STAT 513/413: Lecture 23
Iteratively Reweighted Least Squares

(A dubious relative)

Recall: Cauchy regression

$y_i = x_i^\mathsf{T}\beta + \sigma\varepsilon_i$ with ε_i Cauchy errors (and σ known)

The EM-algorithm: select β_1

Calculate weights $z_i = \dfrac{1}{1 + \dfrac{(y_i - x_i^\mathsf{T}\beta_1)^2}{\sigma^2}}$

Calculate β_2 as a weighted least squares estimate, solving

$$\sum_{i=1}^{n} z_i (y_i - x_i^\mathsf{T}\beta)^2 \looparrowright \min_{\beta}!$$

and repeat...

So it works via EM... but after all, it is nothing but...

IRLS: Iteratively Reweighted Least Squares

(Some call it IWLS: Iteratively Weighted Least Squares)

Weighted least squares

How to solve this?

$$\sum_{i=1}^{n} w_i (y_i - x_i^\top \beta)^2 \looparrowright \min_{\beta}!$$

As in the standard case, just take derivatives

$$\sum_{i=1}^{n} w_i(-x_i)2(y_i - x_i^\top \beta) = 0 \quad \text{that is} \quad \sum_{i=1}^{n} w_i x_i x_i^\top \beta = \sum_{i=1}^{n} w_i x_i y_i$$

And when you think about it, you will see the matrix form

$X^\top W X \beta = X^\top W y$ where, as usual, X is $n \times p$ matrix

and W is an $n \times n$ matrix with the w_i's on the diagonal

Also, the R function `lm()` has a parameter `weights`

Now something else

Now weights - but now squares either; to obtain

$$\sum_{i=1}^{n} \rho(y_i - x_i^{\mathsf{T}}\beta) \looparrowright \min_{\beta}!$$

we may want to solve $$\sum_{i=1}^{n} -x_i \psi(y_i - x_i^{\mathsf{T}}\beta) = 0$$

where $\psi(u) = \rho'(u)$; let us say that $\psi(u) = u\varkappa(u)$

Then it is $$\sum_{i=1}^{n} -x_i(y_i - x_i^{\mathsf{T}}\beta)\varkappa(y_i - x_i^{\mathsf{T}}\beta) = 0$$

which is $$\sum_{i=1}^{n} \varkappa(y_i - x_i^{\mathsf{T}}\beta)x_i x_i^{\mathsf{T}}\beta = \sum_{i=1}^{n} \varkappa(y_i - x_i^{\mathsf{T}}\beta)x_i y_i$$

Recognizing that? It is like $w_i = \varkappa(y_i - x_i^{\mathsf{T}}\beta)$ - the only problem is that w_i depends on the exactly same β we would like to find...

... but maybe we can trick it iteratively

IRLS

Select β_1 (typically: the result of the least-squares fit)

Calculate weights: $w_i = \varkappa(y_i - x_i^\top \beta_1)$

(note: the arguments of $\varkappa$ are *residuals* of the β_1 fit)

Obtain β_2 as the least squares fit with weights w_i

And repeat

For the Cauchy: $\rho(u) = \log(1 + u^2)$

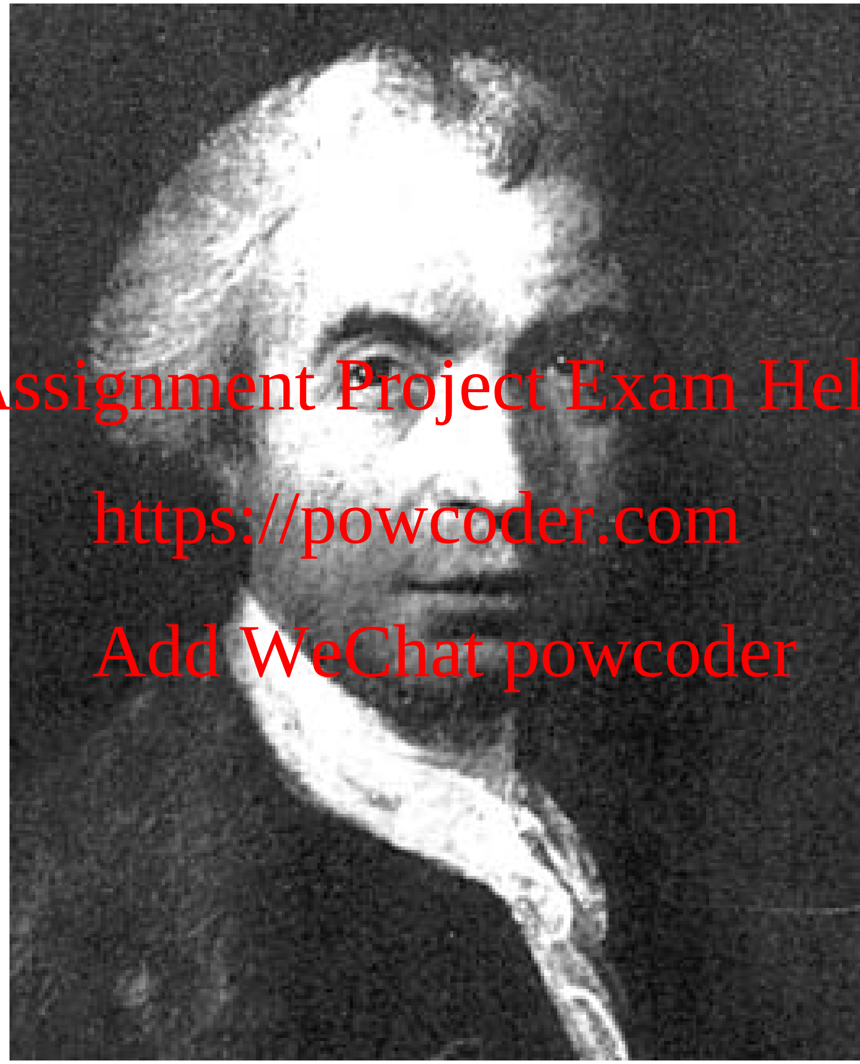
Thus $\quad \psi(u) = \dfrac{2u}{1+u^2} = u\,\dfrac{2}{1+u^2}$

Well, for the EM-algorithm it was $\dfrac{1}{1+u^2}$ $\quad\dots$

Does factor of 2 really matter? Isn't the least-squares fit the same if weighted by w_i or $2w_i$?

Another use: quantile regression

(An insightful alternative to least squares)



Ruđer Josip Bošković (1711-1787)

A highly confidential dataset

The idea

Recall one characterization of the mean: it is the number μ for which the expression $\sum_i (y_1 - \mu)^2$ achieves the minimum

Thus, least-squares regression fits the conditional mean

Does there exist some ρ_p which would yield us p-quantiles in a similar way?

$$q_p \text{ as a solution of } \sum_i \rho_p(y_1 - q) \looparrowright \min_q !$$

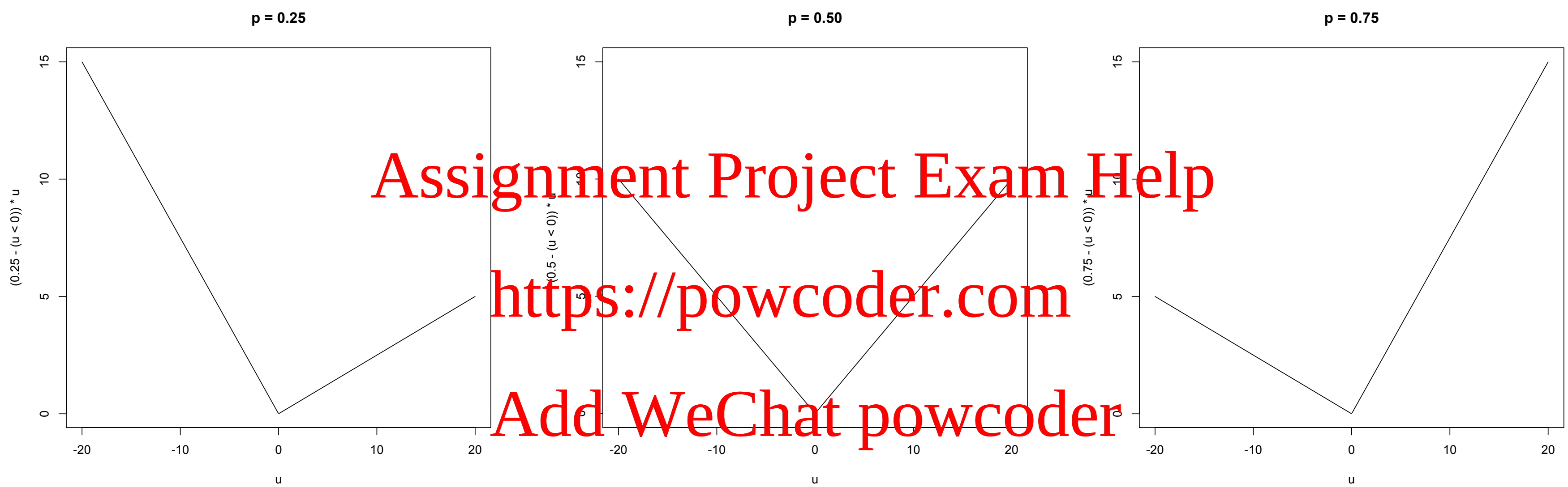
Yes it does!

And allows us to do, in the same manner as least-squares, to do

Quantile regression: fitting conditional quantiles

Check function(s)

$$\rho_p(u) = \begin{cases} (p-1)u & \text{if } u \leqslant 0 \\ pu & \text{if } u \geqslant 0 \end{cases}$$



ρ_p as ρ in the minimization outlined above produces p-quantile(s)

In particular, $\rho_{0.5}(u) = |u|/2$ - which in the minimization is equivalent to $\rho_{0.5}(u) = |u|$ (making an expression minimal is equivalent to making minimal its half)

Summary

For fitting the *regression* p-quantile, take ρ_p and solve

$$\sum_{i=1}^{n} \rho_p(y_i - x_i^{\mathsf{T}}\beta) \looparrowright \min_{\beta}!$$

How? IRLS?

OK, but there may be a problem with 0; there is no derivative...

Be engineers: don't worry. Everywhere else there is

But what if it returns an error that I am dividing by zero?

Right, don't. Add 0.0000001 to every denominator

Recall: weights $\quad w_i = \dfrac{1}{1 + \dfrac{(y_i - x_i^{\mathsf{T}}\beta_1)^2}{\sigma^2}} \quad$ are safe

because of 1 in the denominator; if they were $\quad w_i = \dfrac{1}{\dfrac{(y_i - x_i^{\mathsf{T}}\beta_1)^2}{\sigma^2}}$

then could be a problem, but then we would rather make it...

Mitigating possible division by zero in IRLS

$$w_i = \frac{1}{0.000001 + \frac{(y_i - x_i^T \beta_1)^2}{\sigma^2}}$$

If we have to divide by zero, and we don't want to, we may

- replace zero by some small ε

- have all denominators (if nonnegative) larger by some small ε

- but NEVER EVER just omit the index i in the summation when this happens (thinking that you can return it back in later iterations): this is known to spoil the method!!

Or don't use IRLS for computing regression quantiles at all and instead compute them via linear programming, as implemented in the R package `quantreg`