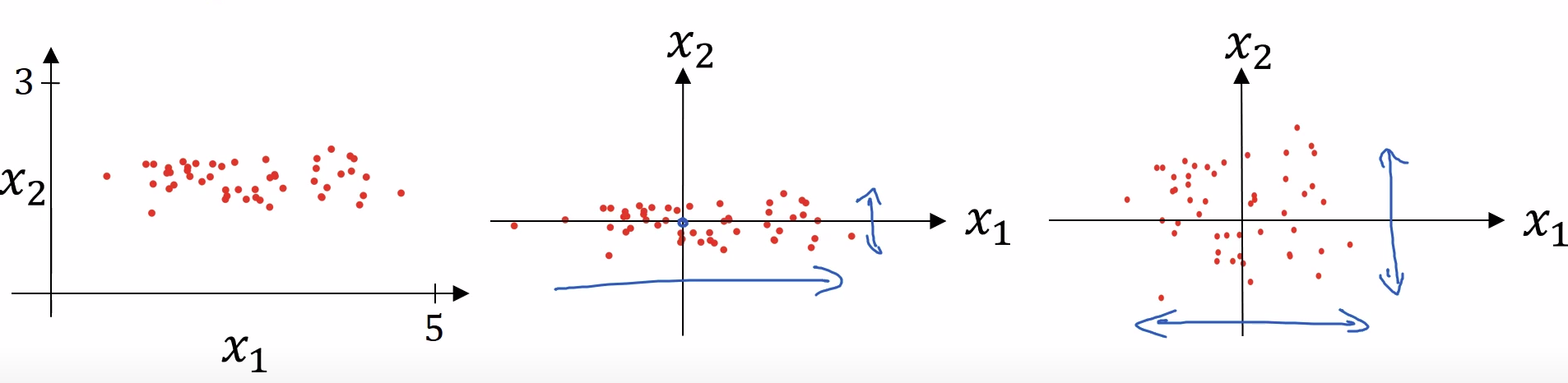
sigmoid(income \* w) <0.5 nếu không trả được nợ

>0.5 nếu trả được nợ

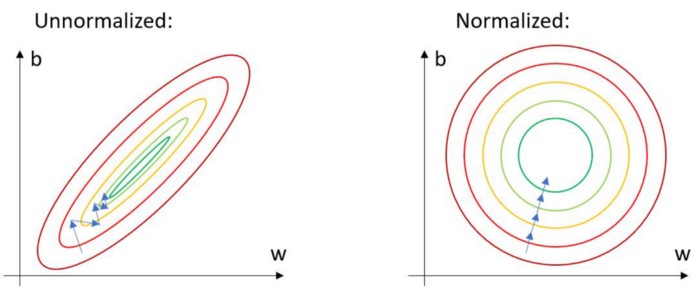
(1)Với số liệu chưa chuẩn hóa: 12000\*w, 18000\*w

(2)Với số liệu đã chuẩn hóa về [0,1] 0.716 \*w, 0.8\*w

Nhận thấy w trong (1) có miền rất rộng, và (2) có miền thấp hơn rất nhiều, vì vậy, quá trình gradients (tìm kiếm cục bộ bằng ngược hướng đạo hàm) sẽ diễn ra nhanh hơn với (2).

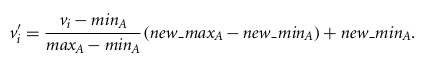


Hình vẽ trên là đưa dữ liệu về phân phối chuẩn

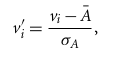
Thuật toán sẽ hội tụ nhanh hơn rất nhiều, hãy tưởng tượng 1 chiếc cốc đặt đứng và 1 chiếc cốc đặt nghiêng, con kiến bò xuống đáy chiếc cốc đặt đứng sẽ nhanh hơn chiếc cốc đặt nghiêng vì kiến phải đi men theo.

2. Một số phương pháp chuẩn hóa

2.1. Min-max normalization



Ví dụ: thu nhập của 1 người dao động từ 12000$ đến 98000$ đô, chúng ta muốn chiếu về hệ quy chiếu trong đoạn [0,1]. Với income = 73600, công thức tính là: (73600-12000)/(98000-12000) \* (1-0) = 0.716

2.2. Z-score normaliazation

Với A gạch là trung bình và sigma là độ lệch tuyệt đối (mean absolute deviation).

Ví du: Trung bình và phương sai của 1 phân phối về thu nhập lần lượt là: 54000 và 16000, như vậy, với giá trị 73600 đưa về z-norm sẽ là: (73600-54000)/16000=1.225

2.3 Decimal Scaling

Ví dụ 2: 1 tập giá trị từ [-986 đến 917]. max là |-986|. Với chuẩn hóa decimal scaling, chúng ta chia mỗi giá trị cho 1000.

3.References

[1] C2W1 Deep Learning, AndrewNg – Coursera

[2] Understand Data Normalization in Machine Learning – Towardsdatasciene

[3] Explore deep learning loss landscape – Javier ideami

[4] K-nearest neightbors – Tiep.VuHuu

[5] CS231N, CS230 – Standford