# ColumbiaX: Machine Learning

Prof. John Paisley

Department of Electrical Engineering
& Data Science Institute

Columbia University

# ColumbiaX: Machine Learning

# Lecture 1

Prof. John Paisley

Department of Electrical Engineering
& Data Science Institute

Columbia University

# Overview

This class will cover model-based techniques for extracting information from data with an end-task in mind. Such tasks include:

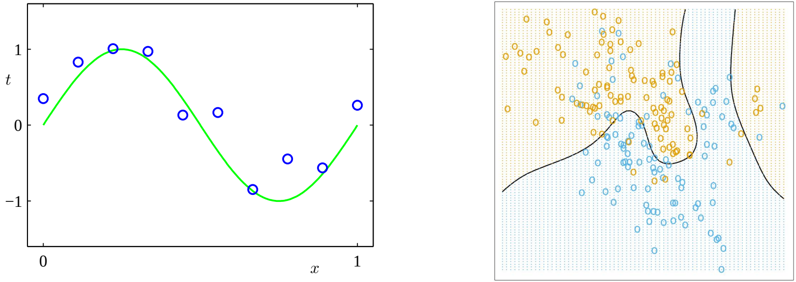
- predicting an unknown "output" given its corresponding "input"
- uncovering information within the data to better understand it
- data-driven recommendation, grouping, classification, ranking, etc.

There are a few ways we can divide up the material as we go along, e.g.,

| | |
|---|---|
| supervised learning | unsupervised learning |
| probabilistic models | non-probabilistic models |
| modeling approach | optimization techniques |

We'll adopt the first method and work in the second two along the way.

# Overview: Supervised learning



(a) Regression

(b) Classification

**Regression**: Using set of inputs, predict real-valued output.

**Classification**: Using set of inputs, predict a discrete label (aka class).

# Example Classification Problem

Given a set of inputs characterizing an item, assign it a label.

## Is this spam?

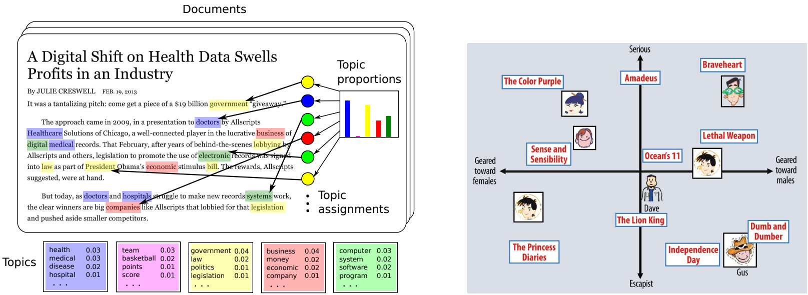
hi everyone,

i saw that close to my hotel there is a pub with bowling (it’s on market between 9th and 10th avenue). meet there at 8:30?

## What about this?

Enter for a chance to win a trip to Universal Orlando to celebrate the arrival of Dr. Seuss’s The Lorax on Movies On Demand on August 21st! Click here now!

# OVERVIEW: UNSUPERVISED LEARNING



A Digital Shift on Health Data Swells Profits in an Industry

By JULIE CRESWELL FEB. 19, 2013

It was a tantalizing pitch: come get a piece of a $19 billion government “giveaway.”

The approach came in 2009, in a presentation to doctors by Allscripts Healthcare Solutions of Chicago, a well-connected player in the lucrative business of digital medical records. That February, after years of behind-the-scenes lobbying by Allscripts and others, legislation to promote the use of electronic records was signed into law as part of President Obama’s economic stimulus bill. The rewards, Allscripts suggested, were at hand.

But today, as doctors and hospitals struggle to make new records systems work, the clear winners are big companies like Allscripts that lobbied for that legislation and pushed aside smaller competitors.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| health | 0.03 | team | 0.03 | government | 0.04 | business | 0.04 | computer | 0.03 |
| medical | 0.03 | basketball | 0.02 | law | 0.02 | money | 0.02 | system | 0.02 |
| disease | 0.02 | points | 0.01 | politics | 0.01 | economic | 0.02 | software | 0.02 |
| hospital | 0.01 | score | 0.01 | legislation | 0.01 | company | 0.01 | program | 0.01 |
| . . . | | . . . | | . . . | | . . . | | . . . | |

(c) topic modeling

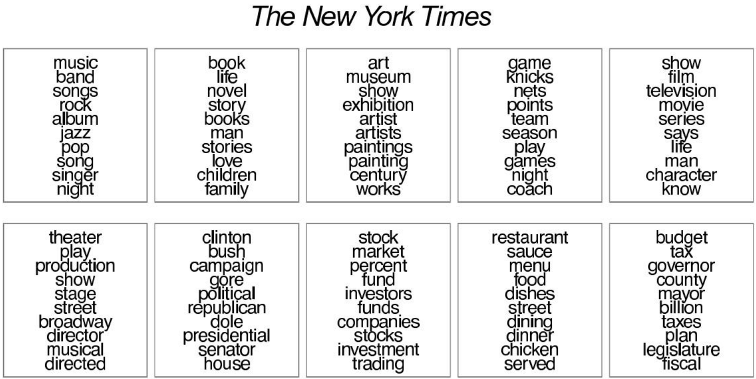
(d) recommendations[1]

With unsupervised learning our goal is often to uncover structure in the data. This helps with predictions, recommendations, efficient data exploration.

[1] Figure from Koren, Y., Robert B., and Volinsky, C.. “Matrix factorization techniques for recommender systems.” Computer 42.8 (2009): 30-37.

# EXAMPLE UNSUPERVISED PROBLEM

**Goal**: Learn the dominant topics from a set of news articles.



*The New York Times*

music
band
songs
rock
album
jazz
pop
song
singer
night

book
life
novel
story
books
man
stories
love
children
family

art
museum
show
exhibition
artist
artists
paintings
painting
century
works

game
knicks
nets
points
team
season
play
games
night
coach

show
film
television
movie
series
says
life
man
character
know

theater
play
production
show
stage
street
broadway
director
musical
directed

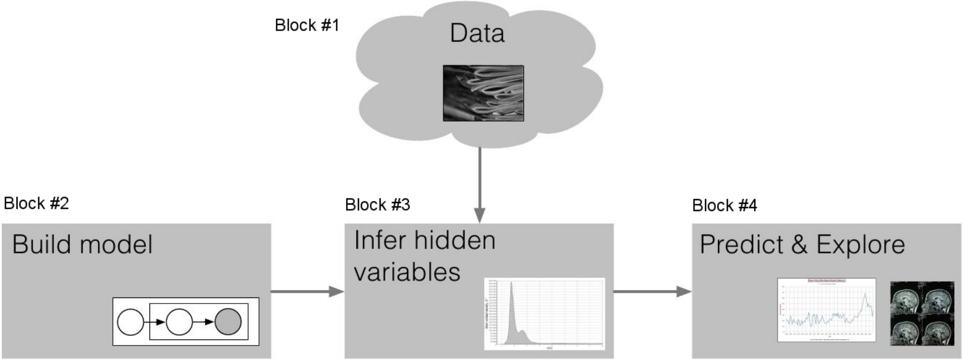
clinton
bush
campaign
gore
political
republican
dole
presidential
senator
house

stock
market
percent
fund
investors
funds
companies
stocks
investment
trading

restaurant
sauce
menu
food
dishes
street
dining
dinner
chicken
served

budget
tax
governor
county
mayor
billion
taxes
plan
legislature
fiscal

# Data Modeling



Block #1

Data

Block #2

Build model

Block #3

Infer hidden variables

Block #4

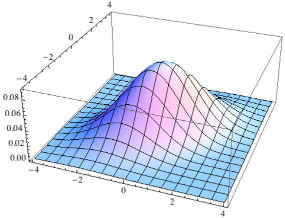
Predict & Explore

- Supervised vs. unsupervised: Blocks #1 and #4
- Probabilistic vs. non-probabilistic: Primarily Block #2 (Some Block #3)
- Model development (Block #2) vs. Optimization techniques (Block #3)

# Gaussian Distribution (multivariate)

## Gaussian density in $d$ dimensions

- Block #1: Data $x_1, \dots, x_n$. Each $x_i \in \mathbb{R}^d$
- Block #2: An i.i.d. Gaussian model
- Block #3: Maximum likelihood
- Block #4: Leave undefined



The density function is

$$p(x|\mu,\Sigma) := \frac{1}{(2\pi)^{\frac{d}{2}}\sqrt{\det(\Sigma)}}\exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big)$$

The central moments are:

$$\mathbb{E}[x] = \int_{\mathbb{R}^d} x\, p(x|\mu,\Sigma)dx = \mu,$$

$$\mathrm{Cov}(x) = \mathbb{E}[(x-\mathbb{E}[x])(x-\mathbb{E}[x])^T] = \mathbb{E}[xx^T] - \mathbb{E}[x]\mathbb{E}[x]^T = \Sigma.$$

# Block #2: A Probabilistic Model

## Probabilistic Models

- A *probabilistic model* is a set of probability distributions, $p(x|\theta)$.
- We pick the *distribution family* $p(\cdot)$, but don't know the parameter $\theta$.

**Example**: Model data with a Gaussian distribution $p(x|\theta)$, $\theta = \{\mu, \Sigma\}$.

## The i.i.d. assumption

Assume data is *independent and identically distributed (iid)*. This is written

$$x_i \overset{iid}{\sim} p(x|\theta), \quad i = 1, \ldots, n.$$

Writing the density as $p(x|\theta)$, then the *joint* density decomposes as

$$p(x_1, \ldots, x_n|\theta) = \prod_{i=1}^{n} p(x_i|\theta).$$

# Block #3: Maximum Likelihood Estimation

## Maximum Likelihood approach

We now need to find $\theta$. *Maximum likelihood* seeks the value of $\theta$ that maximizes the likelihood function:

$$\hat{\theta}_{\mathrm{ML}} := \arg\max_{\theta} \; p(x_1, \ldots, x_n|\theta),$$

This value best explains the data according to the chosen distribution family.

## Maximum Likelihood equation

The analytic criterion for this maximum likelihood estimator is:

$$\nabla_\theta \prod_{i=1}^{n} p(x_i|\theta) = 0.$$

Simply put, the maximum is at a peak. There is no “upward” direction.

# Block #3: Logarithm Trick

## Logarithm trick

Calculating $\nabla_\theta \prod_{i=1}^n p(x_i|\theta)$ can be complicated. We use the fact that the logarithm is monotonically increasing on $\mathbb{R}_+$, and the equality

$$\ln\Big(\prod_i f_i\Big) = \sum_i \ln(f_i).$$

Consequence: Taking the logarithm does not change the *location* of a maximum or minimum:

$$\max_y \ \ln g(y) \neq \max_y \ g(y) \qquad \text{The } \textit{value} \text{ changes.}$$

$$\arg\max_y \ \ln g(y) = \arg\max_y \ g(y) \qquad \text{The } \textit{location} \text{ does not change.}$$

## Maximum likelihood and the logarithm trick

$$\hat{\theta}_{\text{ML}} = \arg\max_{\theta} \prod_{i=1}^{n} p(x_i|\theta) = \arg\max_{\theta} \ \ln\Big(\prod_{i=1}^{n} p(x_i|\theta)\Big) = \arg\max_{\theta} \sum_{i=1}^{n} \ln p(x_i|\theta)$$

To then solve for $\hat{\theta}_{\text{ML}}$, find

$$\nabla_{\theta} \sum_{i=1}^{n} \ln p(x_i|\theta) = \sum_{i=1}^{n} \nabla_{\theta} \ln p(x_i|\theta) = 0.$$

Depending on the choice of the model, we will be able to solve this

1. analytically (via a simple set of equations)
2. numerically (via an iterative algorithm using different equations)
3. approximately (typically when #2 converges to a local optimal solution)

# EXAMPLE: MULTIVARIATE GAUSSIAN MLE

## Block #2: Multivariate Gaussian data model

Model: Set of all Gaussians on $\mathbb{R}^d$ with unknown mean $\mu \in \mathbb{R}^d$ and covariance $\Sigma \in \mathbb{S}^d_{++}$ (positive definite $d \times d$ matrix).

We assume that $x_1, \dots, x_n$ are i.i.d. $p(x|\mu, \Sigma)$, written $x_i \overset{iid}{\sim} p(x|\mu, \Sigma)$.

## Block #3: Maximum likelihood solution

We have to solve the equation

$$\sum_{i=1}^{n} \nabla_{(\mu,\Sigma)} \ln p(x_i|\mu, \Sigma) = 0$$

for $\mu$ and $\Sigma$. (Try doing this without the log to appreciate it's usefulness.)

# EXAMPLE: GAUSSIAN MEAN MLE

First take the gradient with respect to $\mu$.

$$\begin{aligned} 0 &= \nabla_\mu \sum_{i=1}^{n} \ln \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \exp\Big(-\frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)\Big) \\ &= \nabla_\mu \sum_{i=1}^{n} -\frac{1}{2}\ln(2\pi)^d|\Sigma| - \frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu) \\ &= -\frac{1}{2}\sum_{i=1}^{n} \nabla_\mu \Big(x_i^T\Sigma^{-1}x_i - 2\mu^T\Sigma^{-1}x_i + \mu^T\Sigma^{-1}\mu\Big) = -\Sigma^{-1}\sum_{i=1}^{n}(x_i-\mu) \end{aligned}$$

Since $\Sigma$ is positive definite, the only solution is

$$\sum_{i=1}^{n}(x_i-\mu) = 0 \qquad \Rightarrow \qquad \hat{\mu}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Since this solution is independent of $\Sigma$, it doesn't depend on $\hat{\Sigma}_{\text{ML}}$.

# EXAMPLE: GAUSSIAN COVARIANCE MLE

Now take the gradient with respect to $\Sigma$.

$$\begin{aligned} 0 &= \nabla_\Sigma \sum_{i=1}^n -\frac{1}{2}\ln(2\pi)^d|\Sigma| - \frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu) \\ &= -\frac{n}{2}\nabla_\Sigma \ln|\Sigma| - \frac{1}{2}\nabla_\Sigma \text{trace}\Big(\Sigma^{-1}\sum_{i=1}^n (x_i-\mu)(x_i-\mu)^T\Big) \\ &= -\frac{n}{2}\Sigma^{-1} + \frac{1}{2}\Sigma^{-2}\sum_{i=1}^n (x_i-\mu)(x_i-\mu)^T \end{aligned}$$

Solving for $\Sigma$ and plugging in $\mu = \hat{\mu}_{\text{ML}}$,

$$\hat{\Sigma}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^n (x_i - \hat{\mu}_{\text{ML}})(x_i - \hat{\mu}_{\text{ML}})^T.$$

# EXAMPLE: GAUSSIAN MLE (SUMMARY)

So if we have data $x_1, \ldots, x_n$ in $\mathbb{R}^d$ that we hypothesize is i.i.d. Gaussian, the maximum likelihood values of the mean and covariance matrix are

$$\hat{\mu}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \hat{\Sigma}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu}_{\text{ML}})(x_i - \hat{\mu}_{\text{ML}})^T.$$

**Are we done?** There are many assumptions/issues with this approach that makes finding the “best” parameter values not a complete victory.

- We made a model assumption (multivariate Gaussian).
- We made an i.i.d. assumption.
- We assumed that maximizing the likelihood is the best thing to do.

Comment: We often use $\theta_{\text{ML}}$ to make predictions about $x_{new}$ (Block #4).
How does $\theta_{\text{ML}}$ generalize to $x_{new}$?
If $x_{1:n}$ don’t “capture the space” well, $\theta_{\text{ML}}$ can *overfit* the data.

# ColumbiaX: Machine Learning
# Lecture 4

Prof. John Paisley

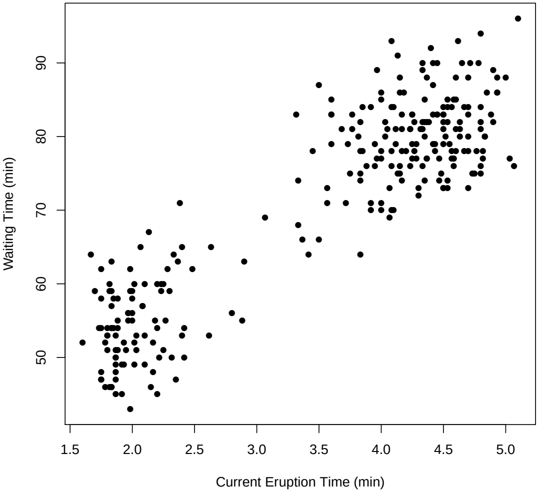
Department of Electrical Engineering
& Data Science Institute

Columbia University

# LINEAR REGRESSION

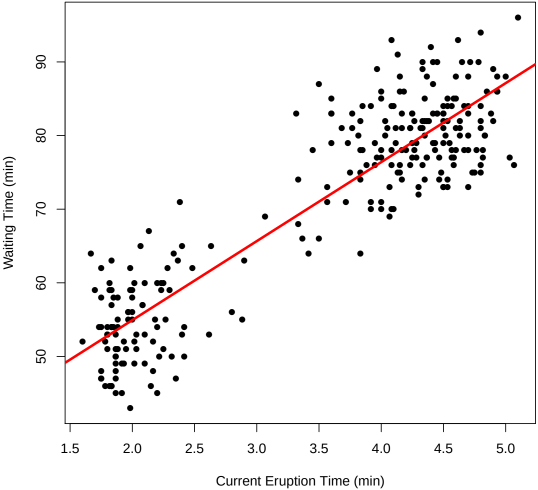
# EXAMPLE: OLD FAITHFUL

# EXAMPLE: OLD FAITHFUL



Can we meaningfully predict the time between eruptions only using the duration of the last eruption?

# EXAMPLE: OLD FAITHFUL



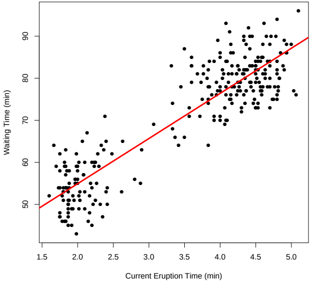
Can we meaningfully predict the time between eruptions only using the duration of the last eruption?

# Example: Old Faithful

## One model for this

$$\text{(wait time)} \approx w_0 + \text{(last duration)} \times w_1$$

- $w_0$ and $w_1$ are to be learned.
- This is an example of linear regression.



## Refresher

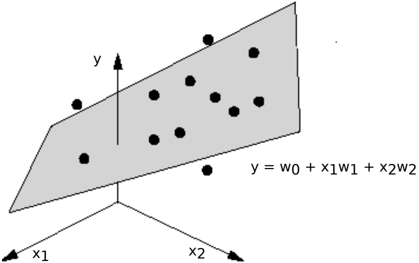
$w_1$ is the slope, $w_0$ is called the intercept, bias, shift, offset.

# HIGHER DIMENSIONS

## Two inputs

(output) $\approx w_0 + (\text{input } 1) \times w_1 + (\text{input } 2) \times w_2$

With two inputs the intuition is the same ⟶



# Regression: Problem Definition

## Data

**Input**: $x \in \mathbb{R}^d$ (i.e., measurements, covariates, features, indepen. variables)
**Output**: $y \in \mathbb{R}$ (i.e., response, dependent variable)

## Goal

Find a function $f : \mathbb{R}^d \to \mathbb{R}$ such that $y \approx f(x; w)$ for the data pair $(x, y)$.
$f(x; w)$ is called a *regression function*. Its free parameters are $w$.

## Definition of linear regression

A regression method is called *linear* if the prediction $f$ is a linear function of the unknown parameters $w$.

# Least squares linear regression model

## Model

The linear regression model we focus on now has the form

$$y_i \;\approx\; f(x_i; w) = w_0 + \sum_{j=1}^{d} x_{ij} w_j.$$

## Model learning

We have the set of *training data* $(x_1, y_1) \dots (x_n, y_n)$. We want to use this data to learn a $w$ such that $y_i \approx f(x_i; w)$. But we first need an *objective function* to tell us what a "good" value of $w$ is.

## Least squares

The *least squares* objective tells us to pick the $w$ that minimizes the sum of squared errors

$$w_{\text{LS}} = \arg\min_{w} \sum_{i=1}^{n} (y_i - f(x_i; w))^2 \equiv \arg\min_{w} \; \mathcal{L}.$$

# LEAST SQUARES IN PICTURES

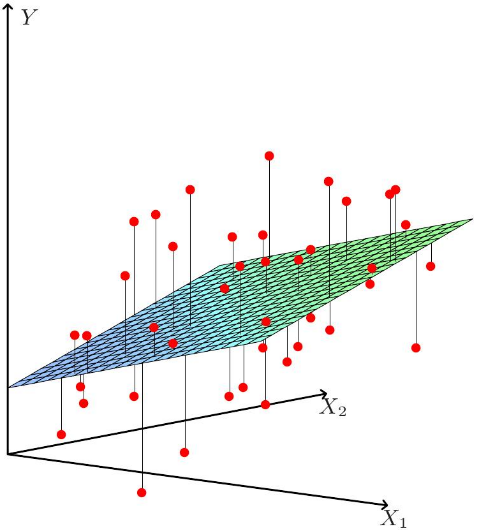
## Observations:

Vertical length is error.

The objective function $\mathcal{L}$ is the sum of all the squared lengths.

Find weights $(w_1, w_2)$ plus an offset $w_0$ to minimize $\mathcal{L}$.

$(w_0, w_1, w_2)$ defines this plane.

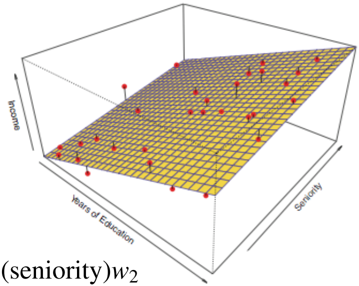


# EXAMPLE: EDUCATION, SENIORITY AND INCOME

## 2-dimensional problem

**Input**: (education, seniority) $\in \mathbb{R}^2$.

**Output**: (income) $\in \mathbb{R}$



**Model**: (income) $\approx w_0 +$ (education)$w_1 +$ (seniority)$w_2$

*Question*: Both $w_1, w_2 > 0$. What does this tell us?

*Answer*: As education and/or seniority goes up, income tends to go up.

(Caveat: This is a statement about correlation, not causation.)

# Least squares linear regression model

## Thus far

We have data pairs $(x_i, y_i)$ of measurements $x_i \in \mathbb{R}^d$ and a response $y_i \in \mathbb{R}$. We believe there is a linear relationship between $x_i$ and $y_i$,

$$y_i = w_0 + \sum_{j=1}^{d} x_{ij} w_j + \epsilon_i$$

and we want to minimize the objective function

$$\mathcal{L} = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - w_0 - \textstyle\sum_{j=1}^{d} x_{ij} w_j)^2$$

with respect to $(w_0, w_1, \dots, w_d)$.

**Can math notation make this easier to look at/work with?**

# NOTATION: VECTORS AND MATRICES

We think of data with $d$ dimensions as a *column* vector:

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix} \quad \text{(e.g.)} \Rightarrow \quad \begin{bmatrix} \text{age} \\ \text{height} \\ \vdots \\ \text{income} \end{bmatrix}$$

A set of $n$ vectors can be stacked into a matrix:

$$\mathbf{X} = \begin{bmatrix} x_{11} & \dots & x_{1d} \\ x_{21} & \dots & x_{2d} \\ \vdots & & \vdots \\ x_{n1} & \dots & x_{nd} \end{bmatrix} = \begin{bmatrix} - \; x_1^T \; - \\ - \; x_2^T \; - \\ \vdots \\ - \; x_n^T \; - \end{bmatrix}$$

Assumptions for now:

- All features are treated as continuous-valued $(x \in \mathbb{R}^d)$
- We have more observations than dimensions ($d < n$)

# NOTATION: REGRESSION (AND CLASSIFICATION)

Usually, for linear regression (and classification) we include an intercept term $w_0$ that doesn't interact with any element in the vector $x \in \mathbb{R}^d$.

It will be convenient to attach a 1 to the first dimension of each vector $x_i$ (which we indicate by $x_i \in \mathbb{R}^{d+1}$) and in the first column of the matrix $X$:

$$x_i = \begin{bmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix}, \qquad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & \dots & x_{1d} \\ 1 & x_{21} & \dots & x_{2d} \\ \vdots & & \vdots & \\ 1 & x_{n1} & \dots & x_{nd} \end{bmatrix} = \begin{bmatrix} 1 - & x_1^T & - \\ 1 - & x_2^T & - \\ & \vdots & \\ 1 - & x_n^T & - \end{bmatrix}.$$

We also now view $w = [w_0, w_1, \dots, w_d]^T$ as $w \in \mathbb{R}^{d+1}$.

# LEAST SQUARES IN VECTOR FORM

Original least squares objective function: $\mathcal{L} = \sum_{i=1}^{n}(y_i - w_0 - \sum_{j=1}^{d} x_{ij}w_j)^2$

Using vectors, this can now be written: $\mathcal{L} = \sum_{i=1}^{n}(y_i - x_i^T w)^2$

## Least squares solution (vector version)

We can find $w$ by setting,

$$\nabla_w \mathcal{L} = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \nabla_w (y_i^2 - 2w^T x_i y_i + w^T x_i x_i^T w) = 0.$$

Solving gives,

$$-\sum_{i=1}^{n} 2y_i x_i + \Big(\sum_{i=1}^{n} 2x_i x_i^T\Big)w = 0 \quad \Rightarrow \quad w_{\text{LS}} = \Big(\sum_{i=1}^{n} x_i x_i^T\Big)^{-1}\Big(\sum_{i=1}^{n} y_i x_i\Big).$$

# LEAST SQUARES IN MATRIX FORM

## Least squares solution (matrix version)

Least squares in matrix form is even cleaner.

Start by organizing the $y_i$ in a column vector, $y = [y_1, \dots, y_n]^T$. Then

$$\mathcal{L} = \sum_{i=1}^{n} (y_i - x_i^T w)^2 = \|y - Xw\|^2 = (y - Xw)^T (y - Xw).$$

If we take the gradient with respect to $w$, we find that

$$\nabla_w \mathcal{L} = 2X^T Xw - 2X^T y = 0 \quad \Rightarrow \quad w_{\text{LS}} = (X^T X)^{-1} X^T y.$$

# RECALL FROM LINEAR ALGEBRA

Recall: Matrix × vector $(X^T y = \sum_{i=1}^{n} y_i x_i)$

$$\begin{bmatrix} | & | & & | \\ x_1 & x_2 & \dots & x_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = y_1 \begin{bmatrix} | \\ x_1 \\ | \end{bmatrix} + y_2 \begin{bmatrix} | \\ x_2 \\ | \end{bmatrix} + \cdots + y_n \begin{bmatrix} | \\ x_n \\ | \end{bmatrix}$$

Recall: Matrix × matrix $(X^T X = \sum_{i=1}^{n} x_i x_i^T)$

$$\begin{bmatrix} | & | & & | \\ x_1 & x_2 & \dots & x_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} - \, x_1^T \, - \\ - \, x_2^T \, - \\ \vdots \\ - \, x_n^T \, - \end{bmatrix} = x_1 x_1^T + \cdots + x_n x_n^T.$$

# LEAST SQUARES LINEAR REGRESSION: KEY EQUATIONS

## Two notations for the *key equation*

$$w_{\text{LS}} = \Big(\sum_{i=1}^{n} x_i x_i^T\Big)^{-1}\Big(\sum_{i=1}^{n} y_i x_i\Big) \quad \Longleftrightarrow \quad w_{\text{LS}} = (X^T X)^{-1} X^T y.$$

## Making Predictions

We use $w_{\text{LS}}$ to make predictions.

Given $x_{\text{new}}$, the least squares prediction for $y_{\text{new}}$ is

$$y_{\text{new}} \approx x_{\text{new}}^T w_{\text{LS}}$$

# LEAST SQUARES SOLUTION

## Potential issues

Calculating $w_{\text{LS}} = (X^T X)^{-1} X^T y$ assumes $(X^T X)^{-1}$ exists.

When doesn't it exist?

> Answer: When $X^T X$ is not a full rank matrix.

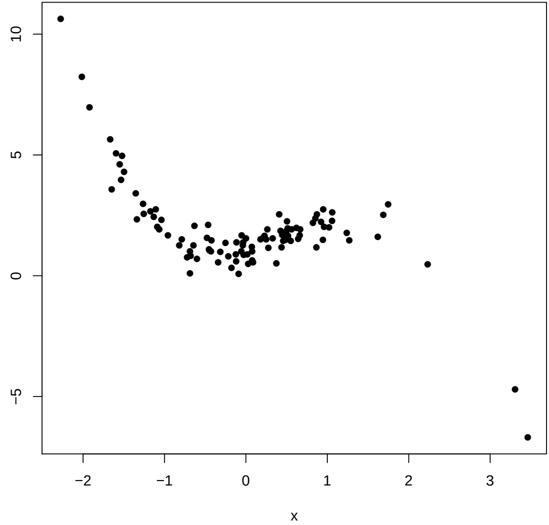
When is $X^T X$ full rank?

> Answer: When the $n \times (d+1)$ matrix $X$ has at least $d+1$ *linearly independent* rows. This means that any point in $\mathbb{R}^{d+1}$ can be reached by a weighted combination of $d+1$ rows of $X$.

Obviously if $n < d+1$, we can't do least squares. If $(X^T X)^{-1}$ doesn't exist, there are an infinite number of possible solutions.

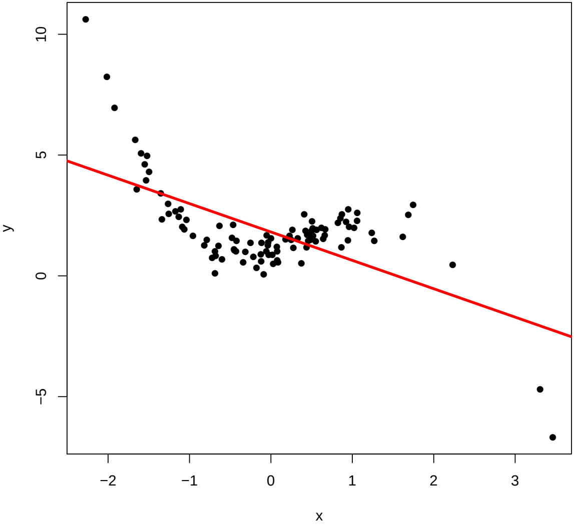
**Takeaway**: We want $n \gg d$ (i.e., $X$ is "tall and skinny").

# BROADENING LINEAR REGRESSION



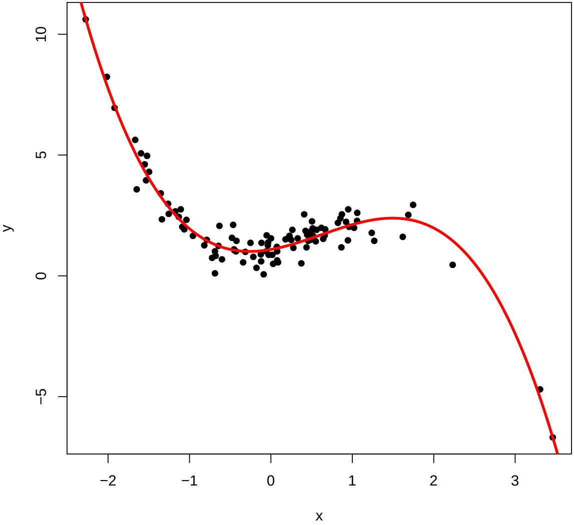
# Broadening linear regression

$$y = w_0 + w_1 x$$



# Broadening linear regression

$$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$



# Polynomial regression in $\mathbb{R}$

## Recall: Definition of linear regression

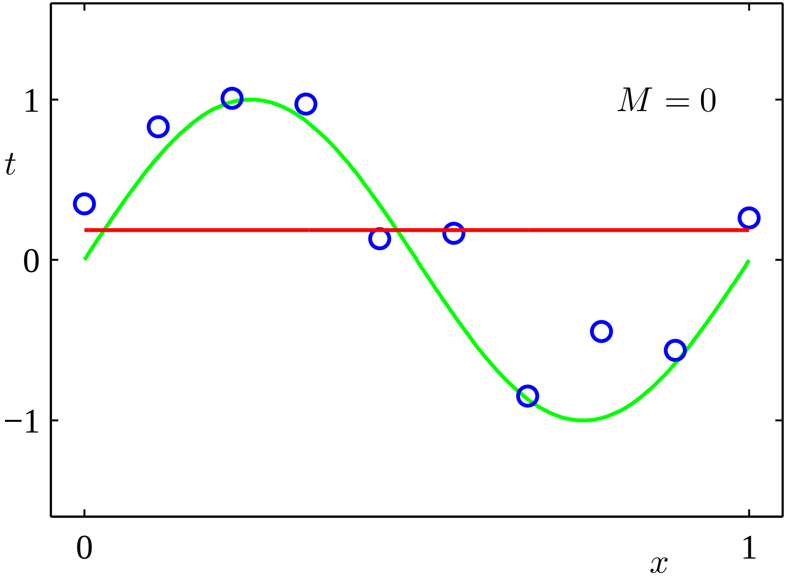
A regression method is called *linear* if the prediction $f$ is a linear function of the unknown parameters $w$.

- Therefore, a function such as $y = w_0 + w_1 x + w_2 x^2$ is *linear* in $w$. The LS solution is the same, only the preprocessing is different.
- E.g., Let $(x_1, y_1) \dots (x_n, y_n)$ be the data, $x \in \mathbb{R}$, $y \in \mathbb{R}$. For a $p$th-order polynomial approximation, construct the matrix

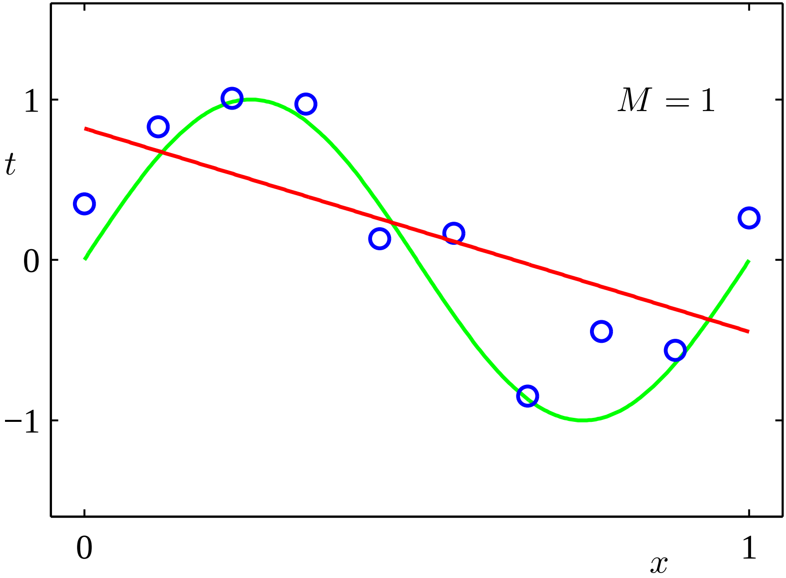
$$\mathbf{X} = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^p \\ 1 & x_2 & x_2^2 & \dots & x_2^p \\ \vdots & & & \vdots & \\ 1 & x_n & x_n^2 & \dots & x_n^p \end{bmatrix}$$

- Then solve exactly as before: $w_{\text{LS}} = (X^T X)^{-1} X^T y$.

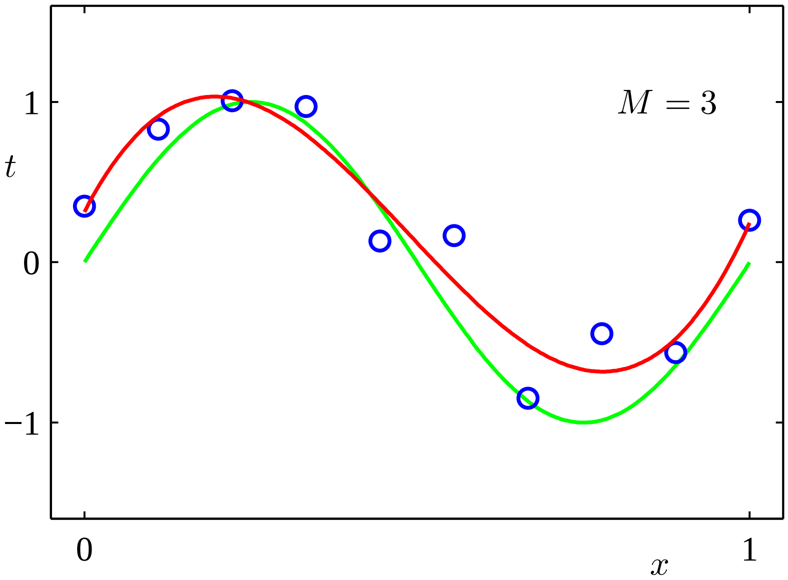
# Polynomial regression ($M$th order)



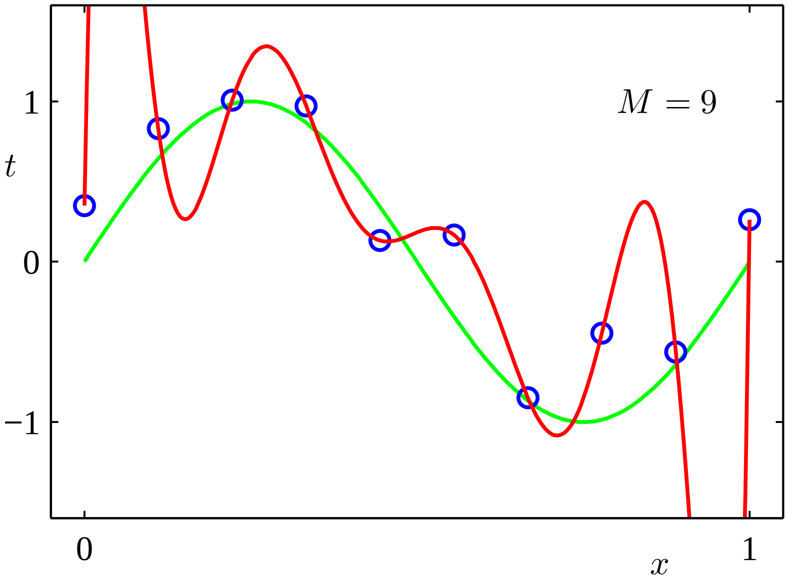
# Polynomial regression ($M$th order)



# Polynomial regression ($M$th order)



# Polynomial regression ($M$th order)



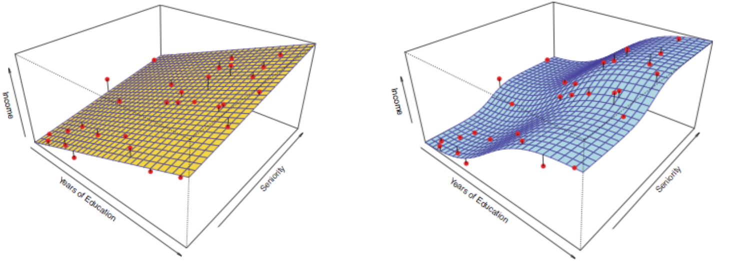
# Polynomial regression in two dimensions

## Example: 2nd and 3rd order polynomial regression in $\mathbb{R}^2$

The width of $X$ grows as (order) × (dimensions) + 1.

$$\text{2nd order:} \qquad y_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i1}^2 + w_4 x_{i2}^2$$

$$\text{3rd order:} \qquad y_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i1}^2 + w_4 x_{i2}^2 + w_5 x_{i1}^3 + w_6 x_{i2}^3$$



(a) 1st order

(b) 3rd order

# Further extensions

More generally, for $x_i \in \mathbb{R}^{d+1}$ least squares linear regression can be performed on functions $f(x_i; w)$ of the form

$$y_i \approx f(x_i, w) = \sum_{s=1}^{S} g_s(x_i) w_s.$$

For example,

$$\begin{aligned} g_s(x_i) &= x_{ij}^2 \\ g_s(x_i) &= \log x_{ij} \\ g_s(x_i) &= \mathbb{I}(x_{ij} < a) \\ g_s(x_i) &= \mathbb{I}(x_{ij} < x_{ij'}) \end{aligned}$$

As long as the function is *linear* in $w_1, \dots, w_S$, we can construct the matrix $X$ by putting the transformed $x_i$ on row $i$, and solve $w_{\text{LS}} = (X^T X)^{-1} X^T y$.

One caveat is that, as the number of functions increases, we need more data to avoid overfitting.

# Geometry of least squares regression

Thinking geometrically about least squares regression helps a lot.

- We want to minimize $\|y - Xw\|^2$. Think of the vector $y$ as a point in $\mathbb{R}^n$. We want to find $w$ in order to get the product $Xw$ close to $y$.
- If $X_j$ is the $j$th *column* of $X$, then $Xw = \sum_{j=1}^{d+1} w_j X_j$.
- That is, we weight the columns in $X$ by values in $w$ to approximate $y$.
- The LS solutions returns $w$ such that $Xw$ is as close to $y$ as possible in the Euclidean sense (i.e., intuitive “direct-line” distance).

# Geometry of least squares regression

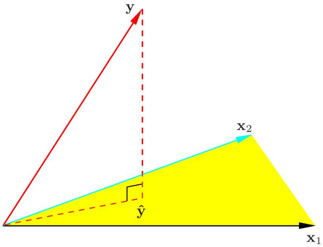
$$\arg\min_w \|y - Xw\|^2 \quad \Rightarrow \quad w_{\text{LS}} = (X^T X)^{-1} X^T y.$$

The columns of $X$ define a $d+1$-dimensional subspace in the higher dimensional $\mathbb{R}^n$.

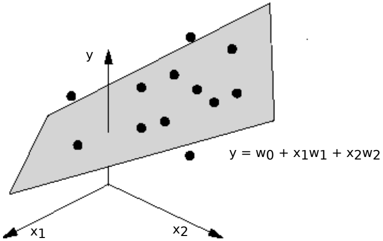
The closest point in that subspace is the *orthonormal projection* of $y$ into the *column space* of $X$.

Right: $y \in \mathbb{R}^3$ and data $x_i \in \mathbb{R}$.
$X_1 = [1, 1, 1]^T$ and $X_2 = [x_1, x_2, x_3]^T$

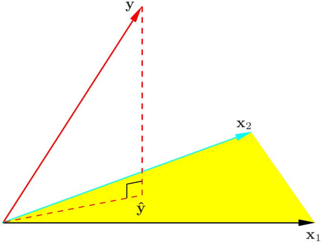
The approximation is $\hat{y} = Xw_{\text{LS}} = X(X^T X)^{-1} X^T y$.



# GEOMETRY OF LEAST SQUARES REGRESSION



(a) $y_i \approx w_0 + x_i^T w$ for $i = 1, \dots, n$



(b) $y \approx Xw$

There are some key difference between (a) and (b) worth highlighting as you try to develop the corresponding intuitions.

(a) Can be shown for all $n$, but only for $x_i \in \mathbb{R}^2$ (not counting the added 1).

(b) This corresponds to $n = 3$ and one-dimensional data: $X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \end{bmatrix}$.