COMP6237 – Linear Regression

Shoaib Ehsan

s.ehsan@soton.ac.uk

Lecture slides available here:

http://comp6237.ecs.soton.ac.uk/

(Credit goes to Jon Hare who developed a large part of the module)

General Plan

- At the start of the course:
  - An introduction to regression techniques
  - An introduction to information theory
- These lectures are based on the stats package R
  - This is free software, you can download it from

    https://www.r-project.org/
  - If you are not familiar with R, follow a tutorial to get some idea:

    https://www.southampton.ac.uk/~mb1a10/Rtutorial/R.html
- At the end of the course:
  - Mining Data Streams

COMP6237: Linear Regression

- Outline:
  - Brief revision of some basic stats
  - Variables and prediction
  - The method of least squares (LS)
  - Practical implementation in R
  - Linear regression in higher dimensions
  - Maximum likelihood estimation (MLE)
  - LS and MLE
  - Weighted LS, Heteroskedasticity, and local linear regression

Reminder of Some Basic Stats (1)

- Suppose we have a set of N observations/measurements $\{X_1, X_2, \ldots, X_N\}$
- Can analyse them via histograms/pdfs
- How to classify distributions?
  - Means (Median, mode, etc.)

$$E[X] = 1/N \sum_{i=1}^{N} X_i$$

  - Variances/standard deviation

$$V[X] = 1/N \sum_{i=1}^{N} (X_i - E[X])^2 = E[X^2] - E[X]^2$$

  - Could use the Central Limit Theorem to argue about standard errors, confidence intervals, etc.

Reminder of Some Basic Stats (2)

- We are often interested in relationships between pairs of observations

$$\{(X_1, Y_1), (X_{2,} Y_2), \ldots, (X_N, Y_N)\}$$

- Can classify relationships in various ways
  - Covariance

$$Cov[X,Y] = 1/N \sum_{i=1}^{N} (X_i - E[X])\,(Y_i - E[Y])$$

  - Correlation coefficients, e.g. Pearson

$$r = \frac{Cov[X,Y]}{\sqrt{V[X]}\sqrt{V[Y]}}$$

  - R^2 … proportion of variance of X explained by knowledge of Y

What about Prediction?

- What about if we want to predict the value of one variable based on the knowledge about another variable?
- This is what regression analysis is all about

Interlude: Types of Variables (1)

- Three types of variables:
  - **Continuous**: real numbered values (e.g., time, mass)
  - **Ordinal**: a numerical variable where a small number of possibilities are ranked (e.g., school grades, Michelin stars)
    - Boundaries between first two types can be blurred (e.g. most “continuous” variables are really ordinal because they can only be measured up to some accuracy
  - **Categorical**: describes membership of a group (but cannot be ranked). e.g., country of birth, gender, etc.
    - Some are binary and others are not

Interlude: Types of Variables (2)

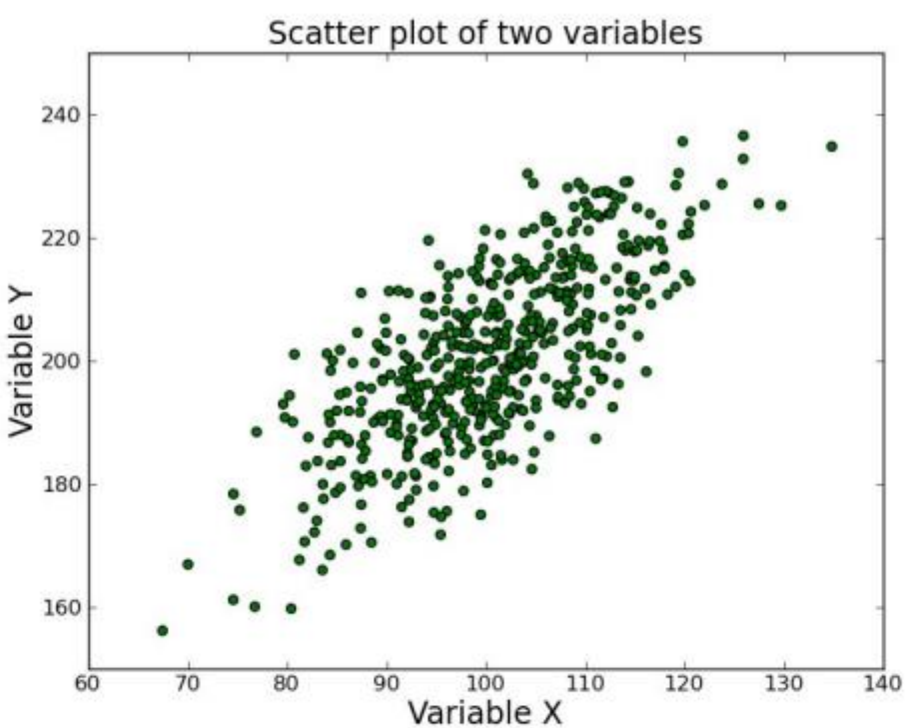
- Whatever form a variable has, they can play different roles when we build statistical models
- **Dependent** or **outcome** variables
  - For our analysis this variable is the focus. It is assumed to be predictable from some other variables.
- **Independent** or **predictor** variables
  - Are assumed to have inherent variation (“they just are”). We will use them to explain the variance in the dependent variables.

Example: Demographic Factors

- Let's assume we want to predict driving test outcomes from demographic data.
- For this particular analysis:
  - Independent variables: demographic factors like gender, age, weight, height, etc.
  - Dependent variables: driving test outcomes like outcome of the practical/theoretical parts
- Role of variables depends on analysis/model we have in mind.
  - We could equally well be interested in the inverse, i.e., predicting demographic factors based on knowledge of driving test outcomes.

Variables in Regression

- Simplest case involves one dependent (Y) and one independent (X) variable
- Both variables are continuous (or at least ordinal)
- We are trying to predict the variation in Y based on the variation in X

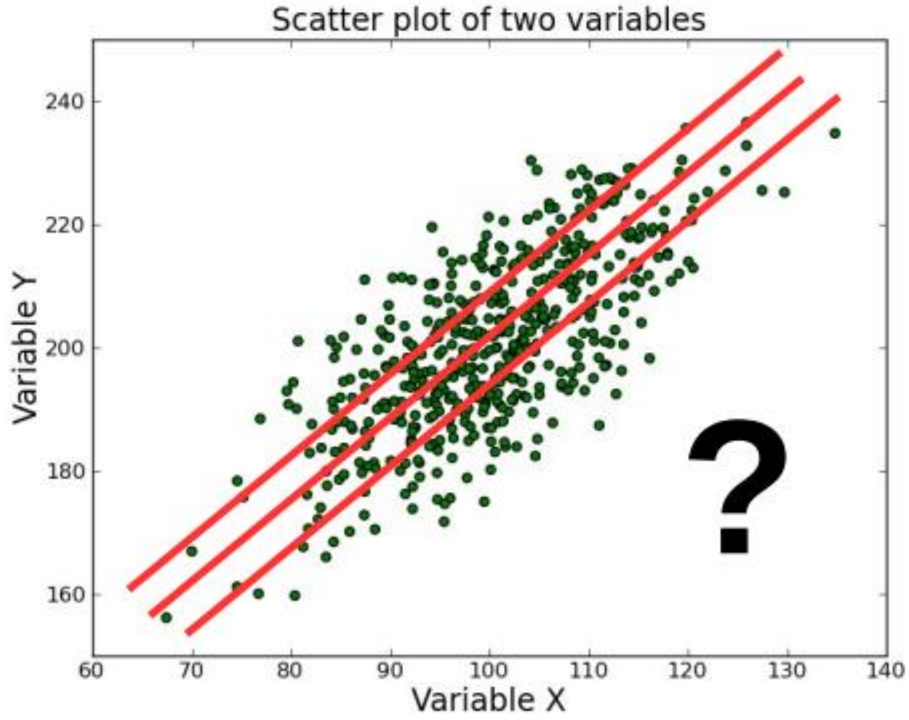


Given a plot like this:
What are your options?

Easiest way:
Assume linear relationship between X and Y

Try to find line of "best guess"

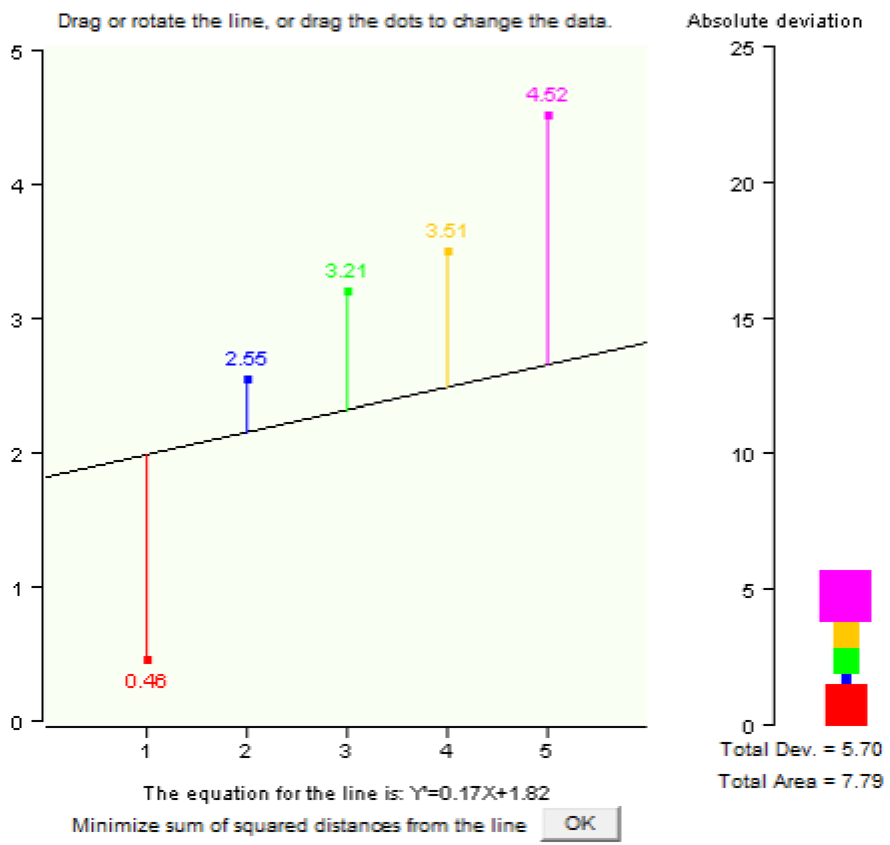
Drawing a Line Through Some Points



- Look for line of “best fit” through cloud of X,Y points
- Y=mX+b, Need to find two parameters m and b
- How to be systematic about it?

Method of Least Squares

- For any given line mX+b we can measure the difference between our actual Y values and those predicted
- **Squared differences** are a reasonable measure of the goodness of fit; regression analysis tries to minimize this difference
- To play around with this explore David Lane's demo:



http://onlinestatbook.com/simulations/reg_least_squares/reg_ls.html

LS as an Optimisation Problem

- One might think that minimizing the squared differences is a difficult combinatorial optimisation problem … it is not:

$$E(m,b) = \sum_{i=1}^{N} (mX_i + b - Y_i)^2 = min!$$

$$\frac{\partial}{\partial b} E(m,b) = 2\sum_{i=1}^{N} (mX_i + b - Y_i) = 0$$

$$\longrightarrow \qquad b = E[Y] - mE[X]$$

- Substituting back:

$$E(m,b) = \sum_{i=1}^{N} \big(m(X_i - E[X]) - (Y_i - E[Y])\big)^2$$

$$\frac{\partial}{\partial m} E(m,b) = 2\sum_{i=1}^{N} \big(m(X_i - E[X]) - (Y_i - E[Y])\big)(X_i - E[X]) = 0$$

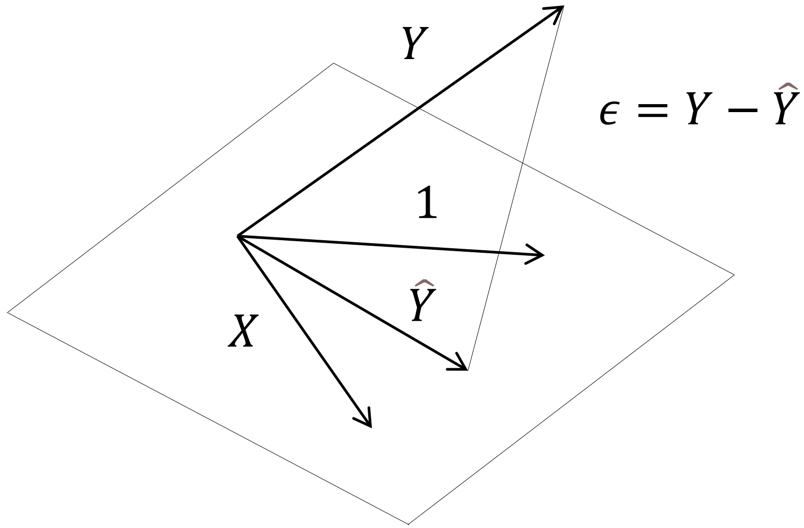
$$m = \frac{\sum_{i=1}^{N} (Y_i - E[Y])(X_i - E[X])}{\sum_{i=1}^{N} (X_i - E[X])^2} = Cov\,[X,Y]/V\,[X]$$

Geometric Interpretation

- Have n data points and want that our best "guess" fulfills $\hat{y_i} = b + mx_i, i = 1, \ldots, n$
- Can define vectors

$$X = (x_1, \ldots, x_n)^T \quad Y = (y_1, \ldots, y_n)^T \quad \hat{Y} = (\widehat{y_1}, \ldots, \hat{y}_n)^T \longrightarrow \hat{Y} = b1 + mX$$

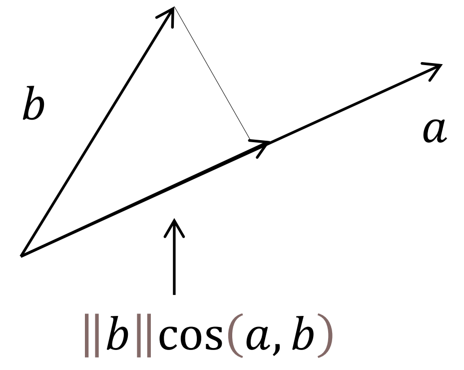
- This says that hat Y is in span {1, X}, but this is usually not the case



- Hat Y that minimizes deviation ε^2 is orthogonal projection of Y into plane spanned by X and 1!

Geometric Interpretation (2)

- Thus, an alternative way of calculating hat Y is by projecting Y into this plane. Strategy:
  - Projection can be calculated by projecting Y onto orthogonal basis vectors of the plane and then adding these projections together
  - Need orthogonal basis vectors of span{X,1}
- Projection of a vector onto another vector?



$$proj_a(b) = \frac{b^T a}{a^T a} a$$

$$proj_a(b) = \frac{a}{\|a\|}\|b\|\cos(a,b) = \frac{b^T a}{\|a\|^2} a = \frac{b^T a}{a^T a} a$$

Geometric Interpretation (3)

- To obtain orthogonal basis vectors of span{1,X} we use 1 and

"centred" vector X

$$X - proj_1(X) = X - \frac{X^T 1}{1^T 1} 1 = X - \frac{\sum_{i=1}^{n} x_i}{n} = X - E[X] = \bar{X}$$

- Hence: $X = E[X]1 + \bar{X}$

$$\hat{Y} = proj_1(Y) + proj_{\bar{X}}(Y)$$

$$= \frac{Y^T 1}{1^T 1} 1 + \frac{Y^T \bar{X}}{\bar{X}^T \bar{X}} \bar{X} = E[Y]1 + \frac{Y^T \bar{X}}{\bar{X}^T \bar{X}} \bar{X}$$

$$= b1 + mX$$

$$= b1 + m(E[X]1 + \bar{X})$$

$$= (b + mE[X])1 + m\bar{X}$$

Geometric Interpretation (3)

- To obtain orthogonal basis vectors of span{1,X} we use 1 and

"centred" vector X

$$X - proj_1(X) = \frac{X^T 1}{1^T 1} 1 = X - \frac{\sum_{i=1}^{n} x_i}{n} = X - E[X]$$

- Hence: $X = E[X]1 + \bar{X}$

$$\hat{Y} = proj_1(Y)1 + proj_{\bar{X}}(Y)\bar{X}$$

$$= \frac{Y^T 1}{1^T 1} 1 + \frac{Y^T \bar{X}}{\bar{X}^T \bar{X}} \bar{X} = E[Y]1 + \frac{Y^T \bar{X}}{\bar{X}^T \bar{X}} \bar{X}$$

$$= b1 + mX \qquad = b1 + m(E[X]1 + \bar{X})$$

$$= (b + mE[X])1 + m\bar{X}$$

$$\longrightarrow \quad b = E[Y] - mE[X] \qquad m = \frac{Y^T \bar{X}}{\bar{X}^T \bar{X}}$$

Geometric Interpretation (4)

- What remains to be checked is whether

$$m = Cov[X,Y]/V[X]$$

- Observe that

$$m = \frac{Y^T \bar{X}}{\bar{X}^T \bar{X}} = \frac{Y^T (X - E[X]1)}{\|X - E[X]1\|^2}$$

$$= \frac{\sum_i x_i y_i - nE[X]E[Y]}{\sum_i (x_i - E[X])^2}$$

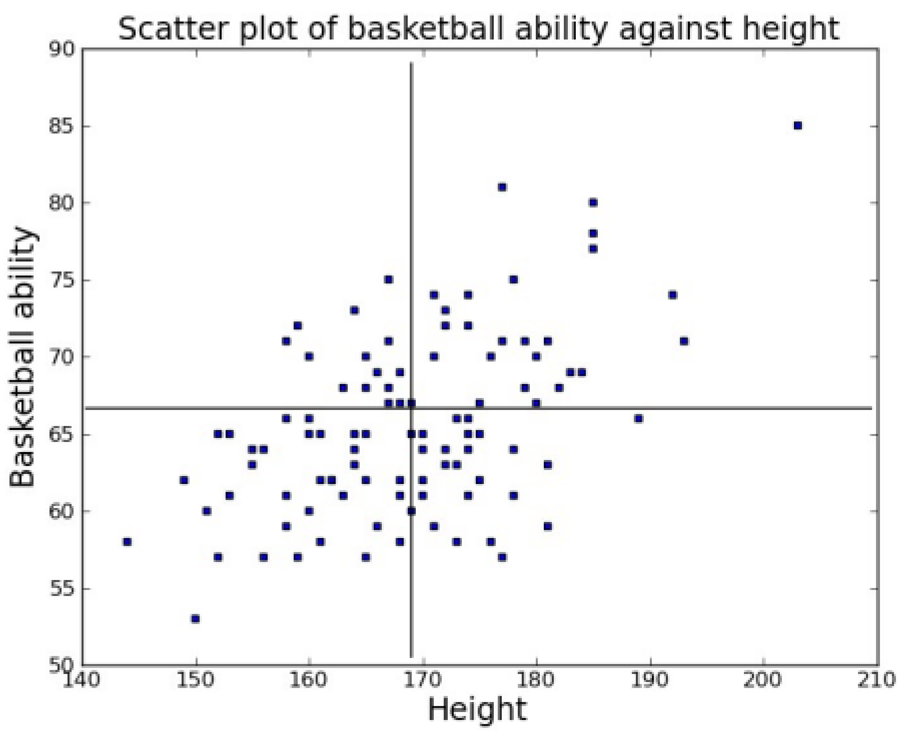
$$= Cov[X,Y]/V[X]$$

Regression in Machine Learning

- Slightly different point of view is that we might consider the pairs (Xi,Yi) as a training set
- Want to learn a mapping y=f(x) from the training set
- Approach often:
  - “somehow” parametrise f(x) (e.g., f(x)=mx+b here) and try to learn best parameters m and b
  - This is often done via minimising some error function E which sums up squared residual errors over training set (i.e., this is the same as above)
  - e.g., in NN f(x) is a nonlinear function

How Do I Run Linear Regression in R?

- Let's start with a fictional data set
  - Say a 100 men have been measured for their height and basketball ability
  - We want to predict basketball skill from height



Mean height is 169cm, stddev 10.5cm
Mean ability is 66 (arbitrary scale), stdev 6

Correlation r=0.52 (i.e. 28% of the variation of basketball ability are explained by height)

R Commands ...

- Read data into R with the usual command read.table (); the variables of interest are Height and BasketballAbility
- Build a regression model with
- regmodel=lm (BasketballAbility ~ Height)
- (lm stands for “linear model”)
- summary (regmodel) gives us most of the information we need ...

The Full Output

```
Residuals:
     Min       1Q   Median       3Q      Max
-11.0733  -3.4851  -0.5733   3.4969  12.9267

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 15.67832    8.22351   1.907   0.0595 .
Height       0.29602    0.04855   6.097 2.15e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.07 on 98 degrees of freedom
Multiple R-squared: 0.275,      Adjusted R-squared: 0.2676
F-statistic: 37.17 on 1 and 98 DF,  p-value: 2.146e-08
```

The Full Output

```
Residuals:
     Min       1Q   Median       3Q      Max 
-11.0733  -3.4851  -0.5733   3.4969  12.9267 
```

Characterizes the distribution of “residuals” (differences between predicted output and actual output)

Should roughly be normally distributed with a mean of zero

The Full Output

This is the important test here, i.e.
the hypothesis of a relationship (slope !=0)
is confirmed.

```
Residuals:
     Min       1Q   Median       3Q      Max
-11.0733  -3.4851  -0.5733   3.4969  12.9267

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 15.67832    8.22351   1.907   0.0595 .
Height       0.29602    0.04855   6.097 2.15e-08 ***
```

The coefficients (“m” and “b”).

Meaning … a man of zero height would have basketball ability 15.7,
Every additional cm of height adds 0.3 to basketball ability.

Also get standard errors, t-tests for hypothesis that values are 0

The Full Output

Summary of the analysis, we get

- an R^2 value (variation explained, as expected, see earlier) and an adjusted R^2 to account for the fact that models with more parameters are expected to perform better
- Finish with an F-test on the model as a whole with degrees of freedom 1 and 98 – tests the significance of the model
- Why 1 and 98?

  In a data set with 100 data points there are 99 free to vary we always want the have the same mean;

  Intercept also not free to vary (goes through joint means)

  → 1 DOF for model, 98 for error variance

```
Residual standard error: 5.07 on 98 degrees of freedom
Multiple R-squared: 0.275,      Adjusted R-squared: 0.2676
F-statistic: 37.17 on 1 and 98 DF,  p-value: 2.146e-08
```

Generalisations

- Multi-dimensional input?
  - "fit" a hyperplane

$\mathbf{x} \in R^D, y \in R$

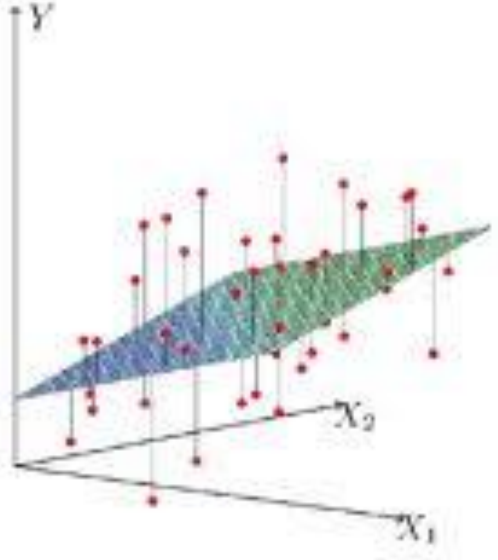


Figure 3.1: *Linear least squares fitting with $X \in \mathbb{R}^2$. We seek the linear function of X that minimizes the sum of squared residuals from Y.*

- A general linear regression function then is

$$f(\vec{x}) = \sum_{j=1}^{D} w_j x_j + b = \mathbf{wx} + b$$

- For simpler notation we could say

$$\widetilde{\mathbf{w}} = [w_1, w_2, \ldots, w_D, b]^T, \tilde{\mathbf{x}} = [x_1, x_2, \ldots, x_D, 1]^T$$

- Hence:

$$E = \sum_{i=1}^{N} (y_i - \widetilde{\mathbf{w}}\widetilde{\mathbf{x}}_i)^2$$

Generalisations (2)

- For a more compact notation say

$$\widetilde{X} = \begin{bmatrix} \widetilde{\boldsymbol{x}}_1^T & . \\ \widetilde{\boldsymbol{x}}_2^T & . \\ . & . \\ . & . \\ \widetilde{\boldsymbol{x}}_N^T & . \end{bmatrix} \qquad \boldsymbol{y} = [y_1, y_2, \ldots, y_N]^T$$

- Then: $E = \sum_{i=1}^{N} (y_i - \widetilde{\boldsymbol{w}}\, \widetilde{\boldsymbol{x}}_i)^2 = \|\boldsymbol{y} - \widetilde{X}\, \widetilde{\boldsymbol{w}}\|^2 = (\boldsymbol{y} - \widetilde{X}\, \widetilde{\boldsymbol{w}})^T (\boldsymbol{y} - \widetilde{X}\, \widetilde{\boldsymbol{w}})$
- Can find w through $dE/dw_i = 0$ ...

$$\widetilde{\boldsymbol{w}} = (\widetilde{X}^T \widetilde{X})^{-1} \widetilde{X}^T \boldsymbol{y} = X^+ \boldsymbol{y}$$

↑

pseudoinverse of X also denoted by X^+

Generalisations (3)

- Some practical considerations
  - The data matrix X can have large dimensions and we might need to calculate an inverse … this can be quite time consuming
- Two ways to go about it:
  - Often via **QR factorization**, factorizing $\tilde{X} = QR$
  - with Q a diagonal matrix and R an upper triangular matrix (can be done via applying Gram-Schmidt procedure to column vectors) → for exact solutions
  - Alternatively: can use **stochastic gradient descent** to numerically minimize residuals

Stochastic Gradient Descent

- Stochastic gradient ascent

$$E = \sum_{i=1}^{N} (y_i - \tilde{\mathbf{w}}\tilde{\mathbf{x}}_i)^2 = \left\| \mathbf{y} - \tilde{X}\tilde{\mathbf{w}} \right\|^2 = \left(\mathbf{y} - \tilde{X}\tilde{\mathbf{w}}\right)^T \left(\mathbf{y} - \tilde{X}\tilde{\mathbf{w}}\right)$$

$$= \mathbf{y}^{\mathbf{T}}\mathbf{y} - 2\tilde{\mathbf{w}}^T \tilde{X}^T \mathbf{y} + \tilde{\mathbf{w}}^T \tilde{X}^T \tilde{X} \tilde{\mathbf{w}}$$

$$\longrightarrow \qquad \frac{\partial E}{\partial \tilde{\mathbf{w}}} = -2\tilde{X}^T \mathbf{y} + 2\tilde{X}^T \tilde{X} \tilde{\mathbf{w}}$$

- Gradient descent: starting from initial weight vector iteratively update

$$\tilde{\mathbf{w}}^{t+1} = \tilde{\mathbf{w}}^t - \eta \frac{\partial E}{\partial \tilde{\mathbf{w}}} = \tilde{\mathbf{w}}^t + \eta \tilde{X}^T \left(\mathbf{y} - \tilde{X}\tilde{\mathbf{w}}^t\right)$$

- Stochastic gradient descent: restrict to a single point of the training data; processing them in random order

Stochastic Gradient Descent

- Stochastic gradient ascent
  - Instead of $\frac{\partial E}{\partial \tilde{\mathbf{w}}} = -\tilde{X}^T \mathbf{y} + \tilde{X}^T \tilde{X} \tilde{\mathbf{w}}$
  - use gradient at training point k: $\frac{\partial E}{\partial \tilde{\mathbf{w}}}(x_k) = -x_k y_k + x_k x_k^T \tilde{\mathbf{w}}$
- Then, e.g., iterate through training points in random order until some tolerance has been reached such that (norm) difference between updates in the w’s becomes small enough.

Generalisations (4)

- D dimensional input, K dimensional output; could say:

$$\boldsymbol{y}=\widetilde{W}^T\widetilde{\boldsymbol{x}} \qquad \widetilde{W}=[\widetilde{\boldsymbol{w}}_1,\widetilde{\boldsymbol{w}}_2,\dots,\widetilde{\boldsymbol{w}}_K]=\begin{bmatrix}\boldsymbol{w}_1 & \boldsymbol{w}_2 & \dots & \boldsymbol{w}_K\\ b_1 & b_2 & \dots & b_K\end{bmatrix}$$

This is useful since: $y_j=\widetilde{w}_j\widetilde{\boldsymbol{x}}$

- An error function can be constructed by summing over all pairs and output dimensions

$$E=\sum_{i=1}^{N}\sum_{j=1}^{K}(y_{i,j}-\widetilde{\boldsymbol{w}}_{\boldsymbol{j}}\widetilde{\boldsymbol{x}}_i)^2$$

- Introduce: $\boldsymbol{y}_{\boldsymbol{j}}'=[y_{1,j},y_{2,j},\dots,y_{N,j}]^T \quad \widetilde{X}^T=[\widetilde{x}_1,\widetilde{x}_2,\dots,\widetilde{x}_N]$

$$\rightarrow E=\sum_{j=1}^{K}\|\boldsymbol{y}'_j-\widetilde{X}\widetilde{\boldsymbol{w}}_j\|^2$$

- i.e. we have K distinct estimation problems with solutions $\widetilde{\boldsymbol{w}}_j=X^+\boldsymbol{y}'_j$

Some Comments

- One may have wondered if different results would have been obtained when making a different choice about the difference function between predictions and data points?
  - i.e., what if one wouldn't use the sum of the squares (~L_2 norm) but some other measure? → other results would have been obtained!
- Least squares is a very common strategy in the sciences
- Another prominent approach is maximum likelihood estimates

Maximum Likelihood Estimation

- Suppose we have a set of n data points X1,…,Xn which are from some pdf f(x;p) which has some "hidden" parameters p. We want to "guess" these parameters.
- How to go about it? Construct a likelihood function:

$$L(X_1, X_2, \ldots, X_n; p) = \prod_{i=1}^{n} f(X_i; p)$$

↗

Likelihood of obtaining the data by sampling from f given the parameter p

- Maximising L will give us a value of p corresponding to the most likely explanation of the data

Example (1)

- Suppose our observations have been generated from a normal distribution with unknown mean and variance, i.e.,

$$X \sim N(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$

- Then the likelihood function is:

$$L(X_1, \ldots, X_n; \mu, \sigma^2) = \sigma^{-n}(2\pi)^{-n/2} \exp\left(\frac{-1}{2\sigma^2}\left((X_1-\mu)^2 + (X_2-\mu)^2 + \ldots + (X_n-\mu)^2\right)\right)$$

- L is max if $\sum_{i=1}^{N}(X_i-\mu)^2 = min!$ which is if $\mu = 1/N \sum_{i=1}^{N} X_i$
- Hence the expectation is a **maximum likelihood estimator** for this example.

Example (2)

- Suppose we have n observations X1,…,Xn that have been sampled from the uniform distribution over [0,N] and N is unknown.

Example (2)

- Suppose we have n observations X1,…,Xn that have been sampled from the uniform distribution over [0,N] and N is unknown.
- Construct likelihood function

$$L(X_1, \dots, X_n; N) = \begin{cases} 0 & any X_i outside [0, N] \\ (1/N)^n & otherwise \end{cases}$$

- Maximum likelihood estimator?

Example (2)

- Suppose we have n observations X1,…,Xn that have been sampled from the uniform distribution over [0,N] and N is unknown.
- Construct likelihood function

$$L(X_1,\ldots,X_n;N) = \begin{cases} 0 & any X_i outside [0,N] \\ (1/N)^n & otherwise \end{cases}$$

- Maximum likelihood estimator is $N = max(X_1, X_2, \ldots, X_n)$
- A problem of MLE is that estimators can be biased … to see this here:

Example (2)

- Construct pdf for N, start with cumulative pdf:

$$P(N \leq x) = P(X_1 \leq x, X_2 \leq x, \ldots, X_n \leq x)$$

$$P(N \leq x) = P(X_1 \leq x) P(X_2 \leq x) \ldots P(X_n \leq x)$$

$$P(N \leq x) = P(X_1 \leq x)^n = (x/N)^n$$

- Obtain the pdf as

$$f(x) = d/dx\, P(N \leq x) = \begin{cases} 0 & x \notin [0, N] \\ n\, x^{n-1}/N^n & x \in [0, N] \end{cases}$$

- And hence

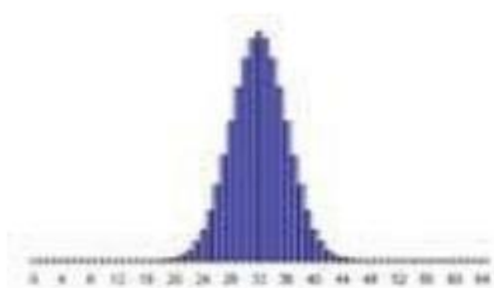
$$E[N] = \int_0^N x\, f(x) dx = \int_0^N n\, x^n / N^n\, dx = nN/(n+1) \neq N!$$

i.e., this ML estimator is not unbiased!

Implied Model of Data Generation in Linear Regression

Intercept +

Slope × This man's height

+ 

Random noise

= His basketball ability score

MLE and LS

- Suppose we know a priori that X and Y have a linear relationship except for some noise, i.e.,

$$Y_i = mX_i + b + \epsilon_i, \epsilon_i \sim N(0, \sigma^2)$$

- where the epsilons are independent. We aim to find m and b via MLE:

$$L(X_1, \ldots, X_n; m, b) = (2\pi)^{-n/2}\sigma^{-n}\exp\left(\frac{-1}{2\sigma^2}\sum_i (Y_i - mX_i - b)^2\right)$$

- To maximise L we need to minimise the sum of squares in the exponent, i.e.,
  - Least squares is equivalent to an MLE estimate for m and b if X and Y are **linearly** related with **Gaussian noise**. Sum of squares has a privileged position ...

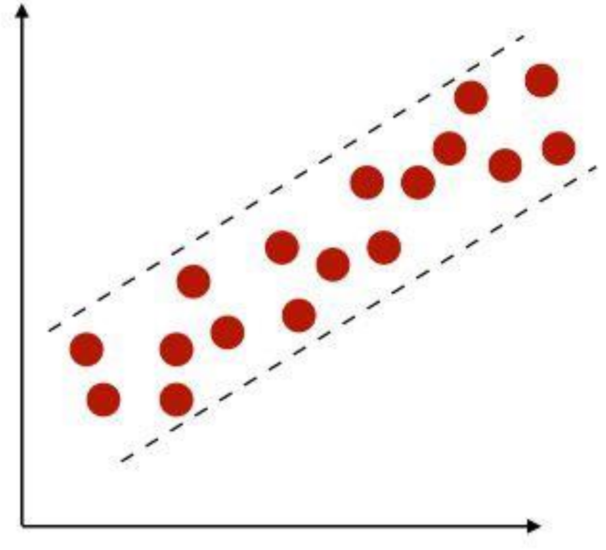
Weighted Least Squares

- Instead of minimising $\sum_{i=1}^{N} (mX_i + b - Y_i)^2 = min!$
- one might want to minimise: $\sum_{i=1}^{N} w_i \, (mX_i + b - Y_i)^2 = min!$
- Why?
  - To focus accuracy – might be more interested in certain X-regions, or errors in some regions might be more costly than in others
  - There is a number of other optimisation problems transformed/approximated by WLS, e.g., generalised linear models where the response is some nonlinear function of a linear predictor (e.g., logistic regression, see later)

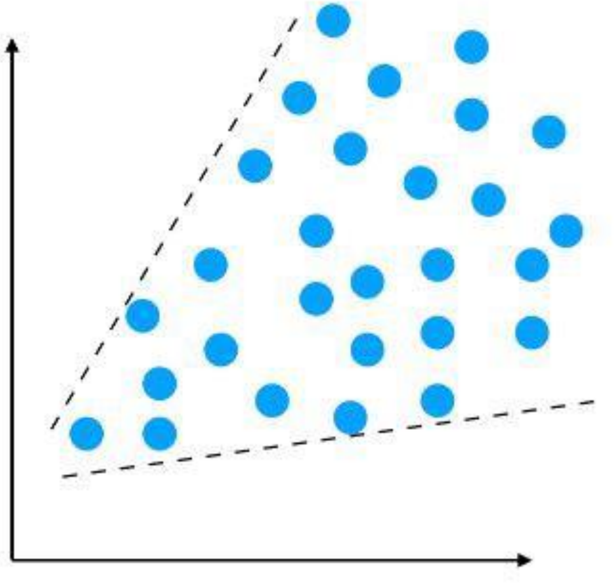
Homo-/Heteroskedasticity

- Discounting imprecision
  - Ordinary least squares assumes y=mx+b+e with e iid Gaussian white noise. This implies that e has constant variance (**homoskedasticity**).
  - Often this is not the case → **heteroskedastic** data
  - Can then set $w_i=1/s^2_i$ so we get the heteroskedastic MLE
  - (in other words: does not make much sense to concentrate on noisy parts of the data, want to use parts with little noise for our estimates)

Homo-/Heteroskedasticity – Examples



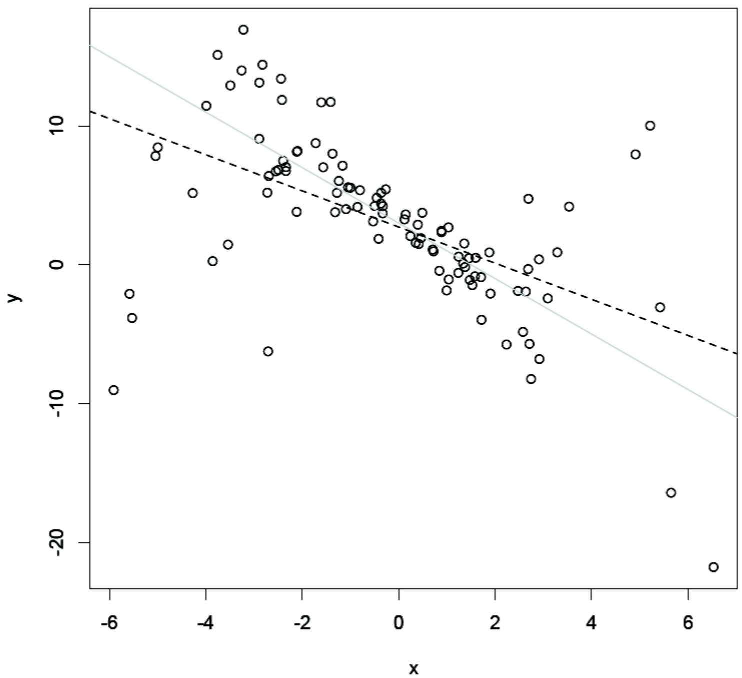
Homoscedasticity

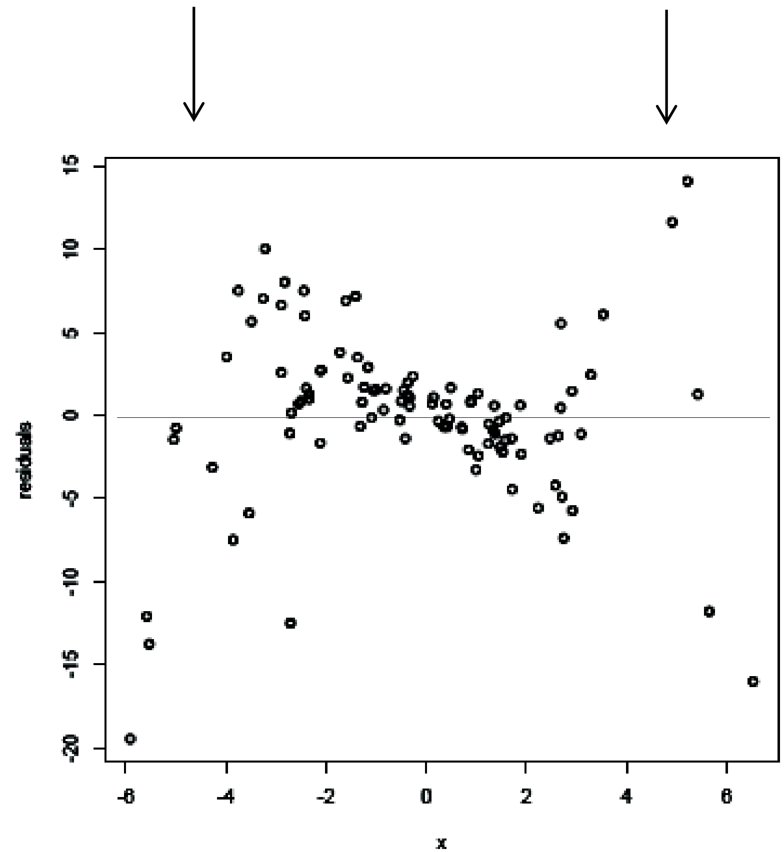


Heteroscedasticity

An Example in R

- Suppose we have a linear relationship y=3-2x between X and Y, and add Gaussian white noise with $s(x)=1+0.5x^2$



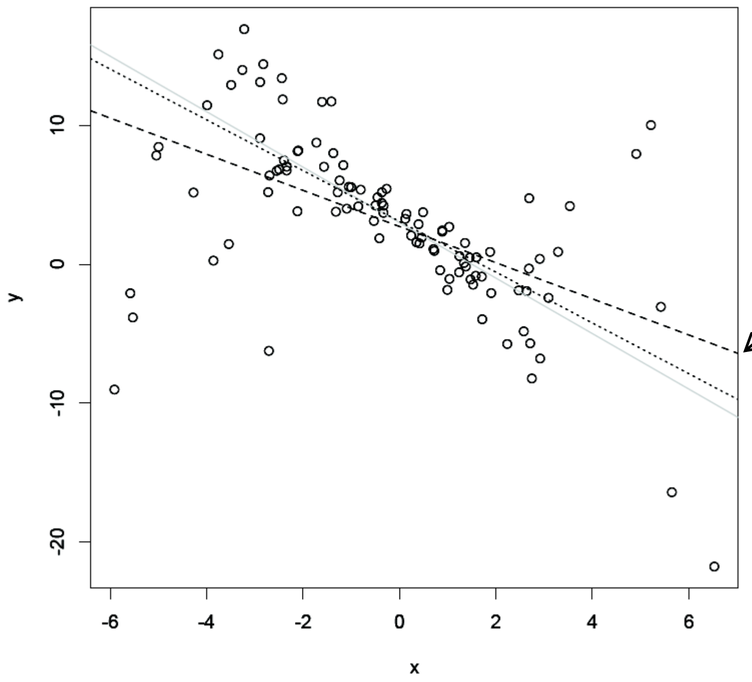



```
x=rnorm(100,0,3)
Y=3-2*x+rnorm(100, 0, sapply(x,function{x}{1+0.5*x^2}))
plot(x,y)
abline(a=3,b=-2,col=”grey”)
fit.ols= lm(y~x)
abline(fit.ols$coefficients,lty=2)
```

```
plot(x,fit.old$residuals)
```

As expected, variance is not constant!
Fit misses the real relationship by some margin.

Weighted Linear Regression



unweighted regression

weighted regression performs better

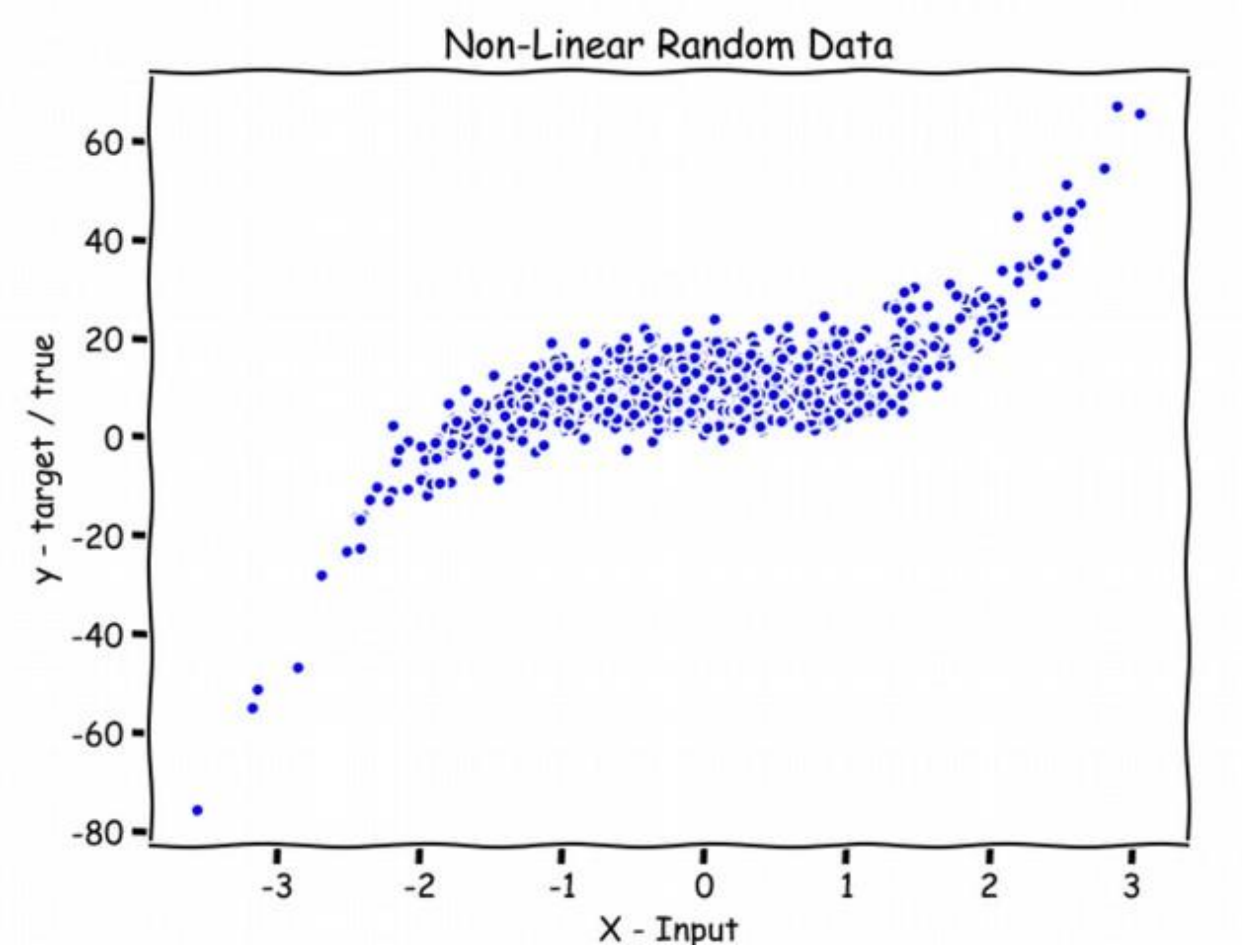
real relationship

```
fit.wls = lm (y~x, weights=1/(1+0.5*x*x))
abline(fit.wls$coefficients, lty=3)
```

How to know the proper weights?

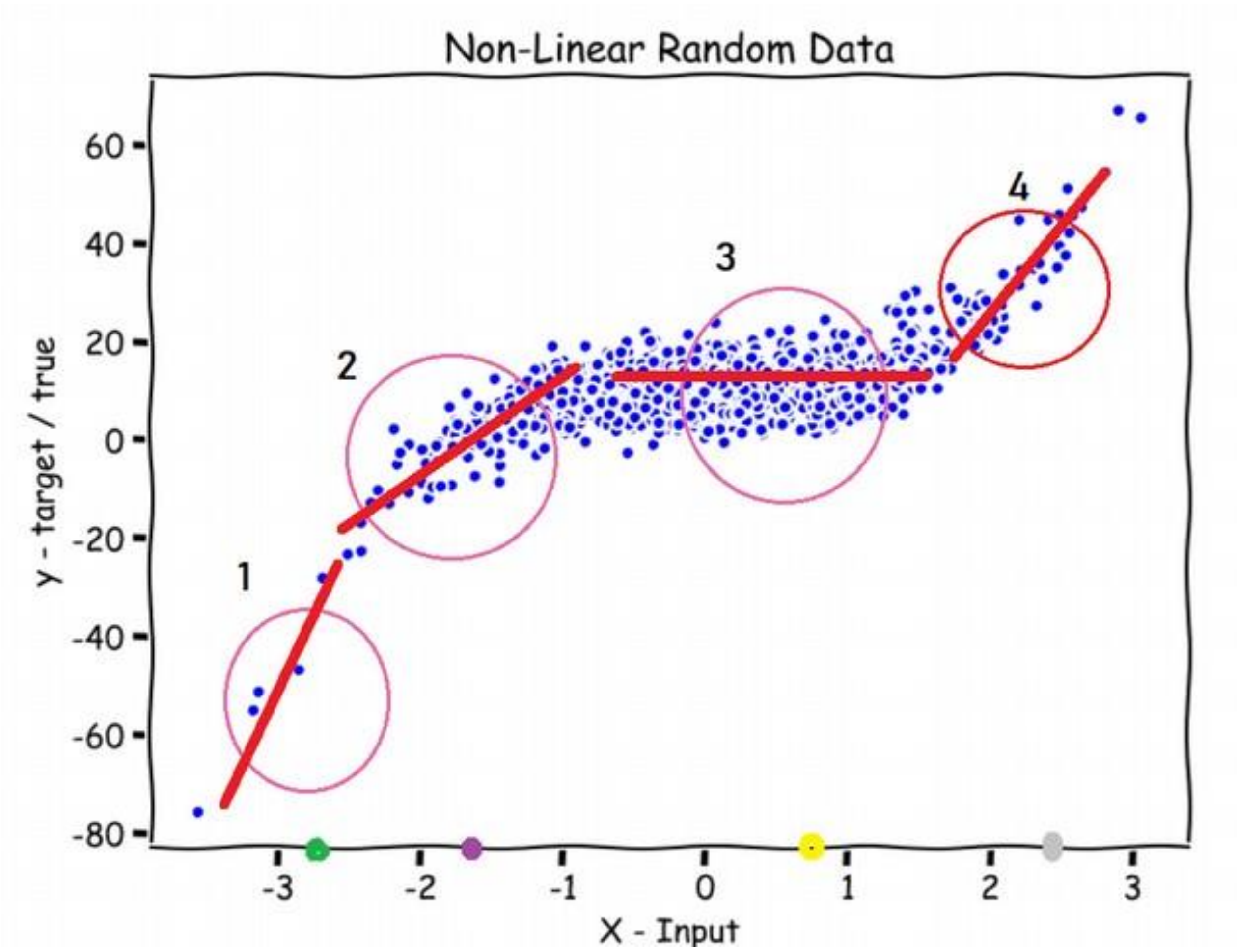
- Somehow know it from measurement device (e.g., know precision of the devise for various ranges)
- e.g., in polls or surveys variance of the proportions we find should be inversely related to sample size, hence can make weights proportional to sample size.
- Try to estimate it from the data, e.g.
  - Estimate y(x)
  - Construct log squared residuals $z_i = \log((y_i - r(x_i))^2)$
  - Estimate mean of the z's → q(x)
  - Use $s_x^2 = \exp q(x)$

What if the data are not linear?



https://towardsdatascience.com/locally-weighted-linear-regression-in-python-3d324108efbf

Local linear regression



https://towardsdatascience.com/locally-weighted-linear-regression-in-python-3d324108efbf

Local Linear Regression

- Linear regression could be justified by looking at a general regression function r(x) and expanding into a Taylor series

$$r(x)=r(x_0)+(x-x_0)r'+1/2(x-x_0)^2 r''(x_0)+\dots(*)$$

  - Then as long as we are close to x0 r' is the regression coefficient, but what if the relationship is not linear?

- Naive approach: use some window (say of size h) around data points and set

$$w_i=\begin{cases}1 & if\,|x_i-x_0|<h\\ 0 & otherwise\end{cases}$$

… and then use weighted least squares

Local Linear Regression (2)

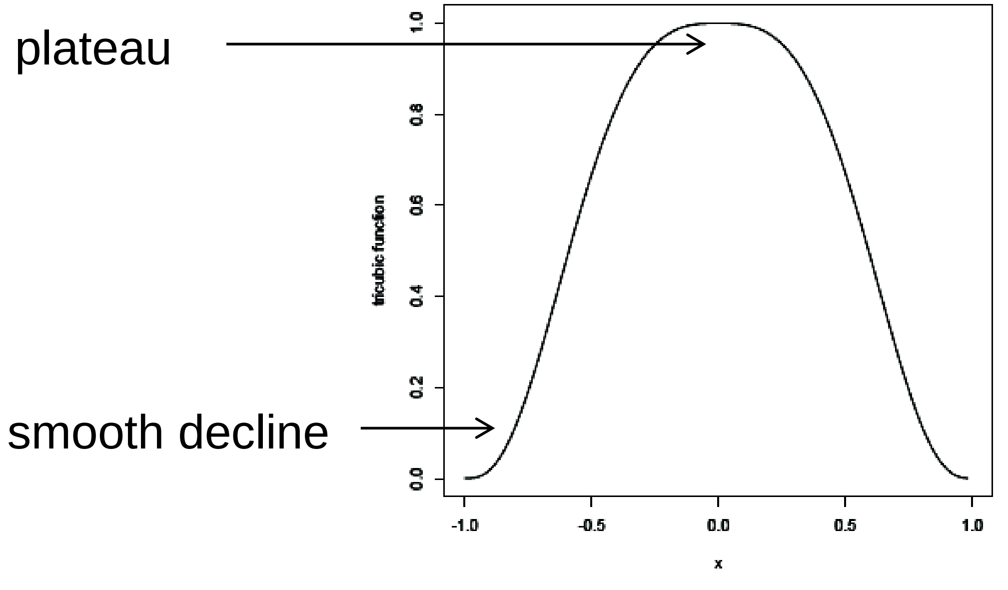
- Often one wants weights to change a bit more smoothly than that.
- Kernel regression:
  - Cut off Taylor expansion (*) after constant term and solve $min_b \sum_{i=1}^{N} w_i(x)(y_i - b)^2$
  - Which is solved by

$$b = \frac{\sum_{i=1}^{N} w_i(x) y_i}{\sum_{i=1}^{N} w_i(x)}$$

$$w_i(x) \propto K(x_i, x)$$

Locally Linear Regression (3)

- Take 0st and 1st order terms from (*)
- Often use a tri-cubic kernel

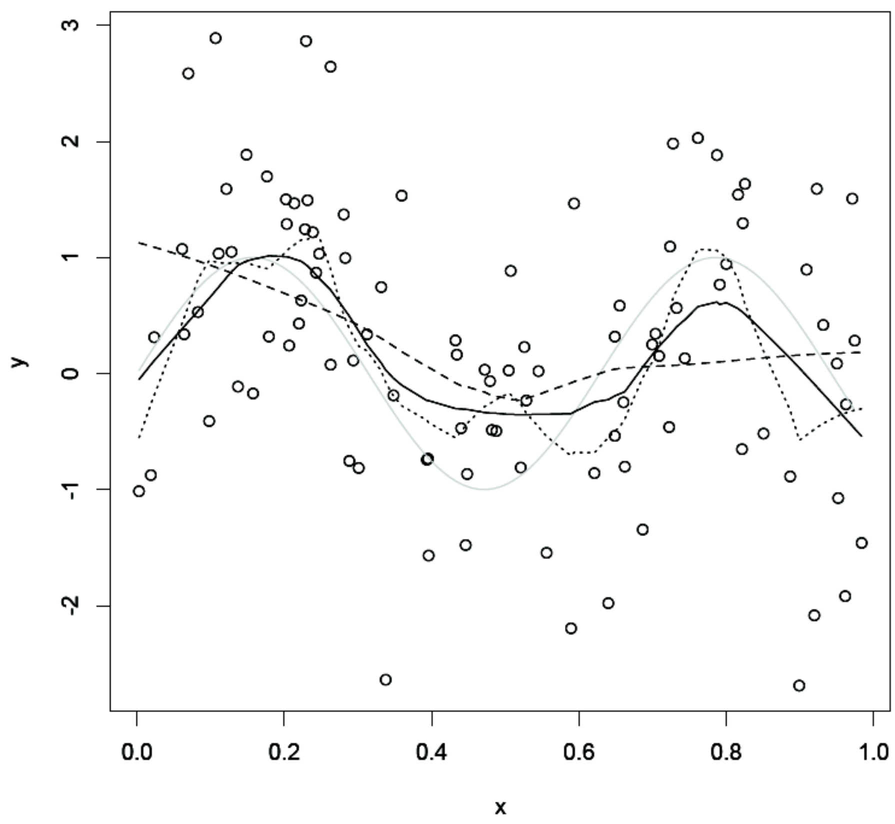


$$K(x_i, x) = \left(1 - \left(\frac{|x - x_0|}{h}\right)^3\right)^3$$

(h=1)

- R functions: lowess (specify fraction f of data points included), loess

Example




```
x.sin=runif(100,0,1)
y.sin=sin(10*x.sin)+rnorm(100,0,1)
plot(x.sin,y.sin, xlab="x", ylab="y")
curve(sin(10*x),col="grey",add="TRUE")
lines(lowess(x.sin, y.sin,f=1/3),lty=1)
lines(lowess(x.sin, y.sin,f=2/3),lty=2)
lines(lowess(x.sin, y.sin,f=1/6),lty=3)
```

Some art in choosing f appropriately; but can do quite a good job.

Local Linear Regression (4)

- When would you actually use this?
  - Number of predictors is small
  - You don’t want to think too hard about what features to use
  - (we will talk about other ways to deal with non-linear data later)
- Cons:
  - Linear regression is a parametric technique: estimate weights from data and can then use to predict
  - This is a non-parametric technique (the “data provide the function” – basically need to keep data points in memory to make predictions)
  - Need more data ...

Summary

- Linear regression
  - What it is, how it works
  - How to judge significance
  - Generalisations to higher d's
- MLE
  - Formalisation
  - Relationship to LR
- Extensions of linear regression
  - Homo vs heteroskedastic data
  - Weighted linear regression

Problem (1)

- A data set is constructed by taking 100 samples from a normal distribution with mean 5 and standard deviation 2 to construct a variable X_i, $i=1,\ldots,100$. Then, a variable Y_i, $i=1,\ldots,100$ is constructed by taking the values of the corresponding X_i and adding one half of a random variate drawn from a normal distribution with mean 5 and standard deviation 2 and thus a set of 100 pairs (X_i,Y_i) is obtained.
- Q: Find the parameters of a linear regression of Y on X (both by doing the numerical experiment and by calculating the result analytically).

Problem (2)

- Some person wants to conduct a least squares regression on a data set of N (X,Y) pairs, but wants attaches varying importance to deviations of various (X,Y) pairs to the line of best fit. The relative importance of deviations of pair (Xi,Yi) are given by a function f(i). Find an expression for the line of best fit generated by this procedure.

Problem (3)

- Repeated coin tossing of an (unfair) coin produces 100 heads up and 120 tails up. Find a maximum likelihood estimate for the probability that a coin toss will result in heads up.
- N variables have been sampled from an exponential distribution with unknown parameter. Find an expression for a maximum likelihood estimate for the parameter characterising the exponential distribution.