Introduction to Machine Learning. Lec.5 Support Vector Regression

Aidos Sarsembayev, IITU, 2018

Regression is…

- a technique for determining the statistical **relationship between** two or more variables where a change in a dependent variable is associated with, and depends on, a change in one or more independent variables.

http://www.businessdictionary.com/definition/regression.html

Types of regression models

- Simple Linear Regression
- Multiple Linear Regression
- Polynomial Regression
- **Support Vector Regression (SVR)**
- Decision Tree Regression
- Random Forest Regression

SLR. Formula

Constant

Coefficient

$$y = b_0 + b_1 * x_1$$

Dependent variable (DV)

Independent variable (IV)

MLR. Formula

$$y = b_0 + b_1*x_1 + b_2*x_2 + b_3*x_3 + \ldots + b_n*x_n$$

PR. Formula

$$y = b_0 + b_1{*}x_1 + b_2{*}x_1^2 + b_3{*}x_1^3 + \dots + b_n{*}x_1^n$$

SVR

- SVR is a type of Support Vector Machine, that supports **linear** and **nonlinear** regression

SVR

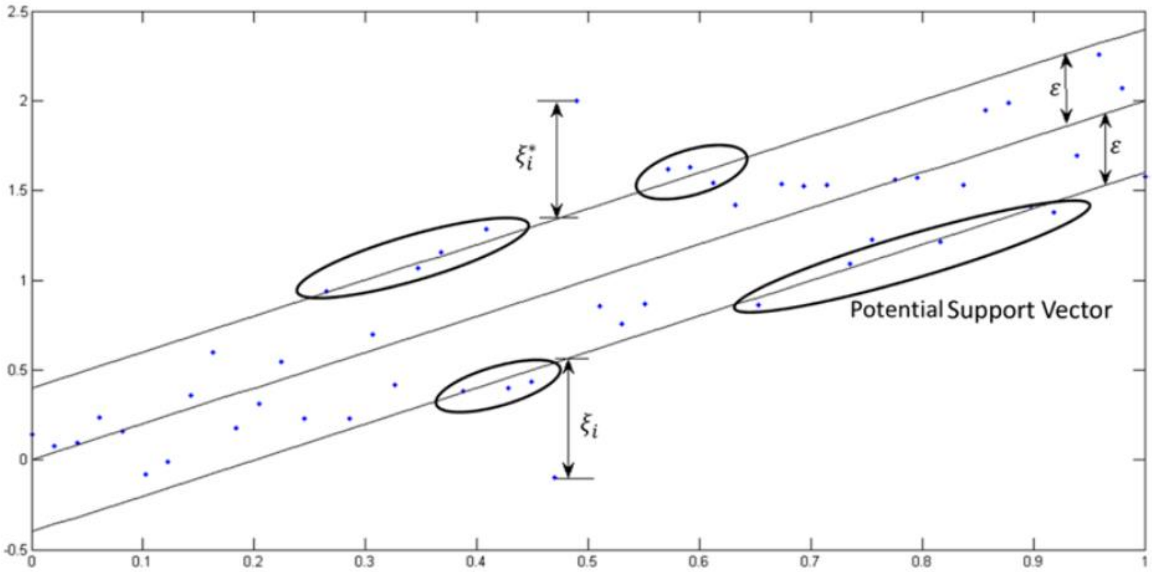
- SVR is a type of Support Vector Machine, that supports **linear** and **nonlinear** regression

SVR

- SVR is a type of Support Vector Machine, that supports **linear** and **nonlinear** regression
- As in linear regression models (i.g. SLR), SVR has kind of a best fitting line. But other than this, it has parallels (supporting vectors).

SVR

- As in linear regression models (i.g. SLR), SVR has kind of a best fitting line. But other than this, it has parallels (supporting vectors).



SVR

- SVR is a type of Support Vector Machine, that supports **linear** and **nonlinear** regression
- As in LR models (i.g. SLR), SVR has kind of a best fitting line. But other than this, it has parallels (supporting vectors).
- SVR has a different goal compared to LR. In LR we are trying to minimize the error between the prediction and data. In SVR our goal is to make sure that errors **do not exceed the threshold**

SVR

- As in LR models (i.g. SLR), SVR has kind of a best fitting line. But other than this, it has parallels (supporting vectors).
- SVR has a different goal compared to LR. In LR we are trying to minimize the error between the prediction and data. In SVR our goal is to make sure that errors **do not exceed the threshold**
- This threshold is controlled by hyper parameter Epsilon (ε, *also called as width of a street*)

SVR. Regression vs. Classification

- The regression problem is a generalization of the classification problem, in which the model returns a continuous-valued output, as opposed to an output from a finite set. In other words, a regression model estimates a continuous-valued multivariate function.

SVM

- SVMs solve binary classification problems by formulating them as convex optimization problems
- The optimization problem entails finding the maximum margin separating the hyperplane, while correctly classifying as many training points as possible
- SVMs represent this optimal hyperplane with support vectors

SVM -> SVR

- SVM generalization to SVR is accomplished by introducing an ε-insensitive region around the function, called the ε-tube
- This tube reformulates the optimization problem to find the tube that best approximates the continuous-valued function, while balancing model complexity and prediction error

SVM -> SVR

More specifically, SVR is formulated

- as an optimization problem by first defining a convex ε -insensitive loss function to be minimized
- and finding the flattest tube that contains most of the training instances.

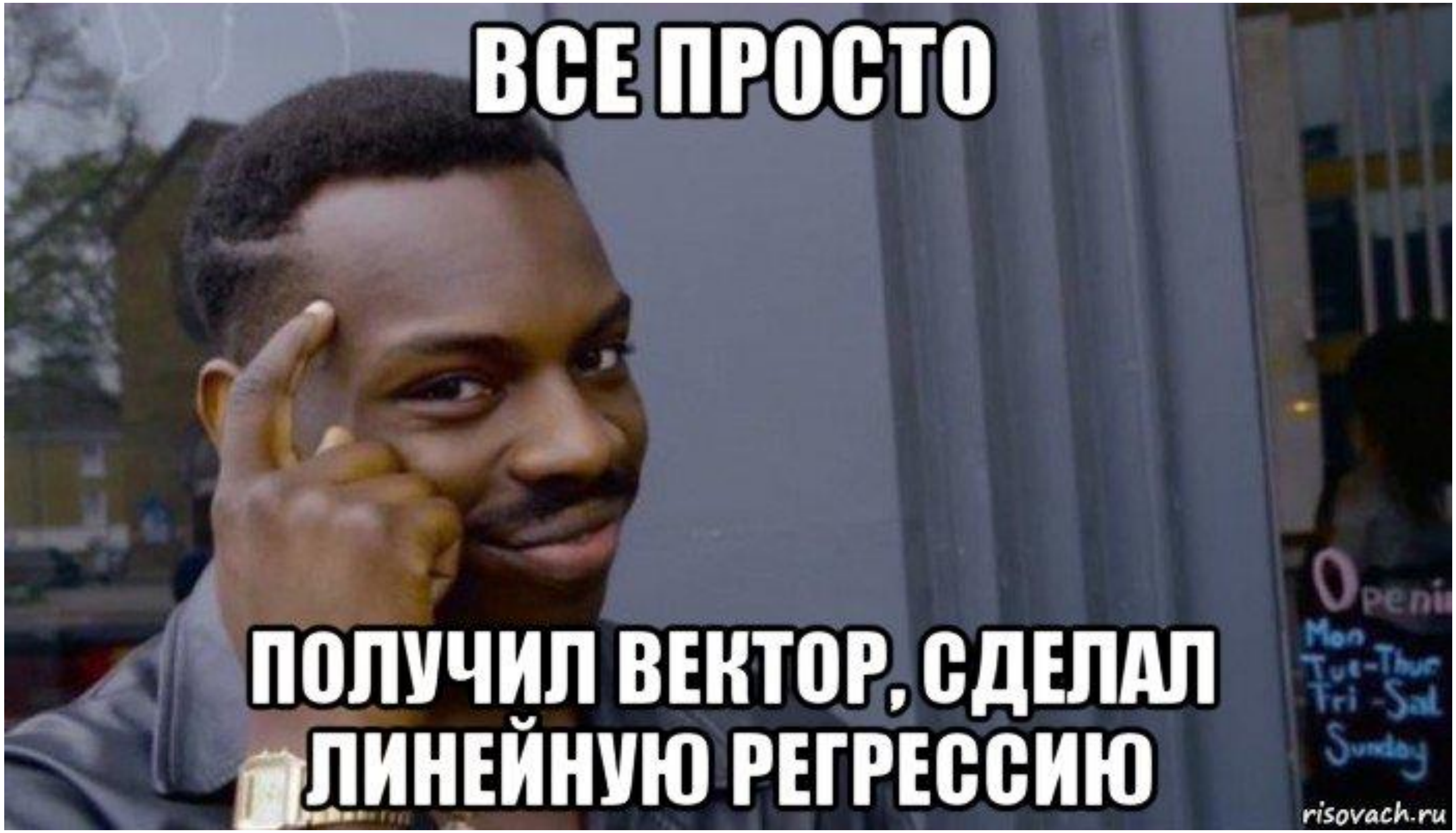
What SVR does?

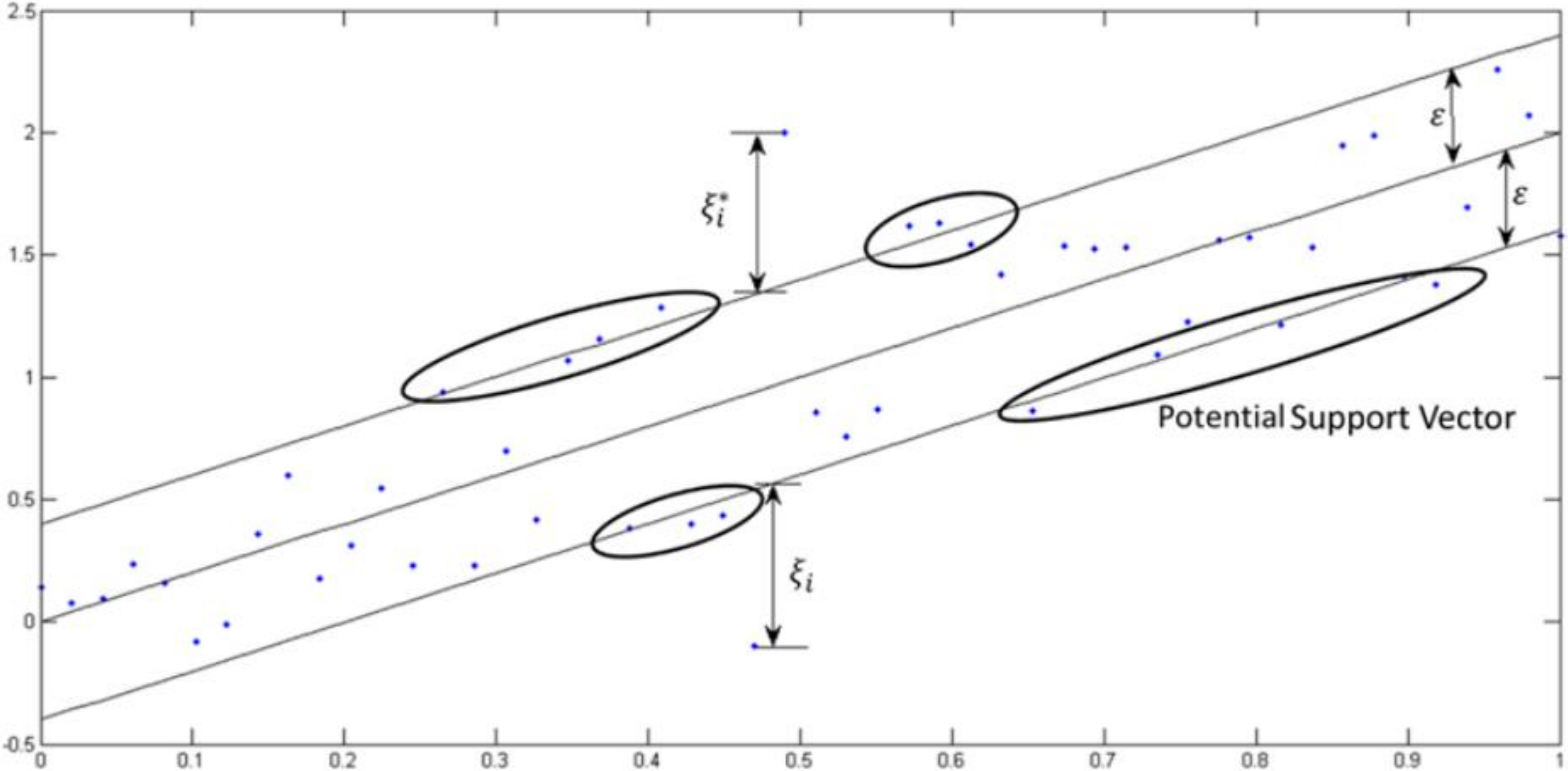
- SVR performs linear regression in a higher dimensional space
- We can think of SVR as if each data point in the training set represents it's own dimension.
- When you evaluate your kernel between a test point and a point in the training set the resulting value gives you the coordinate of your test point in that dimension

What SVR does?

- The vector we get when we evaluate the test point for all points in the training set, $\vec{k}$ is the representation of the test point in the higher dimensional space
- Once you have that vector you then use it to perform a linear regression

What SVR does?





ξ_i^*
ε
ε
Potential Support Vector
ξ_i

General rule

- It requires a training set : $\tau = \{\overrightarrow{X}, \overrightarrow{Y}\}$ which covers the domain of interest and is accompanied by solutions on that domain
- The work of the SVM is to approximate the function we used to generate the training set

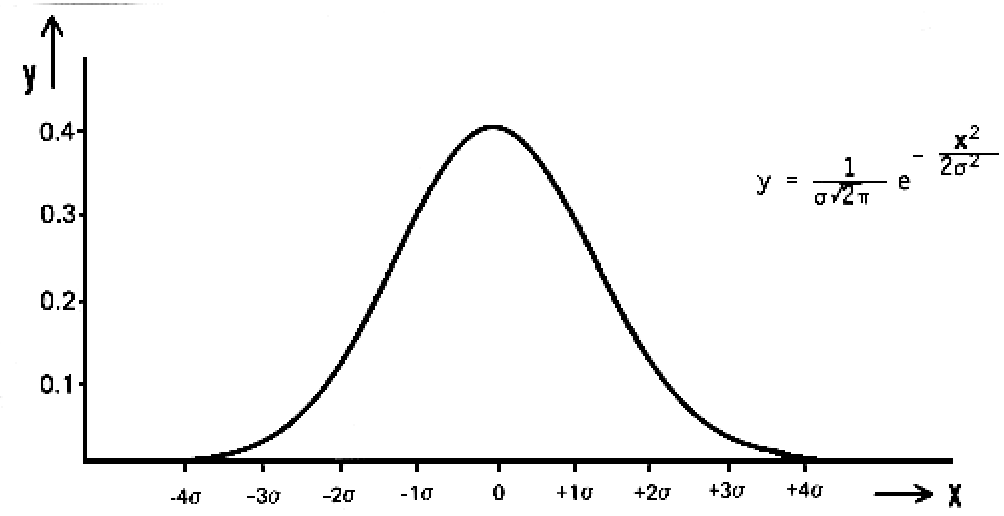
$$F(X) = \overrightarrow{Y}$$

The algorithm of SVR. Step 1

1. Collect a training set $\tau = \{\overrightarrow{X}, \vec{Y}\}$
2. Choose a kernel and it's parameters as well as any regularization needed
3. Form the correlation matrix, $\vec{K}$
4. Train your machine, exactly or approximately, to get contraction coefficients $\vec{\alpha} = \{\alpha_i\}$
5. Use those coefficients, create your estimator $f(\overrightarrow{X}, \vec{\alpha}, x^*) = y^*$

The algorithm of SVR. Step 2

1. Choose a kernel: Gaussian
2. Choose regularization: Noise



Correlation matrix

$$K_{i,j} = \exp\left(\sum_k \theta_k \left| x_k^i - x_k^j \right|^2 \right) + \epsilon \delta_{i,j}$$

- At this time we have a main part of the algorithm

$$\bar{K}\vec{\alpha} = \vec{y}$$

- $\vec{y}$ is the vector of values corresponding the your training set
- $\bar{K}$ is you correlation matrix
- $\vec{\alpha}$ is a set of unknowns we need to solve for.

$$\vec{\alpha} = \bar{K}^{-1}\vec{y}$$

- Once $\vec{\alpha}$ parameters are known – form the estimator
- We use the coefficients we found during the optimization step and the kernel we started with
- To estimate the value y^* for a test point, $\overrightarrow{x^*}$ - compute the correlation vector $\vec{k}$
- $y^* = \vec{\alpha} \cdot \vec{k}$

- Once $\vec{\alpha}$ parameters are known – form the estimator
- We use the coefficients we found during the optimization step and the kernel we started with
- To estimate the value y^* for a test point, $\overrightarrow{x^*}$ - compute the correlation vector $\vec{k}$
- $y^* = \vec{\alpha} \cdot \vec{k}$
- Finally, we compute the element of k

$$k_i = \exp\left(\sum_k \theta_k |x_k^i - x_k^\star|^2 \right)$$