IBM **Research**

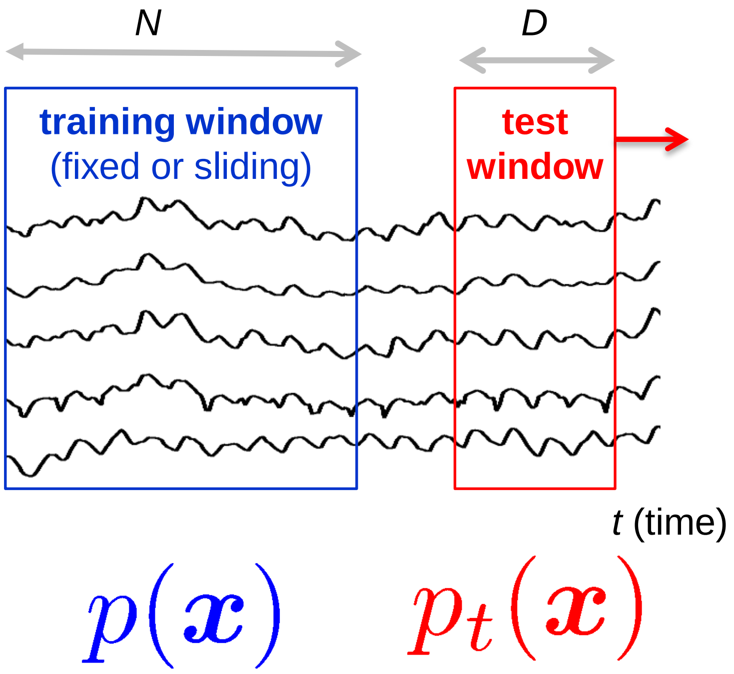
Change Detection Using Directional Statistics

T. Ide ("Ide-san"), D. Phan, J. Kalagnanam
IBM T. J. Watson Research Center

The 25th International Joint Conference on Artificial Intelligence (IJCAI-16)

Problem setting: change detection from multi-variate noisy time-series data

- Change = difference between $p(\boldsymbol{x})$ and $p_t(\boldsymbol{x})$
  - **x**: *M*-dimensional *i.i.d.* observation
  - *p*(**x**): p.d.f. estimated from training window
  - p_t(**x**): p.d.f. estimated from the test window at time *t*
- Question 1: What kind of model should we use for the pdf?
- Question 2: How can we quantify the difference between the p.d.f.'s?

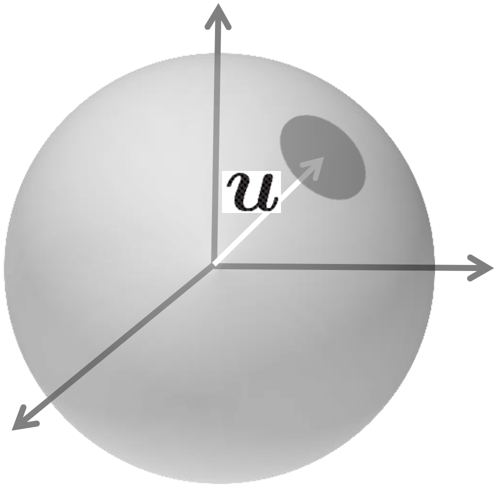


We use von Mises-Fisher distribution to model $p(\boldsymbol{x})$ and $p_t(\boldsymbol{x})$

- vMF distribution: "Gaussian for unit vectors"

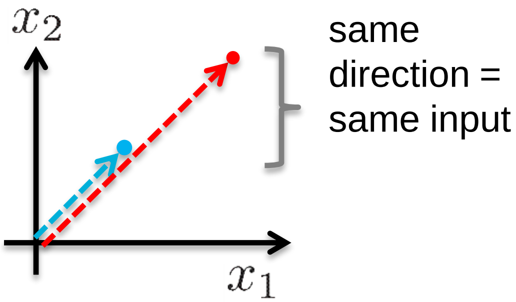
$$p(\boldsymbol{z} \mid \boldsymbol{u}, \kappa) = c_M(\kappa) \exp\left(\kappa \boldsymbol{u}^\top \boldsymbol{z}\right)$$

  - ***z***: random unit vector of ||**z**|| =1
  - ***u***: mean direction
  - κ: "concentration" (~ precision in Gaussian)
  - *M*: dimensionality



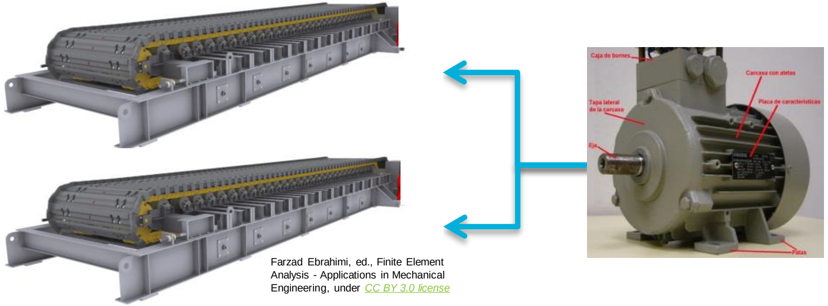
- We are concerned only with the direction of observation ***x***:
  - $$\boldsymbol{z} = \frac{\boldsymbol{x}}{||\boldsymbol{x}||}$$

  • Normalization is always made
  • Do not care about the norm



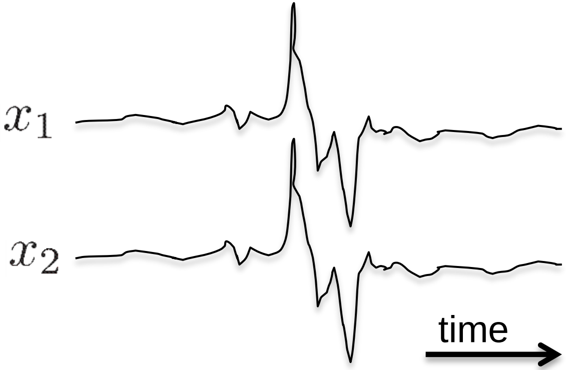
Why do we enforce normalization? Rationale from the real world

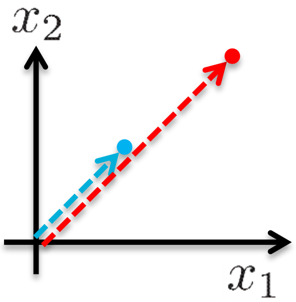
- Real mechanical systems often incur multiplicative noise
  - Example: two belt conveyors operated by the same motor
  - Multiplicative noise equally applied to correlated variables



Farzad Ebrahimi, ed., Finite Element Analysis - Applications in Mechanical Engineering, under CC BY 3.0 license

- Normalization is simple but powerful method for noise reduction





Mean direction *u* is learned via weighted maximum likelihood to down-weight contaminated samples

- Weighted likelihood function

$\|\boldsymbol{x}^{(n)}\|_2$ (normalization factor)

$$L(\boldsymbol{u},\kappa) = \sum_{n=1}^{N} w^{(n)} b^{(n)} \{\ln c_M(\kappa) + \kappa \boldsymbol{u}^\top \boldsymbol{z}^{(n)}\}$$

sample weight

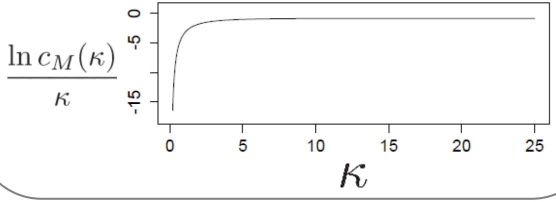
- Regularization over sample weights

$$R(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \nu\|\boldsymbol{w}\|_1$$

- p.d.f. is learned by solving

$$(\boldsymbol{u}^*, \boldsymbol{w}^*) = \arg\max_{\boldsymbol{u},\boldsymbol{w}} \{L(\boldsymbol{u},\kappa) + \lambda R(\boldsymbol{w})\}$$

The term related to κ is less important. κ is treated as a given constant.



Multiple patterns (directions) can be obtained by coupling maximum likelihood equations

- Find orthogonal sequence of the mean direction $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_m$ by coupling the weighted regularized maximum likelihood

$$(\boldsymbol{u}_1^*, \boldsymbol{w}_1^*) = \arg \max_{\boldsymbol{u}_1, \boldsymbol{w}_1} \{L(\boldsymbol{u}_1, \kappa) + \lambda R(\boldsymbol{w}_1)\}$$

$$(\boldsymbol{u}_2^*, \boldsymbol{w}_2^*) = \arg \max_{\boldsymbol{u}_2, \boldsymbol{w}_2} \{L(\boldsymbol{u}_2, \kappa) + \lambda R(\boldsymbol{w}_2)\}$$

$$\vdots$$

$$(\boldsymbol{u}_m^*, \boldsymbol{w}_m^*) = \arg \max_{\boldsymbol{u}_m, \boldsymbol{w}_m} \{L(\boldsymbol{u}_m, \kappa) + \lambda R(\boldsymbol{w}_m)\}$$

Orthogonality condition

$$\boldsymbol{u}_i^\top \boldsymbol{u}_j = \delta_{i,j}$$

Kronecker delta

Iterative sequential algorithm for the coupled maximum likelihood

$$\boldsymbol{w}_1 \rightleftarrows \boldsymbol{u}_1$$

$$\boldsymbol{w}_2 \rightleftarrows \boldsymbol{u}_2 \perp \boldsymbol{u}_1$$

$$\vdots$$

$$\boldsymbol{w}_m \rightleftarrows \boldsymbol{u}_m \perp \boldsymbol{u}_i$$

$$(i = 1, \ldots, m-1)$$

- For each i, $\boldsymbol{w}_i$ and $\boldsymbol{u}_i$ are solved iteratively until convergence
- Analytic solution exists in each step
- Results in very simple fixed point equations

(For reference) Derived fixed-point iteration algorithm

- Example: i =1

Given $\boldsymbol{w}_1$, solve

$$\max_{\boldsymbol{u}_1}\{\kappa \boldsymbol{u}_1^\top \mathsf{X} \boldsymbol{w}_1\} \quad \text{s.t.} \quad \boldsymbol{u}_1^\top \boldsymbol{u}_1 = 1$$

Given $\boldsymbol{u}_1$, solve

$$\min_{\boldsymbol{w}_1}\left\{\frac{1}{2}\|\boldsymbol{w}_1 - \frac{\boldsymbol{q}}{\lambda}\|_2^2 + \nu\|\boldsymbol{w}_1\|_1\right\}$$

$$\boldsymbol{q} \equiv \ln c_M \boldsymbol{b} + \kappa \mathsf{X}^\top \boldsymbol{u}_1$$

This Lasso problem is solved analytically

Algorithm 1 RED algorithm.

Input: Initialized $\boldsymbol{w}$. Regularization parameters λ, ν. Concentration parameter κ. The number of major directional patterns m.
Output: $\mathsf{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$ and $\mathsf{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m]$.
for $j = 1, 2, \ldots, m$ **do**
  while no convergence **do**

$$\boldsymbol{u}_j \leftarrow \kappa[\mathsf{I}_M - \mathsf{U}_{j-1}\mathsf{U}_{j-1}^\top]\mathsf{X}\boldsymbol{w}_j \tag{17}$$

$$\boldsymbol{u}_j \leftarrow \operatorname{sign}(\boldsymbol{u}_j^\top \mathsf{X}\boldsymbol{w}_j)\frac{\boldsymbol{u}_j}{\|\boldsymbol{u}_j\|_2} \tag{18}$$

$$\boldsymbol{q}_j \leftarrow \gamma \boldsymbol{b} + \kappa \mathsf{X}^\top \boldsymbol{u}_j \tag{19}$$

$$\boldsymbol{w}_j \leftarrow \operatorname{sign}(\boldsymbol{q}_j) \odot \max\left\{\frac{|\boldsymbol{q}_j|}{\lambda} - \nu\boldsymbol{1}, \boldsymbol{0}\right\} \tag{20}$$

  end while
end for
Return U and W.

Theoretical property: The algorithm is reduced to the "trust-region subproblem" in $\nu \to 0$

Theorem 2. *When ν tends to 0, the nonconvex problem (5) is reduced to an optimization problem in the form of*

$$\min_{\boldsymbol{u}} \{\boldsymbol{u}^\top \mathsf{Q}\boldsymbol{u} + \boldsymbol{c}^\top \boldsymbol{u}\} \quad s.t. \quad \boldsymbol{u}^\top \boldsymbol{u} = 1, \tag{23}$$

which has a global solution obtained in polynomial time.

Useful to initialize the iterative algorithm

Proof. The non-convex optimization problem (23) is known as the *trust region subproblem*. For polynomial algorithms to the global solution, see [Sorensen, 1997; Tao and An, 1998; Hager, 2001; Toint *et al.*, 2009]. Here we show how the algorithm is reduced to the trust region subproblem.

Change score as parameterized Kullback-Leibler divergence

- With extracted directions, define the change score at time *t* as

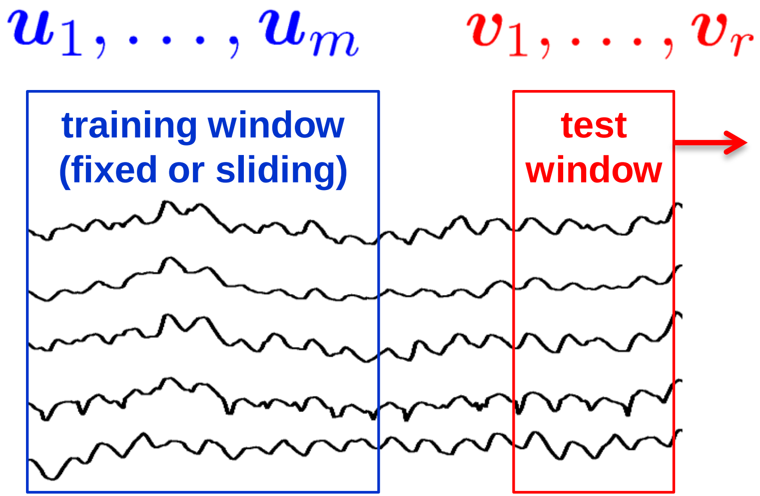
vMF dist. vMF dist.

$$a^{(t)} = \min_{\boldsymbol{f},\boldsymbol{g}} \int \mathrm{d}\boldsymbol{x}\, \mathcal{M}(\boldsymbol{x}|\mathsf{U}\boldsymbol{f},\kappa) \ln \frac{\mathcal{M}(\boldsymbol{x}|\mathsf{U}\boldsymbol{f},\kappa)}{\mathcal{M}(\boldsymbol{x}|\mathsf{U}^{(t)}\boldsymbol{g},\kappa)}$$

vMF dist.

$$\boldsymbol{f}^\top \boldsymbol{f} = 1,\ \boldsymbol{g}^\top \boldsymbol{g} = 1$$

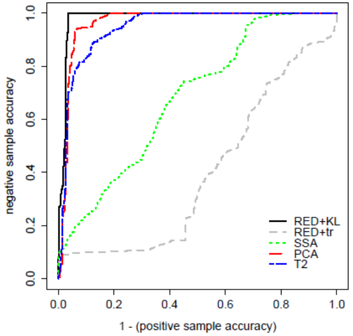
- Concisely represented by the top singular value of $\mathsf{U}^\top \mathsf{U}^{(t)}$



$$\mathsf{U} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m] \quad \mathsf{U}^{(t)} \equiv [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r]$$

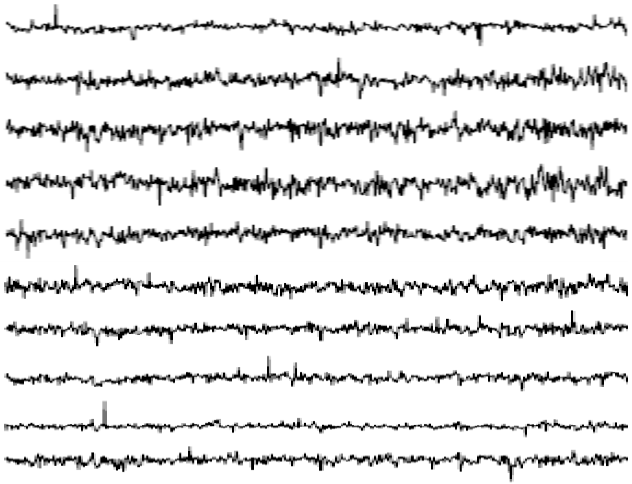
Experiment: Failure detection of ore belt conveyors

- vMF formulation successfully suppressed very noisy non-Gaussian noise of multiplicative nature
- ~40% of samples were automatically excluded from the model
- Better than alternatives
  - PCA, Hoteling T^2
  - Stationary subspace analysis [Blythe et al., 2012]



Learned sample weights

Training data



(simulation data)

Summary – Thank you for your attention!

- Proposed a new change detection algorithm featuring
  - (1) New feature extraction method based on weighted max. likelihood of the vMF distribution
  - (2) Change score based on parameterized KL divergence
- Showed the linkage with the trust region sub-problem