December 19

Lecture on site and online (zoom)

Video of lecture will be made available after the lecture

9h15-12am: ***Handling Imbalanced Datasets***
Enable Incremental Learning
Overview of the course
Exam preparation

- 12am-13: Open Q&A regarding exam and material of the class

Q&A session – exam preparation

December 19, 11h15-13:00

Friday 24, 11h00-12h00 room ME.A3.31

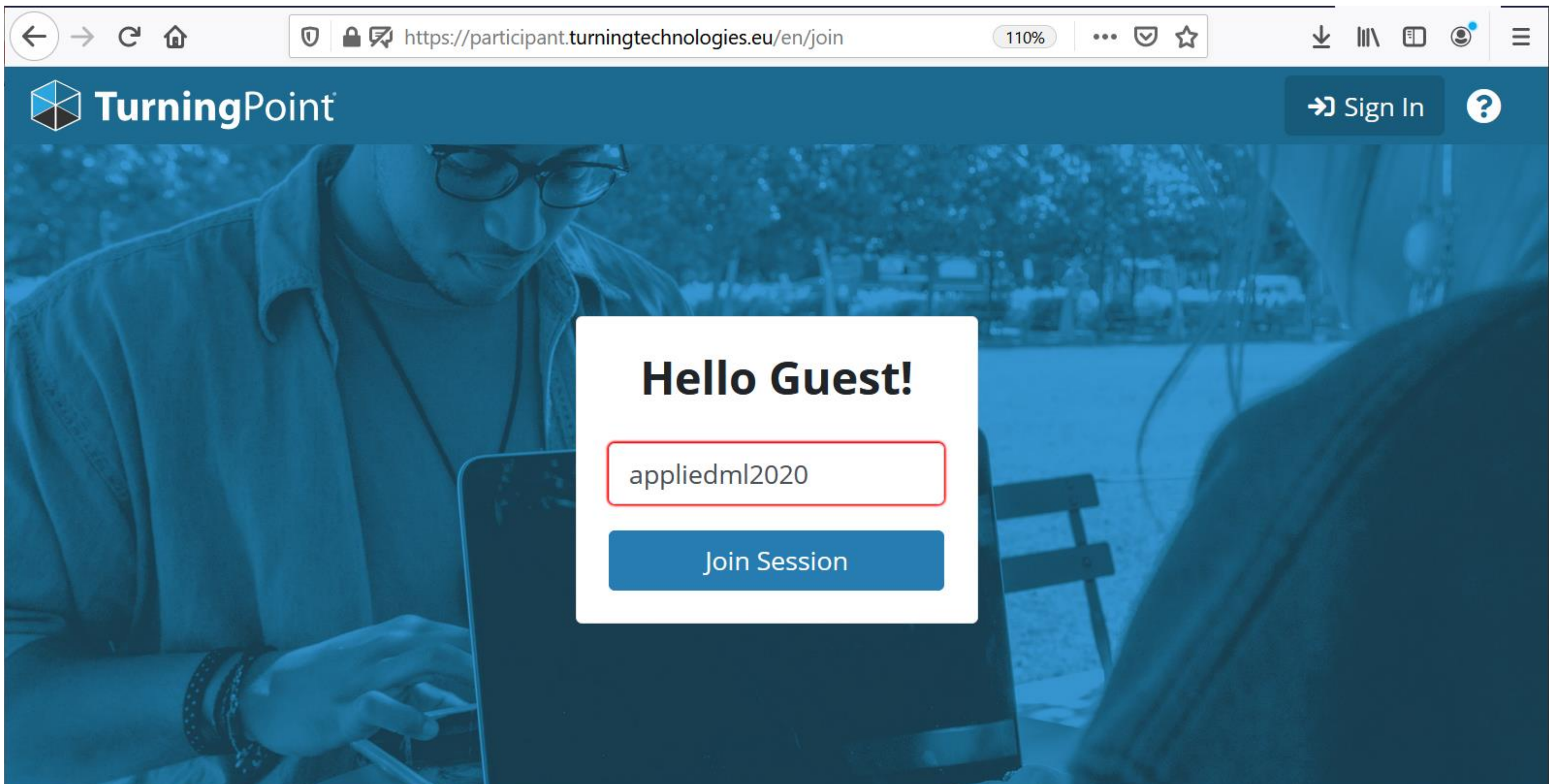
Nonlinear Regression

Interactive Lecture

Launch polling system

https://participant.turningtechnologies.eu/en/join

Acces as GUEST and enter the session id: *appliedml2020*



4

Linear Regression

Find the optimal parameter *w* through least-square regression:

$$w^* = \min_w \left(\sum_{i=1}^{M} \frac{1}{2} \left(w^T x^i - y^i \right)^2 \right)$$

Closed-form solution:

$$w^* = \left(X X^T \right)^{-1} X y$$

Weighted Regression

Find the optimal parameter *w* through least-square regression:

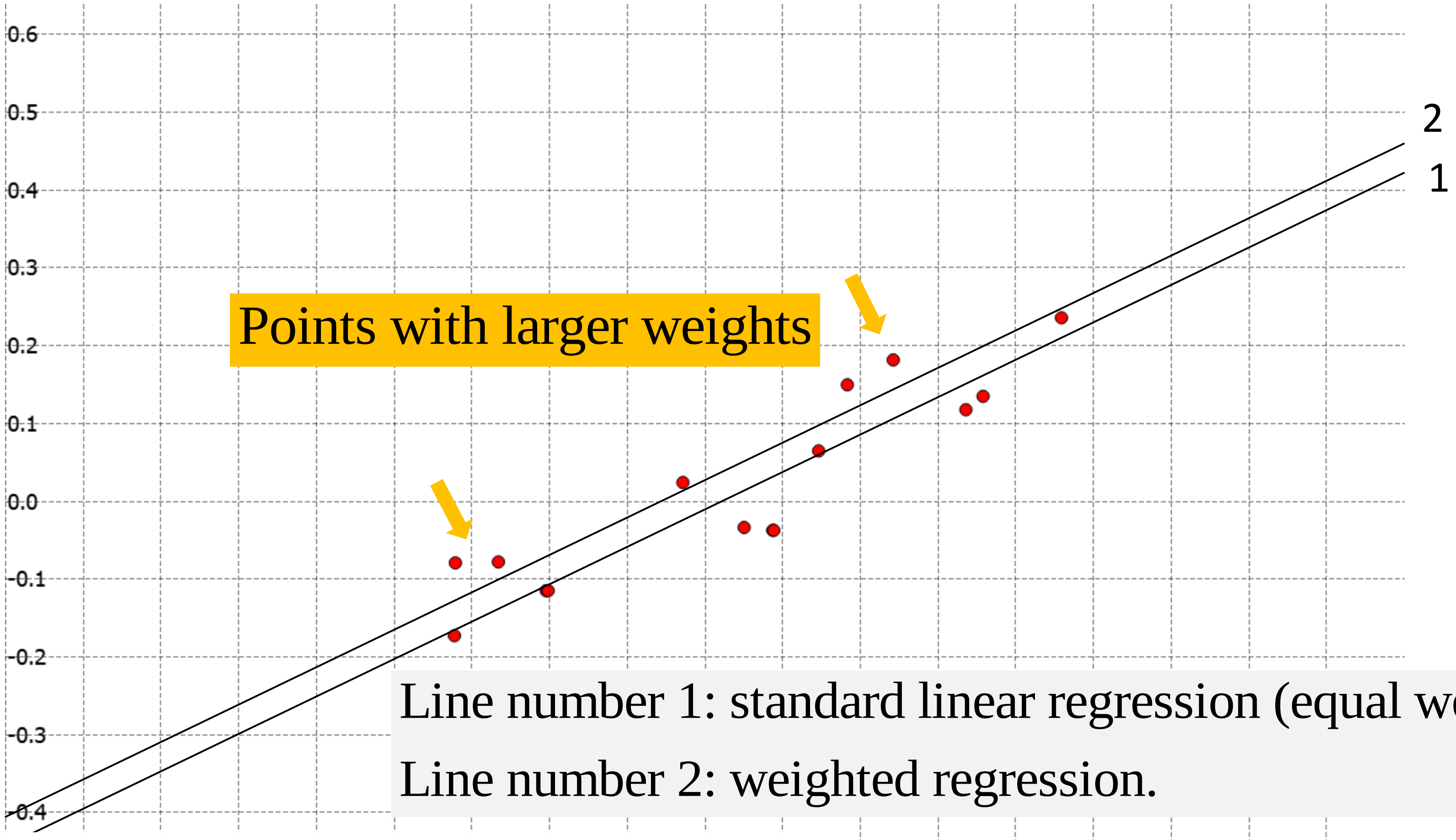
$$w^* = \min_w \left(\sum_{i=1}^{M} \frac{1}{2} \beta_i \left(w^T x^i - y^i \right)^2 \right)$$

Closed-form solution:

$$w^* = \left(Z Z^T \right)^{-1} Z v \; x$$

$$Z = X B^{1/2} \; and \; v = B^{1/2} y$$

Weighted linear regression



Line number 1: standard linear regression (equal weights)

Line number 2: weighted regression.

Least-square weighted regression

$$w^* = \min_w \left(\sum_{i=1}^{M} \frac{1}{2} \beta_i \left(w^T x^i - y^i \right)^2 \right), \qquad \beta_i > 0 \qquad \underbrace{\beta_1 = \beta_2 \ldots = \beta_M}_{\rightarrow \text{ Standard linear regression}}$$

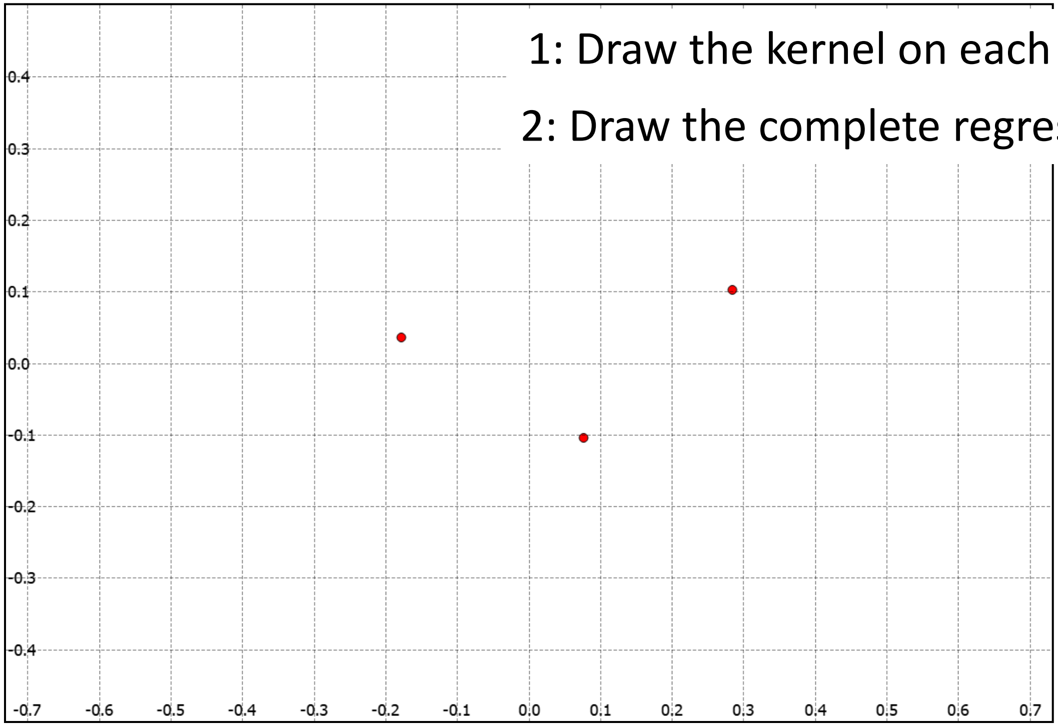
Locally weighted regression

Introduce local solution:

$$\hat{y}(x) = \sum_{i=1}^{M} \beta_i(x) y^i \Big/ \sum_{j=1}^{M} \beta_j(x) \qquad \beta_i(x) \in \mathbb{R}: \text{ weights function of x}$$

$$\beta_i(x) = e^{-\left\|x^i - x\right\|^2}$$

Also closed-form solution, but local regression



1: Draw the kernel on each point

2: Draw the complete regressive function

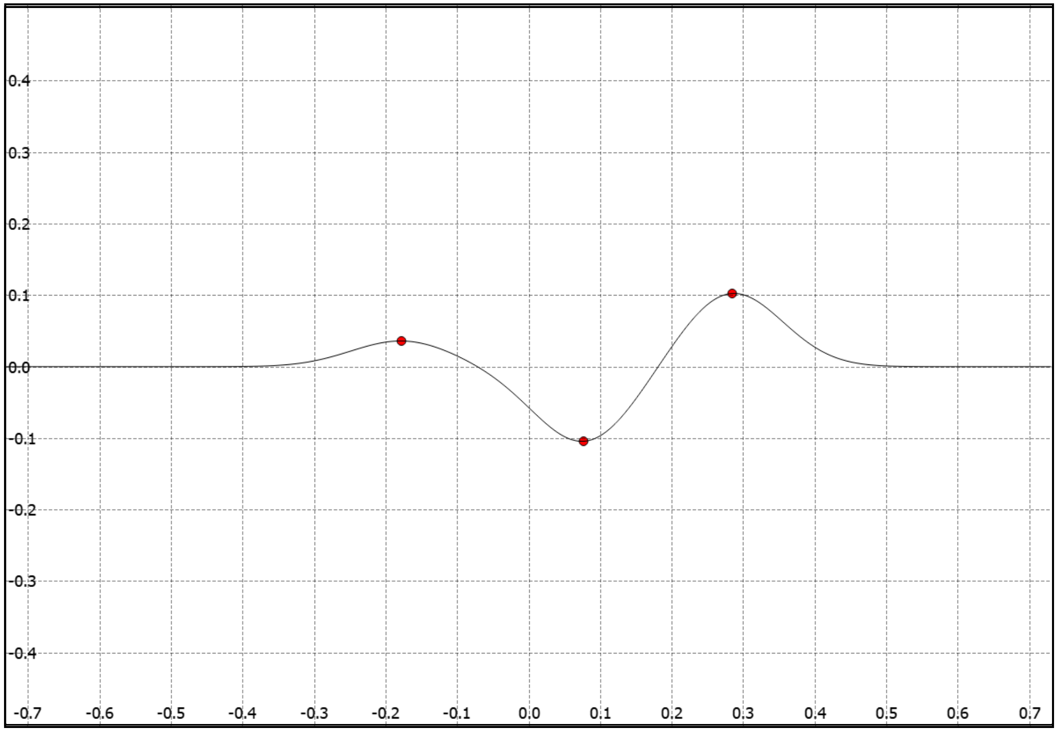
Locally weighted regression

Introduce local solution:

$$\hat{y}(x) = \sum_{i=1}^{M} \beta_i(x)\, y^i \Big/ \sum_{j=1}^{M} \beta_j(x) \qquad \beta_i(x) \in \mathbb{R}: \text{ weights function of x}$$

$$\beta_i(x) = e^{-\left\|x^i - x\right\|^2}$$

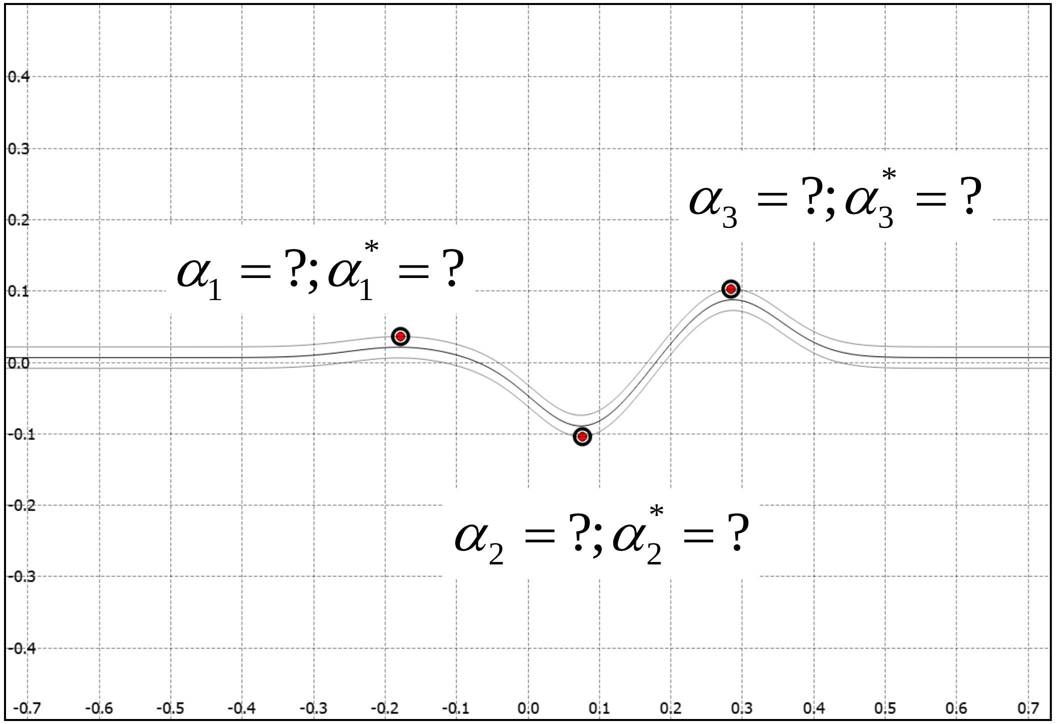
Also closed-form solution, but local regression



Support Vector Regression

SVR determines automatically which point matters for building the regression.

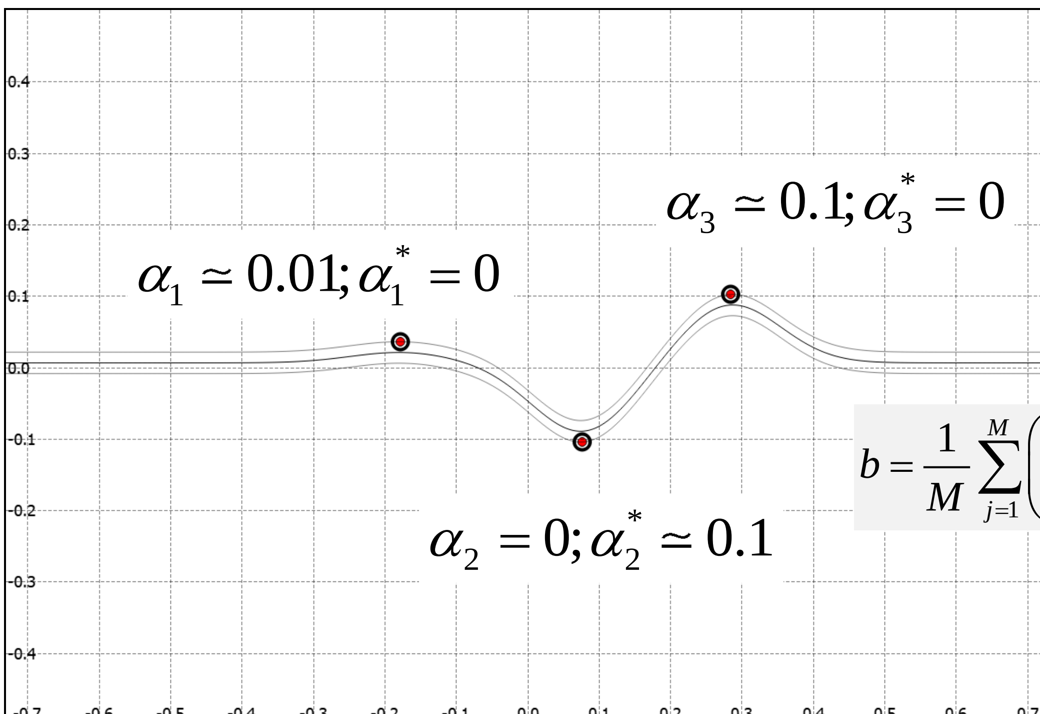
$$y = f(x) = \sum_{i=1}^{M} \left(\alpha_i - \alpha_i^*\right) k\left(x^i, x\right) + b$$



Support Vector Regression

Determines automatically which point matters for building the regression.

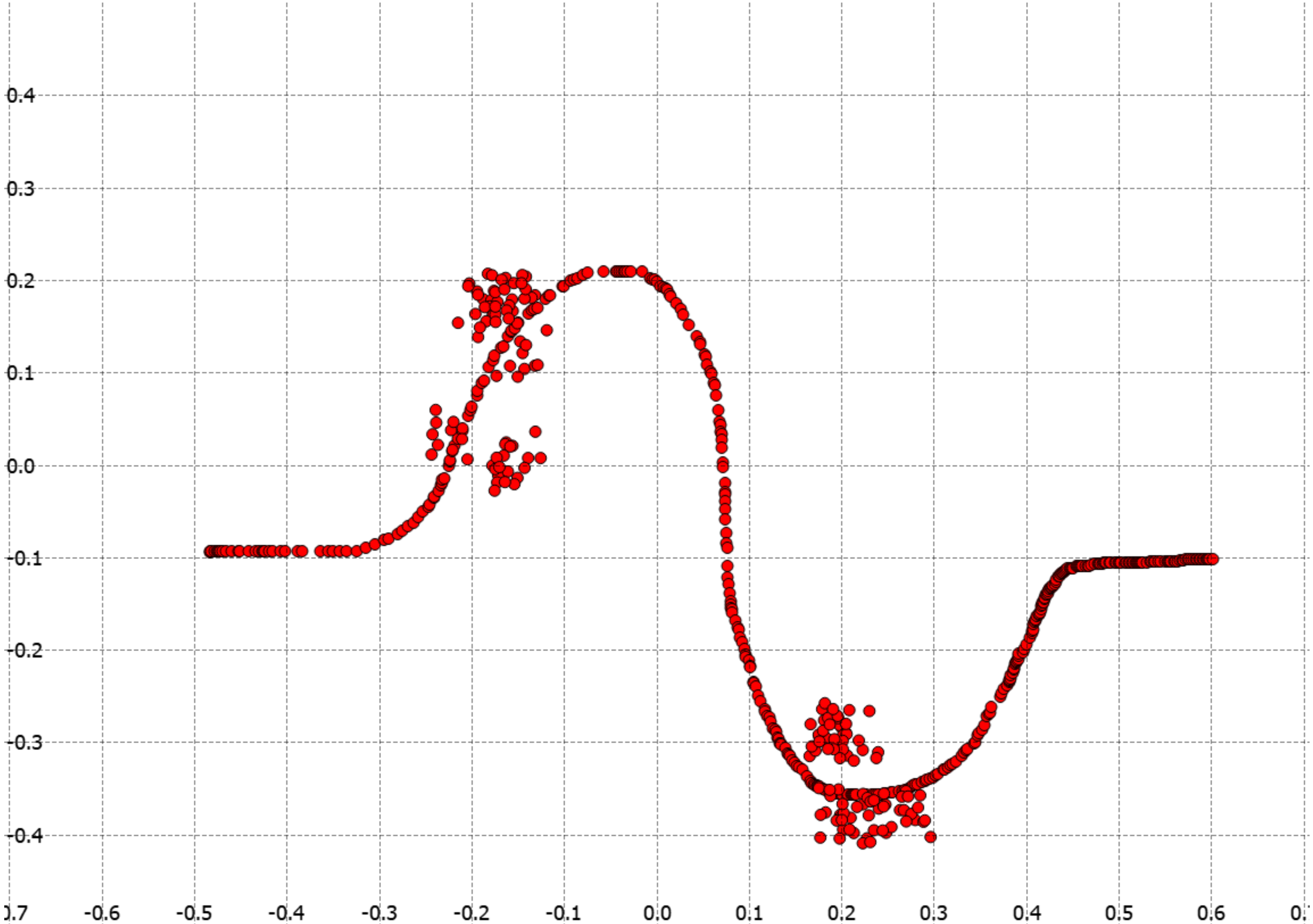
$$y = f(x) = \sum_{i=1}^{M} \left(\alpha_i - \alpha_i^*\right) k\left(x^i, x\right) + b$$



$$b \simeq 0$$

$$b = \frac{1}{M} \sum_{j=1}^{M} \left(y^j - \sum_{i=1}^{M} \left(\alpha_i - \alpha_i^*\right) k\left(x^j, x^i\right) \right)$$

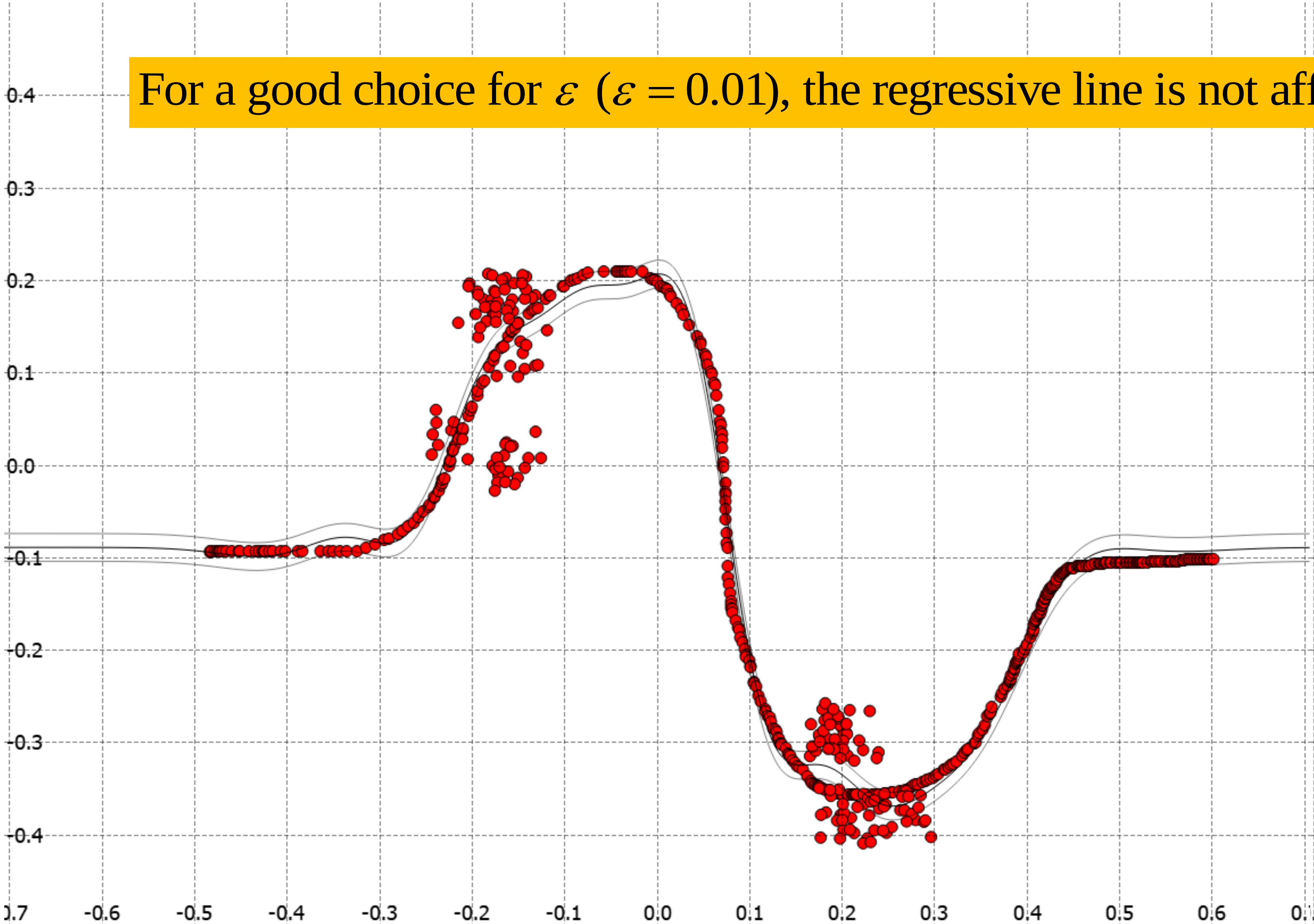
Regression: noise



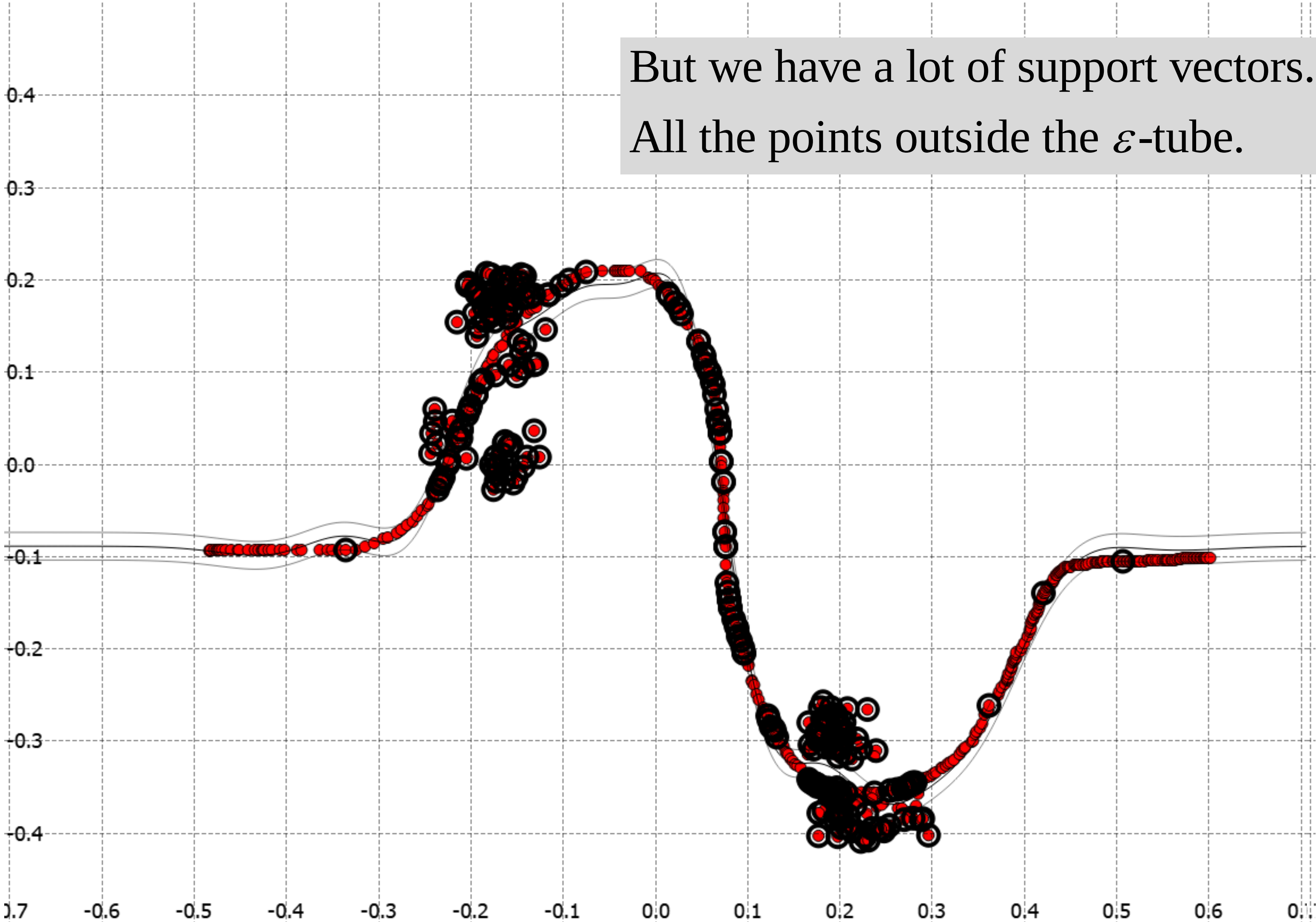
How would SVR handle this noise?

Regression: noise

For a good choice for ε ($\varepsilon = 0.01$), the regressive line is not affected.



Regression: noise

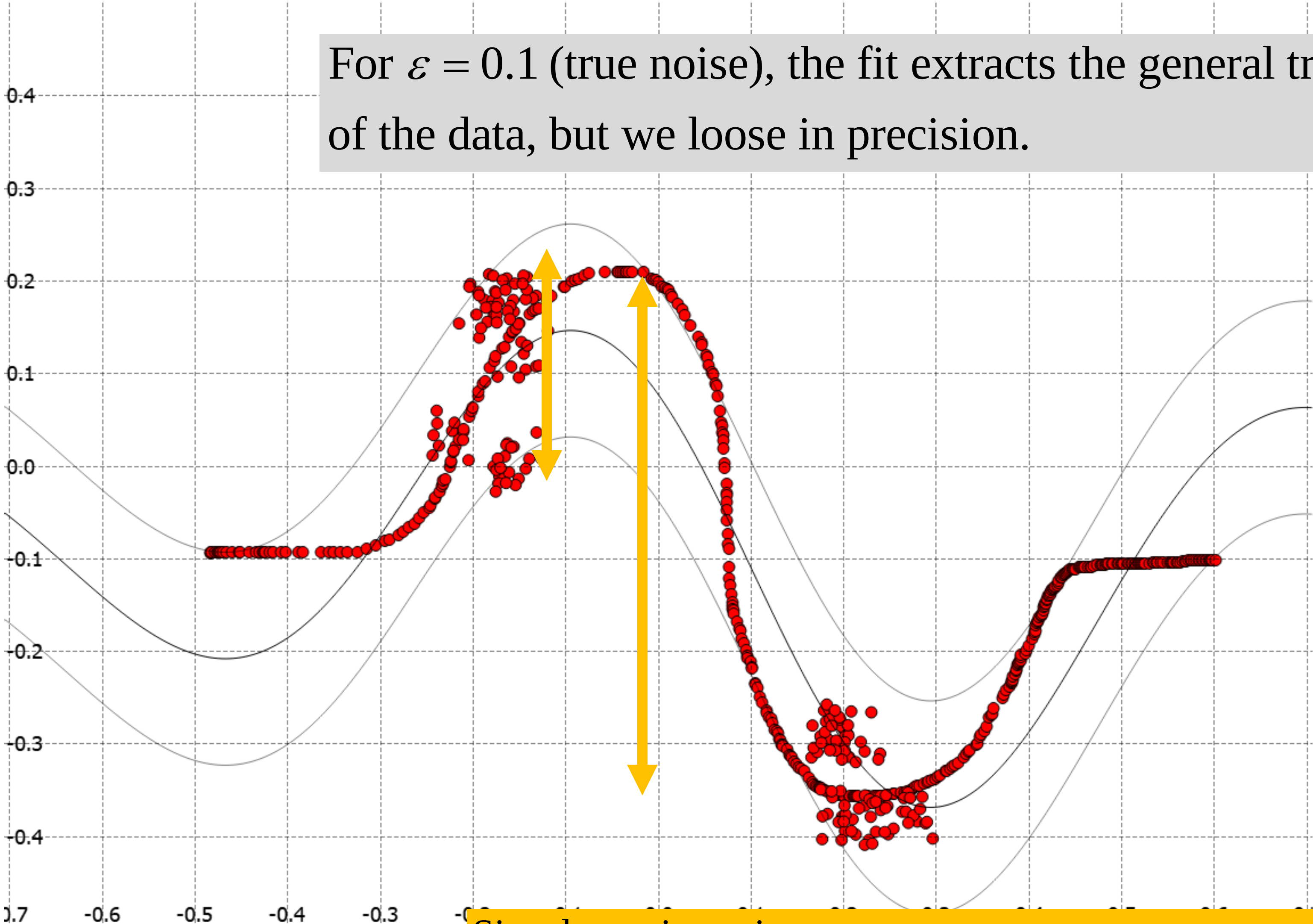


But we have a lot of support vectors.

All the points outside the ε-tube.

Regression: noise

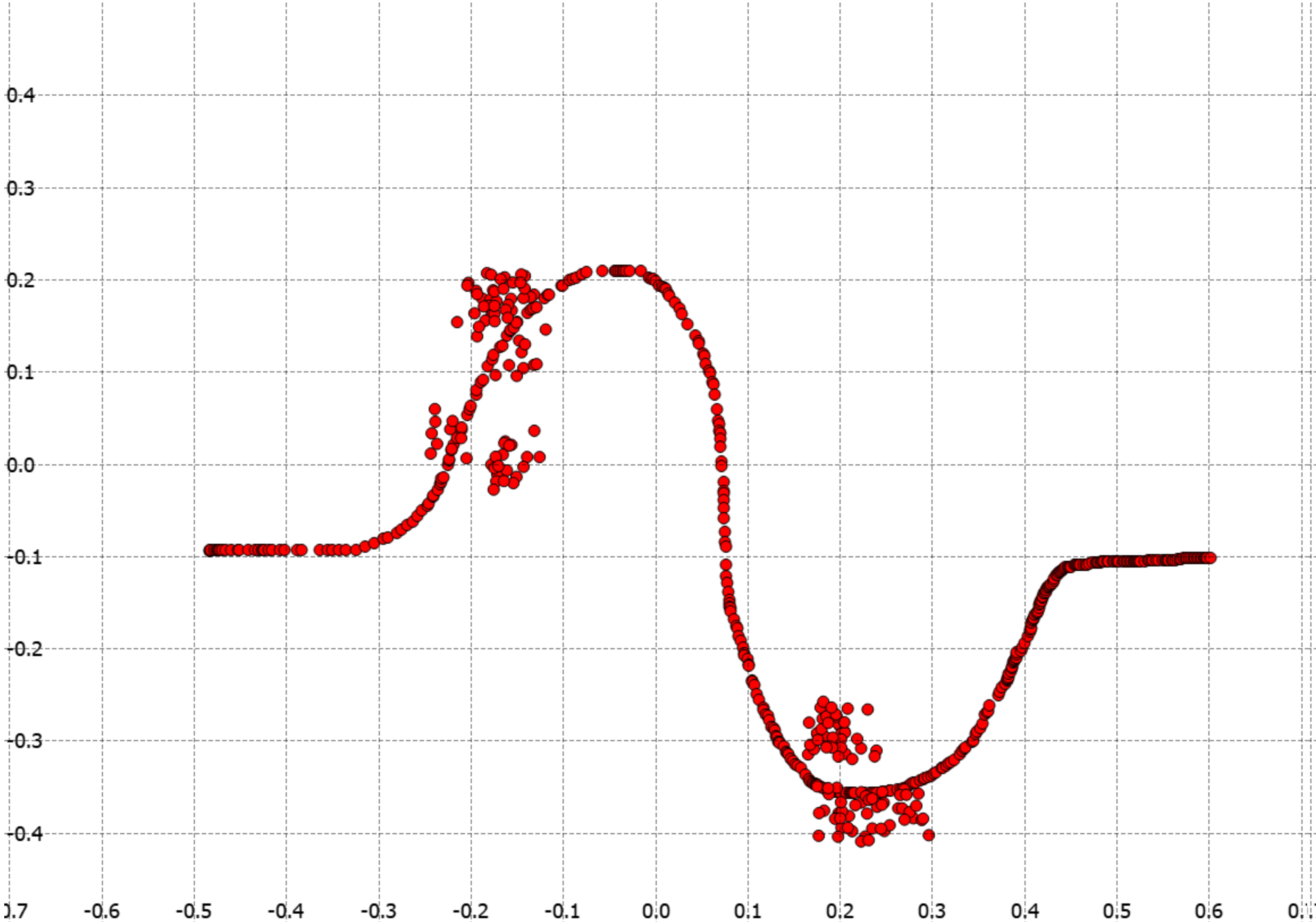
For $\varepsilon = 0.1$ (true noise), the fit extracts the general trend of the data, but we loose in precision.



Signal to noise ratio :

variance of y should be much larger than noise magnitude

Regression: noise



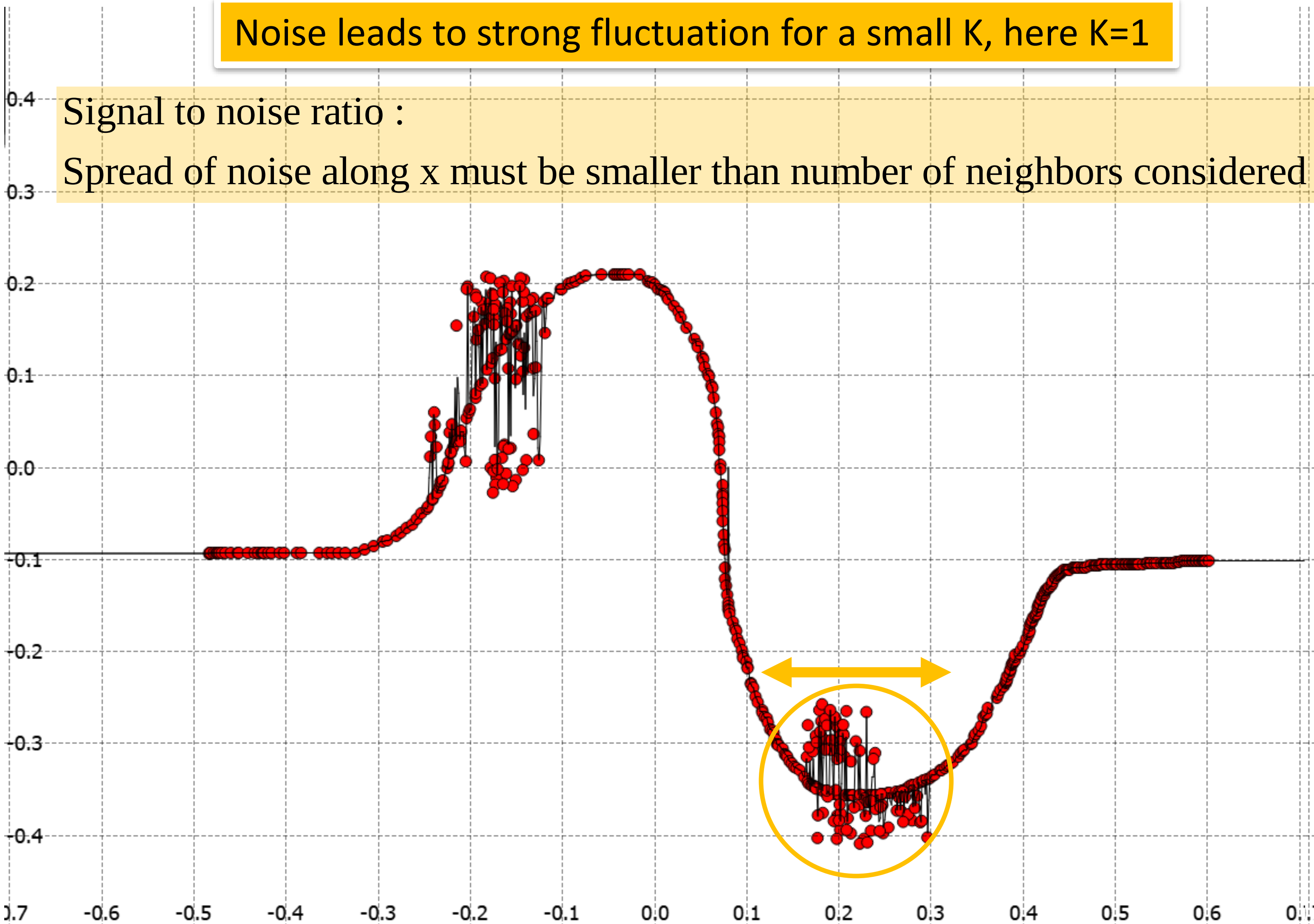
How would KNN handles this noise?

Regression: noise

Noise leads to strong fluctuation for a small K, here K=1

Signal to noise ratio :

Spread of noise along x must be smaller than number of neighbors considered (K)

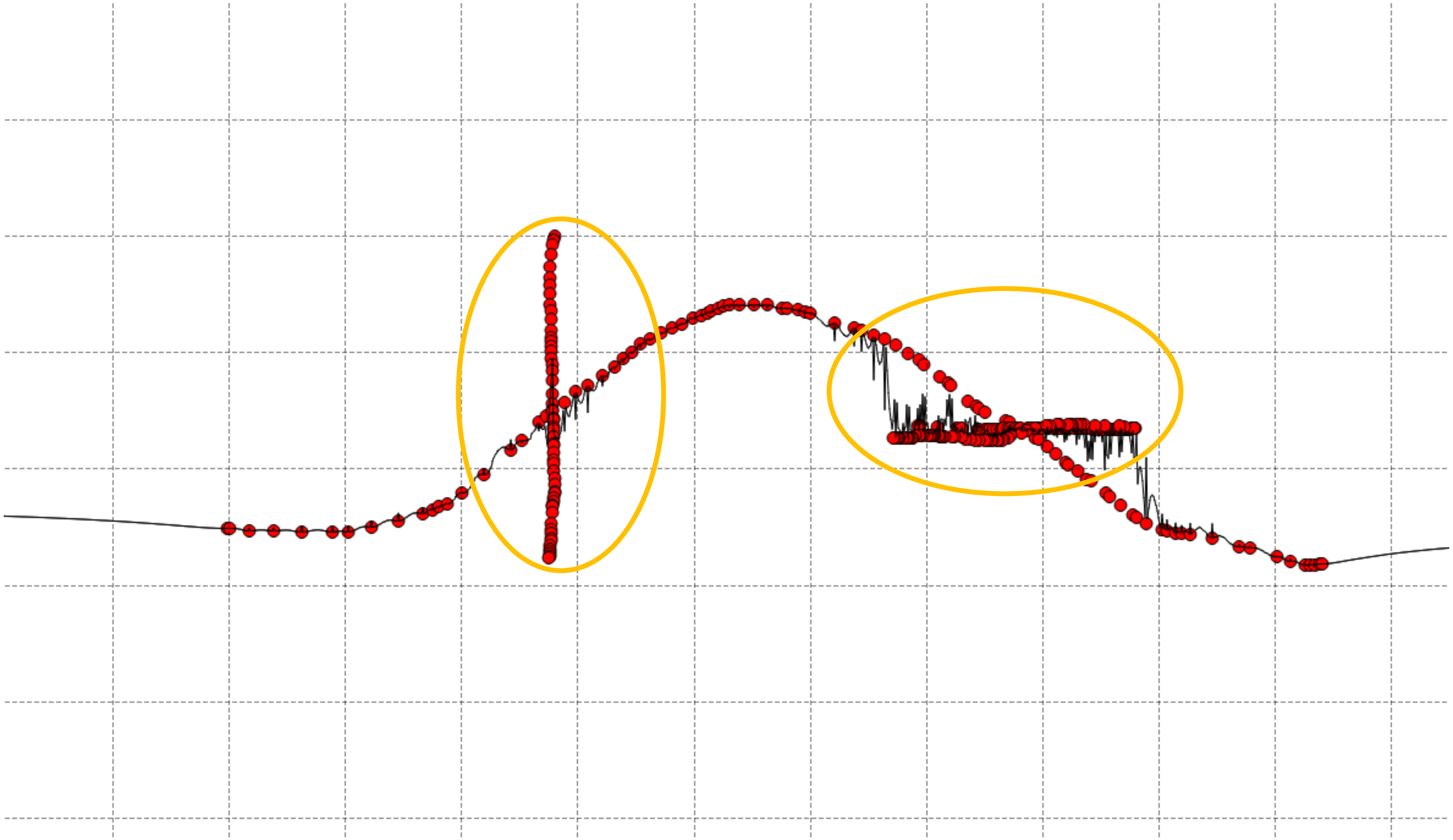


How would KNN handles this noise?

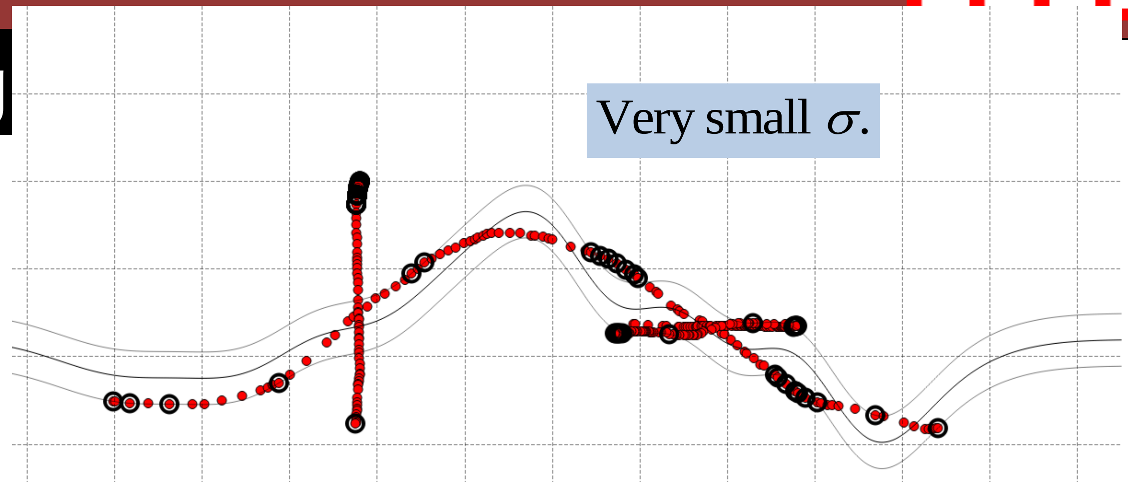
Regression: noise

Signal to noise ratio :

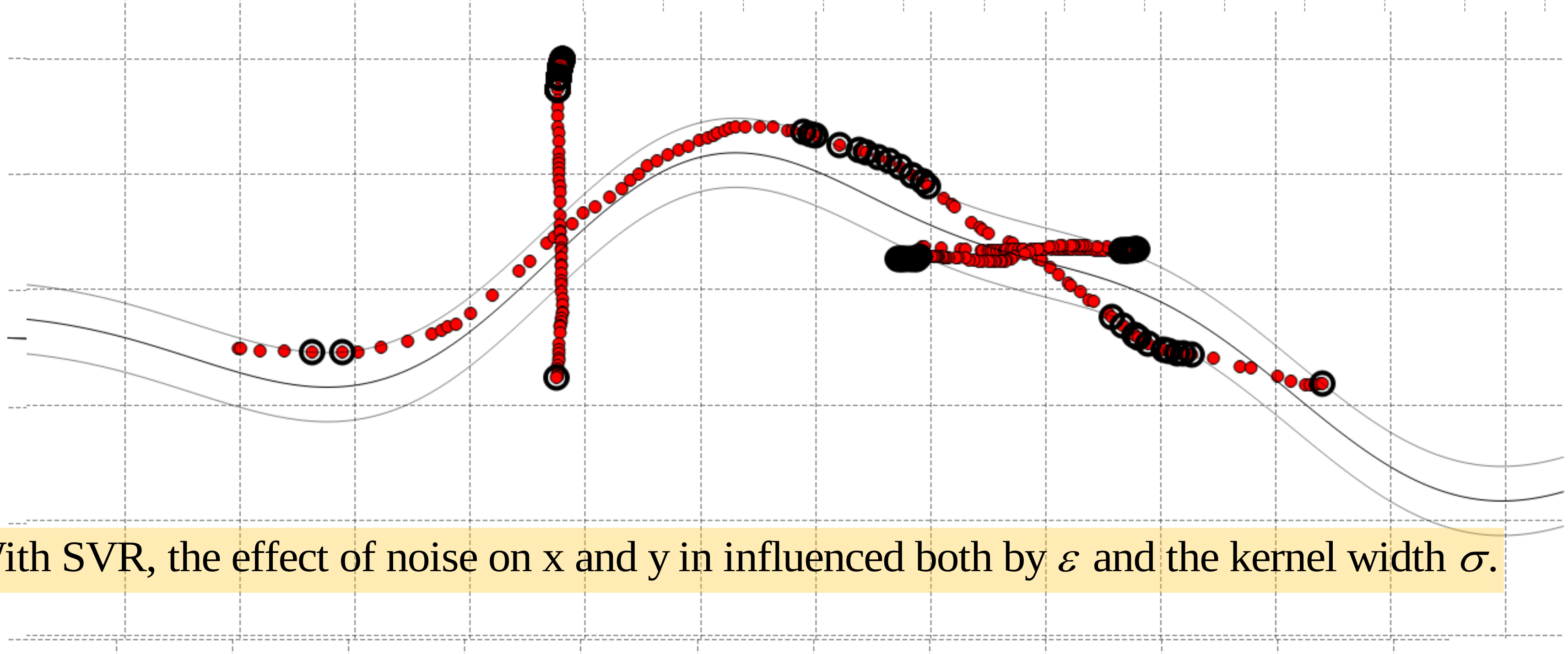
Spread of noise along x must be smaller than number of neighbors considered (K)



Reg

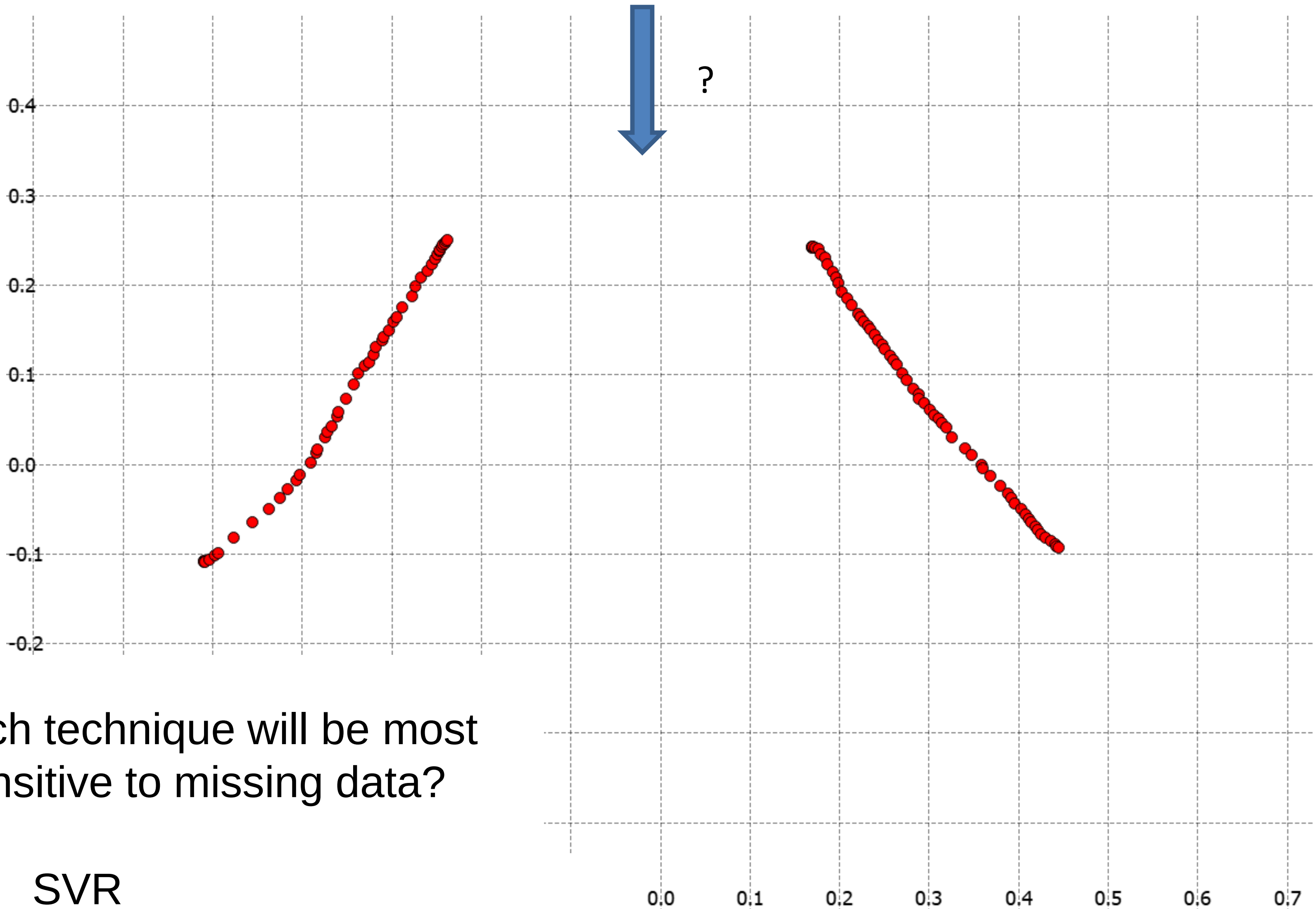


The smaller the kernel width σ, the more the noise on x influences the prediction



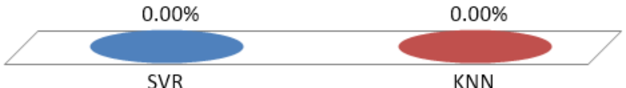
With SVR, the effect of noise on x and y in influenced both by ε and the kernel width σ.

Regression: interpolation



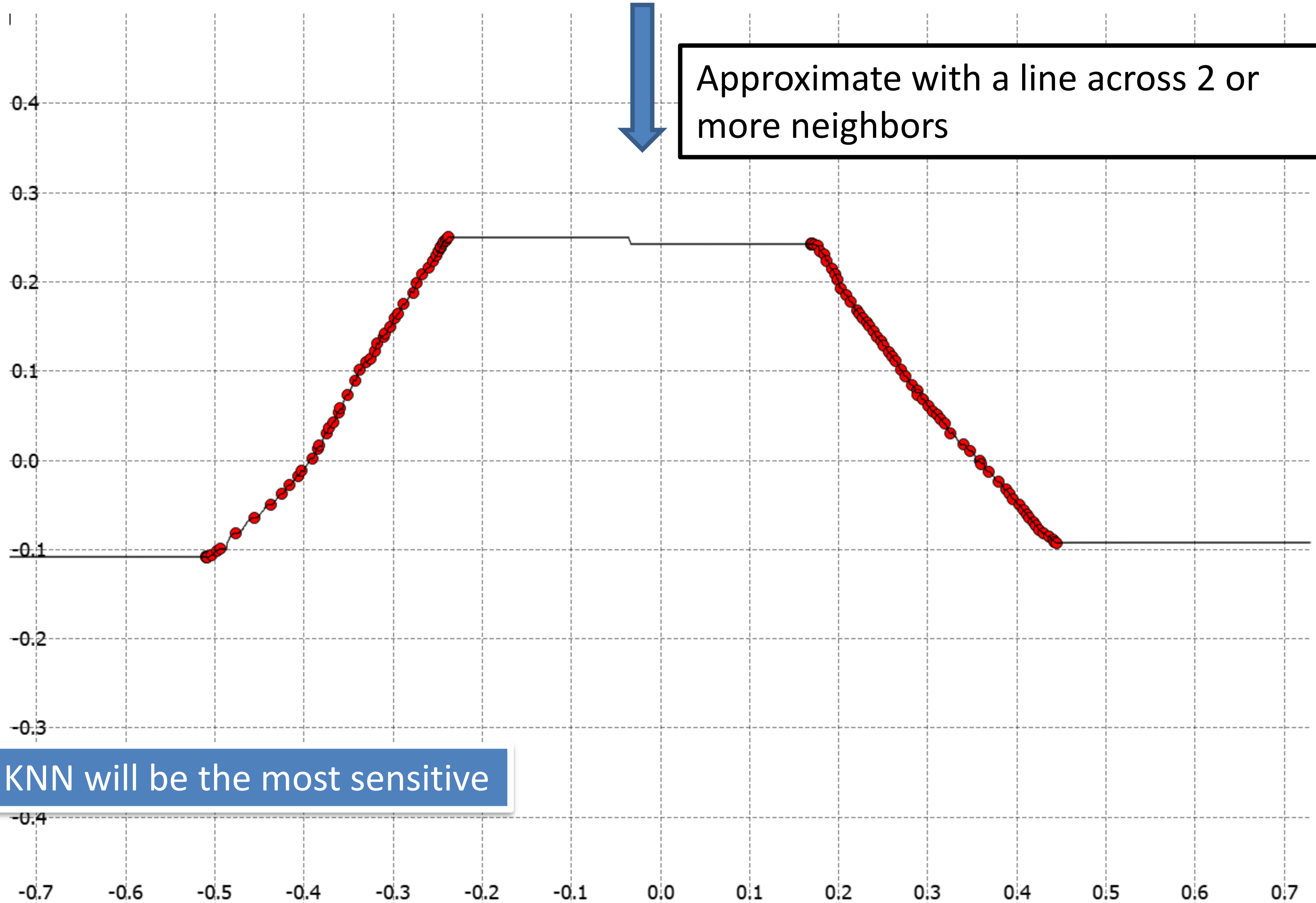
Which technique will be most sensitive to missing data?

A. SVR
B. KNN



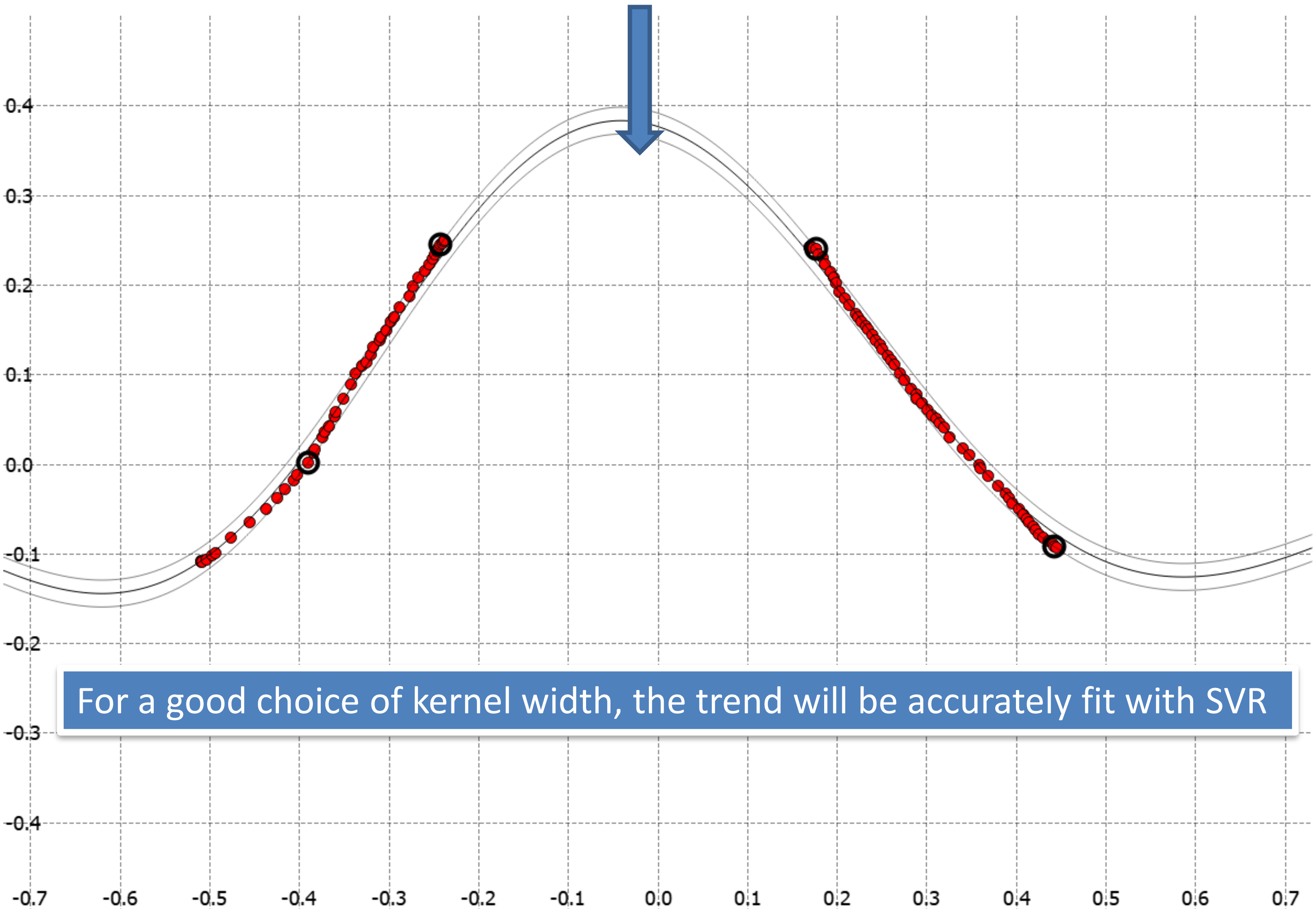
Regression: interpolation

Approximate with a line across 2 or more neighbors



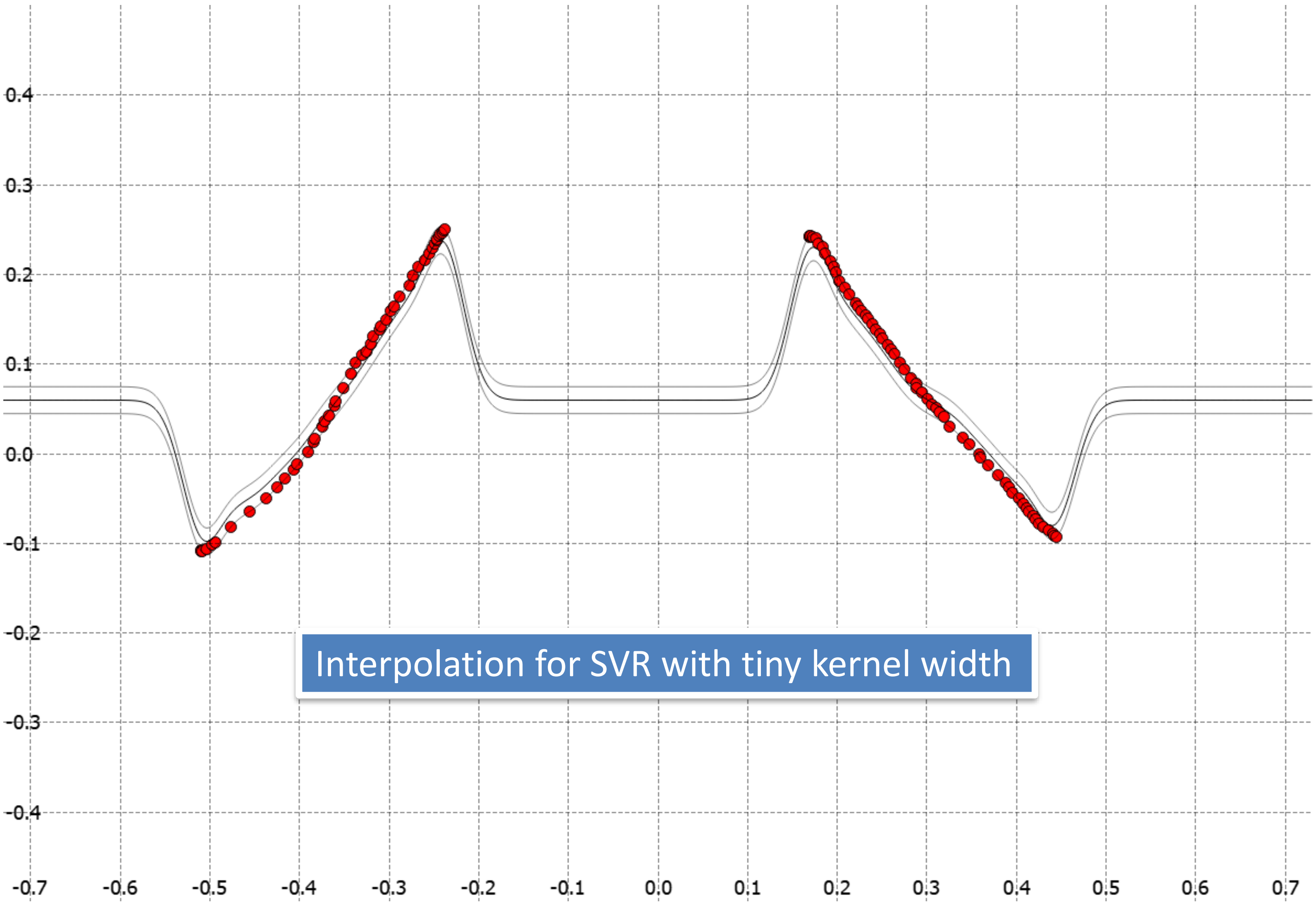
KNN will be the most sensitive

Regression: interpolation



For a good choice of kernel width, the trend will be accurately fit with SVR

Regression: interpolation



Gaussian mixture regression

We must first learn the joint distribution

$$p(x,y)=\sum_{k=1}^{K}\alpha_k \cdot p\left(x,y;\mu^k,\Sigma^k\right)$$

$$\alpha_k \in [0,1]$$

$$p\left(x,y;\mu^k,\Sigma^k\right)=N\left(\mu^k,\Sigma^k\right)$$

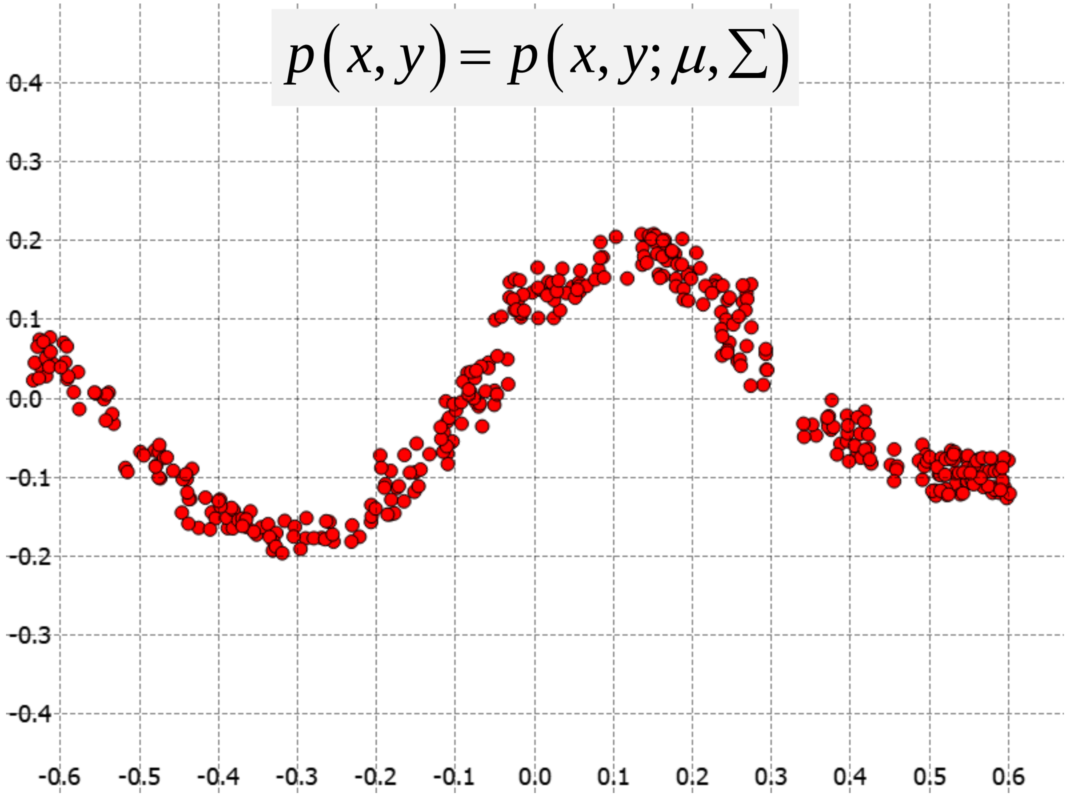
μ^k,Σ^k: mean and covariance matrix of Gaussian k.

And then we compute the regressive signal:

$$y=E\left\{p\left(y\,|\,x\right)\right\}$$

Gaussian mixture regression: 1 Gauss model

$$p(x,y) = p(x,y;\mu,\Sigma)$$



Gaussian mixture regression: 1 Gauss model

$$p(x,y) = p(x,y;\mu,\Sigma)$$

$$y = E\{p(y\,|\,x)\} = \sum_{k=1}^{K} \beta_k(x)\left(\mu_y^k + \Sigma_{yx}^k\left(\Sigma_{xx}^k\right)^{-1}\left(x-\mu_x^k\right)\right)$$

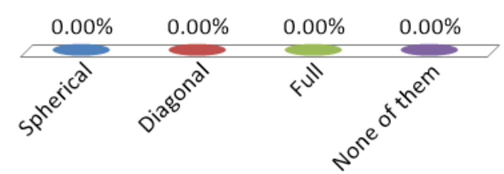
$$K = 1 \Rightarrow y = \mu_y + \Sigma_{yx}\left(\Sigma_{xx}\right)^{-1}\left(x-\mu_x\right)$$

$$\text{with} \quad \beta_k(x) = \frac{\alpha_k \cdot p\left(x;\mu_x^k,\Sigma_x^k\right)}{\sum_{k=1}^{K} \alpha_k \cdot p(x;\mu_x^k,\Sigma_x^k)}$$

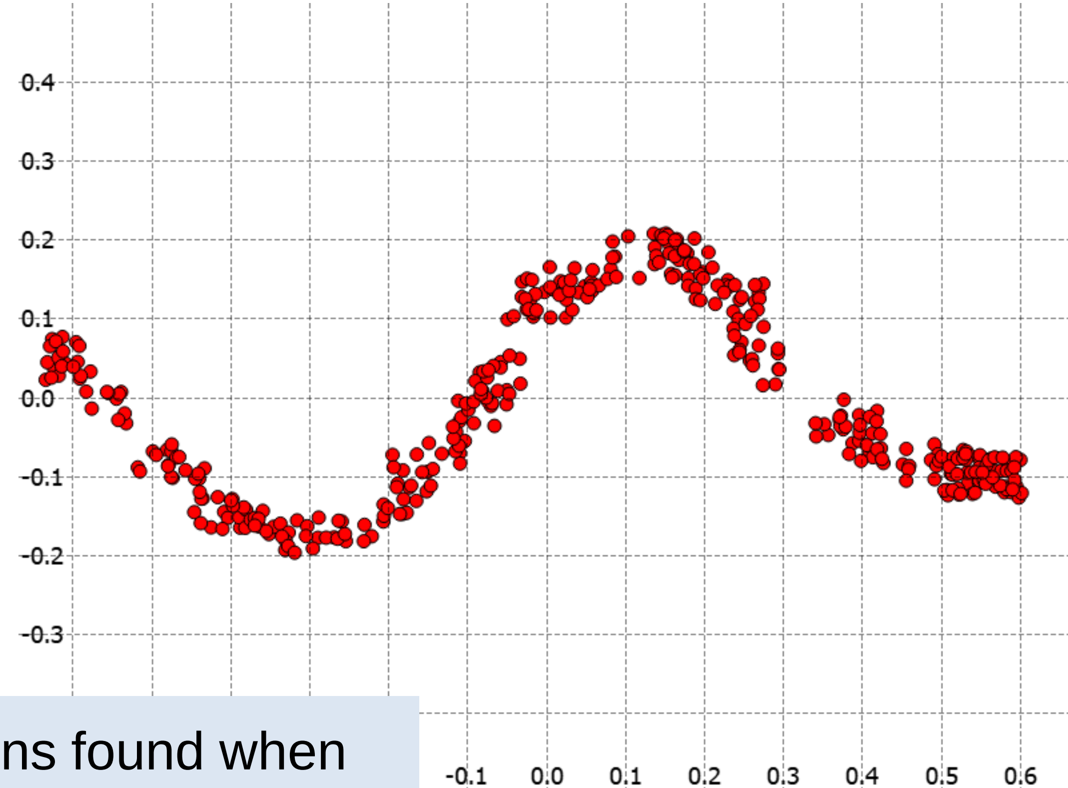
$$y = \Sigma_{yx}\left(\Sigma_{xx}\right)^{-1} x + \mu_y - \Sigma_{yx}\left(\Sigma_{xx}\right)^{-1}\mu_x$$

For which model is the regressive curve a straight line?

A. Spherical ✓
B. Diagonal ✓
C. Full ✓
D. None of them



Gaussian mixture regression: 1 Gauss model

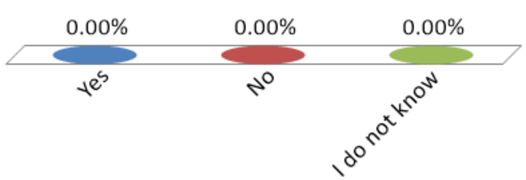


Are the solutions found when using spherical or diagonal covariance matrices different?

A. Yes

B. No ✓

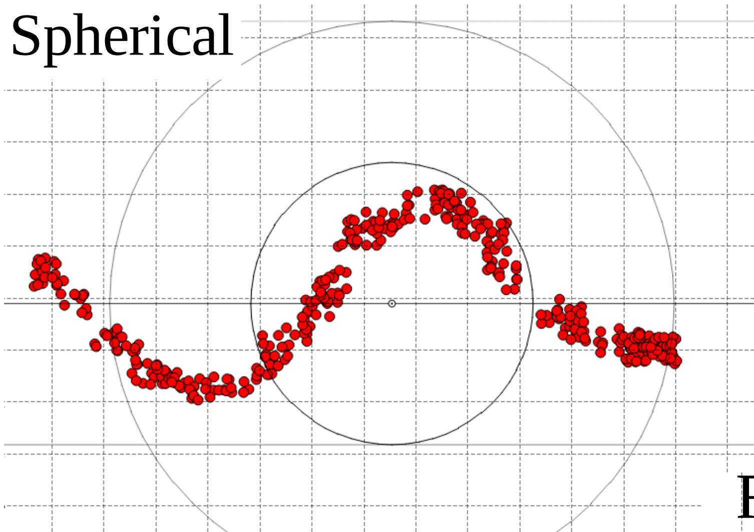
C. I do not know



Gaussian mixture regression: 1 Gauss model

$$y = \Sigma_{yx}\left(\Sigma_{xx}\right)^{-1} x + \mu_y - \Sigma_{yx}\left(\Sigma_{xx}\right)^{-1}\mu_x \qquad \Sigma = \begin{bmatrix} \Sigma_x & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_y \end{bmatrix}$$

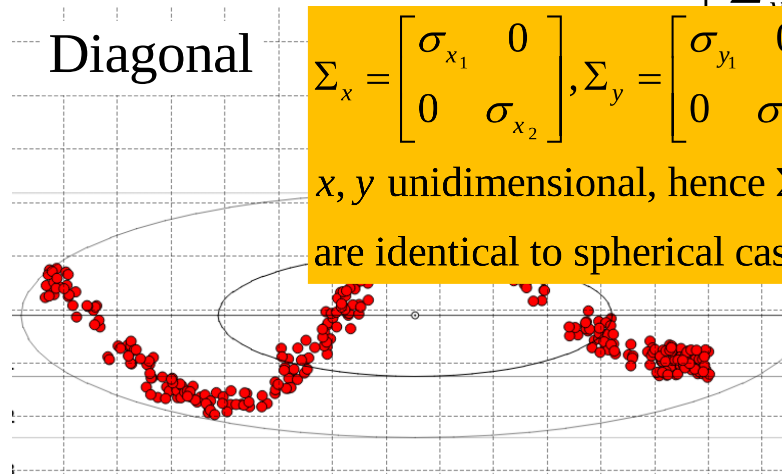
Spherical



$\Sigma_x = \sigma_x I, \Sigma_y = \sigma_y I,\ \Sigma_{xy} = 0$

Σ_{xy} is a matrix for multidimensional $x,\ y$.

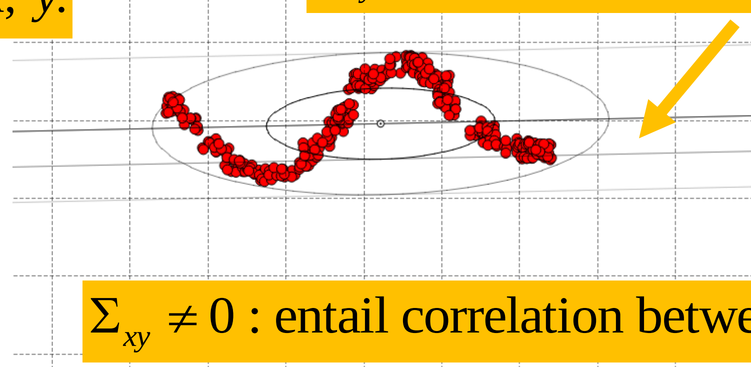
Diagonal



$\Sigma_x = \begin{bmatrix} \sigma_{x_1} & 0 \\ 0 & \sigma_{x_2} \end{bmatrix}, \Sigma_y = \begin{bmatrix} \sigma_{y_1} & 0 \\ 0 & \sigma_{y_2} \end{bmatrix},\ \Sigma_{xy} = 0$

x, y unidimensional, hence Σ_x, Σ_y are identical to spherical case.

Full

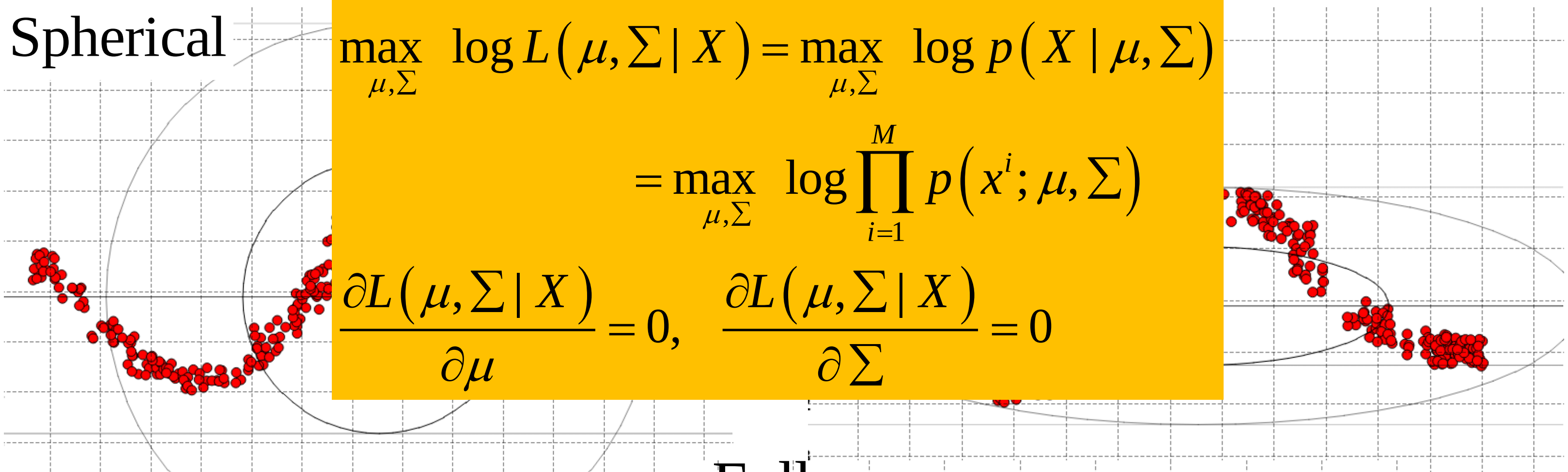


Σ_{xy} entails the correlation across x and y.

$\Sigma_{xy} \neq 0$: entail correlation between x and y.

GMR – 1 Gauss fct: Uniqueness of solution

Spherical



Solution of maximum likelihood problem:

$$\max_{\mu,\Sigma} \log L(\mu,\Sigma \mid X) = \max_{\mu,\Sigma} \log p(X \mid \mu,\Sigma)$$

$$= \max_{\mu,\Sigma} \log \prod_{i=1}^{M} p\left(x^i;\mu,\Sigma\right)$$

$$\frac{\partial L(\mu,\Sigma \mid X)}{\partial \mu} = 0, \quad \frac{\partial L(\mu,\Sigma \mid X)}{\partial \Sigma} = 0$$

Full

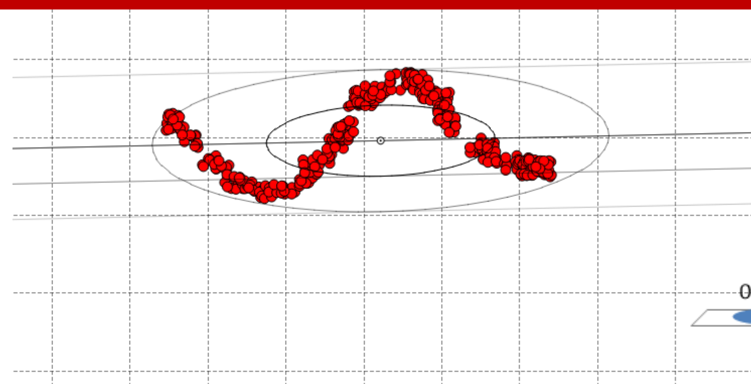
The solution is unique and found in closed-form: mean and variance of dataset, see exercises of pdf lecture

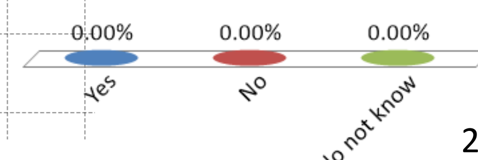
Is the solution unique?

A. Yes

B. No

C. I do not know





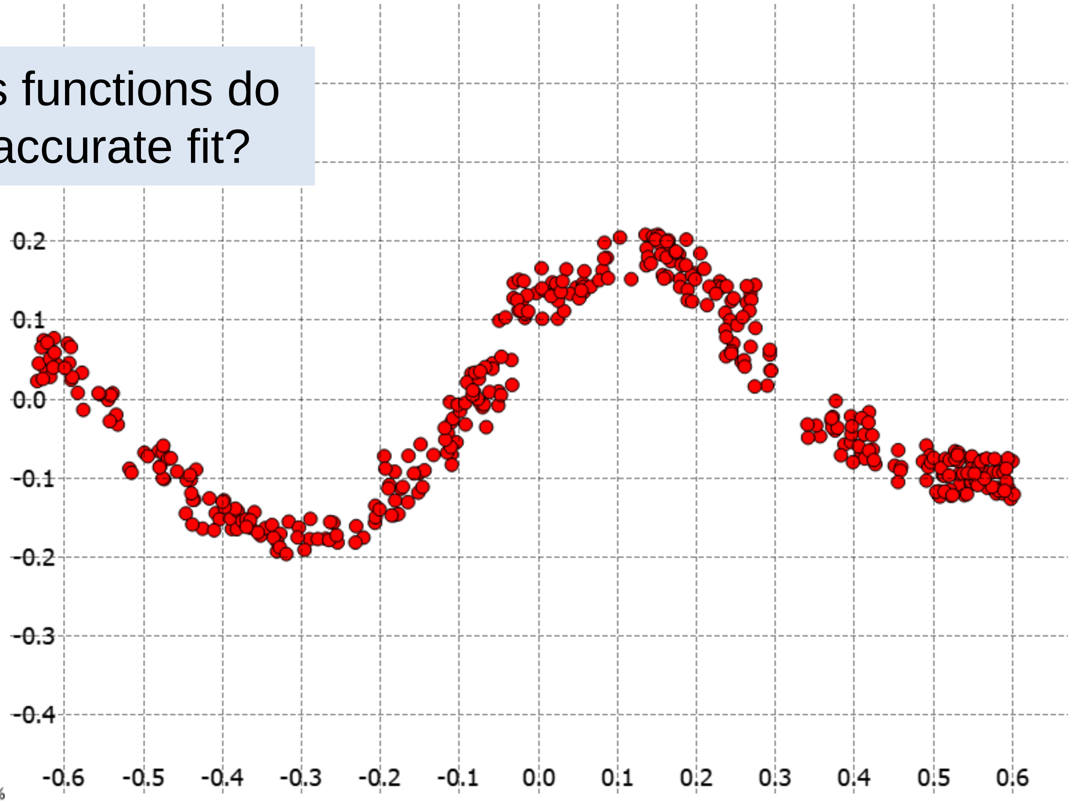
GMR: Multiple Gauss Functions

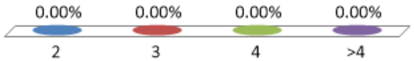
$$p(x,y)=\sum_{k=1}^{K}\alpha_k \cdot p\left(x,y;\mu^k,\Sigma^k\right), \qquad \text{with } p\left(x,y;\mu^k,\Sigma^k\right)=N\left(\mu^k,\Sigma^k\right)$$

μ^i,Σ^i : mean and covariance matrix of Gaussian k.

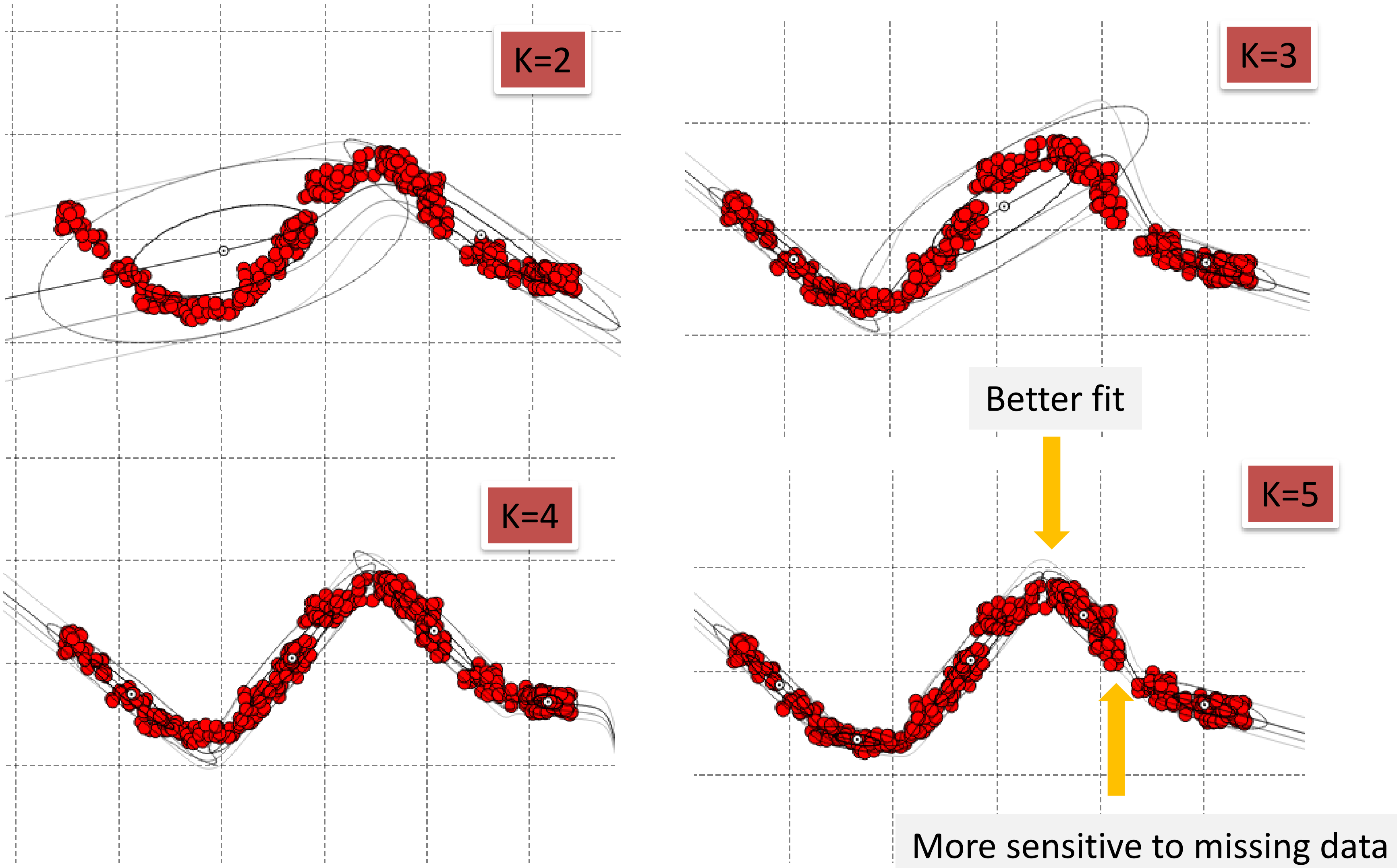
How many Gauss functions do we need for an accurate fit?

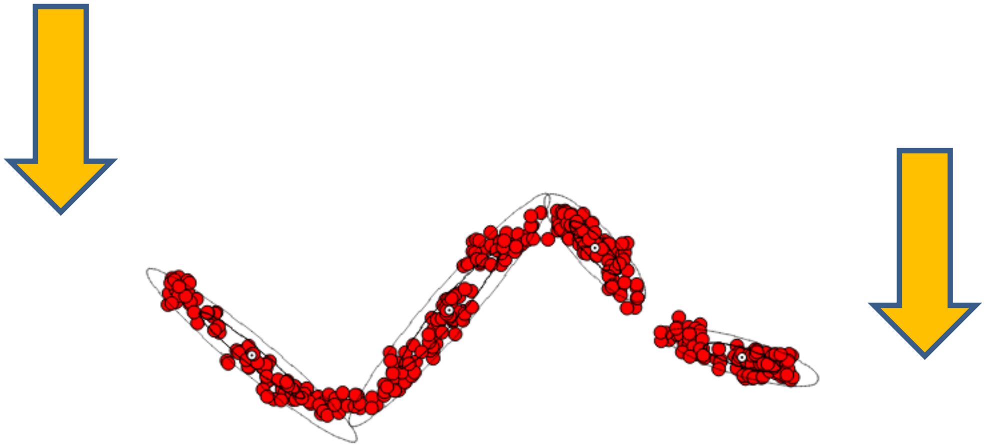
A. 2
B. 3
C. 4
D. >4





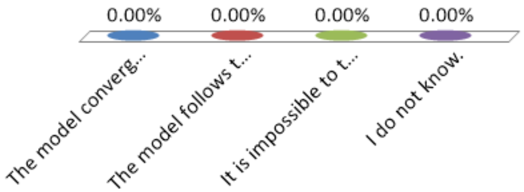
Accuracy of fit with multiple Gauss functions





What does the model predict away from the data?

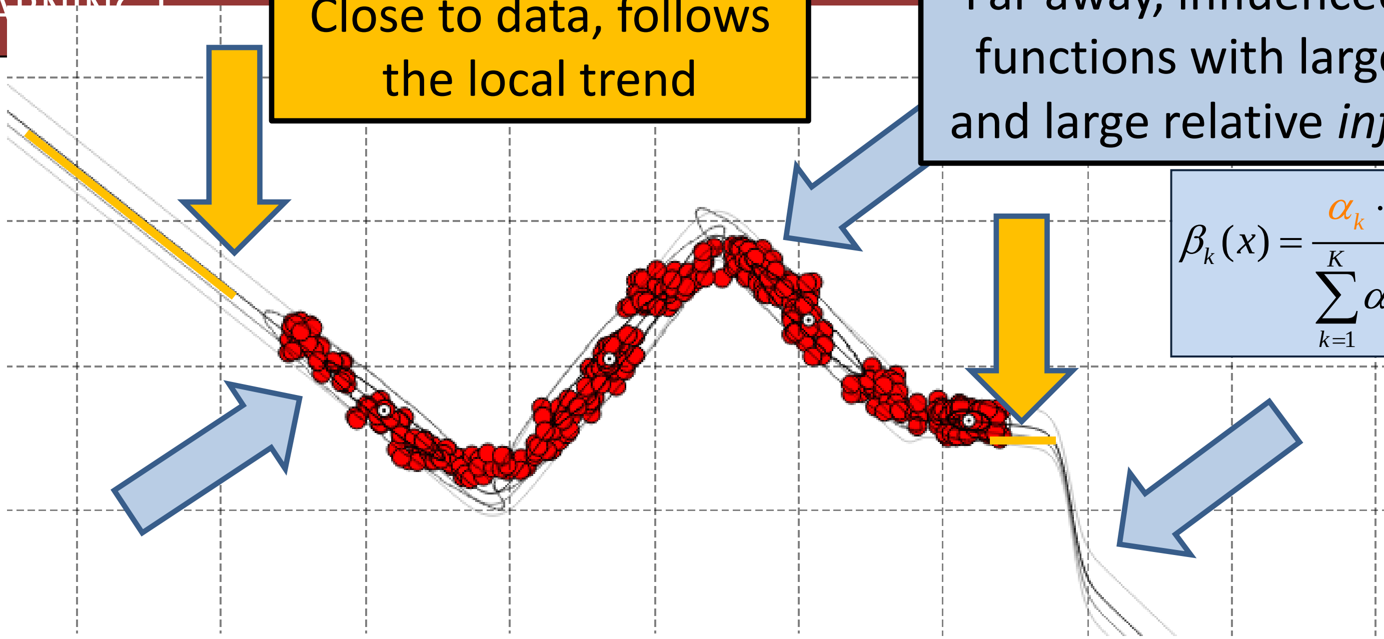
A. The model converges to a single value.
B. The model follows the local trend.
C. It is impossible to tell.
D. I do not know.



Close to data, follows the local trend

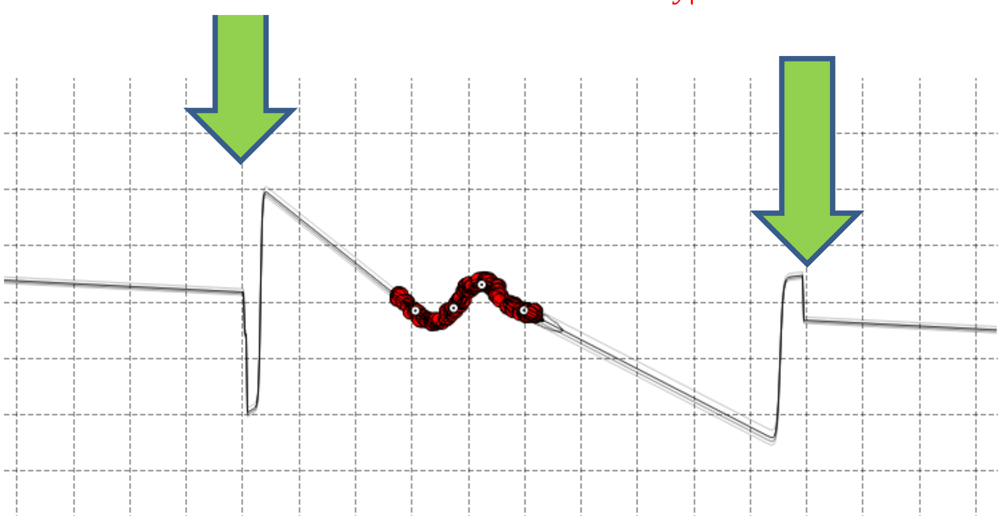
Far away, influenced by Gauss functions with large variance and large relative *influence* (α_k)

$$\beta_k(x) = \frac{\alpha_k \cdot p\left(x; \mu_x^k, \Sigma_x^k\right)}{\sum_{k=1}^{K} \alpha_k \cdot p(x; \mu_x^k, \Sigma_x^k)}$$



$$y = \sum_{k=1}^{K} \beta_k\left(x\right) \underbrace{\left(\mu_y^k + \Sigma_{yx}^k \left(\Sigma_{xx}^k\right)^{-1} \left(x - \mu_x^k\right)\right)}_{\tilde{\mu}_{y|x}^k(x)} \Rightarrow y = \sum_{k=1}^{K} \beta_k\left(x\right) \cdot \tilde{\mu}_{y|x}^k\left(x\right)$$

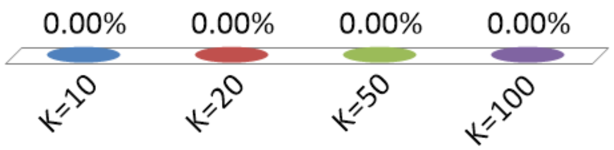
Linear combination of *K local regressive* models

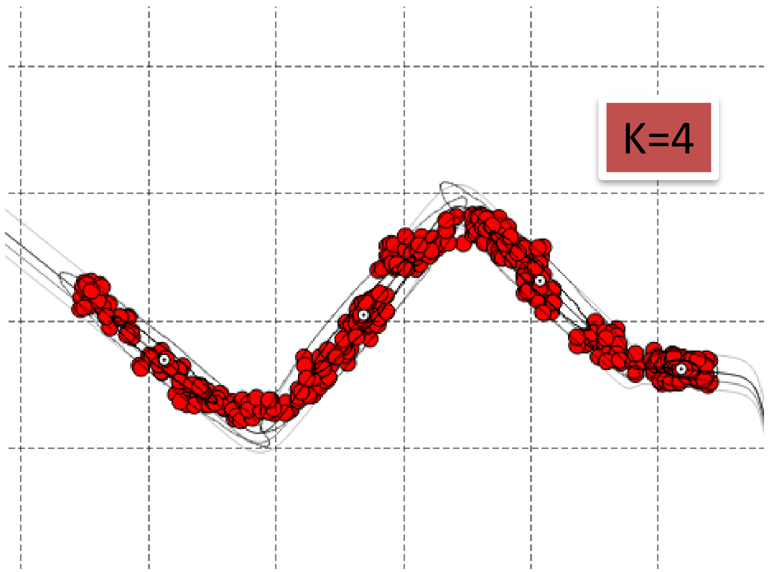


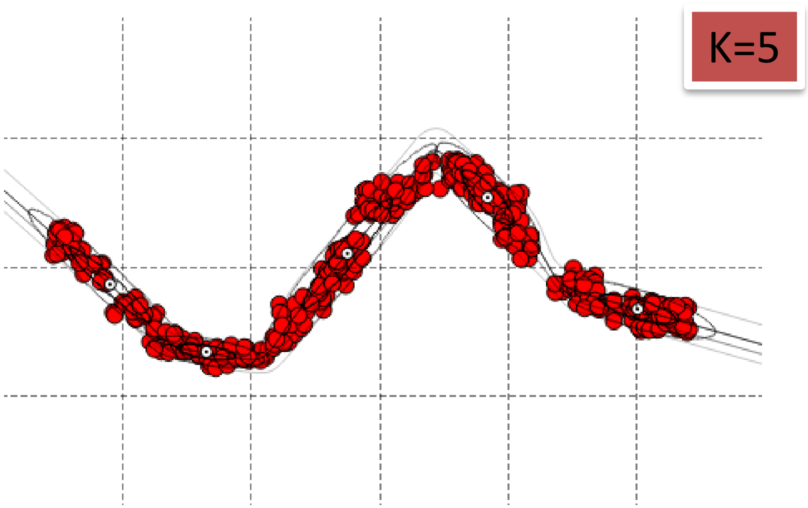
Very far from the data the prediction can be completely unexpected.

Knowing that we have ~100 points and use a 2/3rd training/testing ratio, for which K would we start seeing overfitting?

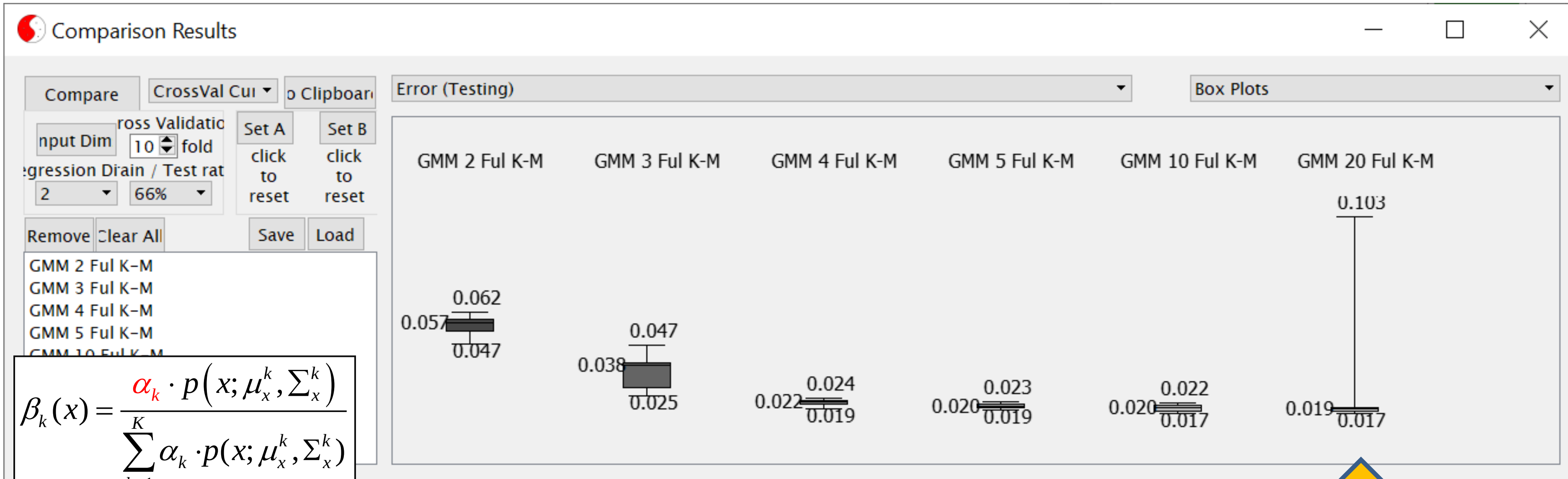
A. K=10
B. K=20
C. K=50
D. K=100







Overfitting with multiple Gauss functions



$$\beta_k(x) = \frac{\alpha_k \cdot p\left(x; \mu_x^k, \Sigma_x^k\right)}{\sum_{k=1}^{K} \alpha_k \cdot p(x; \mu_x^k, \Sigma_x^k)}$$

$$y = \sum_{k=1}^{K} \beta_k(x)\left(\mu_y^k + \Sigma_{yx}^k \left(\Sigma_{xx}^k\right)^{-1}\left(x - \mu_x^k\right)\right)$$

Clear overfit

$$\underset{\text{Priors}}{K} + \underset{\text{Means}}{K \cdot N} + \underbrace{K \cdot \left(\frac{N(N+1)}{2}\right)}_{\text{Full Covariance Matrix}}$$

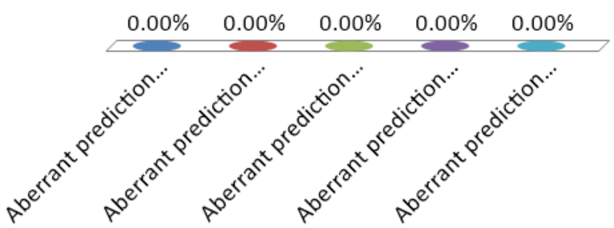
ratio · # datapoints · dimension = 0.66 · 100 · 2 = 132

K=20: # parameters =120

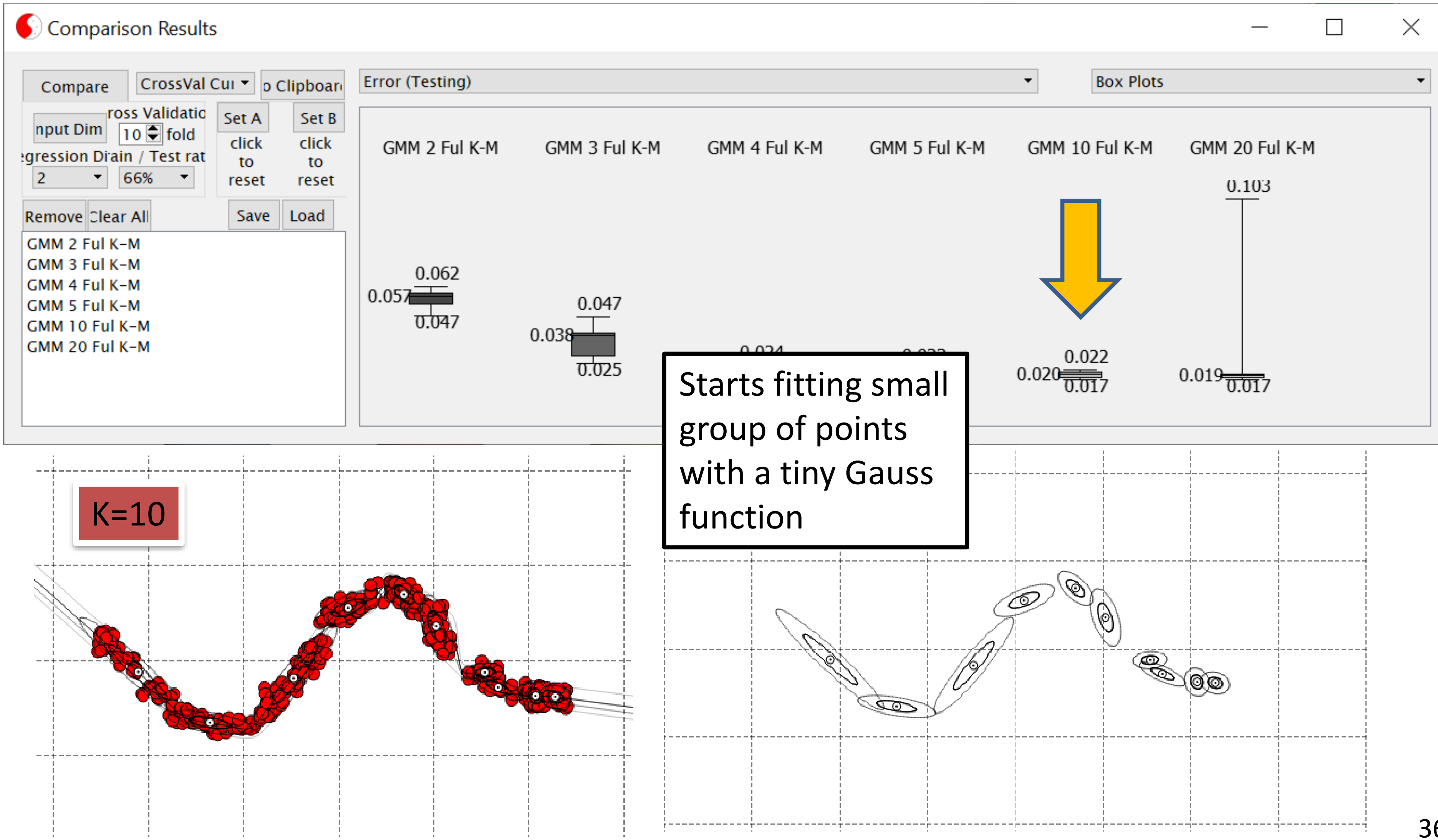
What would be the effect of overfitting in GMR ?

Multiple correct responses

A. Aberrant prediction when far from the dataset
B. Aberrant prediction even for query points close to the dataset
C. Aberrant prediction could be any value, even values never seen at training.
D. Aberrant prediction would be a value that remains within variance of the dataset.
E. Aberrant prediction can only be "zero".

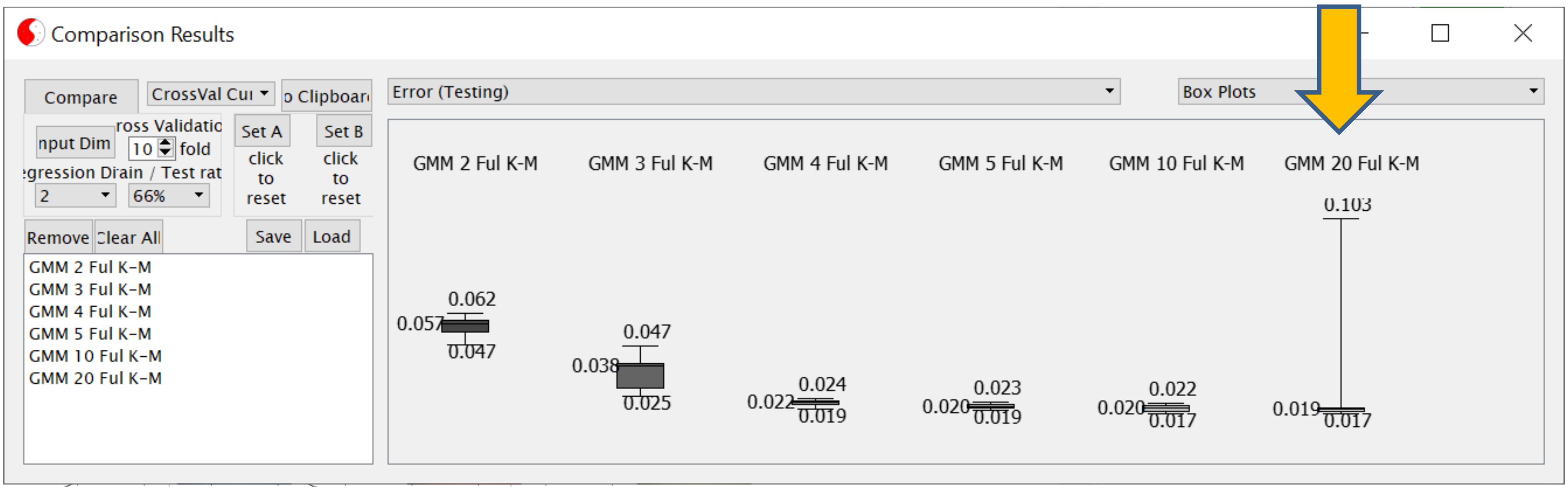


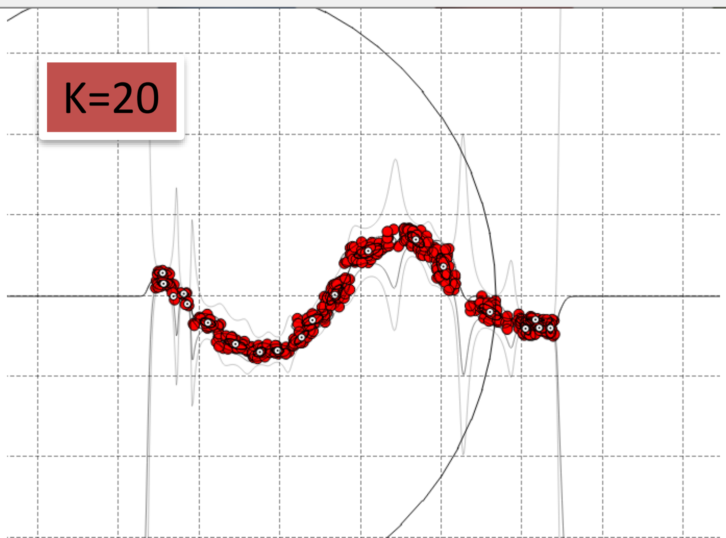
Overfitting with multiple Gauss functions

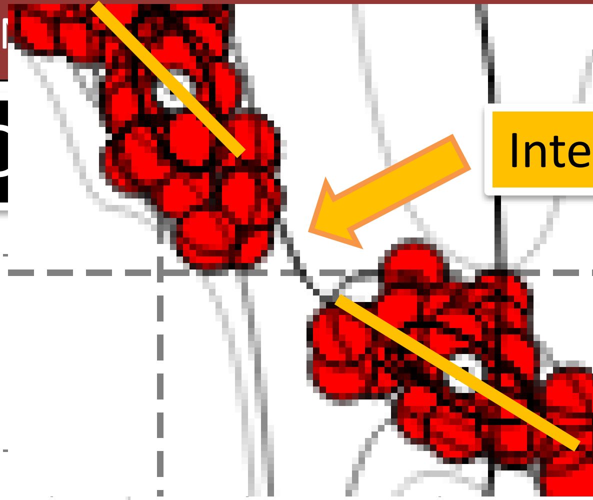


Starts fitting small group of points with a tiny Gauss function

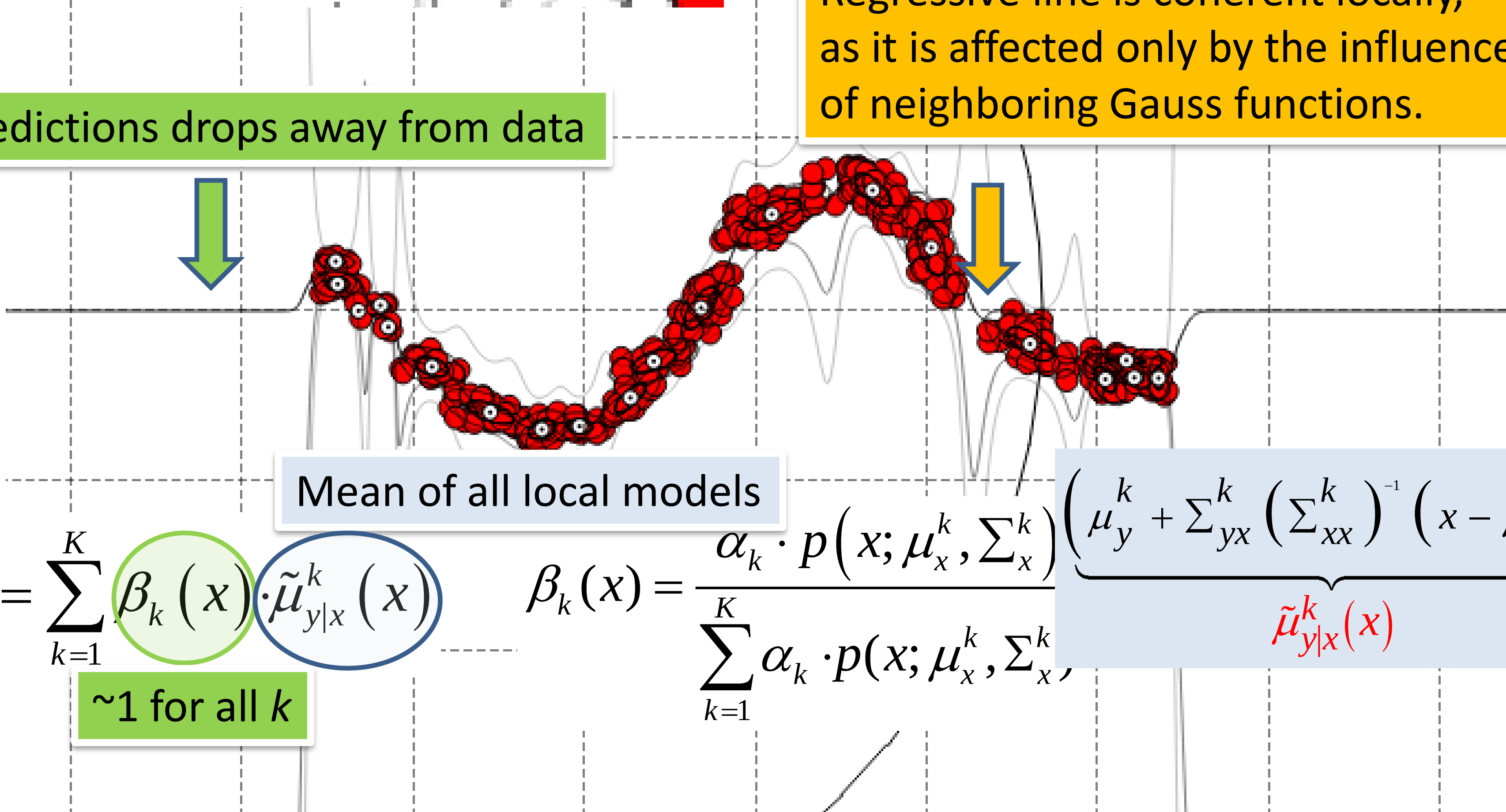
Overfitting with multiple Gauss functions







Interpolates across two neighbor linear models.



Regressive line is coherent locally, as it is affected only by the influence of neighboring Gauss functions.

Predictions drops away from data

Mean of all local models

$$y=\sum_{k=1}^{K}\beta_k(x)\cdot\tilde{\mu}^k_{y|x}(x)$$

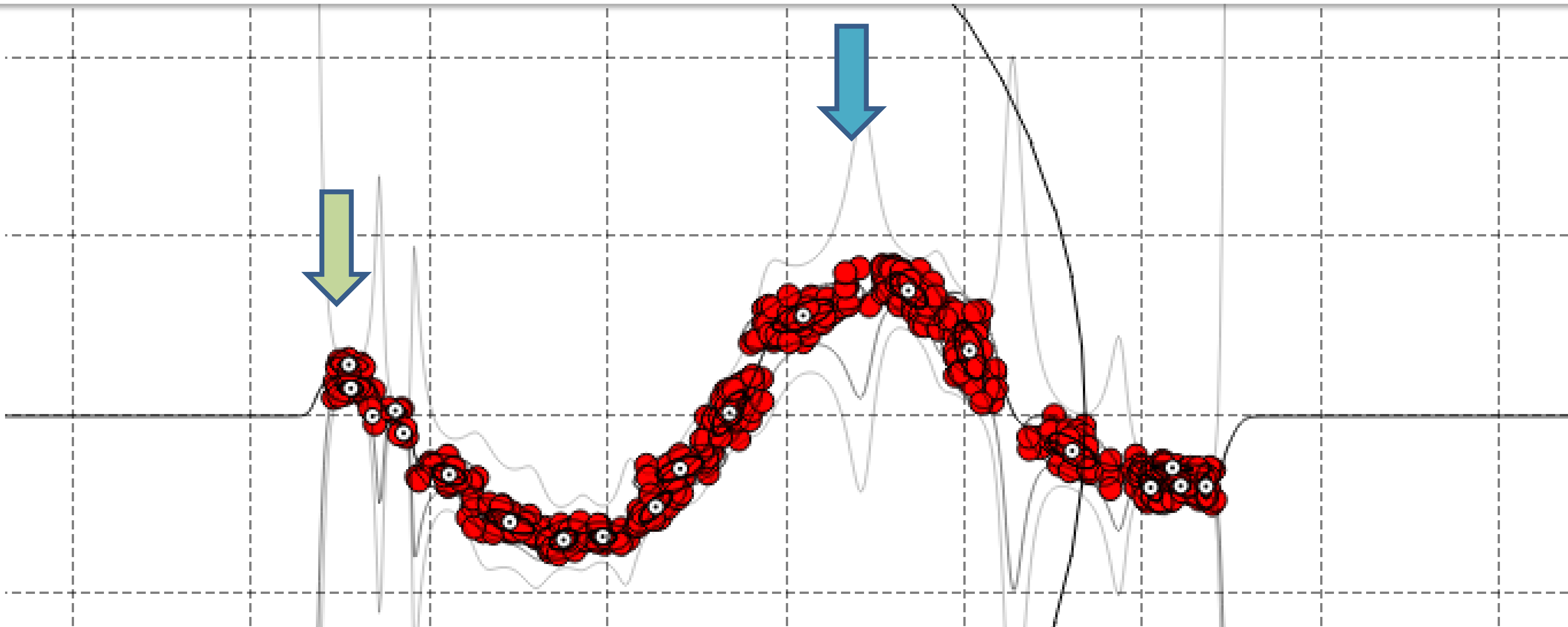
~1 for all *k*

$$\beta_k(x)=\frac{\alpha_k\cdot p\left(x;\mu_x^k,\Sigma_x^k\right)}{\sum_{k=1}^{K}\alpha_k\cdot p(x;\mu_x^k,\Sigma_x^k)}$$

$$\underbrace{\left(\mu_y^k+\Sigma_{yx}^k\left(\Sigma_{xx}^k\right)^{-1}\left(x-\mu_x^k\right)\right)}_{\tilde{\mu}^k_{y|x}(x)}$$

Overfitting with multiple Gauss functions

Aberrant predictions of variance –not enough statistics to estimate all parameters, when single full Gauss function estimated from too few datapoints.

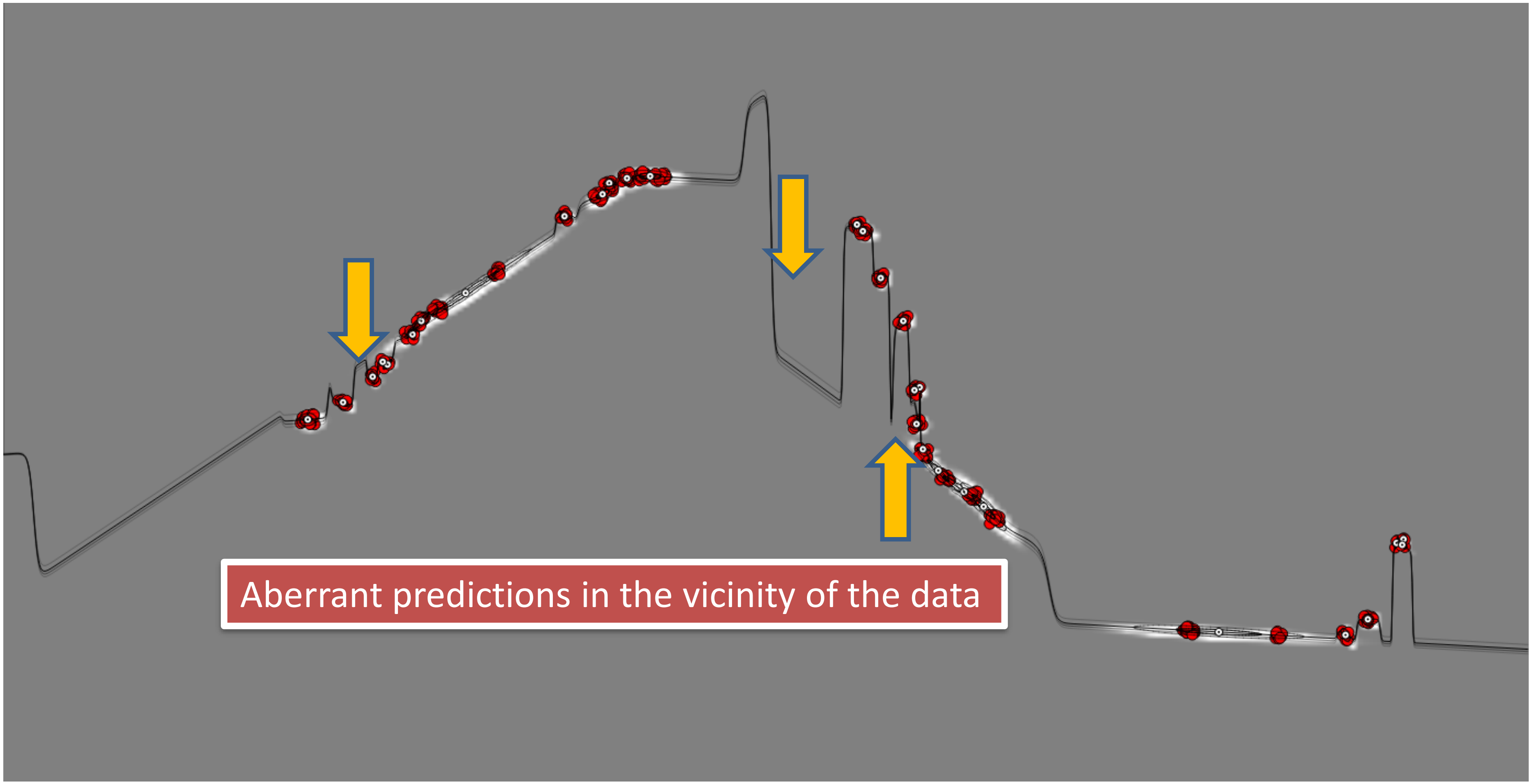


$$\mathrm{var}\left\{p\left(y\,|\,x\right)\right\} = \sum_{k=1}^{K} \beta_k\left(x\right)\cdot\left(\left(\tilde{\mu}^k_{y|x}\left(x\right)\right)^2 + \tilde{\Sigma}^k_{y|x}\right) - \left(\sum_{k=1}^{K}\left(\beta_k\left(x\right)\cdot\tilde{\mu}^k_{y|x}\left(x\right)\right)\right)^2$$

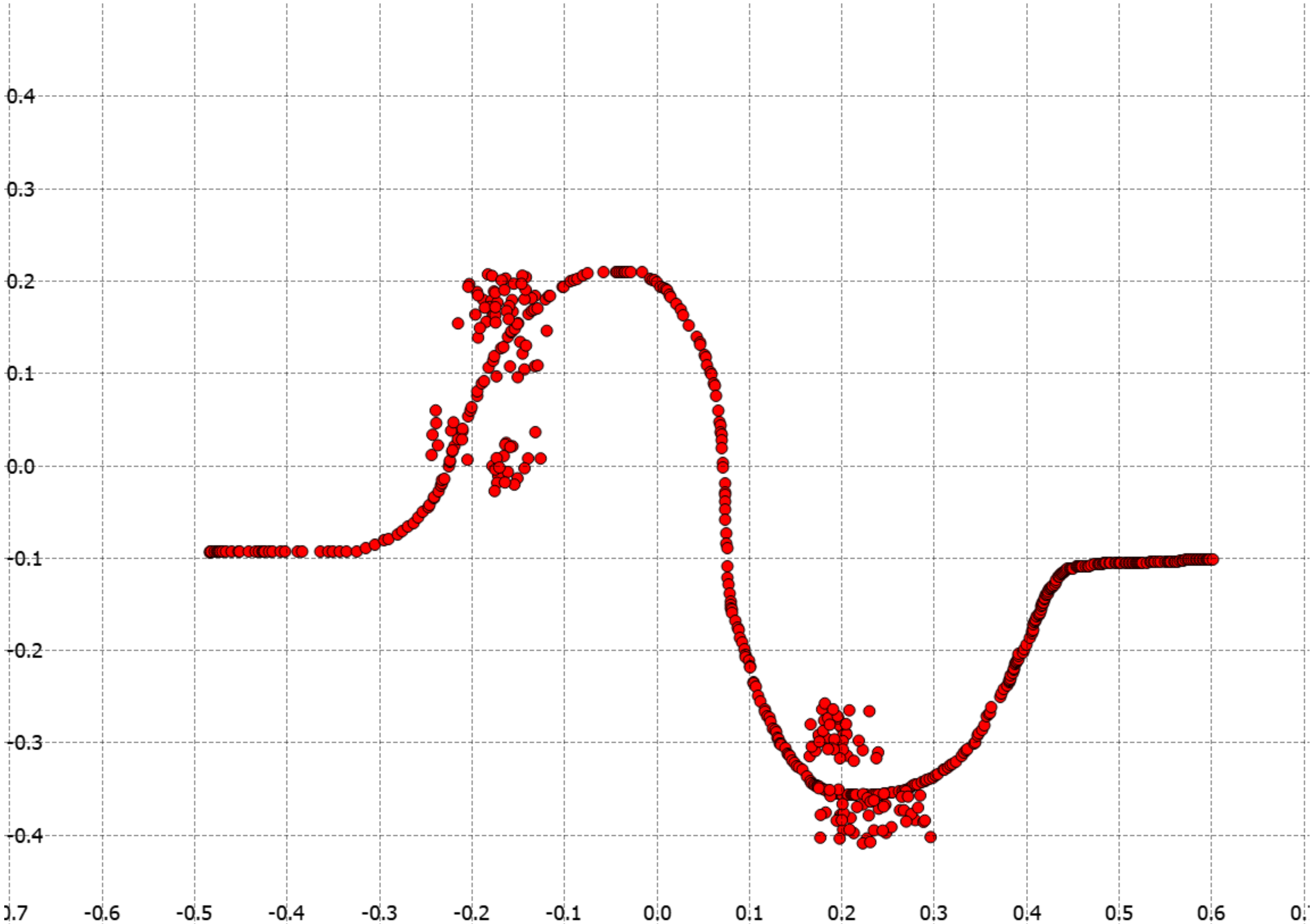
$$\text{with } \tilde{\Sigma}^k_{y|x} = \Sigma^k_{yy} - \Sigma^k_{yx}\left(\Sigma^k_{xx}\right)^{-1}\Sigma^k_{xy}$$

Numerical instabilities

Overfitting with multiple Gauss functions

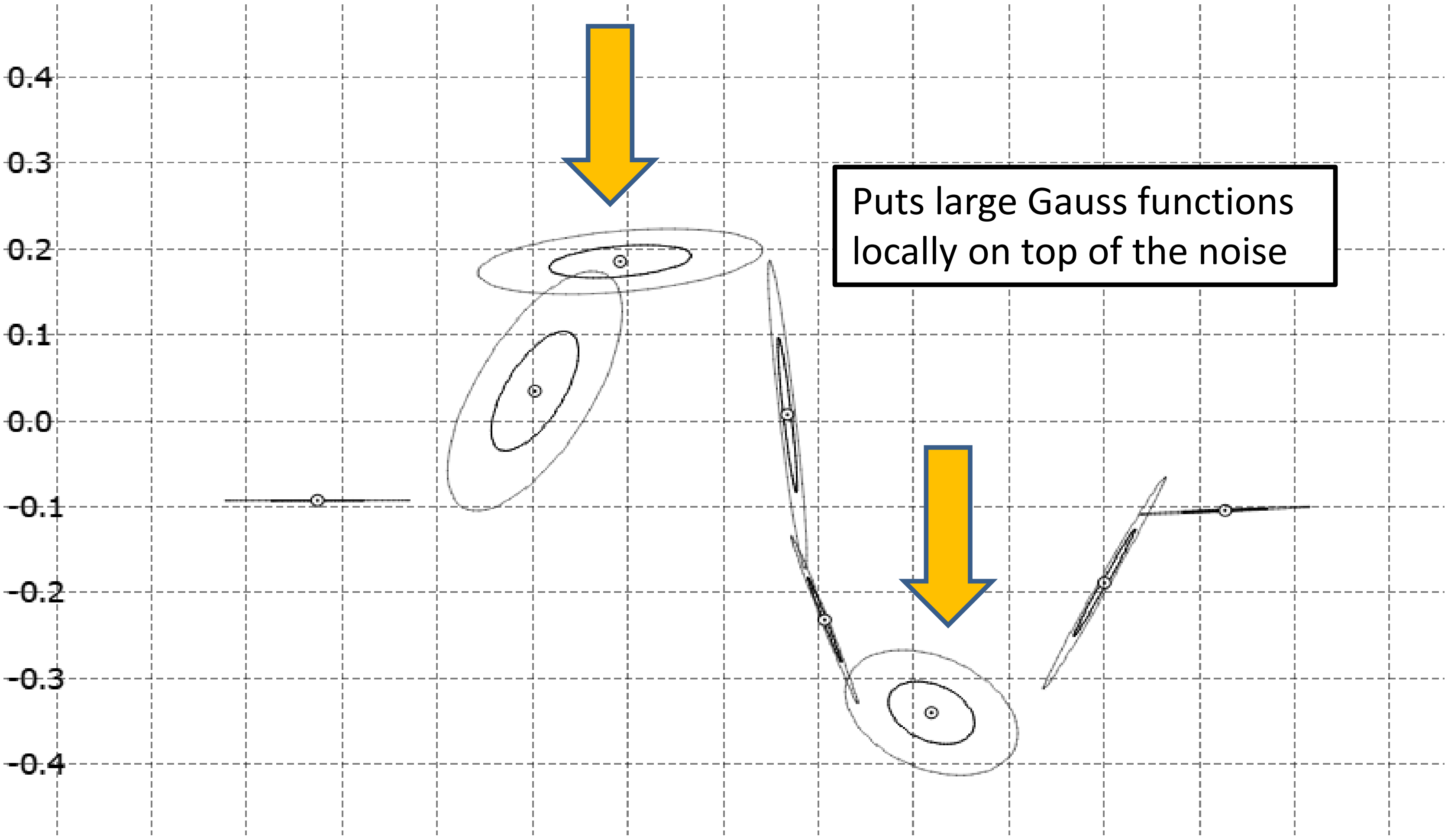


Regression: noise

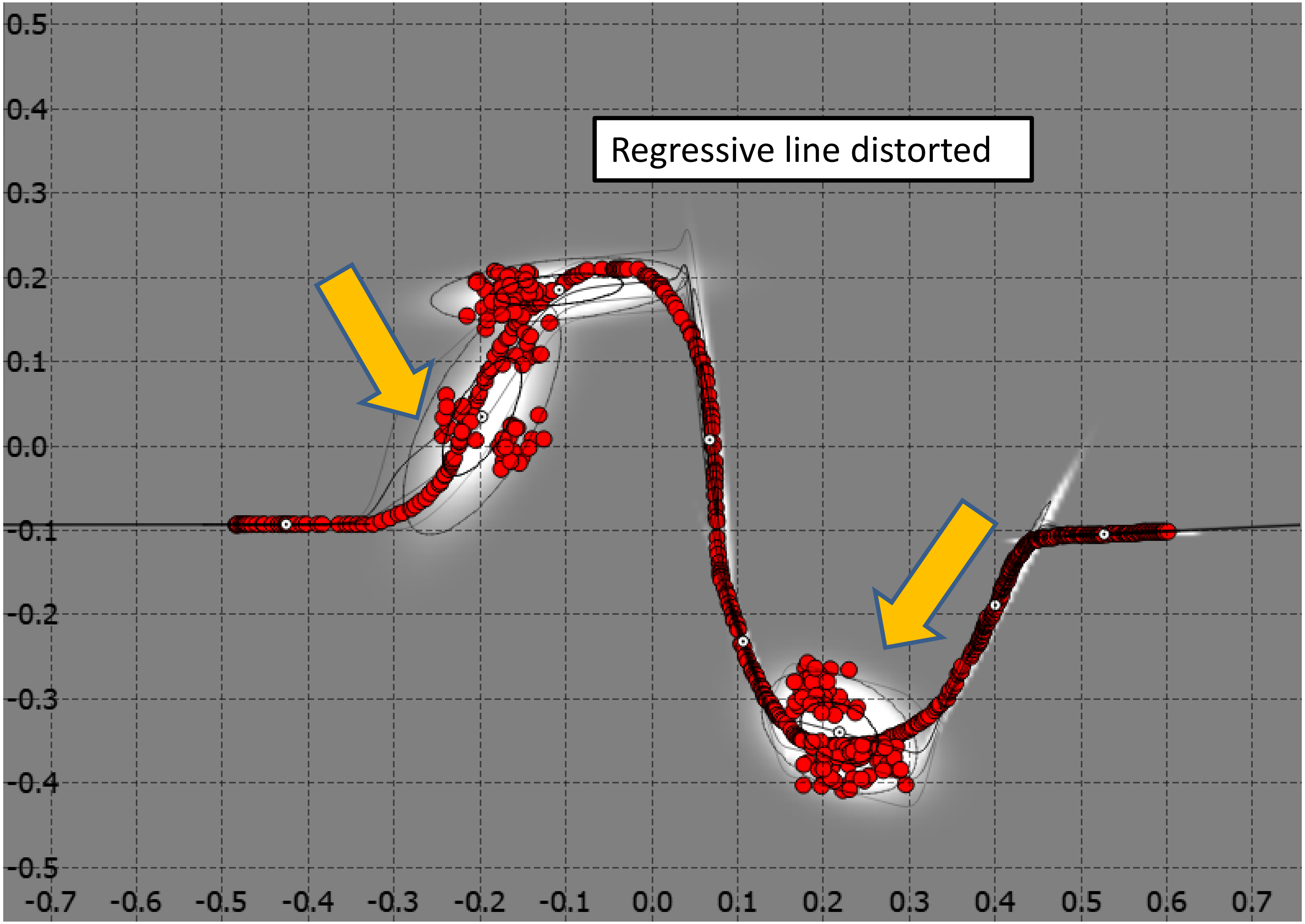


How would GMR handle this noise?

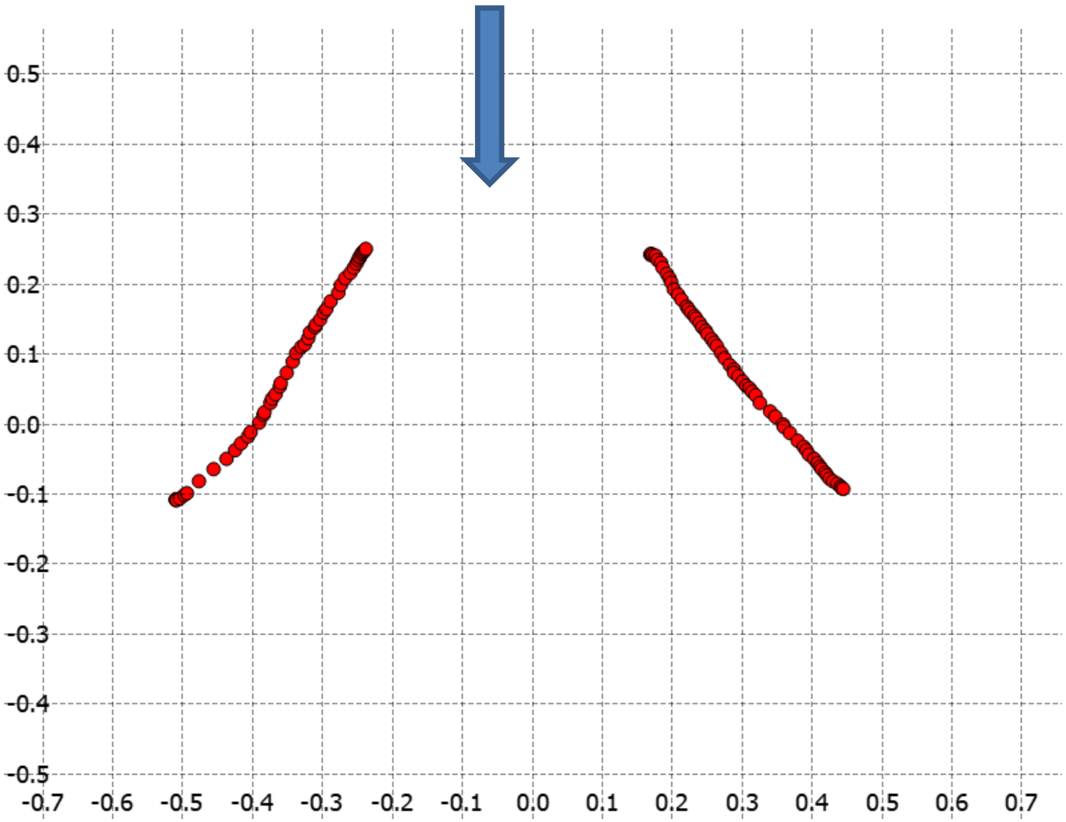
Regression: noise



Regression: noise



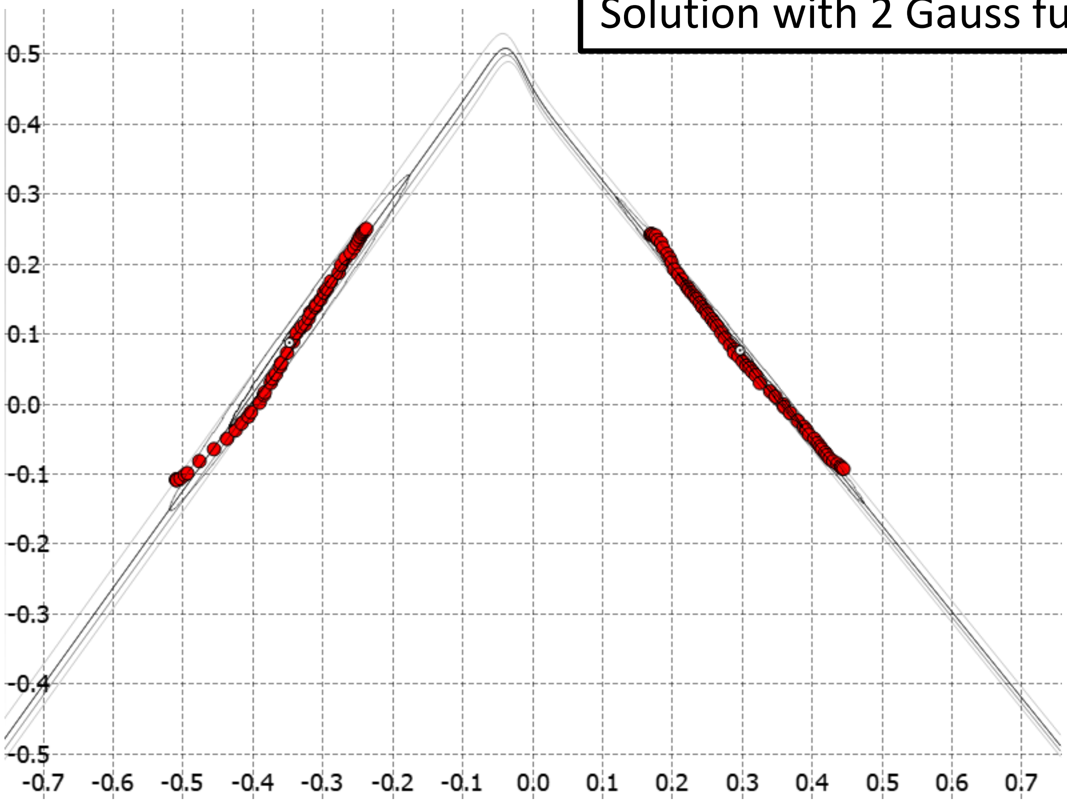
Regression: interpolation



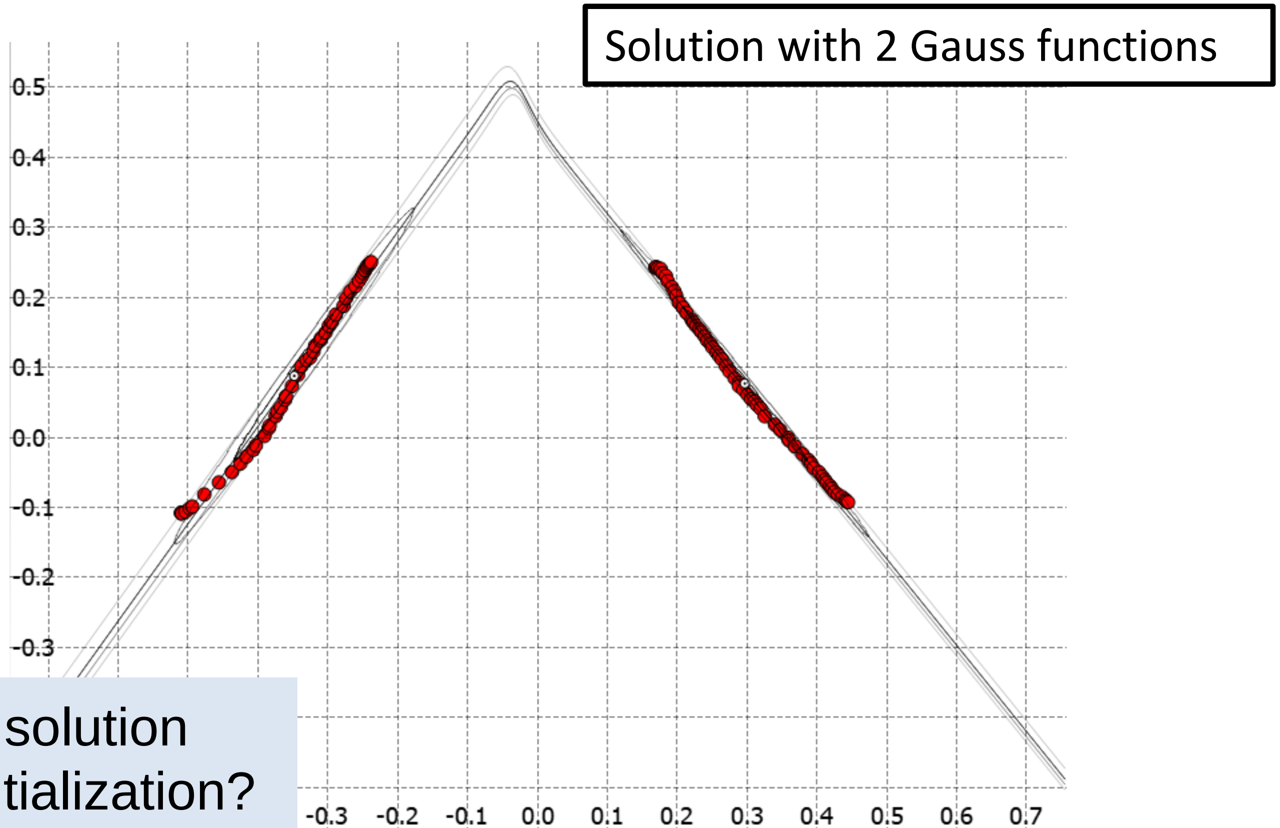
How would GMR handle missing data?

Regression: interpolation

Solution with 2 Gauss functions



GMR interpolates correctly following the trend with a small curvature at the junction

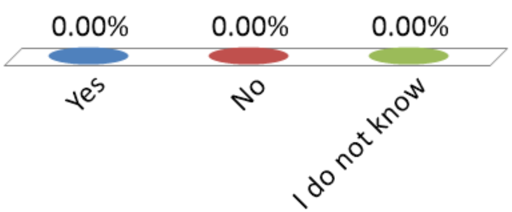


Would the solution depend on initialization?

A. Yes

B. No

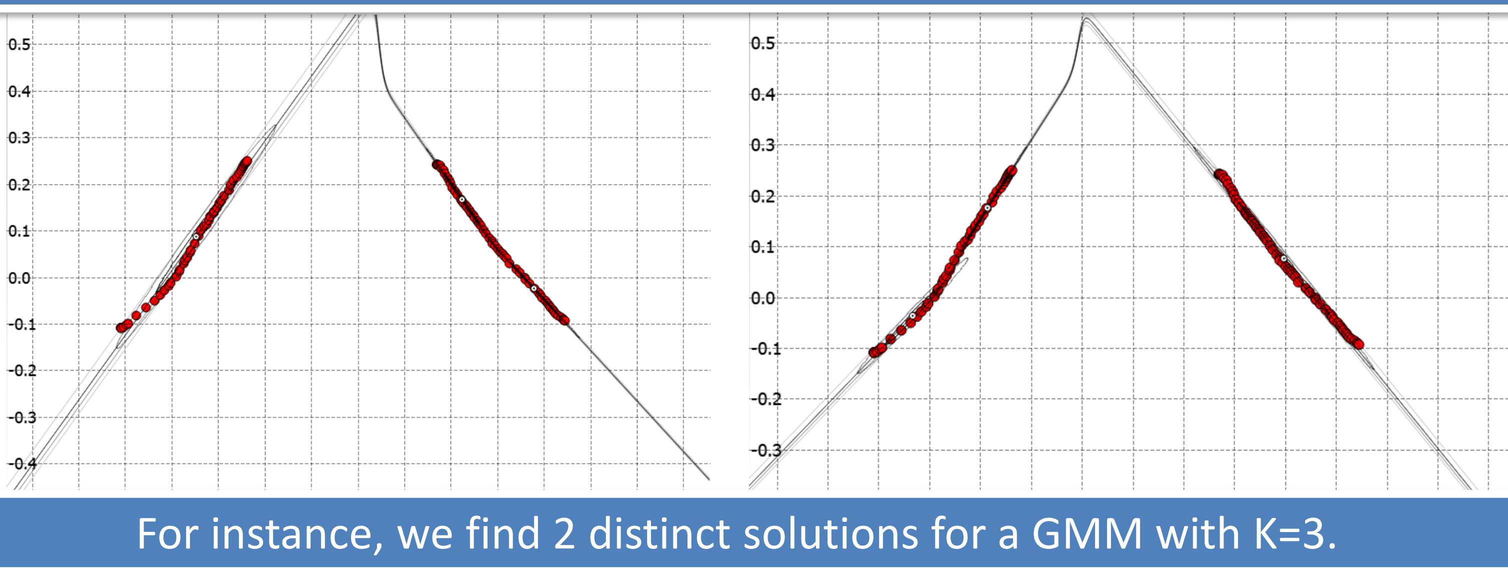
C. I do not know



Regression: interpolation

The solution is identical across runs as there is no other local optimum.

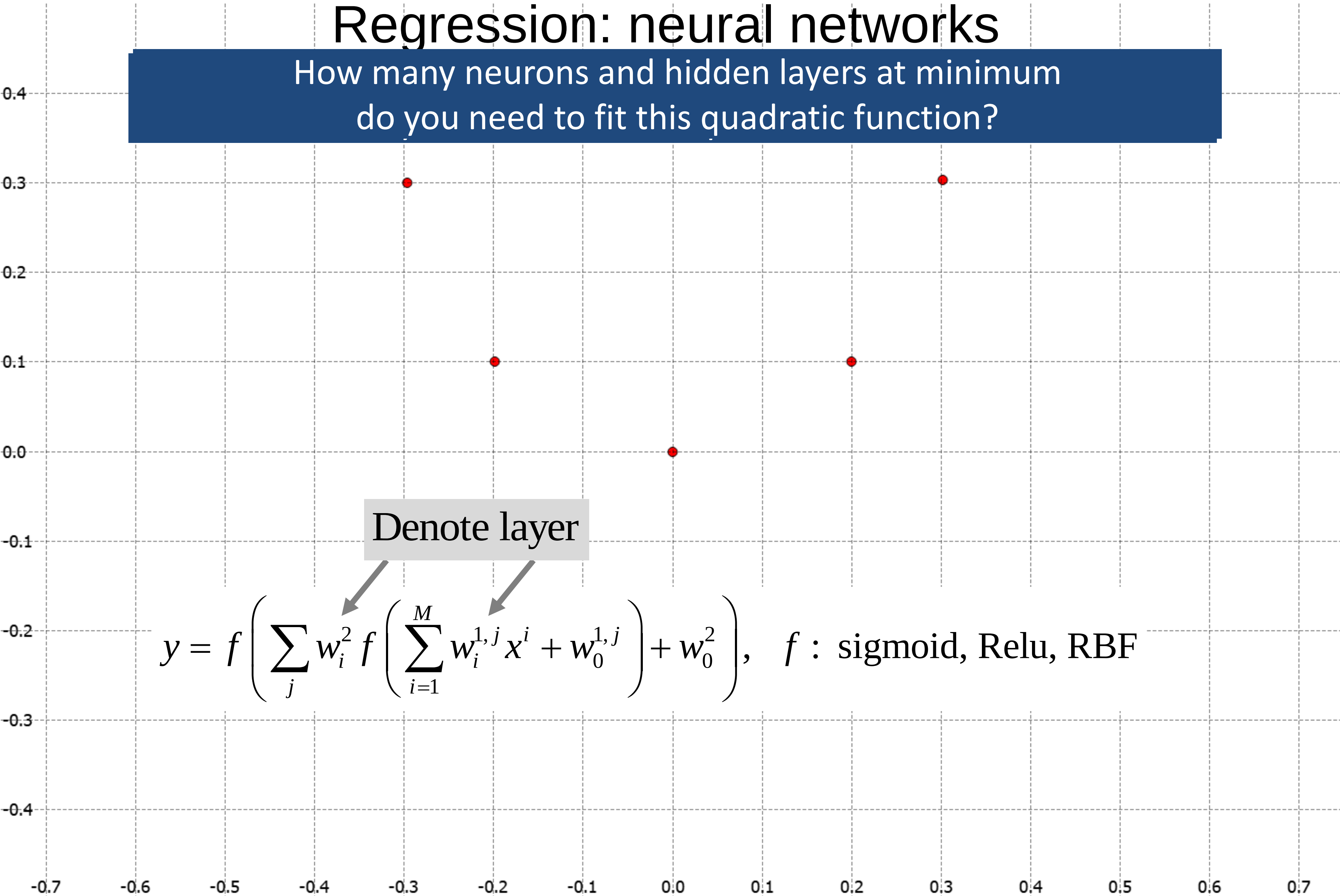
But in principle the solution is not unique and there are often many solutions, each of which corresponds to a local optima on the likelihood.



For instance, we find 2 distinct solutions for a GMM with K=3.

Regression: neural networks

How many neurons and hidden layers at minimum do you need to fit this quadratic function?

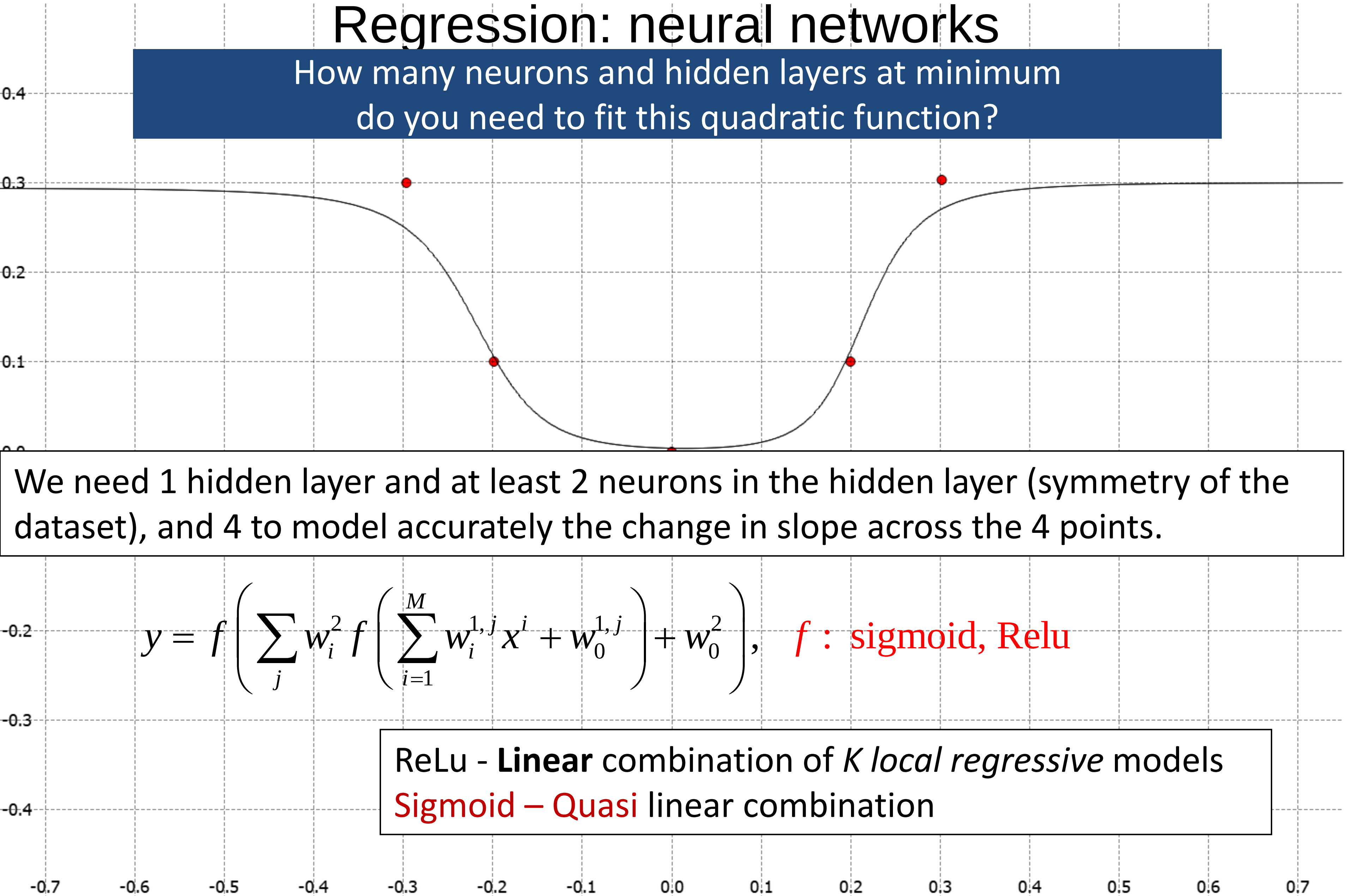


Denote layer

$$y = f\left(\sum_j w_i^2 f\left(\sum_{i=1}^{M} w_i^{1,j} x^i + w_0^{1,j}\right) + w_0^2\right), \quad f: \text{ sigmoid, Relu, RBF}$$

Regression: neural networks

How many neurons and hidden layers at minimum do you need to fit this quadratic function?



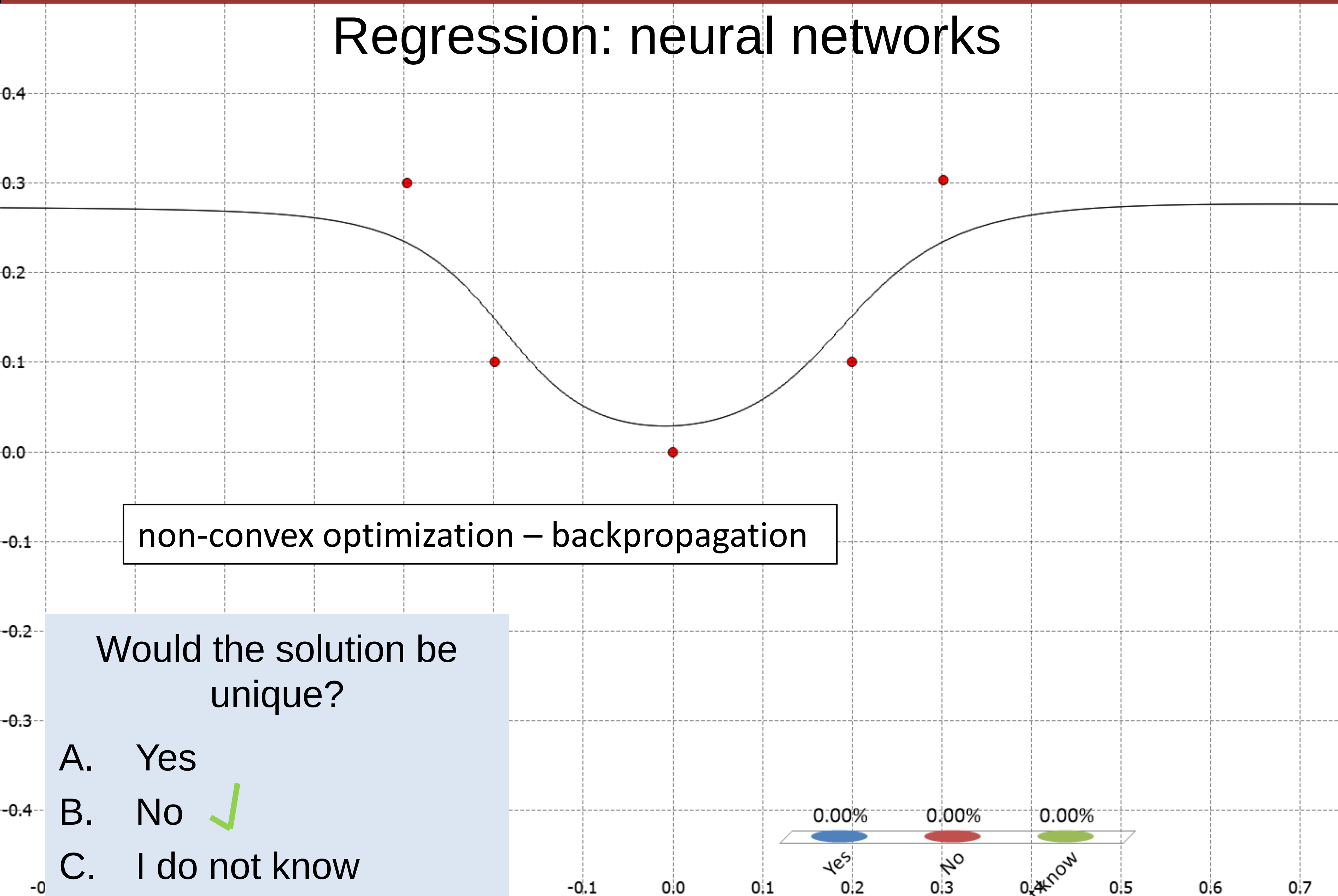
We need 1 hidden layer and at least 2 neurons in the hidden layer (symmetry of the dataset), and 4 to model accurately the change in slope across the 4 points.

$$y = f\left(\sum_j w_i^2 f\left(\sum_{i=1}^{M} w_i^{1,j} x^i + w_0^{1,j}\right) + w_0^2\right), \quad f:\ \text{sigmoid, Relu}$$

ReLu - **Linear** combination of *K local regressive* models

Sigmoid – Quasi linear combination

Regression: neural networks

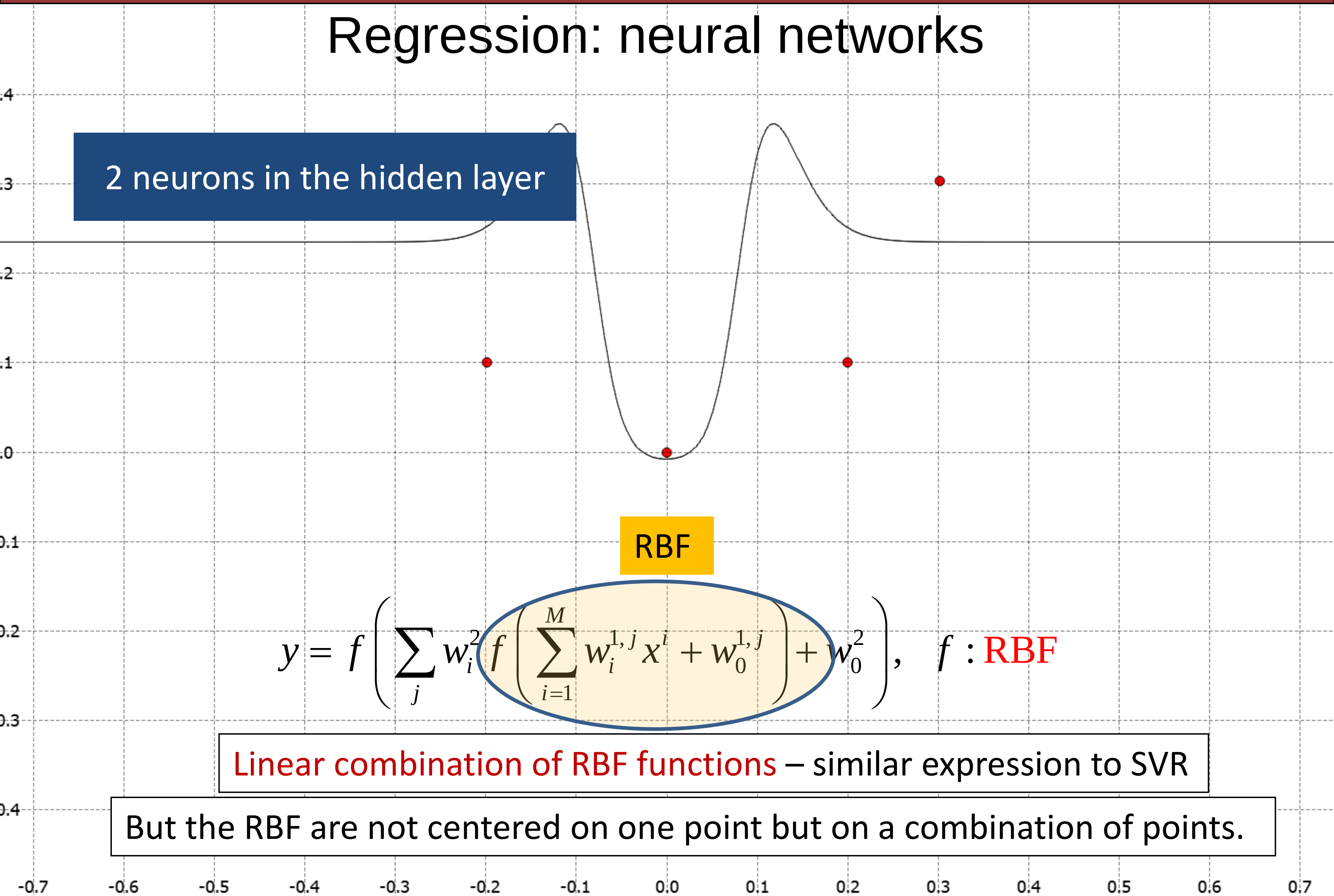


non-convex optimization – backpropagation

Would the solution be unique?

A. Yes
B. No ✓
C. I do not know

Regression: neural networks



2 neurons in the hidden layer

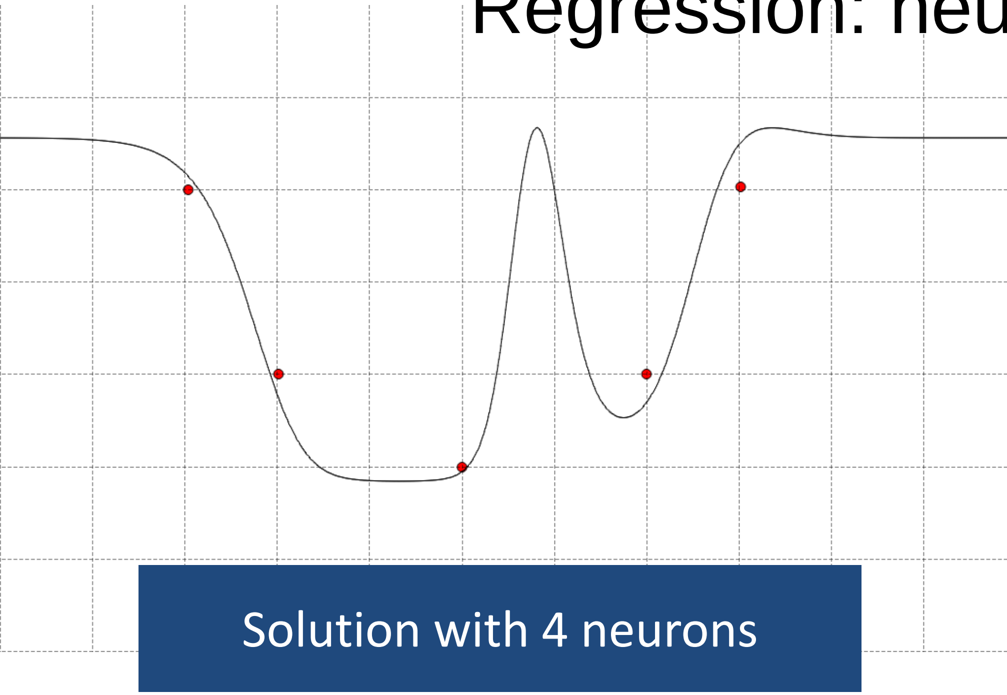
RBF

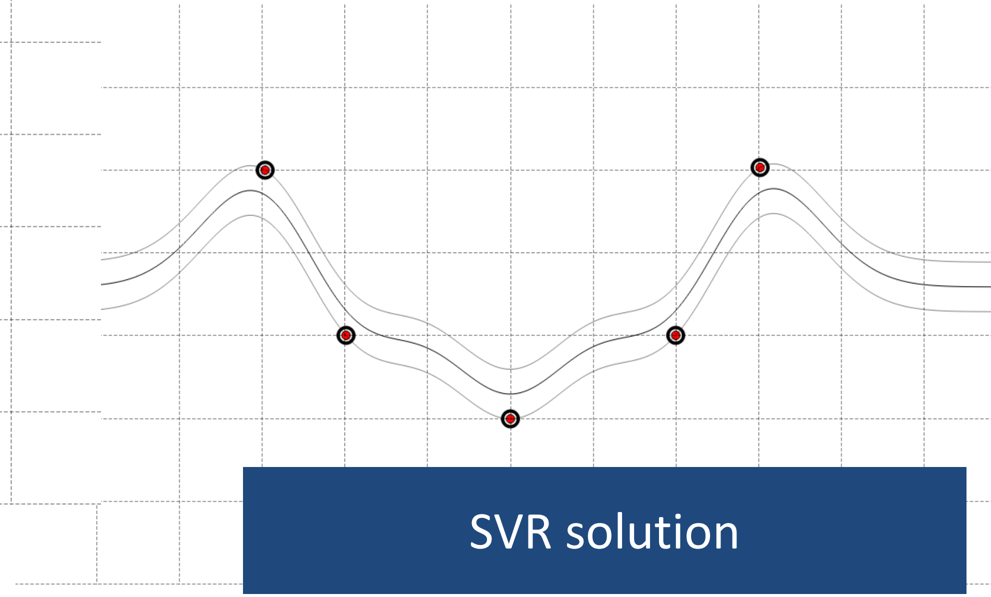
$$y = f\left(\sum_j w_i^2 f\left(\sum_{i=1}^{M} w_i^{1,j} x^i + w_0^{1,j}\right) + w_0^2\right), \quad f : \mathrm{RBF}$$

Linear combination of RBF functions – similar expression to SVR

But the RBF are not centered on one point but on a combination of points.

Regression: neural networks





$$y = f\left(\sum_{j} w_i^2 f\left(\sum_{i=1}^{M} w_i^{1,j} x^i + w_0^{1,j}\right) + w_0^2\right), \quad f : \text{RBF}$$

RBF function – similar expression to SVR

But non-convex optimization – backpropagation, in contrast to SVR