Carnegie Mellon

Center for Automated Learning and Discovery

and Robotics Institute

and School of Computer Science

Efficient Locally Weighted Polynomial Regression Predictions

Andrew Moore   Jeff Schneider   Kan Deng

*Thanks to Mary Soon Lee. Funded by NSF.

www.cs.cmu.edu/~AUTON

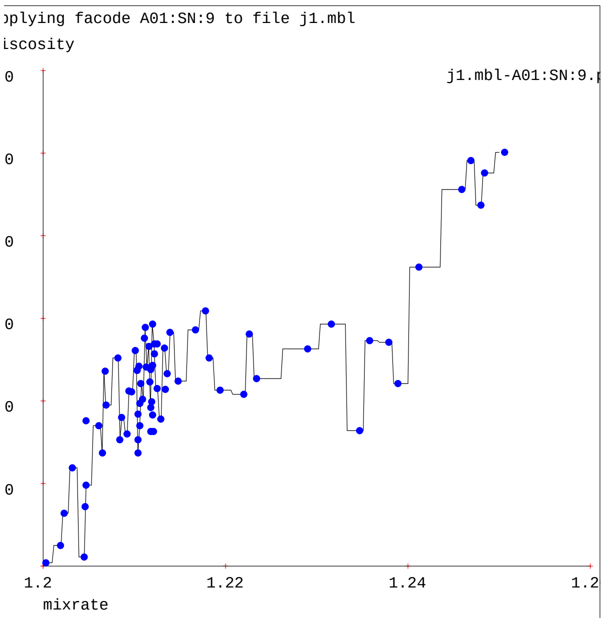
What is L WPR?

Locally weighted polynomial regression is an algorithm for learning non-linear numeric functions.

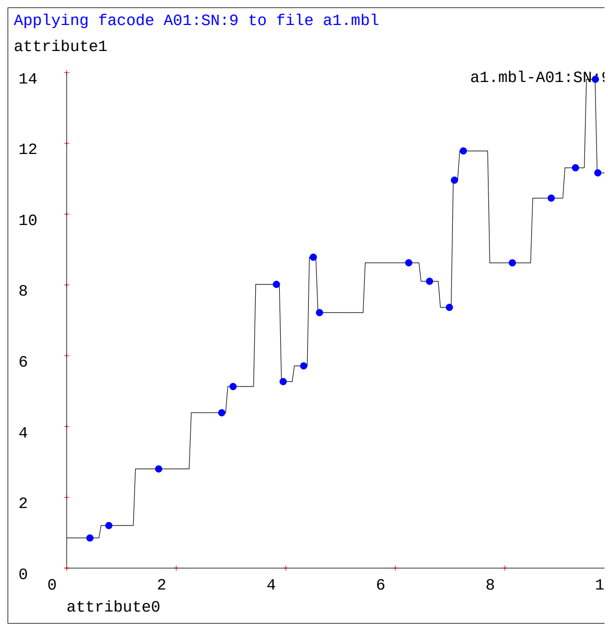
Here , we will...

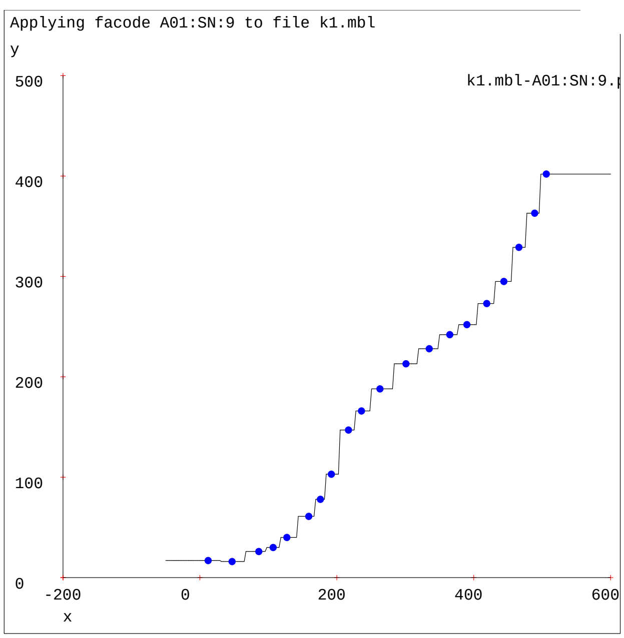
- Review L WPR (tra velling via Nearest Neighbor and K ernel Regression).
- Praise its virtues .
- Mourn its computational expense .
- Discuss recently proposed methods for making it faster .
- Introduce and evaluate a new multiresolution data structure .

Why not just use 1-nearest neighbor?



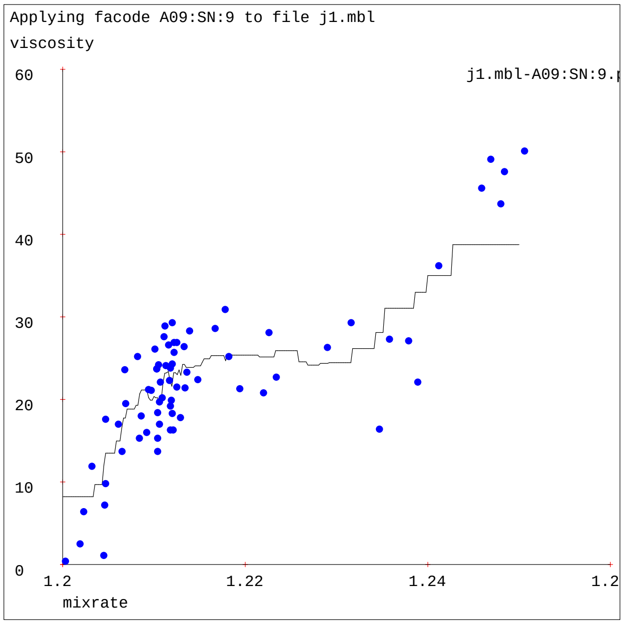
..because it fits noise!



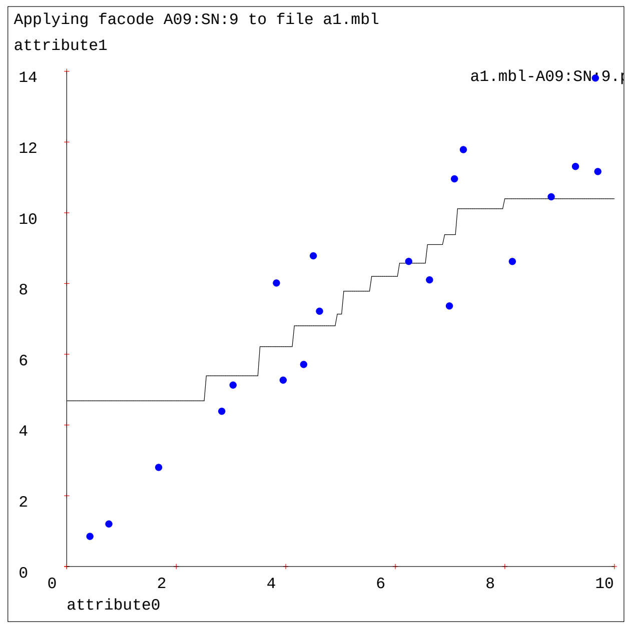


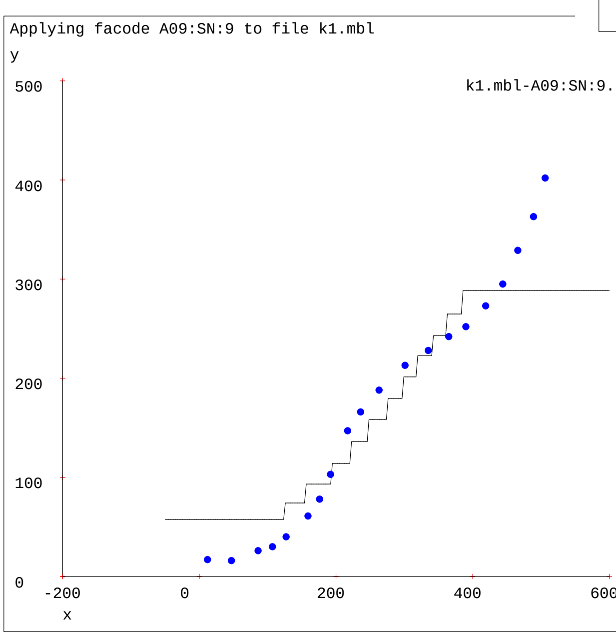
...and doesn't provide continuous predictions or gradient estimates .

Why not use average of 10-nearest neighbors?



..less noise fitting

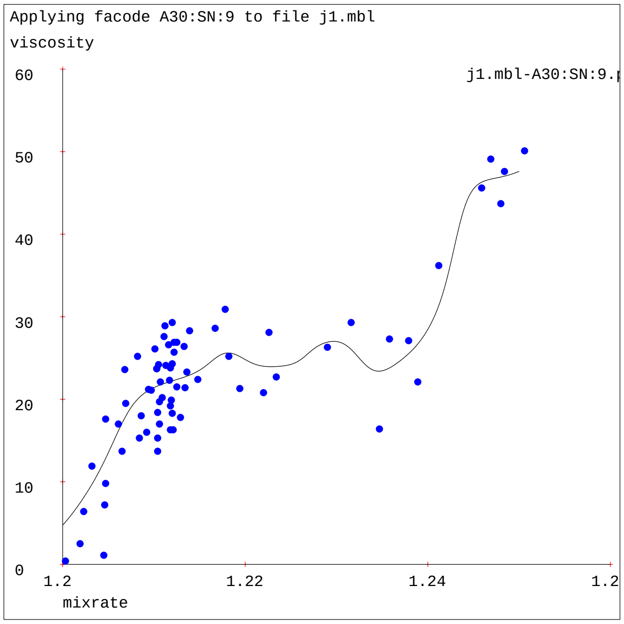


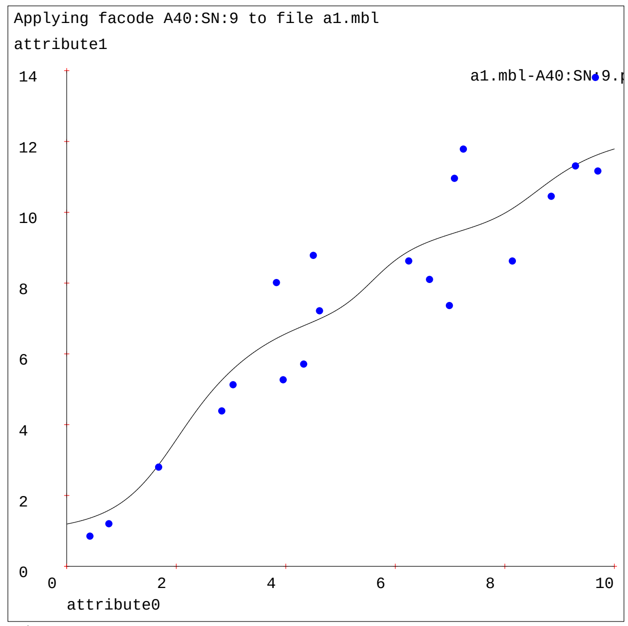


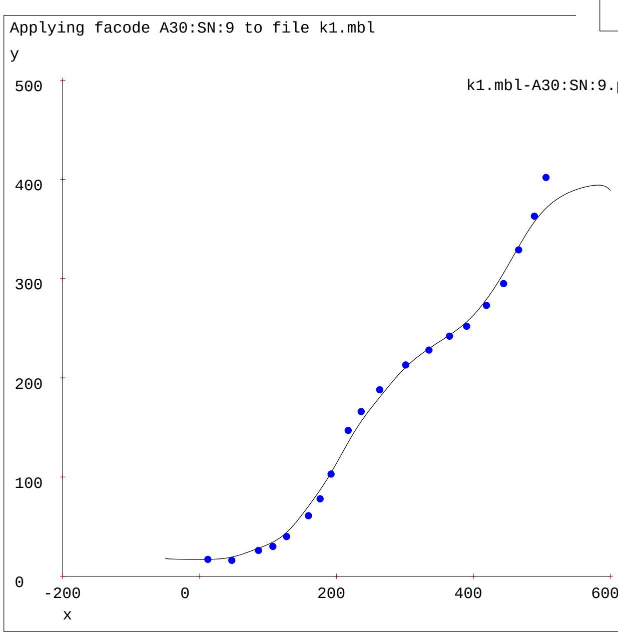
...but the discontinuities and bad extrapolations are embarrassing!

Maybe Kernel Regression?

..nice and smooth





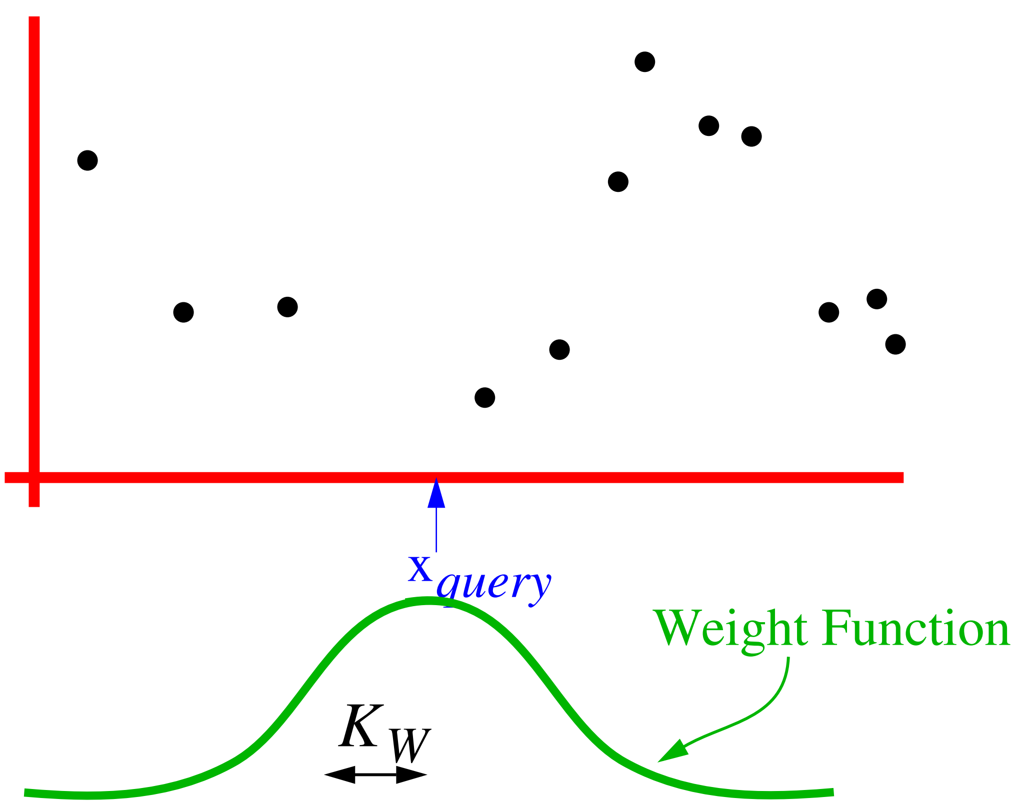


Kernel Regression Reminder

You're about to do a prediction at query x_{query}.

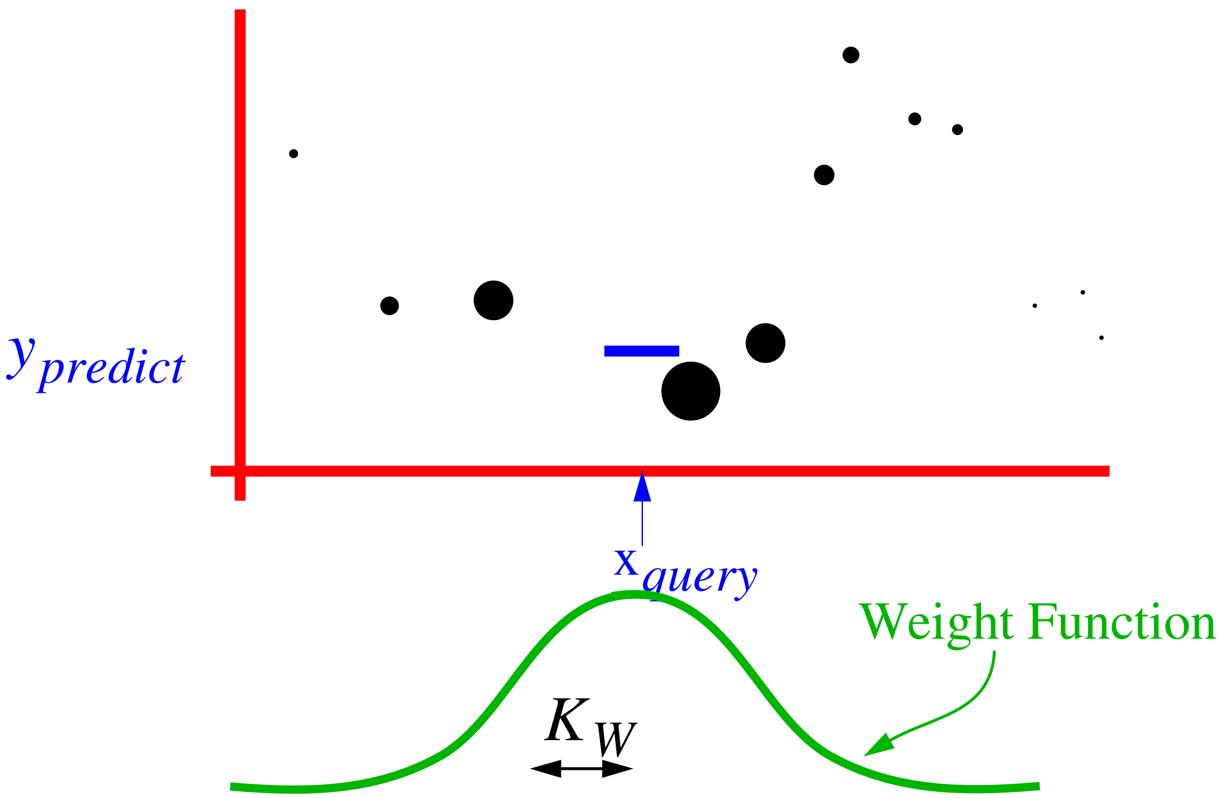
You take every single datapoint and assign it a weight according to how close it is to the query:

$$w_k = \exp(-Distance(x_k, x_{query})^2 / K_W^2)$$



Kernel Regression Reminder

$$w_i = exp(-D(x_i, query)^2 / K_W^2)$$

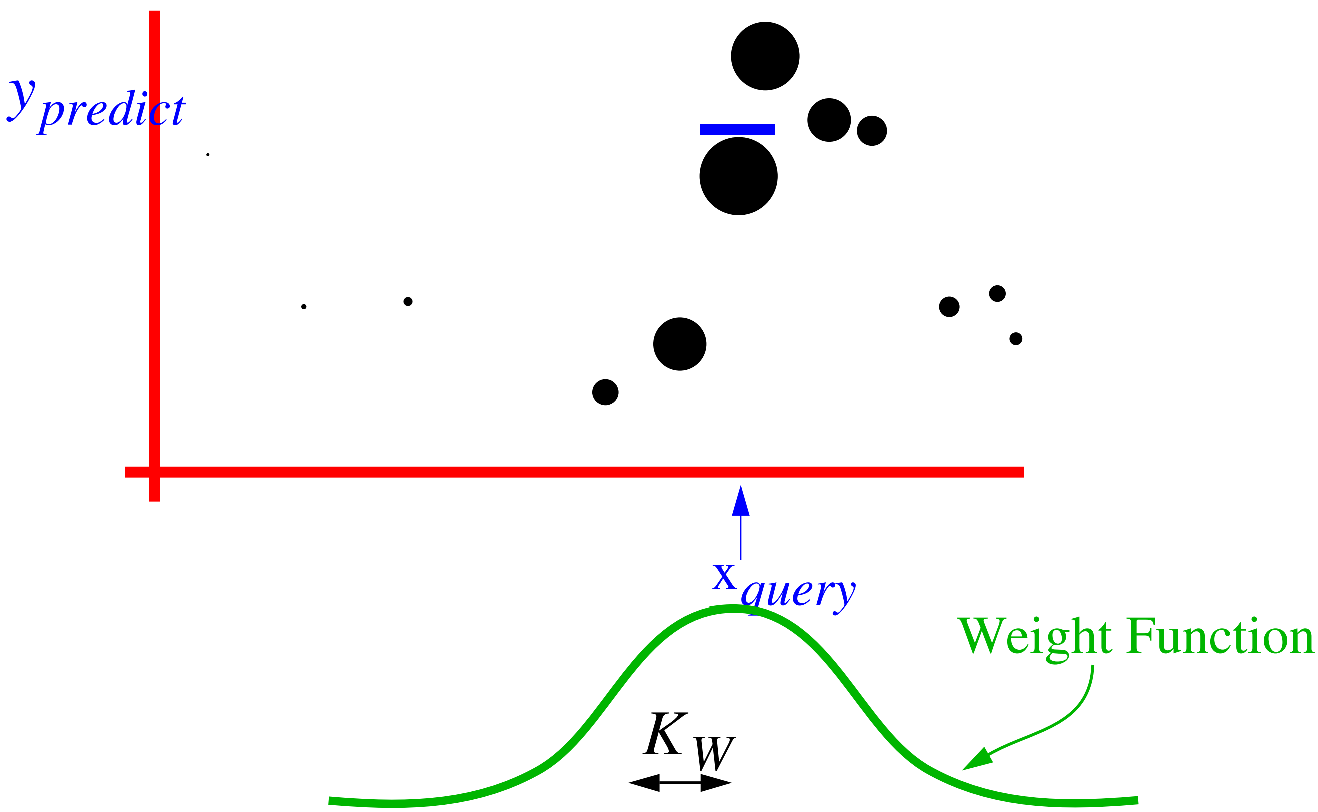


- Having weighted those datapoints, you simply predict the weighted average of all the outputs:

$$y_{predict} = \Sigma_k w_k y_k \,/\, \Sigma_k w_k$$

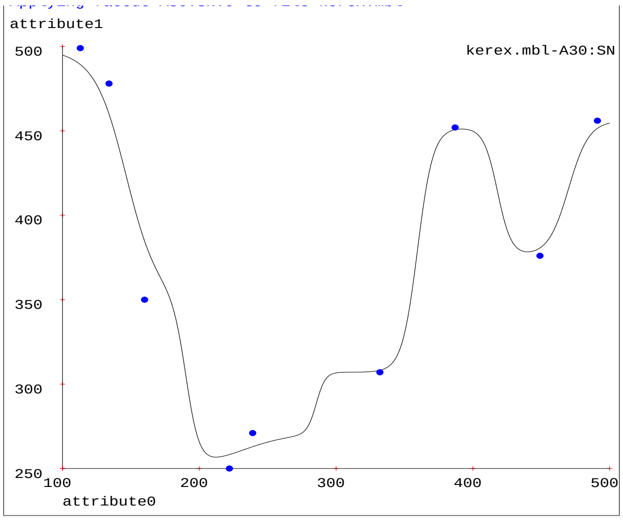
Kernel Regression Reminder

$$w_i = exp(-D(x_i, query)^2 / K_W^2)$$

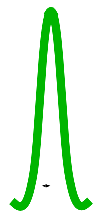


If you move the query , you get different weights , and so you get different predictions .

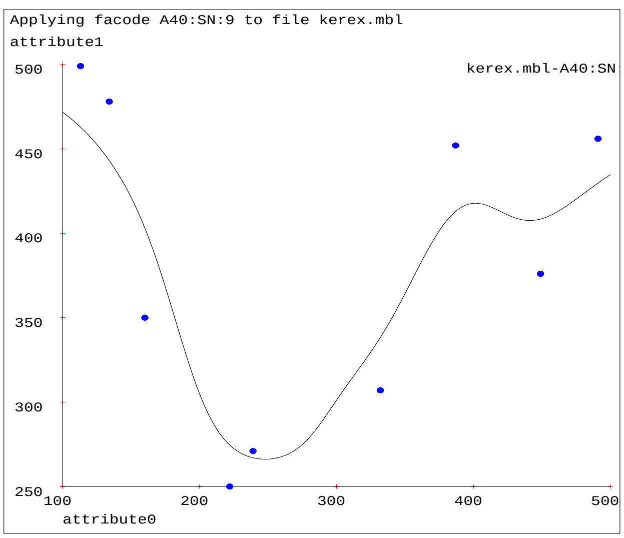
The kernel width matters



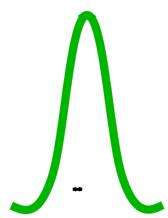
Weight Function



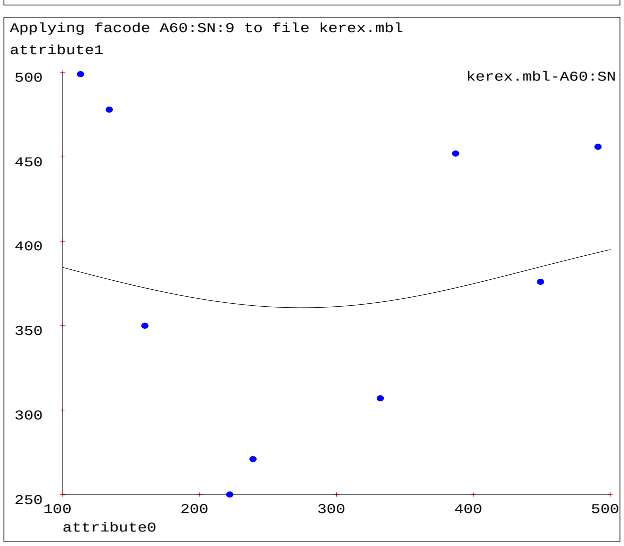
$K_W = 10$



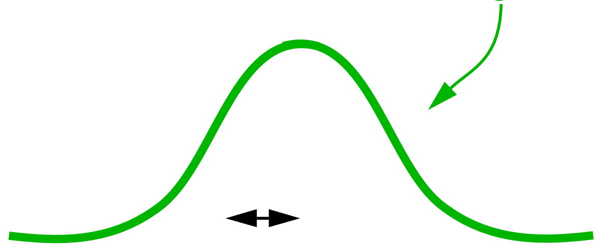
Weight Function



$K_W = 20$



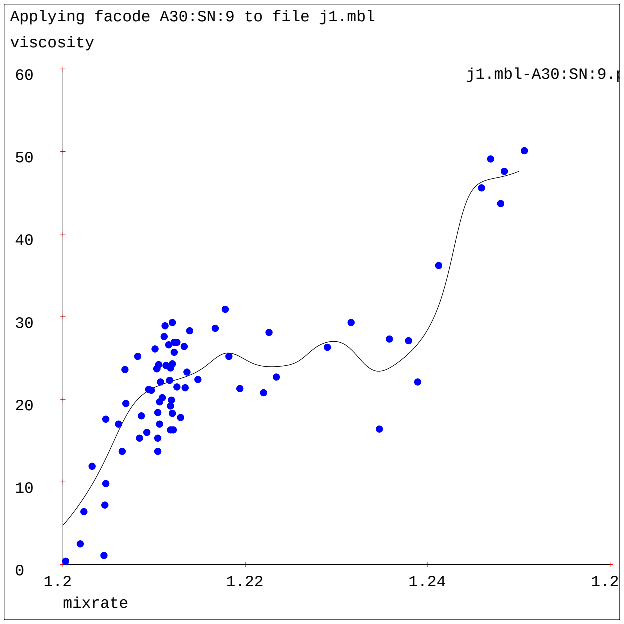
Weight Function

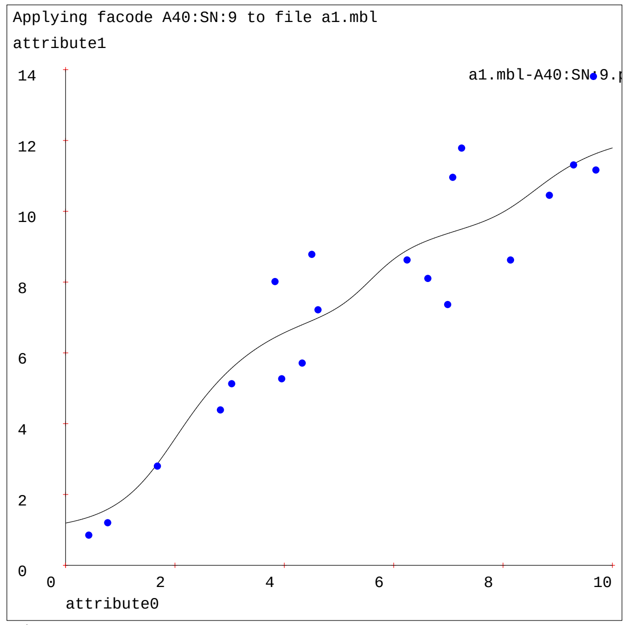


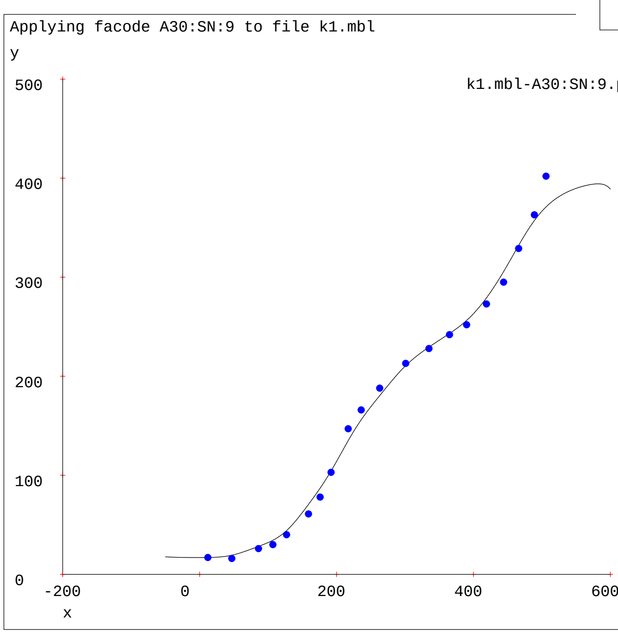
$K_W = 80$

So why not Kernel Regression?

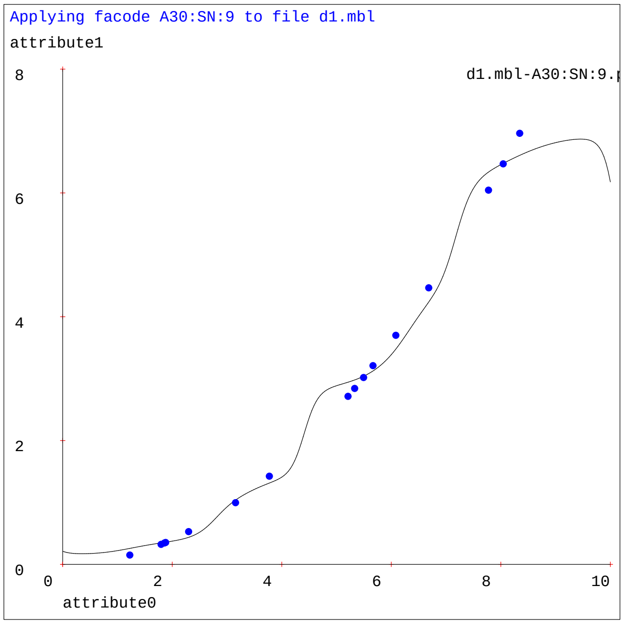
..nice and smooth

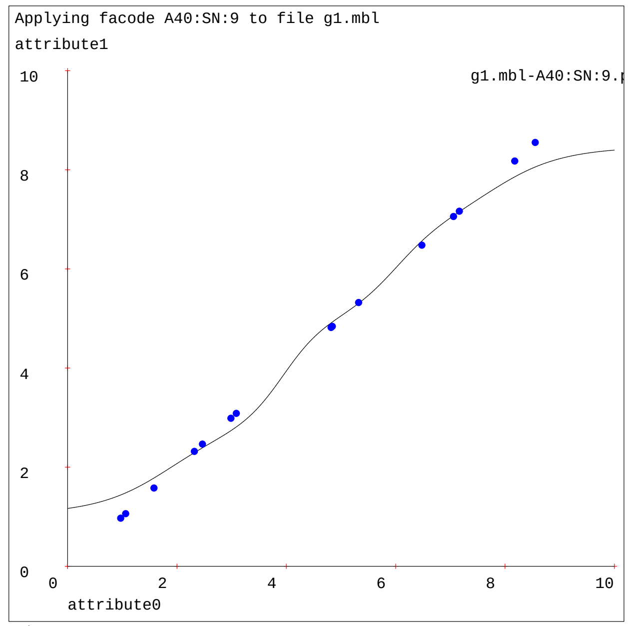


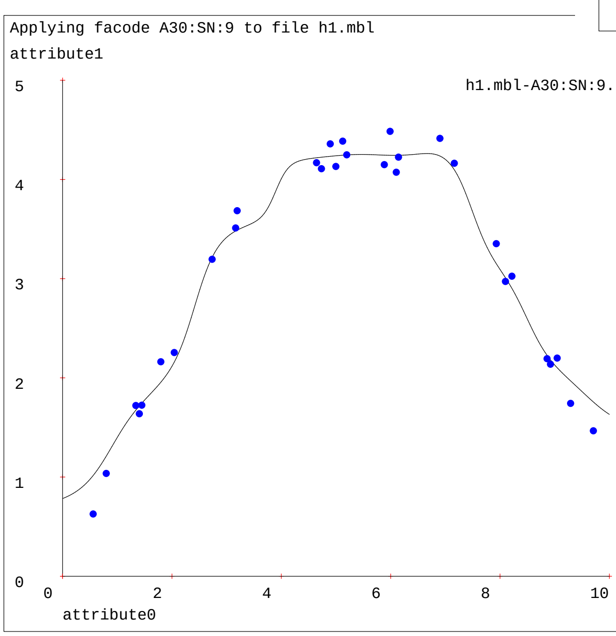




Kernel Regression can look stupid...





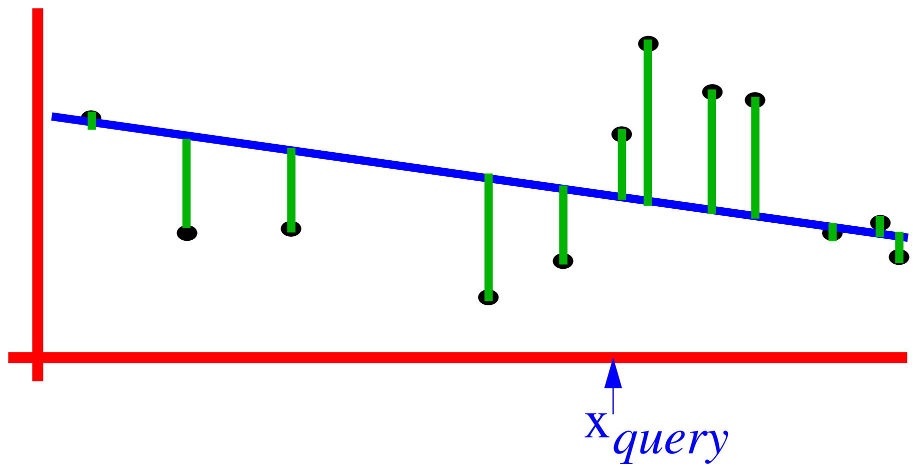


LWPR has several nicer properties ...

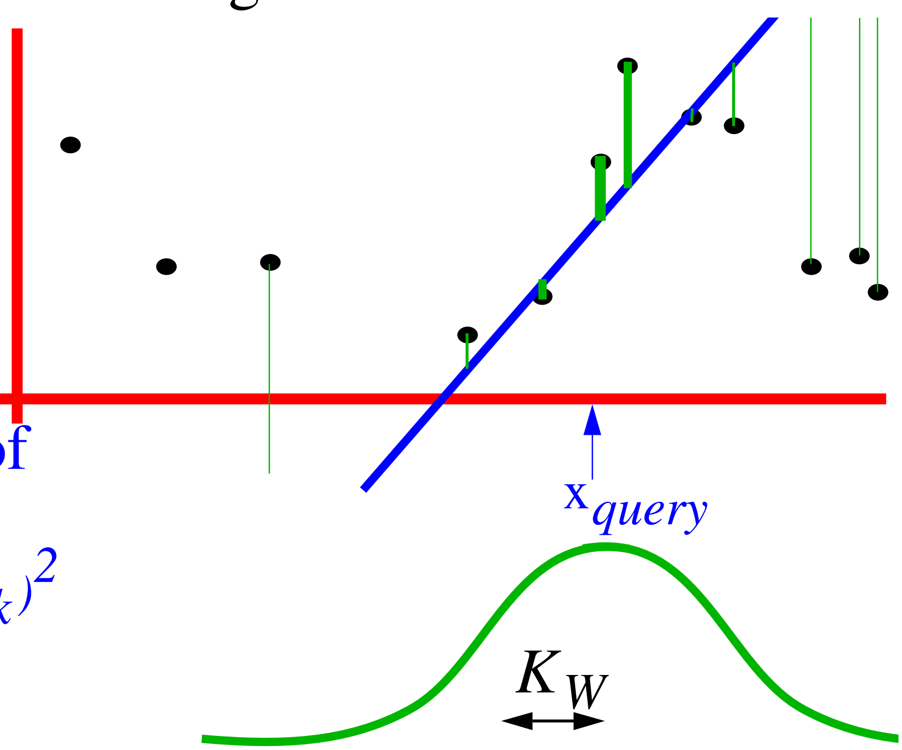
Locally weighted linear regression

Global Regression: ($y = \beta_0 + \beta_1 x + noise$):

Minimizes sum of squared residuals:
$\Sigma_k (y_k - \beta_0 - \beta_1 x_k)^2$



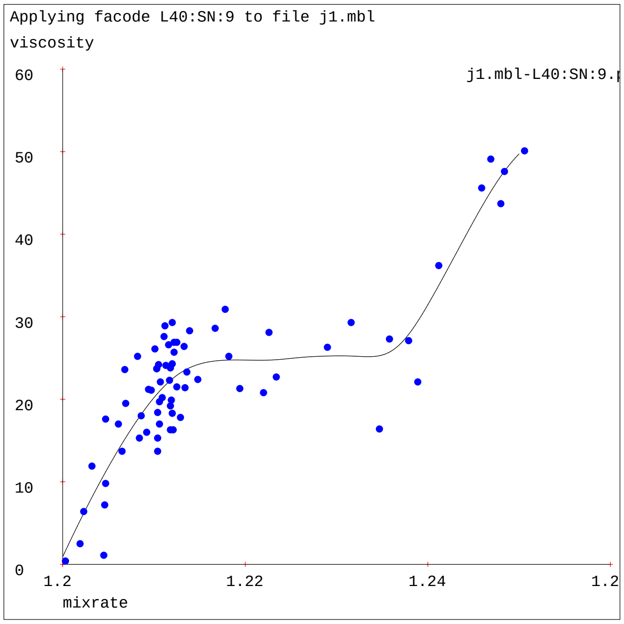
Locally weighted linear regression:

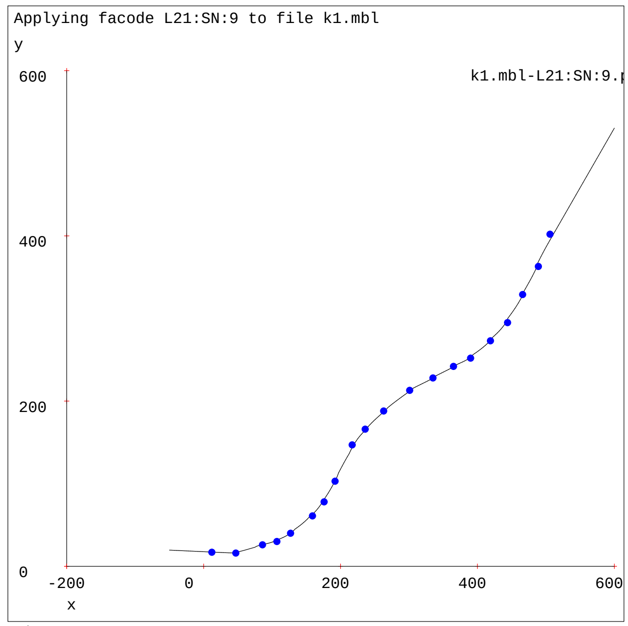


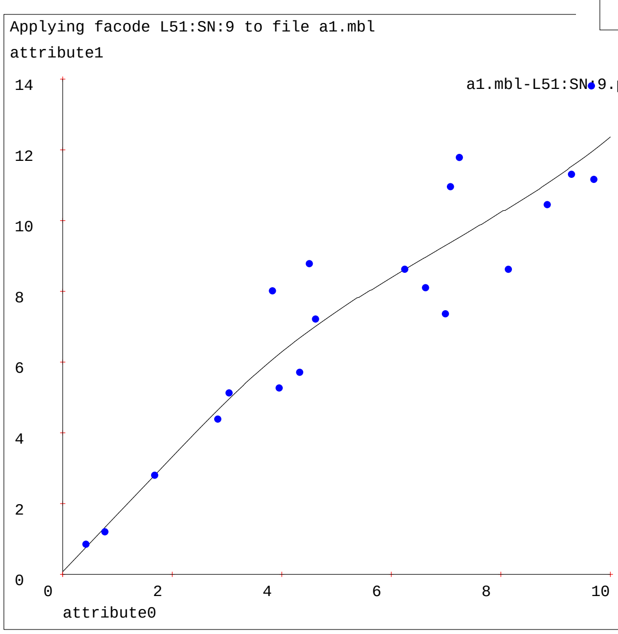
Minimizes **weighted** sum of squared residuals:
$\Sigma_k w_k (y_k - \beta_0 - \beta_1 x_k)^2$

Locally weighted linear regression...

..very nice .







(LWPR goes back to early 1900's. See [Cleveland+Delvin 1988] survey article)