ZUSE INSTITUTE BERLIN

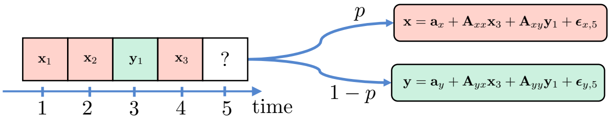
Fast Robust Classifiers for Data Streams

INFORMS 2022

Kartikey Sharma

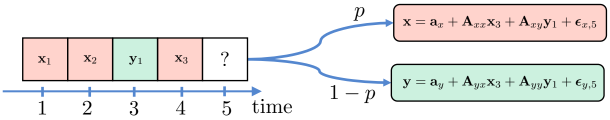
16th October 2022

Data Streams



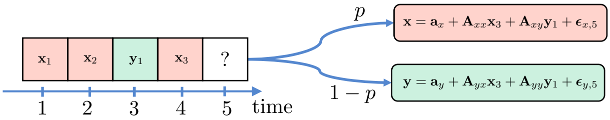
- Data comes in over time
- Distribution of data may change over time
- Classifier needs to adapt to changing distribution

Data Streams



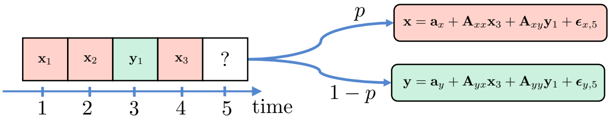
- Data comes in over time
- Distribution of data may change over time
- Classifier needs to adapt to changing distribution

Data Streams



- Data comes in over time
- Distribution of data may change over time
- Classifier needs to adapt to changing distribution

Data Streams



- Data comes in over time
- Distribution of data may change over time
- Classifier needs to adapt to changing distribution

Existing Research

Non parametric methods. Time-windows (Li et al. 2017; Nguyen, Woon, and Ng 2015; Žliobaitė et al. 2013), forgetting methods (Anagnostopoulos et al. 2012; Krawczyk and Woźniak 2015)

Parametric Methods. Time-series and Gaussian processes (Kumagai and Iwata 2016; Kumagai and Iwata 2017; Kumagai and Iwata 2018).

Neural Networks Architectures . LSTMs (Jia et al. 2017), Spiking NNs (Lobo et al. 2020), Limited data (Das et al. 2020; Ksieniewicz et al. 2019).

Existing Research

Non parametric methods. Time-windows (Li et al. 2017; Nguyen, Woon, and Ng 2015; Žliobaitė et al. 2013), forgetting methods (Anagnostopoulos et al. 2012; Krawczyk and Woźniak 2015)

Parametric Methods. Time-series and Gaussian processes (Kumagai and Iwata 2016; Kumagai and Iwata 2017; Kumagai and Iwata 2018).

Neural Networks Architectures . LSTMs (Jia et al. 2017), Spiking NNs (Lobo et al. 2020), Limited data (Das et al. 2020; Ksieniewicz et al. 2019).

Existing Research

Non parametric methods. Time-windows (Li et al. 2017; Nguyen, Woon, and Ng 2015; Žliobaitė et al. 2013), forgetting methods (Anagnostopoulos et al. 2012; Krawczyk and Woźniak 2015)

Parametric Methods. Time-series and Gaussian processes (Kumagai and Iwata 2016; Kumagai and Iwata 2017; Kumagai and Iwata 2018).

Neural Networks Architectures . LSTMs (Jia et al. 2017), Spiking NNs (Lobo et al. 2020), Limited data (Das et al. 2020; Ksieniewicz et al. 2019).

Minimax Probability Machine

Minimax probability machine (Lanckriet, El Ghaoui, and Jordan 2003):

$$\begin{aligned} \max_{\alpha,\mathbf{a},b}\ & \alpha \\ \text{s.t.}\ & \inf_{\mathbb{P}_x \in \mathcal{M}_x} \mathbb{P}_x[\mathbf{a}^\top \mathbf{x} \geq b] \geq \alpha \\ & \inf_{\mathbb{P}_y \in \mathcal{M}_y} \mathbb{P}_y[\mathbf{a}^\top \mathbf{y} \leq b] \geq \alpha. \end{aligned}$$

$$\begin{aligned} \min_{r,\mathbf{u},\mathbf{v}}\ & r \\ \text{s.t.}\ & \mu_x + \Sigma_x^{\frac{1}{2}}\mathbf{u} = \mu_y + \Sigma_y^{\frac{1}{2}}\mathbf{w} \\ & \|\mathbf{u}\|_2 \leq r \\ & \|\mathbf{w}\|_2 \leq r. \end{aligned}$$

Here, $\mathcal{M}_i$ represents distributions with mean $\boldsymbol{\mu}_i$ and covariance $\boldsymbol{\Sigma}_i$.

Minimax Probability Machine

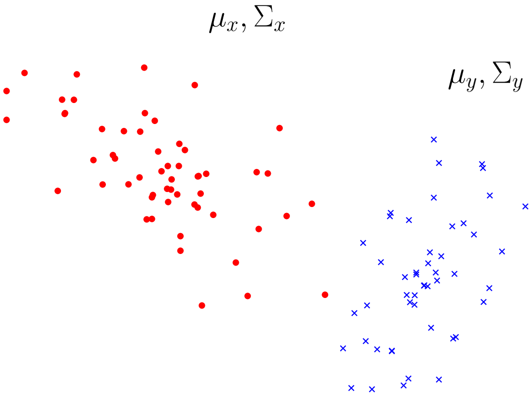
Minimax probability machine (Lanckriet, El Ghaoui, and Jordan 2003):

$$\begin{aligned} \max_{\alpha,\mathbf{a},b}\ & \alpha \\ \text{s.t.}\ & \inf_{\mathbb{P}_x \in \mathcal{M}_x} \mathbb{P}_x[\mathbf{a}^\top \mathbf{x} \geq b] \geq \alpha \\ & \inf_{\mathbb{P}_y \in \mathcal{M}_y} \mathbb{P}_y[\mathbf{a}^\top \mathbf{y} \leq b] \geq \alpha. \end{aligned}$$

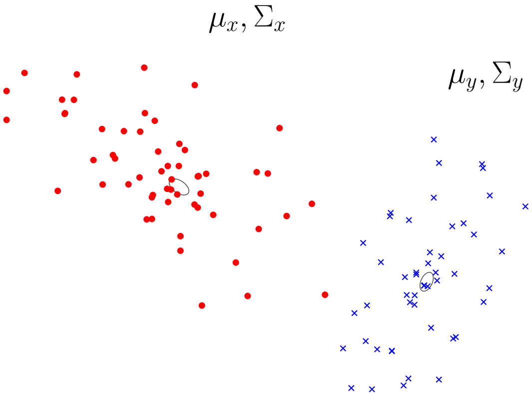
$$\begin{aligned} \min_{r,\mathbf{u},\mathbf{v}}\ & r \\ \text{s.t.}\ & \boldsymbol{\mu}_x + \Sigma_x^{\frac{1}{2}} \mathbf{u} = \boldsymbol{\mu}_y + \Sigma_y^{\frac{1}{2}} \mathbf{w} \\ & \|\mathbf{u}\|_2 \leq r \\ & \|\mathbf{w}\|_2 \leq r. \end{aligned}$$

Here, $\mathcal{M}_i$ represents distributions with mean $\boldsymbol{\mu}_i$ and covariance $\boldsymbol{\Sigma}_i$.

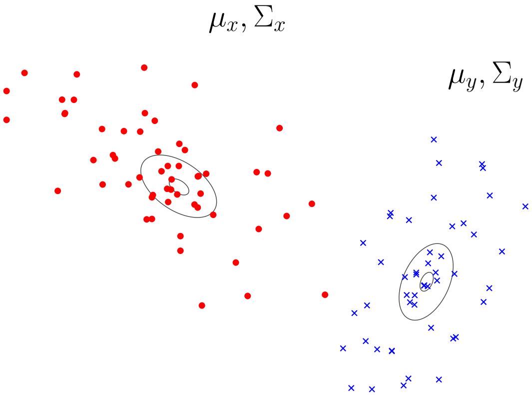
Classifying Surface



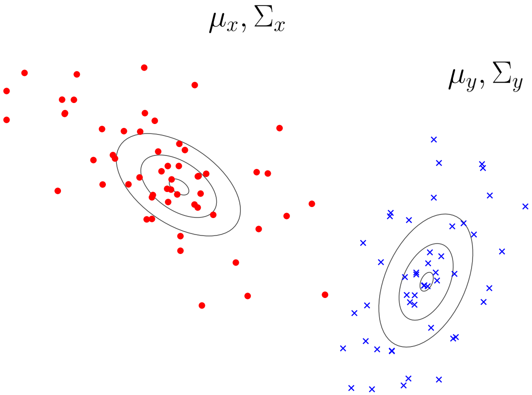
Classifying Surface



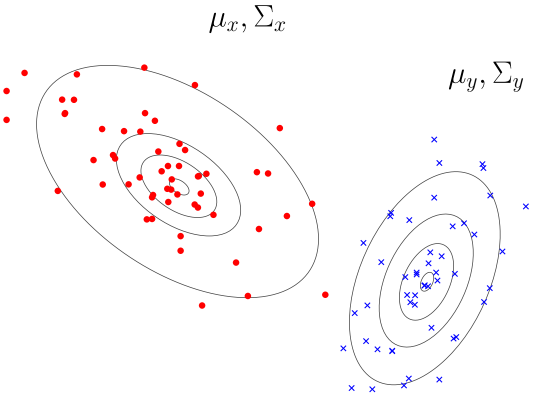
Classifying Surface



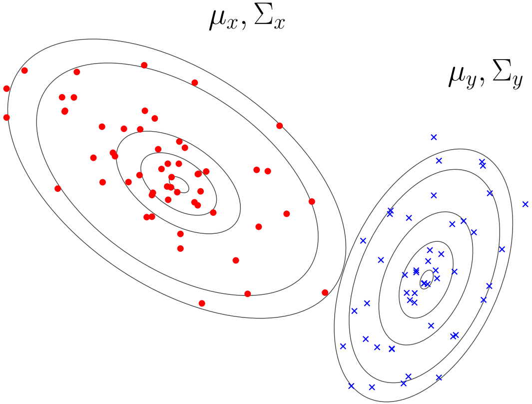
Classifying Surface



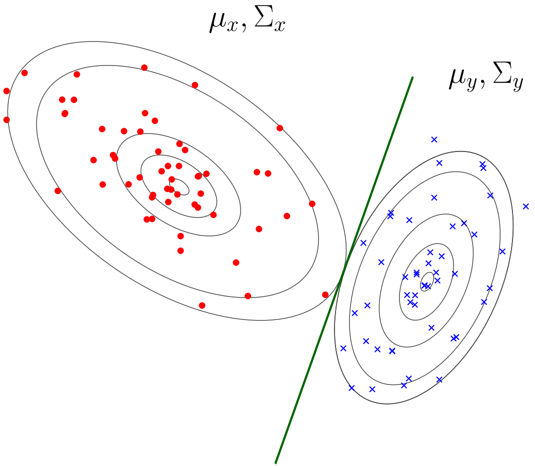
Classifying Surface



Classifying Surface

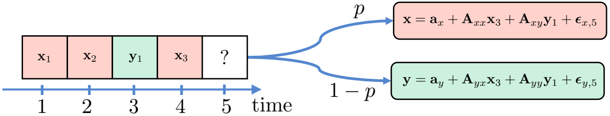


Classifying Surface



Data Stream Model

- Time series model of data

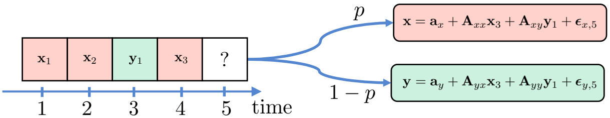


- Allows for correlation over time and cross class dependence

$$
\begin{aligned}
\mu_x(\hat{\mathbf{x}}, \hat{\mathbf{y}}) &= \mathbf{E}[\mathbf{x} | \mathbf{x}_{\tau_x(t)} = \hat{\mathbf{x}}, \mathbf{y}_{\tau_y(t)} = \hat{\mathbf{y}}] \\
&= \mathbf{a}_x + \mathbf{A}_{xx}\hat{\mathbf{x}} + \mathbf{A}_{xy}\hat{\mathbf{y}} \\
\mu_y(\hat{\mathbf{x}}, \hat{\mathbf{y}}) &= \mathbf{E}[\mathbf{y} | \mathbf{x}_{\tau_x(t)} = \hat{\mathbf{x}}, \mathbf{y}_{\tau_y(t)} = \hat{\mathbf{y}}] \\
&= \mathbf{a}_y + \mathbf{A}_{yx}\hat{\mathbf{x}} + \mathbf{A}_{yy}\hat{\mathbf{y}}.
\end{aligned}
$$

Data Stream Model

- Time series model of data

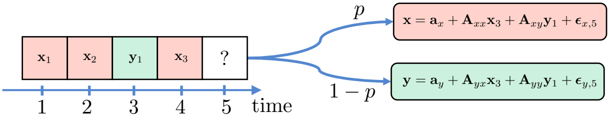


- Allows for correlation over time and cross class dependence

$$
\begin{aligned}
\mu_x(\hat{\mathbf{x}}, \hat{\mathbf{y}}) &= \mathbf{E}[\mathbf{x} | \mathbf{x}_{\tau_x(t)} = \hat{\mathbf{x}}, \mathbf{y}_{\tau_y(t)} = \hat{\mathbf{y}}] \\
&= \mathbf{a}_x + \mathbf{A}_{xx}\hat{\mathbf{x}} + \mathbf{A}_{xy}\hat{\mathbf{y}} \\
\mu_y(\hat{\mathbf{x}}, \hat{\mathbf{y}}) &= \mathbf{E}[\mathbf{y} | \mathbf{x}_{\tau_x(t)} = \hat{\mathbf{x}}, \mathbf{y}_{\tau_y(t)} = \hat{\mathbf{y}}] \\
&= \mathbf{a}_y + \mathbf{A}_{yx}\hat{\mathbf{x}} + \mathbf{A}_{yy}\hat{\mathbf{y}}.
\end{aligned}
$$

Data Stream Model

- Time series model of data



- Allows for correlation over time and cross class dependence

$$\begin{aligned}
\mu_x(\hat{\mathbf{x}}, \hat{\mathbf{y}}) &= \mathbf{E}[\mathbf{x} | \mathbf{x}_{\tau_x(t)} = \hat{\mathbf{x}}, \mathbf{y}_{\tau_y(t)} = \hat{\mathbf{y}}] \\
&= \mathbf{a}_x + \mathbf{A}_{xx}\hat{\mathbf{x}} + \mathbf{A}_{xy}\hat{\mathbf{y}} \\
\mu_y(\hat{\mathbf{x}}, \hat{\mathbf{y}}) &= \mathbf{E}[\mathbf{y} | \mathbf{x}_{\tau_x(t)} = \hat{\mathbf{x}}, \mathbf{y}_{\tau_y(t)} = \hat{\mathbf{y}}] \\
&= \mathbf{a}_y + \mathbf{A}_{yx}\hat{\mathbf{x}} + \mathbf{A}_{yy}\hat{\mathbf{y}}.
\end{aligned}$$

Explicit Dependence

- Find policy for changing the classifying surface
- Classifying surface dependent upon point of contact

$$
\begin{aligned}
&\min_{\substack{r,\mathbf{u}\\ \mathbf{w}}} r \\
&\text{s.t. } \boldsymbol{\mu}_x + \Sigma_x^{1/2}\mathbf{u} = \boldsymbol{\mu}_y + \Sigma_y^{1/2}\mathbf{w} \\
&\qquad \|\mathbf{u}\|_2 \leq r \\
&\qquad \|\mathbf{w}\|_2 \leq r.
\end{aligned}
$$

Explicit Dependence

- Find policy for changing the classifying surface
- Classifying surface dependent upon point of contact

$$
\begin{aligned}
\min_{\substack{r,\mathbf{u}\\ \mathbf{w}}} \; & r \\
\text{s.t. } & \mu_x + \Sigma_x^{1/2}\mathbf{u} = \mu_y + \Sigma_y^{1/2}\mathbf{w} \\
& \|\mathbf{u}\|_2 \leq r \\
& \|\mathbf{w}\|_2 \leq r.
\end{aligned}
$$

Explicit Dependence

- Find policy for changing the classifying surface
- Classifying surface dependent upon point of contact

$$
\begin{aligned}
&\min_{\substack{r,\mathbf{u}\\ \mathbf{w}}} r \\
&\text{s.t. } \boldsymbol{\mu}_x + \boldsymbol{\Sigma}_x^{1/2}\mathbf{u} = \boldsymbol{\mu}_y + \boldsymbol{\Sigma}_y^{1/2}\mathbf{w} \\
&\qquad \|\mathbf{u}\|_2 \leq r \\
&\qquad \|\mathbf{w}\|_2 \leq r.
\end{aligned}
$$

Explicit Dependence

- Find policy for changing the classifying surface
- Classifying surface dependent upon point of contact

$$
\begin{aligned}
\min_{\substack{r(\mathbf{x},\mathbf{y}),\mathbf{u}(\mathbf{x},\mathbf{y})\\ \mathbf{w}(\mathbf{x},\mathbf{y})}} \quad & r(\mathbf{x},\mathbf{y}) \\
\text{s.t.}\ \boldsymbol{\mu}_x(\mathbf{x},\mathbf{y}) + \Sigma_x^{1/2}\mathbf{u}(\mathbf{x},\mathbf{y}) &= \boldsymbol{\mu}_y(\mathbf{x},\mathbf{y}) + \Sigma_y^{1/2}\mathbf{w}(\mathbf{x},\mathbf{y}) \\
\|\mathbf{u}(\mathbf{x},\mathbf{y})\|_2 &\leq r(\mathbf{x},\mathbf{y}) \\
\|\mathbf{w}(\mathbf{x},\mathbf{y})\|_2 &\leq r(\mathbf{x},\mathbf{y}).
\end{aligned}
$$

Explicit Dependence

- Find policy for changing the classifying surface
- Classifying surface dependent upon point of contact

$$
\begin{aligned}
\min_{\substack{r(\mathbf{x},\mathbf{y}),\mathbf{u}(\mathbf{x},\mathbf{y})\\ \mathbf{w}(\mathbf{x},\mathbf{y})}} \; & \max_{\substack{\mathbf{x}\in\boldsymbol{U}_x\\ \mathbf{y}\in\boldsymbol{U}_y}} r(\mathbf{x},\mathbf{y}) \\
\text{s.t. } & \boldsymbol{\mu}_x(\mathbf{x},\mathbf{y}) + \boldsymbol{\Sigma}_x^{1/2}\mathbf{u} = \boldsymbol{\mu}_y(\mathbf{x},\mathbf{y}) + \boldsymbol{\Sigma}_y^{1/2}\mathbf{w}(\mathbf{x},\mathbf{y}) && \forall \mathbf{x} \in \boldsymbol{U}_x \; \forall \mathbf{y} \in \boldsymbol{U}_y \\
& \|\mathbf{u}(\mathbf{x},\mathbf{y})\|_2 \leq r(\mathbf{x},\mathbf{y}) && \forall \mathbf{x} \in \boldsymbol{U}_x, \;\; \mathbf{y} \in \boldsymbol{U}_y \\
& \|\mathbf{w}(\mathbf{x},\mathbf{y})\|_2 \leq r(\mathbf{x},\mathbf{y}) && \forall \mathbf{x} \in \boldsymbol{U}_x, \;\; \mathbf{y} \in \boldsymbol{U}_y.
\end{aligned}
$$

Affine Approximation

- Affine policy to approximate point of contact

$$
\begin{aligned}
\mathbf{u}(\mathbf{x},\mathbf{y}) &= \mathbf{u}_0 + \mathbf{U}_x\mathbf{x} + \mathbf{U}_y\mathbf{y} \\
&= \mathbf{u}(\xi) = [\mathbf{u}_0 \mid \mathbf{U}_x \mid \mathbf{U}_y]\xi = \overline{\mathbf{U}}\xi
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{w}(\mathbf{x},\mathbf{y}) &= \mathbf{w}_0 + \mathbf{W}_x\mathbf{x} + \mathbf{W}_y\mathbf{y} \\
&= \mathbf{w}(\xi) = [\mathbf{w}_0 \mid \mathbf{W}_x \mid \mathbf{W}_y]\xi = \overline{\mathbf{W}}\xi
\end{aligned}
$$

$$
\mathbf{r}(\mathbf{x},\mathbf{y}) = r,
$$

$$
\begin{aligned}
\min_{r,\overline{\mathbf{U}},\overline{\mathbf{V}}} \quad & r \\
\text{s.t.} \quad & \overline{\mathbf{X}}\xi + \Sigma_x^{\frac{1}{2}}\overline{\mathbf{U}}\xi = \overline{\mathbf{Y}}\xi + \Sigma_y^{\frac{1}{2}}\overline{\mathbf{W}}\xi && \forall \xi \in \boldsymbol{U} \\
& \|\overline{\mathbf{U}}\xi\|_2 \le r && \forall \xi \in \boldsymbol{U} \\
& \|\overline{\mathbf{W}}\xi\|_2 \le r && \forall \xi \in \boldsymbol{U}.
\end{aligned}
$$

Affine Approximation

- Affine policy to approximate point of contact

$$
\begin{aligned}
\mathbf{u}(\mathbf{x},\mathbf{y}) &= \mathbf{u}_0 + \mathbf{U}_x\mathbf{x} + \mathbf{U}_y\mathbf{y} \\
&= \mathbf{u}(\xi) = [\mathbf{u}_0 \mid \mathbf{U}_x \mid \mathbf{U}_y]\xi = \overline{\mathbf{U}}\xi
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{w}(\mathbf{x},\mathbf{y}) &= \mathbf{w}_0 + \mathbf{W}_x\mathbf{x} + \mathbf{W}_y\mathbf{y} \\
&= \mathbf{w}(\xi) = [\mathbf{w}_0 \mid \mathbf{W}_x \mid \mathbf{W}_y]\xi = \overline{\mathbf{W}}\xi
\end{aligned}
$$

$$
\mathbf{r}(\mathbf{x},\mathbf{y}) = r,
$$

$$
\begin{aligned}
\min_{r,\overline{\mathbf{U}},\overline{\mathbf{V}}} \quad & r \\
\text{s.t.} \quad & \overline{\mathbf{X}}\xi + \Sigma_x^{\frac{1}{2}}\overline{\mathbf{U}}\xi = \overline{\mathbf{Y}}\xi + \Sigma_y^{\frac{1}{2}}\overline{\mathbf{W}}\xi && \forall \xi \in \boldsymbol{U} \\
& \|\overline{\mathbf{U}}\xi\|_2 \le r && \forall \xi \in \boldsymbol{U} \\
& \|\overline{\mathbf{W}}\xi\|_2 \le r && \forall \xi \in \boldsymbol{U}.
\end{aligned}
$$

Affine Approximation

- Affine policy to approximate point of contact

$$\begin{aligned} \mathbf{u}(\mathbf{x},\mathbf{y}) &= \mathbf{u}_0 + \mathbf{U}_x\mathbf{x} + \mathbf{U}_y\mathbf{y} \\ = \mathbf{u}(\xi) &= [\mathbf{u}_0 \mid \mathbf{U}_x \mid \mathbf{U}_y]\xi = \overline{\mathbf{U}}\xi \end{aligned}$$

$$\begin{aligned} \mathbf{w}(\mathbf{x},\mathbf{y}) &= \mathbf{w}_0 + \mathbf{W}_x\mathbf{x} + \mathbf{W}_y\mathbf{y} \\ = \mathbf{w}(\xi) &= [\mathbf{w}_0 \mid \mathbf{W}_x \mid \mathbf{W}_y]\xi = \overline{\mathbf{W}}\xi \end{aligned}$$

$$\mathbf{r}(\mathbf{x},\mathbf{y}) = r,$$

$$\begin{aligned} \min_{r,\overline{\mathbf{U}},\overline{\mathbf{V}}} \quad & r \\ \text{s.t.} \quad & \overline{\mathbf{X}}\xi + \Sigma_x^{\frac{1}{2}}\overline{\mathbf{U}}\xi = \overline{\mathbf{Y}}\xi + \Sigma_y^{\frac{1}{2}}\overline{\mathbf{W}}\xi && \forall \xi \in \boldsymbol{U} \\ & \|\overline{\mathbf{U}}\xi\|_2 \leq r && \forall \xi \in \boldsymbol{U} \\ & \|\overline{\mathbf{W}}\xi\|_2 \leq r && \forall \xi \in \boldsymbol{U}. \end{aligned}$$

AjRC Algorithm

Algorithm 1 The steps of AjRC for classifying streaming data.

1: Initialize: $t = 1$ and previous data points $\mathbf{x}_0$ and $\mathbf{y}_0$
2: Solve the SDP formulation and obtain policy matrices $\overline{\mathbf{U}}$ and $\overline{\mathbf{W}}$.
3: **while** New streaming data point $\mathbf{z}_t$ is available **do**
4: Let $\xi_t = (1, \mathbf{x}_{\tau_x(t)}, , \mathbf{y}_{\tau_y(t)})$.
5: Estimate contact point $\mathbf{v} = \overline{\mathbf{X}}\xi + \mathbf{L}_x\overline{\mathbf{U}}\xi$.
6: Calculate classifiers $\mathbf{g}_1 = 2\Sigma_x^{-1}\mathbf{L}_x\overline{\mathbf{U}}\xi$ and $\mathbf{g}_2 = 2\Sigma_y^{-1}\mathbf{L}_y\overline{\mathbf{W}}\xi$ with $c_i = \mathbf{g}_i^\top\mathbf{v}$.
7: Evaluate probabilities $\mathbb{P}(\mathbf{g}_i) = p \cdot \mathbb{P}(\mathbf{g}_i^\top\mathbf{x} \geq c_i) + (1-p) \cdot \mathbb{P}(\mathbf{g}_i^\top\mathbf{y} \leq c_i)$.
8: Use surface with higher probability to classify $\mathbf{z}_t$.
9: Record the observed true class of $\mathbf{z}_t$.
10: $t = t + 1$
11: **end while**

AjRC Algorithm

Algorithm 2 The steps of AjRC for classifying streaming data.

Initialize: $t = 1$ and previous data points $\mathbf{x}_0$ and $\mathbf{y}_0$

Solve the SDP formulation and obtain policy matrices $\overline{\mathbf{U}}$ and $\overline{\mathbf{W}}$.

while New streaming data point $\mathbf{z}_t$ is available **do**

  Let $\xi_t = (1, \mathbf{x}_{\tau_x(t)}, , \mathbf{y}_{\tau_y(t)})$.

  Estimate contact point $\mathbf{v} = \overline{\mathbf{X}}\xi + \mathbf{L}_x\overline{\mathbf{U}}\xi$.

  Calculate classifiers $\mathbf{g}_1 = 2\Sigma_x^{-1}\mathbf{L}_x\overline{\mathbf{U}}\xi$ and $\mathbf{g}_2 = 2\Sigma_y^{-1}\mathbf{L}_y\overline{\mathbf{W}}\xi$ with $c_i = \mathbf{g}_i^\top \mathbf{v}$.

  Evaluate probabilities $\mathbb{P}(\mathbf{g}_i) = p \cdot \mathbb{P}(\mathbf{g}_i^\top \mathbf{x} \geq c_i) + (1-p) \cdot \mathbb{P}(\mathbf{g}_i^\top \mathbf{y} \leq c_i)$.

  Use surface with higher probability to classify $\mathbf{z}_t$.

  Record the observed true class of $\mathbf{z}_t$.

  $t = t + 1$

end while

AjRC Algorithm

Algorithm 3 The steps of AjRC for classifying streaming data.

1: Initialize: $t = 1$ and previous data points $\mathbf{x}_0$ and $\mathbf{y}_0$
2: Solve the SDP formulation and obtain policy matrices $\overline{\mathbf{U}}$ and $\overline{\mathbf{W}}$.
3: **while** New streaming data point $\mathbf{z}_t$ is available **do**
4: Let $\xi_t = (1, \mathbf{x}_{\tau_x(t)}, , \mathbf{y}_{\tau_y(t)})$.
5: Estimate contact point $\mathbf{v} = \overline{\mathbf{X}}\xi + \mathbf{L}_x\overline{\mathbf{U}}\xi$.
6: Calculate classifiers $\mathbf{g}_1 = 2\Sigma_x^{-1}\mathbf{L}_x\overline{\mathbf{U}}\xi$ and $\mathbf{g}_2 = 2\Sigma_y^{-1}\mathbf{L}_y\overline{\mathbf{W}}\xi$ with $c_i = \mathbf{g}_i^\top\mathbf{v}$.
7: Evaluate probabilities $\mathbb{P}(\mathbf{g}_i) = p \cdot \mathbb{P}(\mathbf{g}_i^\top\mathbf{x} \geq c_i) + (1-p) \cdot \mathbb{P}(\mathbf{g}_i^\top\mathbf{y} \leq c_i)$.
8: Use surface with higher probability to classify $\mathbf{z}_t$.
9: Record the observed true class of $\mathbf{z}_t$.
10: $t = t + 1$
11: **end while**

AjRC Algorithm

Algorithm 4 The steps of AjRC for classifying streaming data.

Initialize: $t = 1$ and previous data points $\mathbf{x}_0$ and $\mathbf{y}_0$
Solve the SDP formulation and obtain policy matrices $\overline{\mathbf{U}}$ and $\overline{\mathbf{W}}$.
while New streaming data point $\mathbf{z}_t$ is available **do**
    Let $\xi_t = (1, \mathbf{x}_{\tau_x(t)}, , \mathbf{y}_{\tau_y(t)})$.
    Estimate contact point $\mathbf{v} = \overline{\mathbf{X}}\xi + \mathbf{L}_x\overline{\mathbf{U}}\xi$.
    Calculate classifiers $\mathbf{g}_1 = 2\Sigma_x^{-1}\mathbf{L}_x\overline{\mathbf{U}}\xi$ and $\mathbf{g}_2 = 2\Sigma_y^{-1}\mathbf{L}_y\overline{\mathbf{W}}\xi$ with $c_i = \mathbf{g}_i^\top \mathbf{v}$.
    Evaluate probabilities $\mathbb{P}(\mathbf{g}_i) = p \cdot \mathbb{P}(\mathbf{g}_i^\top \mathbf{x} \geq c_i) + (1-p) \cdot \mathbb{P}(\mathbf{g}_i^\top \mathbf{y} \leq c_i)$.
    Use surface with higher probability to classify $\mathbf{z}_t$.
    Record the observed true class of $\mathbf{z}_t$.
    $t = t + 1$
end while

AjRC Algorithm

Algorithm 5 The steps of AjRC for classifying streaming data.

Initialize: $t = 1$ and previous data points $\mathbf{x}_0$ and $\mathbf{y}_0$
Solve the SDP formulation and obtain policy matrices $\overline{\mathbf{U}}$ and $\overline{\mathbf{W}}$.
while New streaming data point $\mathbf{z}_t$ is available **do**
    Let $\xi_t = (1, \mathbf{x}_{\tau_x(t)}, , \mathbf{y}_{\tau_y(t)})$.
    Estimate contact point $\mathbf{v} = \overline{\mathbf{X}}\xi + \mathbf{L}_x\overline{\mathbf{U}}\xi$.
    Calculate classifiers $\mathbf{g}_1 = 2\Sigma_x^{-1}\mathbf{L}_x\overline{\mathbf{U}}\xi$ and $\mathbf{g}_2 = 2\Sigma_y^{-1}\mathbf{L}_y\overline{\mathbf{W}}\xi$ with $c_i = \mathbf{g}_i^\top \mathbf{v}$.
    Evaluate probabilities $\mathbb{P}(\mathbf{g}_i) = p \cdot \mathbb{P}(\mathbf{g}_i^\top \mathbf{x} \geq c_i) + (1 - p) \cdot \mathbb{P}(\mathbf{g}_i^\top \mathbf{y} \leq c_i)$.
    Use surface with higher probability to classify $\mathbf{z}_t$.
    Record the observed true class of $\mathbf{z}_t$.
    $t = t + 1$
end while

AjRC Algorithm

Algorithm 6 The steps of AjRC for classifying streaming data.

```
Initialize: $t = 1$ and previous data points $\mathbf{x}_0$ and $\mathbf{y}_0$
Solve the SDP formulation and obtain policy matrices $\overline{\mathbf{U}}$ and $\overline{\mathbf{W}}$.
while New streaming data point $\mathbf{z}_t$ is available do
    Let $\xi_t = (1, \mathbf{x}_{\tau_x(t)}, , \mathbf{y}_{\tau_y(t)})$.
    Estimate contact point $\mathbf{v} = \overline{\mathbf{X}}\xi + \mathbf{L}_x\overline{\mathbf{U}}\xi$.
    Calculate classifiers $\mathbf{g}_1 = 2\Sigma_x^{-1}\mathbf{L}_x\overline{\mathbf{U}}\xi$ and $\mathbf{g}_2 = 2\Sigma_y^{-1}\mathbf{L}_y\overline{\mathbf{W}}\xi$ with $c_i = \mathbf{g}_i^\top\mathbf{v}$.
    Evaluate probabilities $\mathbb{P}(\mathbf{g}_i) = p \cdot \mathbb{P}(\mathbf{g}_i^\top\mathbf{x} \geq c_i) + (1-p) \cdot \mathbb{P}(\mathbf{g}_i^\top\mathbf{y} \leq c_i)$.
    Use surface with higher probability to classify $\mathbf{z}_t$.
    Record the observed true class of $\mathbf{z}_t$.
    $t = t + 1$
end while
```

Experimental Setup

Visualization

$$\mathbf{x}_t = [-0.4, -0.4] + 0.2\,\mathbf{x}_{\tau_x(t)} + 0.05\,\mathbf{y}_{\tau_y(t)} + \epsilon_{x,t}$$
$$\mathbf{y}_t = [0, 0.15] + 0.05\,\mathbf{x}_{\tau_x(t)} + 0.2\,\mathbf{y}_{\tau_y(t)} + \epsilon_{y,t}.$$

Synthetic Data. 2 randomly generated time series with Gaussian errors at increasing distances.

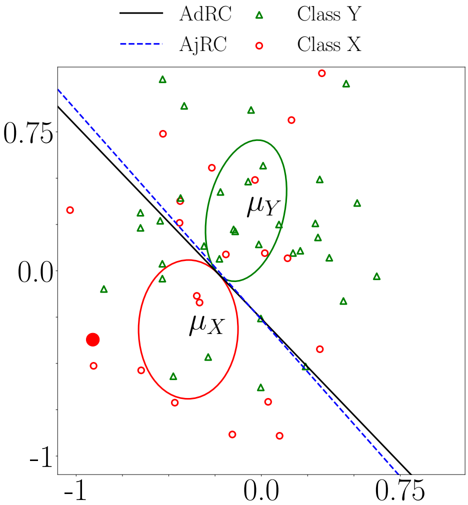
Experimental Setup

Visualization

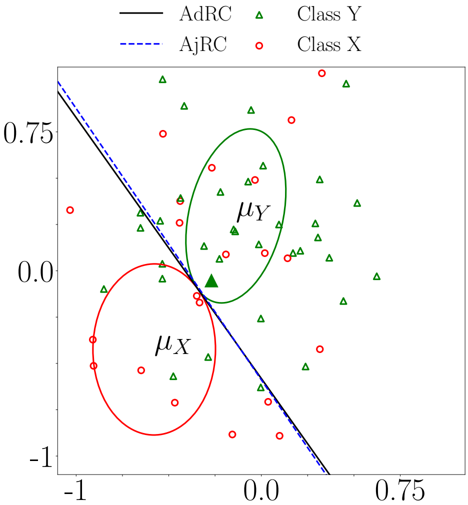
$$\mathbf{x}_t = [-0.4, -0.4] + 0.2\,\mathbf{x}_{\tau_x(t)} + 0.05\,\mathbf{y}_{\tau_y(t)} + \epsilon_{x,t}$$
$$\mathbf{y}_t = [0, 0.15] + 0.05\,\mathbf{x}_{\tau_x(t)} + 0.2\,\mathbf{y}_{\tau_y(t)} + \epsilon_{y,t}.$$

Synthetic Data. 2 randomly generated time series with Gaussian errors at increasing distances.

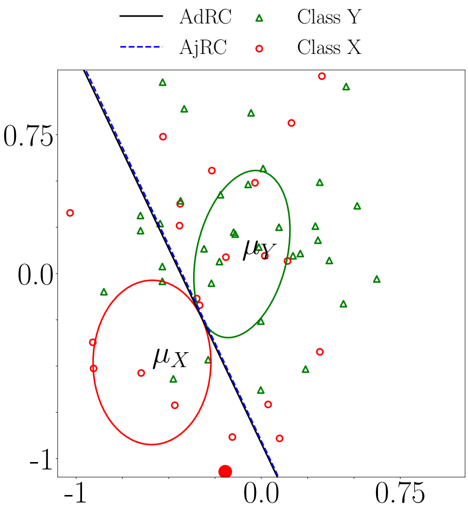
Visualization



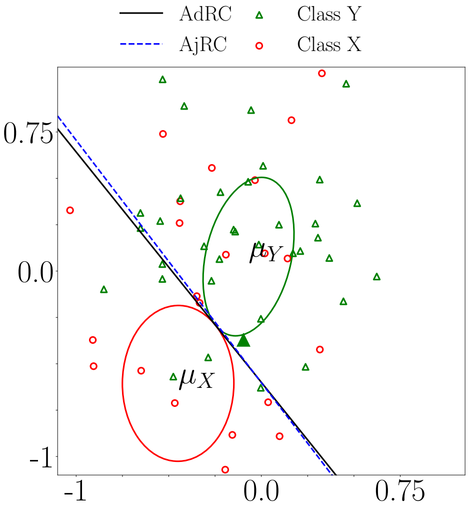
Visualization



Visualization



Visualization



Synthetic Data

Table: Comparison of the three data stream classifiers on randomly generated time series.

distance	AdRC Accuracy	Time [ms]	AjRC Accuracy	Time [ms]	SAMkNN Accuracy	Time [ms]
2	76.45	10.3	76.6	0.09	97.3	1.017
2.25	74.65	7.93	74.3	0.05	95.05	0.888
2.5	73.1	8.47	73.05	0.08	96.85	1.177
2.75	72.7	7.65	72.95	0.12	96.2	1.044
3	75.85	8.12	75.3	0.02	95.8	0.884
3.25	94.8	8.23	94.4	0.04	99.05	1.485
3.5	88.65	8.6	88.8	0.03	98.5	1.545
3.75	92.85	8.35	91.85	0.07	99.25	1.287
4	89.65	7.87	90.45	0.02	99.4	1.

Synthetic Data

Table: Comparison of the three data stream classifiers on randomly generated time series.

distance	AdRC Accuracy	Time [ms]	AjRC Accuracy	Time [ms]	SAMkNN Accuracy	Time [ms]
2	76.45	10.3	**76.6**	0.09	97.3	1.017
2.25	74.65	7.93	**74.3**	0.05	95.05	0.888
2.5	73.1	8.47	**73.05**	0.08	96.85	1.177
2.75	72.7	7.65	**72.95**	0.12	96.2	1.044
3	75.85	8.12	**75.3**	0.02	95.8	0.884
3.25	94.8	8.23	**94.4**	0.04	99.05	1.485
3.5	88.65	8.6	**88.8**	0.03	98.5	1.545
3.75	92.85	8.35	**91.85**	0.07	99.25	1.287
4	89.65	7.87	**90.45**	0.02	99.4	1.

Comparison to other methods

Evaluation of synthetic data on different classifiers.

Table: Comparison to Scikit Multiflow package.

distance	method	accuracy	kappa	time [ms]
3.5	AdRC	90.1	82.3	1.38
3.5	AjRC	88.8	77.4	0.026
3.5	KNN	95.7	91.4	0.378
3.5	HoeffdingTree	91.4	82.8	0.578
3.5	NaiveBayes	83.7	68.0	0.43
3.5	HATT	93.3	86.5	23.8

Comparison to other methods

Evaluation of synthetic data on different classifiers.

Table: Comparison to Scikit Multiflow package.

distance	method	accuracy	kappa	time [ms]
3.5	AdRC	90.1	82.3	1.38
3.5	**AjRC**	**88.8**	**77.4**	**0.026**
3.5	KNN	95.7	91.4	0.378
3.5	HoeffdingTree	91.4	82.8	0.578
3.5	NaiveBayes	83.7	68.0	0.43
3.5	HATT	93.3	86.5	23.8

Conclusion

1. We present an extension of the MPM model to classify streaming observations.
2. The affinely adjustable classifier directly embeds the time series into the optimization problem and leverages decision rules for adjusting the classifier to new observations.
3. We evaluate the performance of these models on numerical experiments and illustrate their benefits on synthetic data.

References

Anagnostopoulos, Christoforos et al. (2012). "Online linear and quadratic discriminant analysis with adaptive forgetting for streaming classification". In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 5.2, pp. 139–166.

Das, Monidipa et al. (2020). "A Skip-Connected Evolving Recurrent Neural Network for Data Stream Classification under Label Latency Scenario". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 04, pp. 3717–3724.

Jia, Xiaowei et al. (2017). "Incremental dual-memory lstm in land cover prediction". In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 867–876.

Krawczyk, Bartosz and Michał Woźniak (2015). "One-class classifiers with incremental learning and forgetting for data streams with concept drift". In: *Soft Computing* 19.12, pp. 3387–3400.

Ksieniewicz, Paweł et al. (2019). "Data stream classification using active learned neural networks". In: *Neurocomputing* 353, pp. 74–82.

References

- Kumagai, Atsutoshi and Tomoharu Iwata (2016). "Learning Future Classifiers without Additional Data.". In: *AAAI*, pp. 1772–1778.
- – (2017). "Learning Non-Linear Dynamics of Decision Boundaries for Maintaining Classification Performance.". In: *AAAI*, pp. 2117–2123.
- – (2018). "Learning Dynamics of Decision Boundaries without Additional Labeled Data". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, pp. 1627–1636.
- Lanckriet, Gert RG, Laurent El Ghaoui, and Michael I Jordan (2003). "Robust novelty detection with single-class MPM". In: *Advances in neural information processing systems*, pp. 929–936.
- Li, Hu et al. (2017). "Multi-window based ensemble learning for classification of imbalanced streaming data". In: *World Wide Web* 20.6, pp. 1507–1525.
- Lobo, Jesus L et al. (2020). "Spiking neural networks and online learning: An overview and perspectives". In: *Neural Networks* 121, pp. 88–100.

References

Nguyen, Hai-Long, Yew-Kwong Woon, and Wee-Keong Ng (2015). “A survey on data stream clustering and classification”. In: *Knowledge and information systems* 45.3, pp. 535–569.

Žliobaitė, Indrė et al. (2013). “Active learning with drifting streaming data”. In: *IEEE transactions on neural networks and learning systems* 25.1, pp. 27–39.

Wind Speed Classification

Table: Accuracy of meteorology data stream classification for wind speeds.

AutoCorr	AdRC Accuracy	Time [ms]	AjRC Accuracy	Time [ms]	SAMkNN Accuracy	Time [ms]
1	58.3878	0.253	76.07843	0.0349	88.061	2.399
2	58.91068	0.236	73.20261	0.0453	88.061	2.816
3	63.44227	0.257	71.15468	0.0418	88.061	1.846
4	64.18301	0.231	61.04575	0.0453	88.061	1.929
5	66.05664	0.248	74.16122	0.0440	88.061	1.483
6	67.40741	0.278	73.24619	0.0684	88.061	1.834
7	68.49673	0.298	68.23529	0.0697	88.061	1.829