Analysis and Design of Deep Neural Networks

Chapter 2

Complexity Indices and Data analysis

Fall 2023

2 Complexity Indices and Data analysis

2.1. Complexity Indices

2.1.1 Separation index and methods

2.1.2 Smoothness index and methods

2.1.3 Linear Density Index index and methods

1.1.3 Data Analysis

2.2.1 Dataset evaluation and Scoring

2.2.2 Supervised Feature Selection

2.2.4 Data Connectivity Matrix (Smi Table)

2.2.5 Data Clustering

2.2.3 Unsupervised Feature Selection

2.1 Complexity Indices

Supervised Indices: 1- Separation Index (Classification Prob.), 2-Smoothness Index(Regression Prob.)

2.1.1 Separation index(SI)

- First order SI
- High order SI
- High order soft SI
- Center Based SI
- Cross SI
- Anti SI
- Self Supervised SI

2.1.2 Smoothness index(SmI)

- First order SmI
- High order SmI
- High order soft SmI
- Cross SmI
- Global SmI
- Data Connectivity SmI

Complexity measures

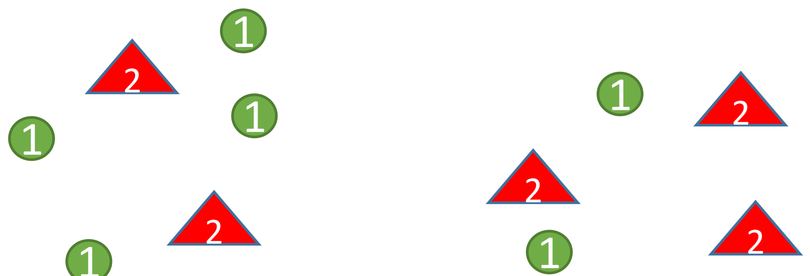
Complexity measures	Overall evaluating approach
√ Feature-based	Discovering informative features by evaluating each feature independently (Orriols-Puig et al., 2010; Cummins, 2013))
√ Linearity separation	Evaluating the linearly separation of different classes (Bottou & Lin, 2007)
√ Neighborhood	Evaluating the shape of the decision boundary to distinguish different classes overlap (Lorena et al., 2012; Leyva et al., 2014)
√ Network	Evaluating the data dataset structure and relationships by representing it as a graph (Garcia et al., 2015)
√ Dimensionality	Evaluating the sparsity of the data and the average number of features at each dimension (Lorena et al., 2012; Basu & Ho, 2006)
√ Class imbalanced	Evaluating the proportion of dataset number between different classes (Lorena et al., 2012)

Table 1. Some complexity measures and their evaluating approaches in a classification problem

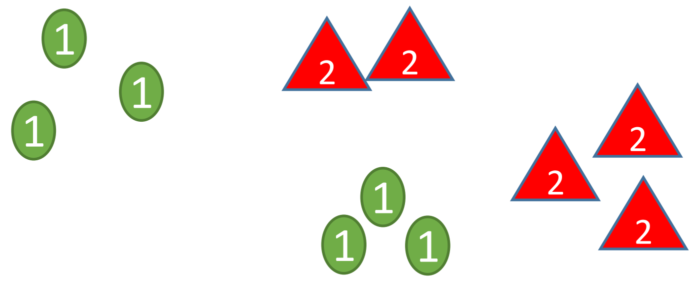
Two Supervised Complexity measures

1. **A separation measure** (in classification problems)

   It shows that how much input data points separate the labels from each others.



(Less separated)

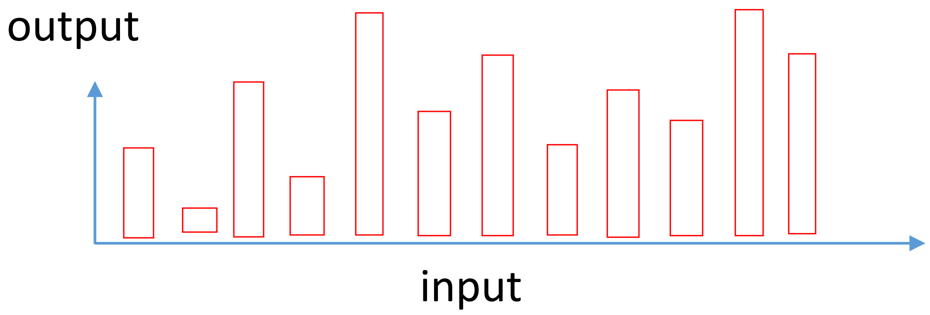


(More separated)

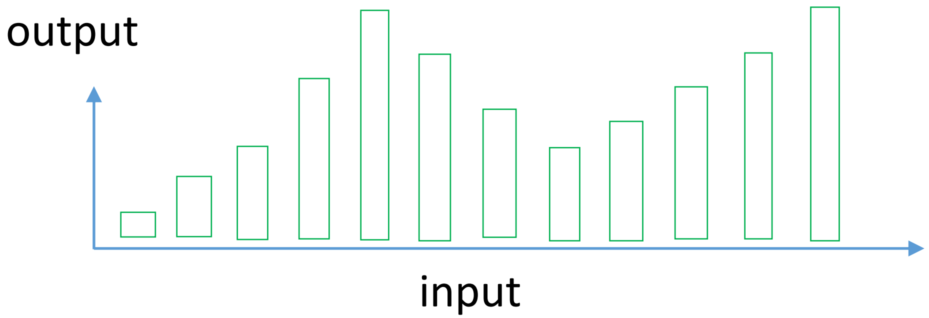
2. **An smoothness measure** (in regression problems)

   It shows that how much input data points make the output targets smooth

(Less Smooth)



(More Smooth)



Separation index (SI)

“SI” measures that how much input data points separate class labels from each others.

2.1. Separation index (SI)

1. First order SI

$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m} \forall i: \boldsymbol{x}^i \epsilon R^{n\times 1}$ $l_i \epsilon \{1,2,\dots,n_C\}$ n_C:number of classes

*it is assumed that “Data” is a measured sample from a domain with high enough diversity.

*$\boldsymbol{x}_i$ may have any format (video, image, time series, etc.) ; however, to compute SI, it must be reshaped as a vector.

$$\mathrm{SI}(Data) = \frac{1}{m}\sum_{i=1}^{m} \delta(l_i, l_{i^*})$$

$$i^* = \underset{\forall q \neq i}{\arg\, min} \left\| \boldsymbol{x}_i - \boldsymbol{x}_q \right\| \qquad \delta(l_i, l_{i^*}) = \begin{cases} 1 & if\ l_i = l_{i^*} \\ 0 & else \end{cases}$$

kronecker delta

*$\|\cdot\|$ denotes Euclidian distance (L_2 norm) but it may be another distance definition such as L_p norm:

$$\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|_{L_p} = \sqrt[p]{\sum_{k=1}^{n} \left| \boldsymbol{x}_i(k) - \boldsymbol{x}_j(k) \right|^p}$$

** It is assumed that the input data is normalized at each dimension just before computing separation index.

Some notes

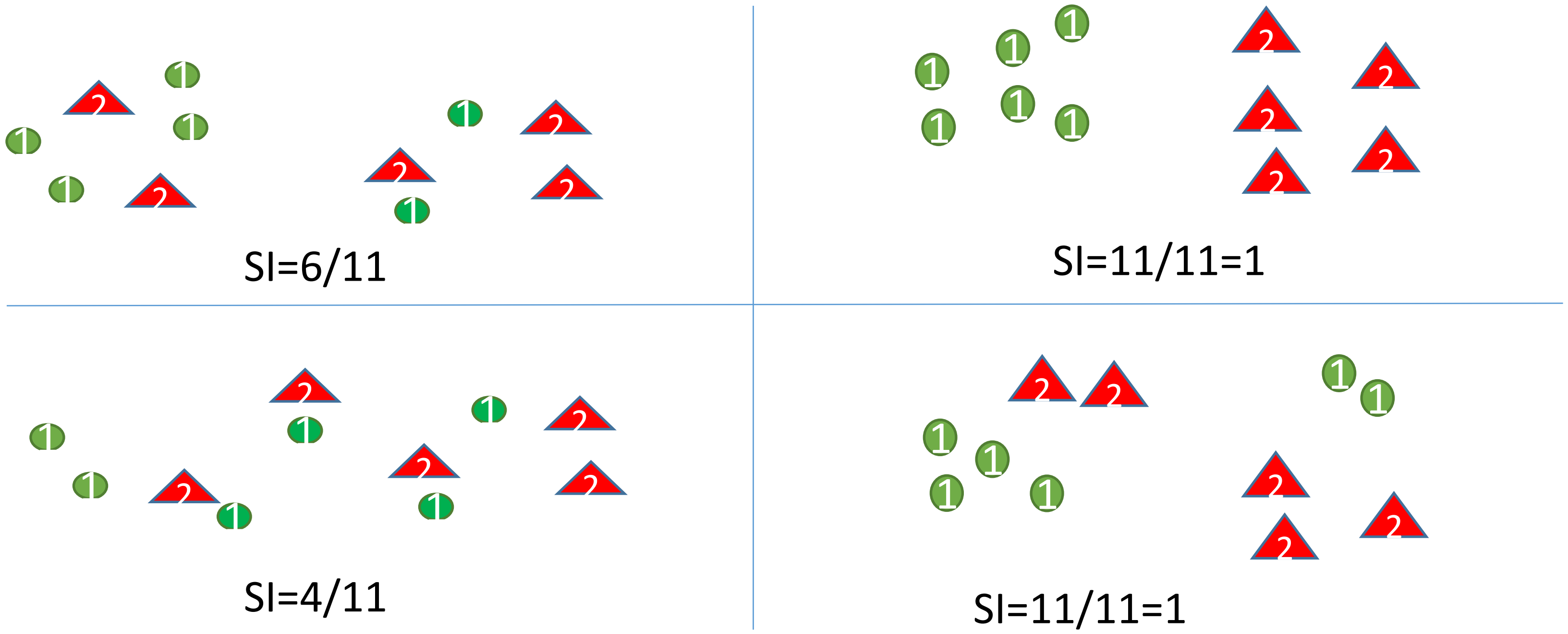
1. “SI” is a normalized index between zero and one: $\mathrm{SmI} \in [0,1]$
2. $SI \to 1$ $(Sepration\ is\ maximmum)$ and $SI \to 0 (Sepration\ is\ minimmum)$
3. “SI” counts (average of) all data points whose nearest neighbors have the same label
4. “SI” is equal to the accuracy of the nearest neighbor classifier as a non-parametric model. Hence, SI is an informative index having strong correlation with the best accuracy one can access by a model without filter process.
5. SI does not change against shift and scales of data points.

$$\forall \beta \neq 0, \forall \alpha \neq 0, \forall x_0, \forall l_0 \qquad SI\left(\left\{\left(\boldsymbol{x}^i, l^i\right)\right\}_{i=1}^{m}\right) = SI\left(\left\{\left(\beta x_i + \boldsymbol{x}_0, \alpha l_i + l_0\right)\right\}_{i=1}^{m}\right)$$

6. Separatin index of $the\ target\ labels\ with\ themselves\ is\ maximum$: $SI\left(\left\{\left(l_i, l_i\right)\right\}_{i=1}^{m}\right)$=1;

it means that how input data become more similar to labels the separation index will increase.

Two dimensional examples (binary classification)



Some notes

- To have a high SI, It is enough that examples of each class become near and near together in some regions
- The number of regions is not important but each region must have at least two members.
- The shape of each region is not important.

The distance matrix

- To achieve SI, matrix distance of all data points must be computed (to get nearest neighbor for each data point)

$$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m} \qquad \boldsymbol{x}^i \epsilon R^{n\times 1}$$

Distance matrix: $D = [d_{ij}]$ $d_{ij} = \left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2$

Steps

1- Provide data Matrix:$\mathrm{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_m]^T$, $\mathrm{X}\epsilon R^{m\times n}$

2- M= $\mathrm{XX^T}$, $\mathrm{M}\epsilon R^{m\times m}$

3- $\mathrm{d} = \mathrm{diag(M)}$, $\mathrm{d}\epsilon R^{m\times 1}$

4- W=[d,d,…,d], $\mathrm{W}\epsilon R^{m\times m}$

5- Distance matrix is computed as follows:

$$\mathrm{D} = \mathrm{W} + \mathrm{W^T} - 2\mathrm{M}$$

Separation index of Each data point

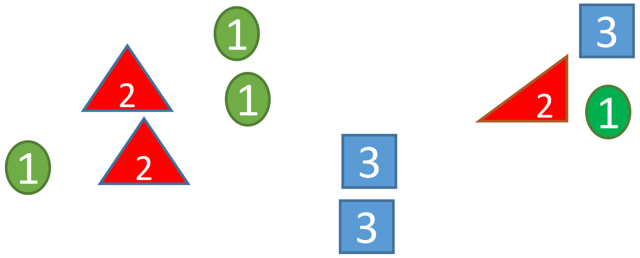
$* \; SI(Data) = \frac{1}{m} \sum_i SI_i^{data} \; , \; SI_i^{data} = \delta(l_i, l_{i^*})$

SI definition with data distribution: $SI(x \; with \; pdf{:} \; p(x)) = Exp_{p(x)}\left(SI^{data}(x)\right)$

Challenge: to compute $SI^{data}(x)$ it is required to have x^*as the nearest neighbor of x.

❖ For a sample of data with high enough diversity SI can be approximated by equation *

A two dimensional illustrative example



i=1,2,….,10

$SI_i^{data} = 0, \quad i$=1,8,9,10

$SI_i^{data} = 1, \quad i = 2,3,4,5,6,7$

$SI(Data)$ =0.6

Separation index of Each Class

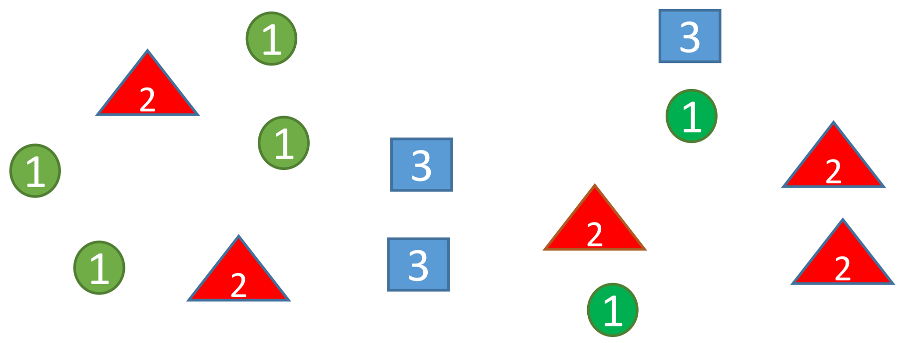
$\mathrm{SI}_c^{class}(Data) = \frac{1}{m_c}\sum_i \delta(l_i, c)\delta(l_i, l_{i^*})$ $\quad$ $c=1,2,..,n_C$

$m_c=\sum_i \delta(l_i, c)$ $\quad$ m_c: number of all data points x^i which $l^i = c$

Relation between "total SI" and "SI of classes"

$$SI(Data) = \frac{1}{m}\sum_{c=1}^{n_C} m_c SI_c^{class}(Data) \qquad \sum_{c=1}^{n_C} m_c = m$$

A two dimensional illustrative example



$n_C = 3, \quad c = 1,2,3$

$SI_1^{class}(Data) = \frac{4}{6}$

$SI_2^{class}(Data) = 2/5$

$SI_3^{class}(Data) = 2/3$

SI=(4+2+2)/(6+5+3)=8/14

* For when for each class c: $m_c = \frac{m}{n_C}$ and a sufficient high number of data points are distributed with a *uniformly distributed random* variable then it is expected that $SI \to 1/n_C$

2. High order SI

$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m}$ $\forall i$: $\boldsymbol{x}_i \epsilon R^{n\times 1}$ $l_i \epsilon \{1,2,\dots,n_C\}$ n_C:number of classes

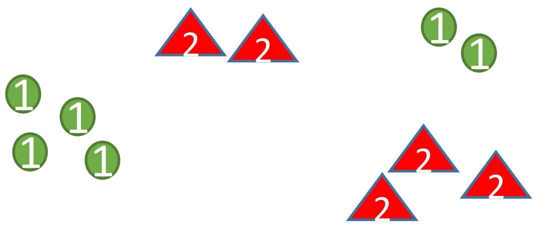
$$\mathrm{SI^r}(Data) = \frac{1}{m}\sum_{i=1}^{m}\prod_{j=1}^{r}\delta\left(l_i, l_{i_j^*}\right)$$ r: the order of "SI"

$$i_j^* = \underset{\forall q \neq i, i_1^*, \cdots, i_{j-1}^*}{\arg\min}\left\|\boldsymbol{x}_i - \boldsymbol{x}_q\right\|$$ $\mathrm{SI^r} \in [0,1]$

- "$\mathrm{SI^r}$" counts (average of) all data points whose all "r" nearest neighbors have the same label
- $\mathrm{SI^r}$ considers more restricted condition of separation than $\mathrm{SI^j}$ $(j < r)$.
- For each "Data" we have: $\mathrm{SI^r} \le \mathrm{SI^{r-1}} \le \cdots \le \mathrm{SI^1}$ $\mathrm{SI^1} = \mathrm{SI}$

Two illustrative Examples

SI^1 =11/11
SI^2 =11/11
SI^3 =11/11
SI^4 =11/11



SI^1 =11/11
SI^2 =7/11
SI^3 =4/11
SI^4 =0

Some notes

1. To increase high order SI, different regions of data points with the same label should merge together and make a hyper-circle shape distribution. In a such case, we will have n_C hyper-circle shape which can separated, linearly from each other.
2. If in a classification problem, the high order SI $\mathrm{SI}^{\mathrm{r}}(r \to \infty) \to 1$, the data points of any pair of classes become more linearly separable.
3. If in a classification problem, the high order SI $\mathrm{SI}^{\mathrm{r}}(r \to \infty) \to 1$, then there is a global separation index (gsi).

3. High order soft SI

$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m} \forall i: \boldsymbol{x}_i \epsilon R^{n\times 1}$ $\quad l_i \epsilon \{1,2,\ldots,n_C\}$ $\quad n_C$:number of classes

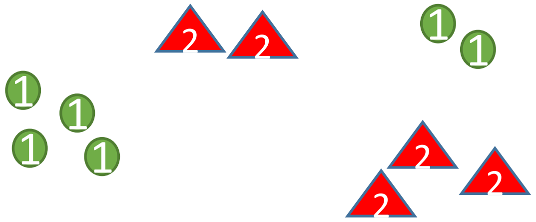
$$\mathrm{SI}_{\mathrm{soft}}^{\mathrm{r}}(Data) = \frac{1}{m\times r}\sum_{i=1}^{m}\sum_{j=1}^{r}\delta(l_i,, l_{i_j^*})$$

r: the order of SI $\quad i_j^* = \underset{\forall q\neq i, i_1^*,\cdots,i_{j-1}^*}{\arg} \; min\|\boldsymbol{x}_i - \boldsymbol{x}_q\|$ $\quad \mathrm{SI}_{\mathrm{soft}}^{\mathrm{r}} \in [0,1]$

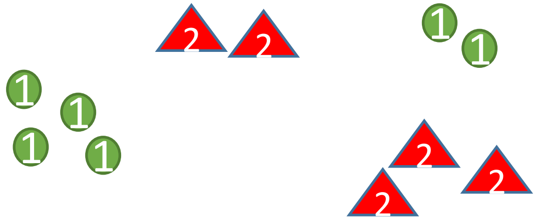
- $\mathrm{SI}_{\mathrm{soft}}^{\mathrm{r}}$ considers less restricted condition of separation than SI^{r}

$\mathrm{SI}_{\mathrm{soft}}^{\mathrm{r}} \geq \mathrm{SI}^{\mathrm{r}}$ and $\mathrm{SI}_{\mathrm{soft}}^{1} = \mathrm{SI}^{1}$

Two illustrative Examples



SI^1 =11/11
SI^2 =7/11
SI^3 =4/11
SI^4 =0



SI^1_{soft} =11/11
SI^2_{soft}=(4+3+0.5+0.5)/11=8/11
SI^3_{soft}=(4+3(2/3)+4*(1/3)/11=8.33/11
SI^4_{soft}=(4*(3/4)+2*(1/4)+2*(1/4)+3*(2/4))/11
=6.5/11

4. Center based Separation Index (CSI)

$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m} \forall i: \boldsymbol{x}^i \epsilon R^{n\times 1}$ $\quad l_i \epsilon \{1,2,\ldots,n_C\}$ $\quad n_C$:number of classes

Center of each class is the mean of all input data points having the label of that class:

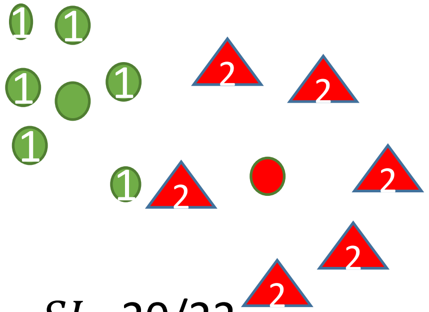
$$\boldsymbol{\mu}_c = \frac{1}{m_c}\sum_{i=1}^{m} \boldsymbol{x}_i \delta(l_i, c), \quad c = 1,2,..,n_C \qquad m_c = \sum_{i=1}^{m} \delta(l_i, c)$$

$$\mathrm{CSI}(Data) = \frac{1}{m}\sum_{i=1}^{m} \delta(l_i, c^*)$$

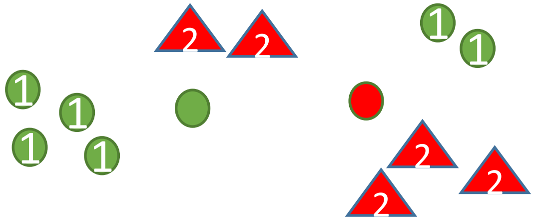
$$c^* = \underset{\forall c}{\arg min} \|\boldsymbol{x}_i - \boldsymbol{\mu}_c\|$$

- CSI is computed much faster than SI because $n_C \ll m$ and you only need to compute the distance matrix of input data points to center of classes.
- It is suggested to compute CSI instead of SI in cases that examples of each class have all exclusive features of that class and do not have any common feature with examples of other classes.

An illustrative Examples



SI =20/22
CSI=22/22



$SI = 1$
CSI=7/11

5. **Self** supervised SI (SSSI)

- $Data = \{(\boldsymbol{x}_i, ?)\}_{i=1}^{m} \forall i: \boldsymbol{x}_i \epsilon R^{n\times 1}$ *Labels are unknown*
- For each $\boldsymbol{x}_i$ we generate some augmented data points: $x_{i_h}, h \in \{1,2,..,n_i\}$
- It is assumed that each x_{i_h} inherits at least an exclusive feature of $\boldsymbol{x}_i$
- An exclusive feature of x_{i_h} is a feature that is sufficient to reveal the label of $\boldsymbol{x}_i$.

$$Data_{\text{aug}} = \{\overbrace{\{(\boldsymbol{x}_{i_h}, \boldsymbol{i})\}_{h=1}^{n_i}}^{i\text{th (self) class}}\}_{i=1}^{m}$$

$$\text{SSSI}^{\text{r}}(Data) = \text{SI}^{\text{r}}\left(Data_{\text{aug}}\right),\ n_C = m,\ m_{aug} = \sum_{i=1}^{m} n_i$$

Cross SI

$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m}$ $\quad D_{test} = \{(\breve{\boldsymbol{x}}_i, \breve{l}_i)\}_{i=1}^{m_{test}}$

$$SI_{cross}(D_{test}, Data) = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} \delta(\breve{l}_i, l_{i^{\#}})$$

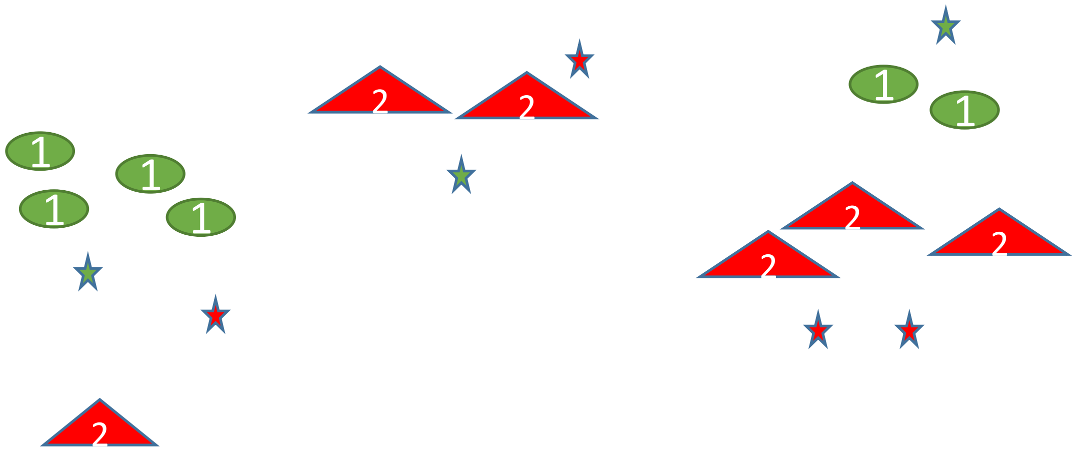
$i^{\#} = \arg\min_{\forall q} \|\breve{\boldsymbol{x}}_i - \boldsymbol{x}_q\|$

Cross SI measures the separation index of a test domain of dataset D_{test} based on the main domain of dataset $Data$.

It can be shown that:

$$SI_{cross}(D_{test}, Data) = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} SI_{cross}\left((\breve{\boldsymbol{x}}_i, \breve{l}_i), Data\right)$$

An illustrative Examples



$$SI = \frac{12}{13} \qquad Corss\ SI = \frac{5}{7}$$

Anti SI

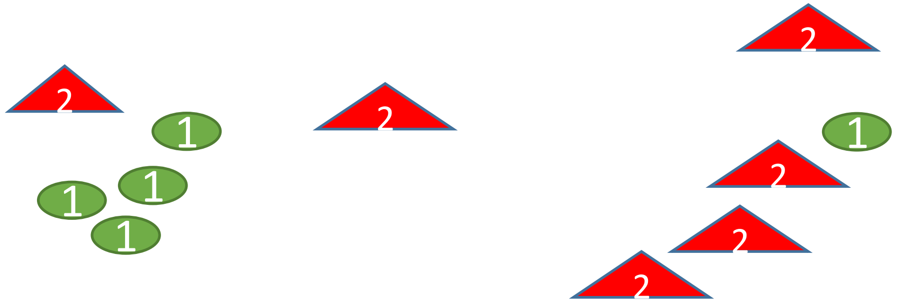
$Data = \{(\boldsymbol{x}_i, l_i)\}_{i=1}^{m}$ $\forall i$: $\boldsymbol{x}_i \epsilon R^{n\times 1}$ $l_i \epsilon \{1,2,\dots,n_C\}$ n_C:number of classes

$$\mathrm{anti\_SI}^{\mathrm{r}}(Data) = \frac{1}{m}\sum_{i=1}^{m}\prod_{j=1}^{r}\left(1-\delta\left(l_i, l_{i_j^*}\right)\right) \quad \mathrm{r}\text{: the order of "anti\_SI"}$$

$$i_j^* = \underset{\forall q\neq i, i_1^*,\cdots,i_{j-1}^*}{\arg} \ min\left\|\boldsymbol{x}_i - \boldsymbol{x}_q\right\| \qquad \mathrm{anti\_SI}^{\mathrm{r}} \in [0,1]$$

- "$\mathrm{anti\ SI}^{\mathrm{r}}$" counts (average of) all data points whose all "r" nearest neighbors have different labels with those data points
- Actually data points having higher anti si make hard examples in a data set
- They may be risky examples which experts have labeled them, incorrectly and they should be removed in a "data cleaning process"
- For when data points are images and other spatial or temporal formats, before to score them by "SI" or "anti SI", one must encode them.

An illustrative Examples



$SI^1 = 7/11$

$anti\_SI^1 = 1 - SI^1 = 4/11$

$SI^2 = 5/11$

$anti\_SI^2 = 2/11$

Smoothens index (SmI)

SmI measures how much input data points make the output targets smooth

2.2 Smoothness index (SI)

A linera smoothness measure for regression problem

1. First order SI

$Data = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{m}$ $\forall i: \boldsymbol{x}_i \epsilon R^{n\times 1}, \boldsymbol{y}_i \epsilon R^{o\times 1}$ o :number of outputs

*it is assumed that Data is a measured sample with high enough diversity.

*$\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ may have any format (video, image, time series, etc.) ; however, to compute SmI, it must be reshaped as a vector.

$$\mathrm{SmI}(Data) = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{\boldsymbol{d}_{imax} - \boldsymbol{d}_{i^*}}{\boldsymbol{d}_{imax} - \boldsymbol{d}_{imin}}\right)$$

$$i^* = \underset{\forall q\neq i}{\arg min}\left\|\boldsymbol{x}_i - \boldsymbol{x}_q\right\| \qquad \boldsymbol{d}_{imax} = \max_{\forall q}\left\|\boldsymbol{y}_i - \boldsymbol{y}_q\right\| \qquad \boldsymbol{d}_{imin} = \min_{\forall q\neq i}\left\|\boldsymbol{y}_i - \boldsymbol{y}_q\right\|$$

$$\boldsymbol{d}_{i^*} = \left\|\boldsymbol{y}_i - \boldsymbol{y}_{i^*}\right\|$$

*$\|\cdot\|$ denotes Euclidian distance (L_2 norm) but it may be another distance definition such as L_p norm.

** It is assumed that the input and target output data are normalized at each dimension just before computing the smoothness index.

*** the above definition of SmI can be biased by outliers.

A modified “linear SmI”

$$\mathrm{SmI}(Data) = \frac{1}{m}\sum_{i=1}^{m} relu\left(1 - \frac{\boldsymbol{d}_{i^*} - \boldsymbol{d}_{imin}}{\boldsymbol{d}_{imean}}\right)$$

$$i^* = \underset{\forall q \neq i}{\arg min}\left\|\boldsymbol{x}_i - \boldsymbol{x}_q\right\| \qquad \boldsymbol{d}_{imean} = \frac{1}{m}\sum_{q=1}^{m}\left\|\boldsymbol{y}_i - \boldsymbol{y}_q\right\| \qquad \boldsymbol{d}_{imin} = \min_{\forall q \neq i}\left\|\boldsymbol{y}_i - \boldsymbol{y}_q\right\|$$

$$\boldsymbol{d}_{i^*} = \left\|\boldsymbol{y}_i - \boldsymbol{y}_{i^*}\right\|$$

Some notes:

1. the above definition of SmI is not affected by outliers due to using mean of syance instead of maximum distance.
2. The “relu” function actually assign zero smoothness index to all data points whose their nearest neighbors have far enough targets with them.

A modified Exponential SmI

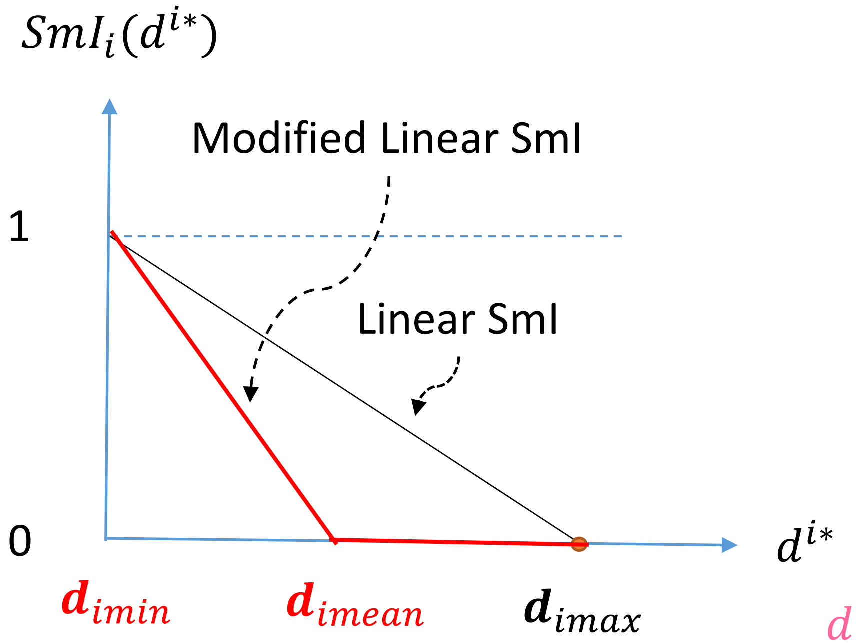
$$\mathrm{SmI}(Data) = \frac{1}{m}\sum_{i=1}^{m} SmI^{i} \quad SmI^{i} = \exp\left(-\gamma\,\frac{\boldsymbol{d}_{i^*} - \boldsymbol{d}_{imin}}{\boldsymbol{d}_{imean}}\right)$$

$$\boldsymbol{d}_{imean} = \frac{1}{m}\sum_{q=1}^{m}\left\|\boldsymbol{y}^{i} - \boldsymbol{y}^{q}\right\| \qquad \text{,Smoothness rate } \gamma>0$$

Some Notes:

1. Exponential SmI is not sensitivity to outliers.
2. For when $\gamma \to \infty$, any distance variation: $\left(\frac{\boldsymbol{d}_{i^*} - \boldsymbol{d}_{imin}}{\boldsymbol{d}_{imean}}\right)$ drops the SmI, significantly.
3. Here we have an exponential smoothness with an optional smoothness rate.
4. To have more restricted definition of SmI, the smoothness rate must be chosen high or γ>>1.

SmI Diagrams versus d^{i*}

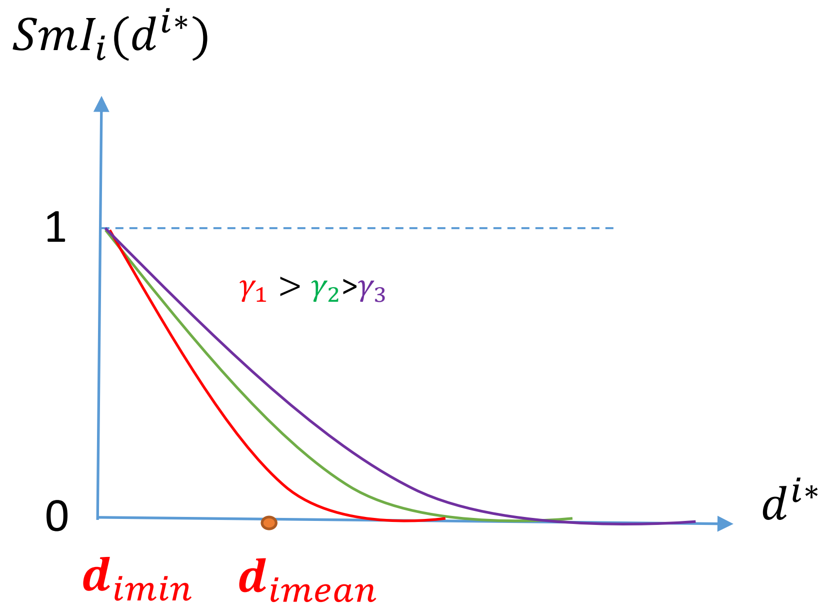


Linear SmI

$$d^{i*} = \|y_i - y_{i^*}\|$$

$$\boldsymbol{d}_{imax} = \max_{\forall q}\|\boldsymbol{y}_i - \boldsymbol{y}_q\|$$

$$\boldsymbol{d}_{imin} = \min_{\forall q \neq i}\|\boldsymbol{y}_i - \boldsymbol{y}_q\|$$



Exponential SmI

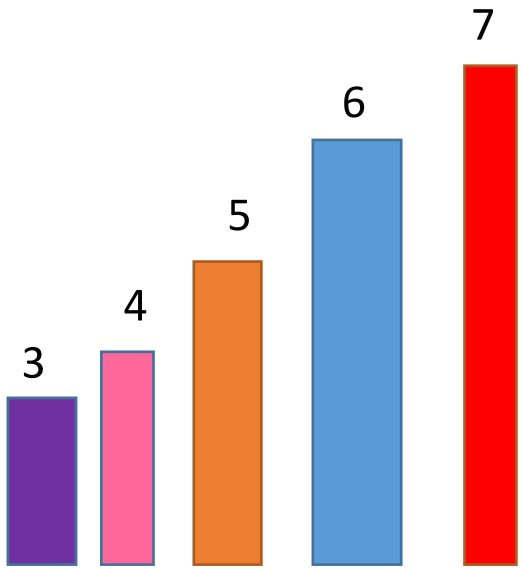
$$\boldsymbol{d}_{imean} = \frac{1}{m}\sum_{q=1}^{m}\|\boldsymbol{y}^i - \boldsymbol{y}^q\|$$

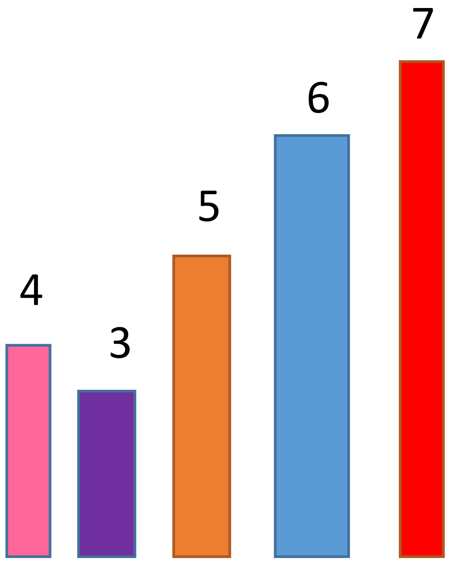
Some notes

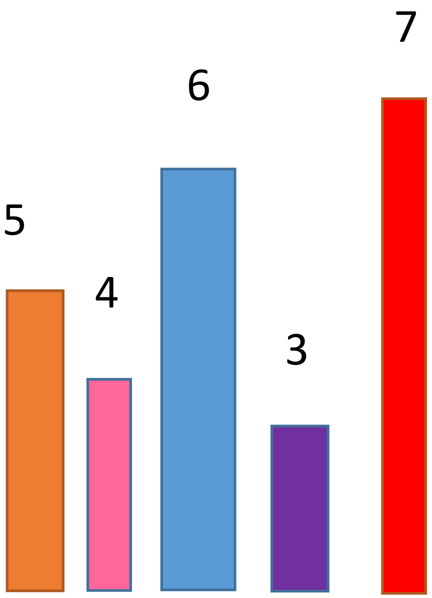
1. "SmI" is a normalized index between zero and one: $\mathrm{SmI} \in [0,1]$
2. $SmI \to 1$ $(Smoothness\ is\ maximmum)$ and $SmI \to 0$ $(Smoothness\ is\ minimmum)$
3. "SmI" measures that how nearness of input data leads to nearness of target data.
4. Assuming, the target outputs are outputs of a classification problem in "one-hot" format, SmI is actually measure the separation index: $\mathrm{SmI} = \mathrm{SI}$
5. Increasing the number of classes and considering a nearness among every two classes, SI is interpreted as a smoothness index. Actually, SmI shows in average that how neighboring examples in input space have classes with near distances in output.
6. SmI does not change for arbitrary position shift and (scalar) scale of the data
   $\forall\beta \neq 0, \forall\alpha \neq 0, \forall x_0, \forall y_0$ $\quad \mathrm{SmI}\left(\left\{\left(x^i, y^i\right)\right\}_{i=1}^m\right) = SmI\left(\left\{\left(\beta \boldsymbol{x}_i + x_0, \alpha \boldsymbol{y}_i + y_0\right)\right\}_{i=1}^m\right)$

7. Smoothness index of target outputs $with\ themselves\ is$ $maximum$: $SmI\left(\left\{\left(\boldsymbol{y}_i, \boldsymbol{y}_i\right)\right\}_{i=1}^m\right)$=1; it means that how input data become more similar to output the smoothness index will increase.

One-dimensional illustrative examples







$$SmI = \frac{1}{5}\left(\frac{4-1}{4-1} + \frac{3-1}{3-1} + \frac{2-1}{2-1} + \frac{3-1}{3-1} + \frac{4-1}{4-1}\right)$$

$$SmI = \frac{1}{5}\left(\frac{3-1}{3-1} + \frac{4-1}{4-1} + \frac{2-2}{2-1} + \frac{3-2}{3-1} + \frac{4-1}{4-1}\right)$$

$$SmI = \frac{1}{5}\left(\frac{2-1}{2-1} + \frac{3-1}{3-1} + \frac{3-2}{3-1} + \frac{4-3}{4-1} + \frac{4-4}{4-1}\right)$$

$SmI = 1$

SmI=0.7

SmI=0.566

2. High order SmI

$Data = \{(x_i, y_i)\}_{i=1}^{m} \forall i: x_i \epsilon R^{n\times 1} \quad \mathrm{y_i} \epsilon R^{o\times 1}$

$$\mathrm{SmI^r}(Data) = \frac{1}{m}\sum_{i=1}^{m} \min_{\forall j \epsilon \{1,..,r\}} \left(\frac{\boldsymbol{d}_{imax} - \boldsymbol{d}_{i_j^*}}{\boldsymbol{d}_{imax} - \boldsymbol{d}_{imin_j}} \right)$$

r: the order of "SmI"

$$i_j^* = \underset{\forall q \neq i, i_1^*, \cdots, i_{j-1}^*}{\arg min} \left\| \boldsymbol{x}_i - \boldsymbol{x}_q \right\| \qquad imin_j = \underset{\forall q \neq i, imin_1, \cdots, imin_{j-1}}{\arg min} \left\| \boldsymbol{y}_i - \boldsymbol{y}_q \right\|$$

$$\boldsymbol{d}_{imin_j} = \left\| \boldsymbol{y}_i - \boldsymbol{y}_{imin_j} \right\| \qquad \boldsymbol{d}_{i^*} = \left\| \boldsymbol{y}_i - \boldsymbol{y}_{i_j^*} \right\|$$

- $\mathrm{SmI^r} \in [0,1]$
- $\mathrm{SmI^r}$ considers more restricted condition of smoothness than $\mathrm{SmI^j}$ $(j < r)$.
- For each "Data" we have: $\mathrm{SmI^r} \leq \mathrm{SmI^{r-1}} \leq \cdots \leq \mathrm{SmI^1}$ $\qquad$ $\mathrm{SmI^1}$=SmI

3. High order soft SmI

$Data = \{(\boldsymbol{x}^i, \boldsymbol{y}^i)\}_{i=1}^{m} \quad \forall i:\ x^i \epsilon R^{n\times 1} \quad y^{\mathrm{i}} \epsilon R^{o\times 1}$

$$\mathrm{SmI}^{\mathrm{r}}_{\mathrm{soft}}(Data) = \frac{1}{m\times r}\sum_{i=1}^{m}\sum_{j=1}^{r}\left(\frac{\boldsymbol{d}_{imax} - \boldsymbol{d}_{i^*_j}}{\boldsymbol{d}_{imax} - \boldsymbol{d}_{imin_j}}\right) \mathrm{j} = 1,2,\ldots,\mathrm{r} \quad \mathrm{r}: \text{the order of "SmI"}$$

- $\mathrm{SmI}^{\mathrm{r}}_{\mathrm{soft}} \in [0,1]$
- $\mathrm{SmI}^{\mathrm{r}}_{\mathrm{soft}}$ considers less restricted condition of smoothness than $\mathrm{SmI}^{\mathrm{r}}$

$\mathrm{SmI}^{\mathrm{r}}_{\mathrm{soft}} \geq \mathrm{SmI}^{\mathrm{r}}$ and $\mathrm{SmI}^{1}_{\mathrm{soft}} = \mathrm{SmI}^{1}$

4. Cross SmI

$Data = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m}$ $\quad D_{test} = \{(\breve{\boldsymbol{x}}_i, \breve{y}_i)\}_{i=1}^{m_{test}}$

$$SmI_{cross}(D_{test}, Data) = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} \left(\frac{\boldsymbol{d}_{imax} - \boldsymbol{d}_{i^{\#}}}{\boldsymbol{d}_{imax} - \boldsymbol{d}_{imin_j}} \right)$$

$i^{\#} = \arg\min_{\forall q} \left\| \breve{\boldsymbol{x}}_i - \boldsymbol{x}_q \right\|$

Cross SI measures the separation index of a test domain of dataset D_{test} based on the main domain of dataset $Data$.

It can be shown that:

$$SmI_{cross}(D_{test}, Data) = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} SmI_{cross}\left((\breve{\boldsymbol{x}}_i, \breve{l}_i), Data \right)$$

Global SmI

- For the Data:$\{(\mathrm{x_i}, y_i\ )\}_{i=1}^{m}$ we have Global SmI when $SmI^m(Data) = 1.$
- For Data with Global SmI, One can show that for each example x_i and two other examples x_{i_1} and x_{i_2}:

$$if\ \|x_i - x_{i_1}\| \leq \|x_i - x_{i_2}\| \text{ then } \|y_i - y_{i_1}\| \leq \|y_i - y_{i_2}\|$$

- For Data$(\mathrm{x_i}, y_i\ )\}_{i=1}^{m}$ where $y_i = \Psi x_i$ and Ψ have orthogonal columns with equal norms, we have global SmI.

Similarity transformation in "SI" and "SmI"

- Show that for all possible r

$$SI^r(\{(x_i, l_i)\}_{i=1}^m) = SI^r(\{(\Psi_1 x_i, \Psi_2 l_i)\}_{i=1}^m)$$
$$SmI^r(\{(x_i, y_i)\}_{i=1}^m) = SmI^r(\{(\Psi_1 x_i, \Psi_2 y_i)\}_{i=1}^m)$$

where

Ψ_h (h=1,2) have orthogonal columns with equal norms.

Data Connectivity SmI

- $Data^{k}$:$\{(\mathrm{x}_{\mathrm{i}}^{k})\}_{i=1}^{m}\}$, k=1,2,…,N

$$DC = \left[smI_{k_1,k_2}\right]_{N\times N}$$

$$smI_{k_1,k_2}=\mathrm{smI}(Data^{k_1}, Data^{k_2})$$

Thank you