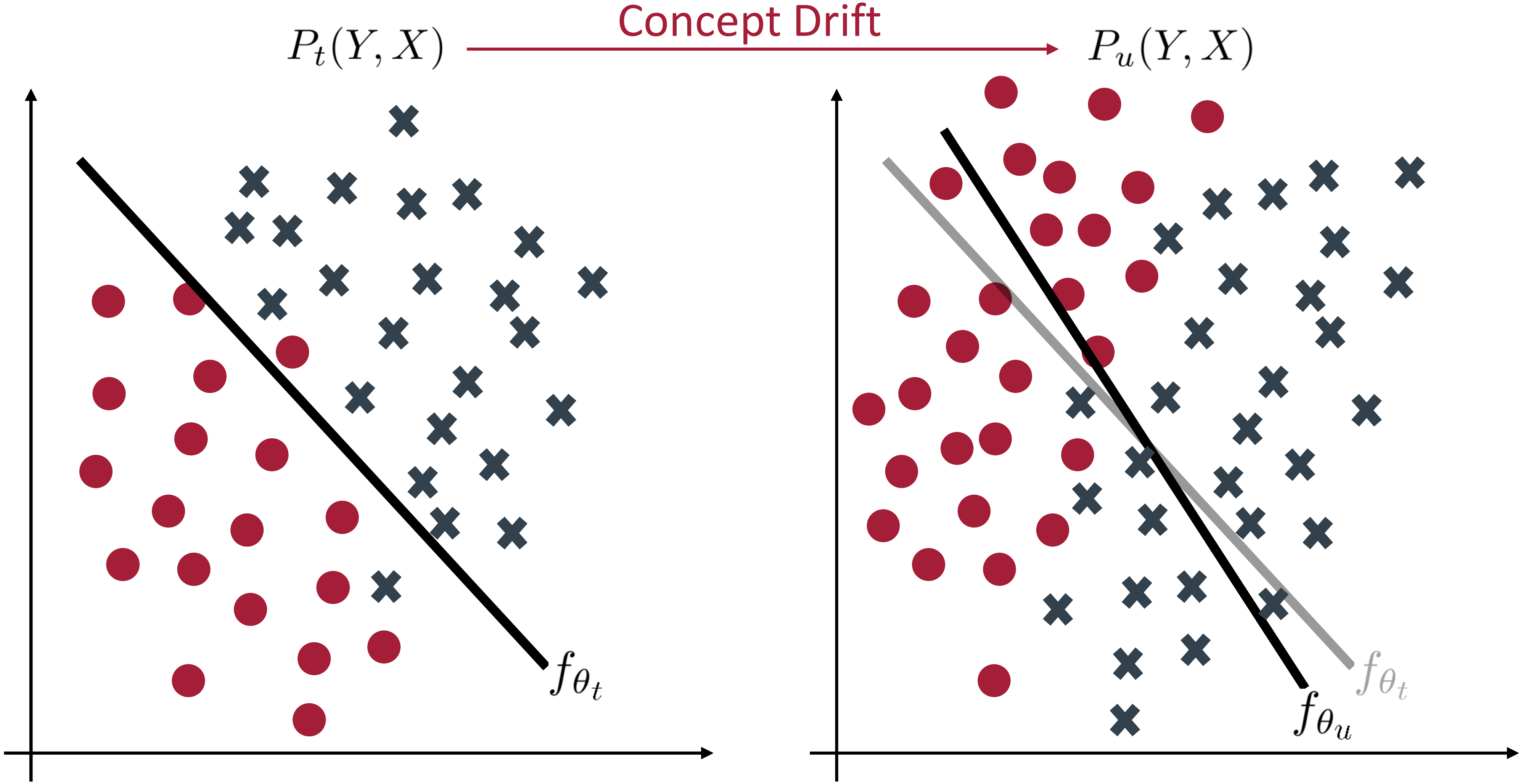
Learning Parameter Distributions to Detect Concept Drift in Data Streams

Johannes Haug and Gjergji Kasneci

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

25th International Conference on Pattern Recognition (ICPR 2020)

Motivation (Binary Classification at Time Step *t* and *u*)



Data Streams & Concept Drift

- Data streams are **potentially unbounded.**
- Data streams **underlie change** (e.g. due to changing customer behaviour).
- We call a distributional change between time step *t* and *u* **concept drift**, i.e.

$$P_t(Y,X) \neq P_u(Y,X)$$
$$\Leftrightarrow P_t(Y|X)P_t(X) \neq P_u(Y|X)P_u(X).$$

- Concept drift may significantly **reduce predictive performance**, if the predictive model *f* does not adapt (fast enough).

A Possible Solution: **Concept Drift Detection Models**

Requirements

Concept drift detection models should …

- … detect distributional change with a **short delay.**
- … produce **few false positives**, i.e. be **robust to random perturbations.**
- … be time/memory **efficient.**

Additionally, we propose that concept drift detection should …

- … consider changes of the **optimal model parameters** θ.

-> Property 1: Model-Aware Concept Drift Detection

- … should be **explainable**.

-> Property 2: Explainable Concept Drift Detection

The ERICS Model (I)

Intuitively, if we train f_{θ_t} at every time step *t,* we **optimize the parameters** θ_t **with respect to the active concept** $P_t(Y, X)$.

Hence, changes in the distribution (uncertainty) of $\theta \sim P(\theta;\psi)$ indicate **real concept drift.**

Taking inspiration from [1], we model concept drift as a **difference in the marginal likelihood** w.r.t ψ_t and ψ_u :

$$\left| \int P(Y|X,\theta)\big[P(\theta;\psi_t) - P(\theta;\psi_u)\big] \; d\theta \right| > 0.$$

[1] Haug, Johannes, et al. "Leveraging Model Inherent Variable Importance for Stable Online Feature Selection." Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020.

The ERICS Model (II)

We show that this difference of marginal likelihoods can be expressed in terms of the **differential entropy *h* and Kullback-Leibler divergence D_{KL}**:

$$\left| \underbrace{h[P(\theta;\psi_u)] - h[P(\theta;\psi_t)]}_{\Delta\text{Uncertainty}} + \underbrace{D_{KL}[P(\theta;\psi_u) \| P(\theta;\psi_t)]}_{\Delta\text{Distribution}} \right|$$

At every time step *t*, we compute the moving average MA_t of the above expression. We then **detect concept drift** as follows:

$$\sum_{j=t-W+1}^{t} \left(\text{MA}_j - \text{MA}_{j-1}\right) > \alpha_t \Leftrightarrow \text{Drift at } t$$

where $\alpha_t \geq 0$ is dynamically updated.

Experiments (I)

ERICS is **model-agnostic**.

For illustration, we specified …

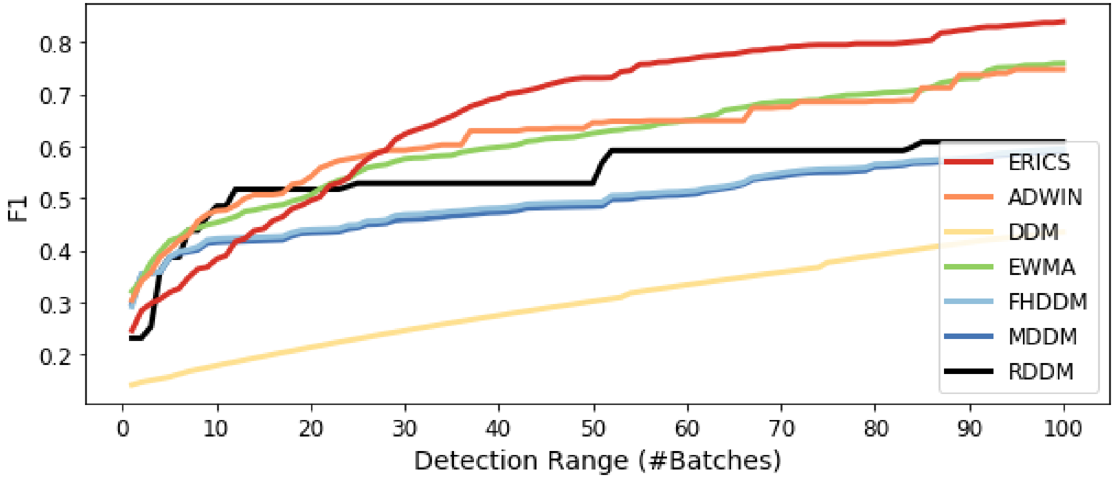
- … a **binary target** variable.
- … a Generalized Linear Model with a **Probit** link.
- … **normally distributed** model parameters, i.e. $P(\theta;\psi_t) = \mathcal{N}(\psi_t = (\mu_t, \Sigma_t))$.

Besides, we used **10 synthetic and real-world data sets** and compared our framework to **6 related concept drift detection models.**

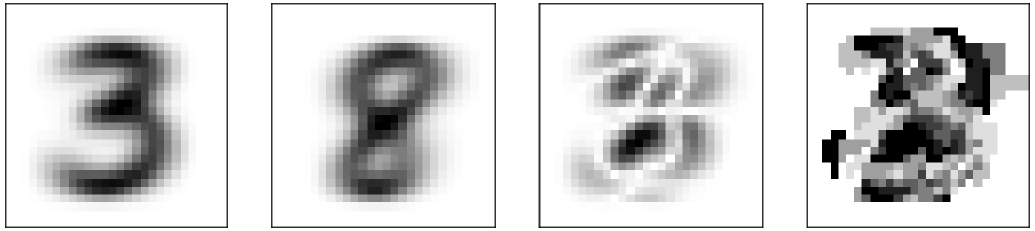
Experiments (II)

	Drift Detection Models						
Datasets	ERICS	ADWIN	DDM	EWMA	FHDDM	MDDM	RDDM
SEA	52.75	34.55	16.45	**0.26**	178.61	178.58	7.42
Agrawal	71.33	16.80	**0.00**	0.31	0.023	0.20	137.22
Hyperplane	**2.00**	42.27	2.23	2.36	11.24	11.22	2.42
Mixed	34.00	27.95	**0.34**	11.55	75.26	75.25	227.98
Spambase	**7.35**	79.0	60.23	29.96	71.63	71.58	117.45
Adult	**2.61**	63.15	488.39	172.57	488.39	488.39	488.39
HAR	13.05	372.45	372.45	**1.55**	372.45	372.45	77.10
KDD	22.01	50.35	**0.00**	0.55	100.25	100.21	13.21
Dota	44.04	25.32	514.72	43.40	3.56	**3.54**	116.46
Mean	**27.68**	79.09	161.65	29.17	144.60	144.60	131.96
Rank	**1**	3	7	2	5	5	4

Average delay in no. of batches.



We counted **drift alerts (TP, FP, FN)** w.r.t. **different detection ranges** (i.e. permitted delay in no. of batches).
We then computed the **F1 score** (i.e. the harmonic mean of precision and recall)



Before Drift ($y_{true} = 3$) After Drift ($y_{true} = 8$) Abs. Difference Detected Drift

Partial Drift Detection on MNIST (Explainability Property).

Summary

ERICS is a **generic concept drift detection framework** that …

- … can **exploit the parameters** of almost any predictive model.
- … provides **partial drift detection** at the input level.
- … **can outperform state-of-the-art models** w.r.t. both the delay and precision of concept drift alerts.

https://github.com/haugjo/erics

www.uni-tuebingen.de/dsar

johannes-christian.haug@uni-tuebingen.de