

EFFICIENT DISTRIBUTION-DERIVED FEATURES FOR HIGH-SPEED ENCRYPTED FLOW CLASSIFICATION

JOHAN GARCIA TOPI KORHONEN

DEPARTMENT OF COMPUTER SCIENCE
KARLSTAD UNIVERSITY, SWEDEN



PRESENTATION OUTLINE

- Problem formulation and specifics
- Distributional attributes
- The KSD approach for discretization
- Synthetic dataset evaluation
- Empirical dataset evaluation
- Conclusions and observations

Thanks to:

KK-stiftelsen



SANDVINE



Chalmers Centre for
Computational Science
and Engineering



SNIC



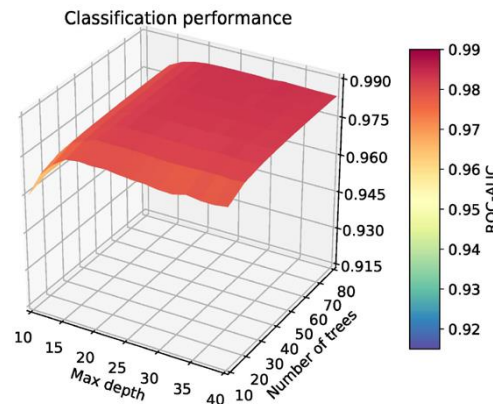
PROBLEM FORMULATION

- Flow classification is useful to ensure efficient network resource usage and support QoE
- Traffic is increasingly becoming encrypted by default
- Flow classification based on traditional Deep packet inspection (DPI) becomes unfeasible with encrypted flows
- Machine Learning on content-independent traffic characteristics can be used for classification of encrypted flows
- A subset of features used for classification are distribution-derived
- **Q: How can we best describe distribution-derived features?**

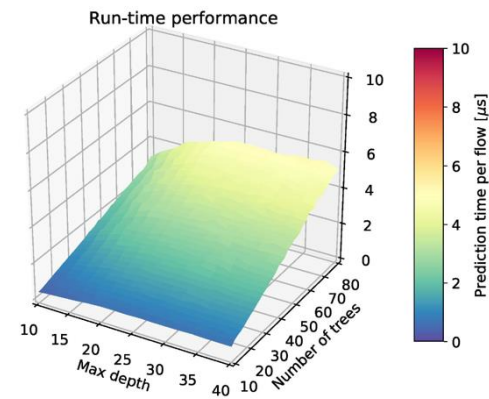
PROBLEM SPECIFICS

Target use case

- Flow level (i.e. 5-tuple) characterization, not session level
- Focus on early flow classification: ≤ 50 packets
- High speed: Up to 1 million flows per second in one box



RF classification perf. vs model complexity



Optimized RF runtime perf. vs model complexity

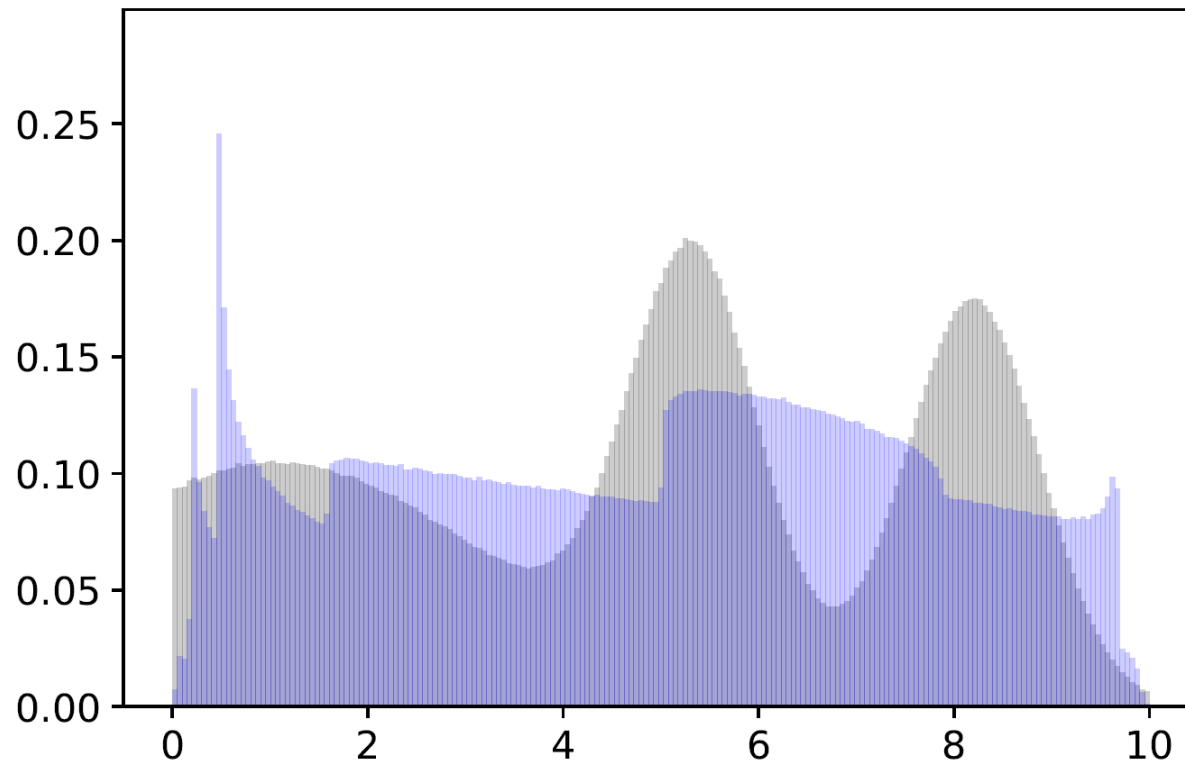
J Garcia, T Korhonen, R Andersson, F Västlund. Towards Video Flow Classification at One Million Encrypted Flows per Second. IEEE AINA 2018

Distributional attributes

DISTRIBUTIONAL ATTRIBUTES OF FLOWS

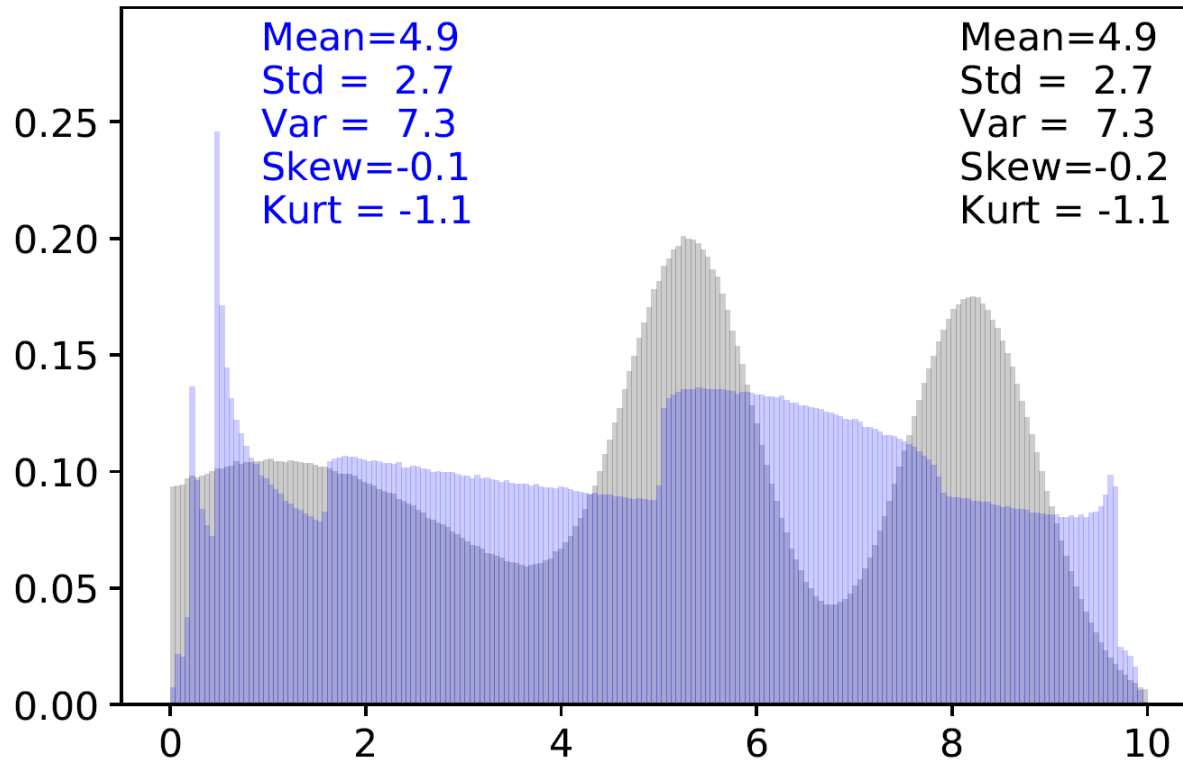
- Distributional attributes of N first packets of a flow:
 - Packet sizes
 - Interarrival times
 - Burst-lengths (in seconds and/or bytes)
 - Inter-burst lengths (in seconds)
- Distributional feature descriptors:
 - Basic: Min/mean/max
 - Moments-based: Standard deviation, variance, skew, kurtosis
 - Histogram based: Linear, Probabilistic, MDLP, or KSD discretization
- Bin-boundary placement, i.e. discretization, quantization, multi-splitting, ...
- Different discretization goals:
 - Encoding a scalar value
 - Describing a distribution
 - **Maximizing the discriminative power between two distributions**

DESCRIBING DISTRIBUTIONAL ATTRIBUTES



A mixture of Gaussian distribution (gray), and a mixture of Beta distributions (blue)

DESCRIBING DISTRIBUTIONAL ATTRIBUTES



A mixture of Gaussian distribution (gray), and a mixture of Beta distributions (blue)

STATISTICAL MOMENTS MAY NOT ALWAYS CAPTURE THE FULL DISTRIBUTIONAL DIFFERENCE

KSD

Kolmogorov-Smirnov

Discretization

KSD ALGORITHM EXAMPLE

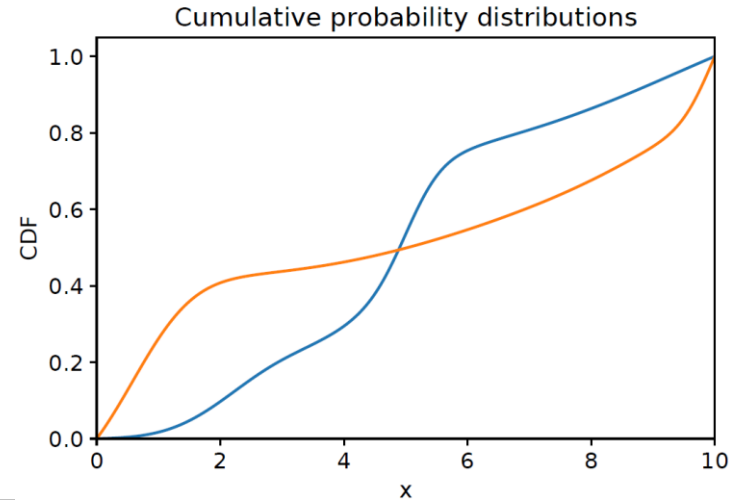
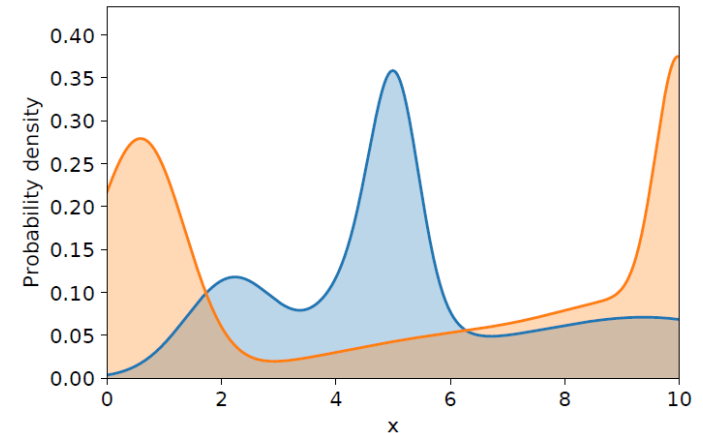
- PDF of two Gaussian mixtures
- CDF

o_i : i th observation

Indicator function: $\mathbf{I}(A) \begin{cases} 1 & A = \text{True} \\ 0 & A = \text{False} \end{cases}$

CDF: $\hat{F}_X(t) = n^{-1} \sum_{i=1}^n \mathbf{I}(o_i \leq t)$

$KS = \arg \max_t |\hat{F}_1(t) - \hat{F}_2(t)|$



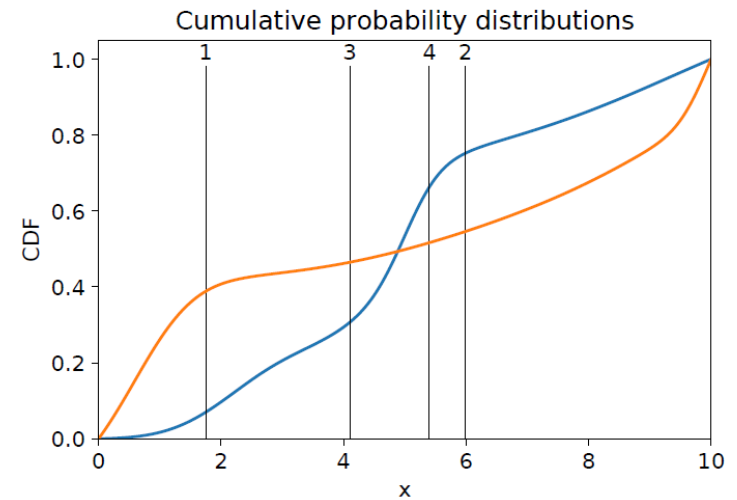
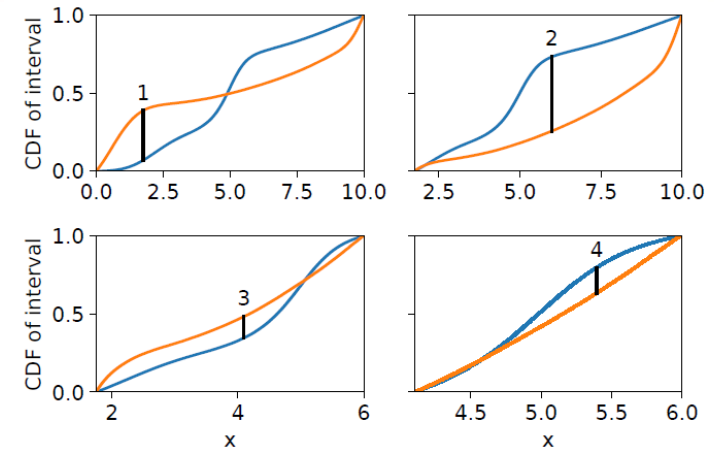
KSD ALGORITHM EXAMPLE

Algorithm 1 KSD bin placement algorithm - Bin limited variant

```

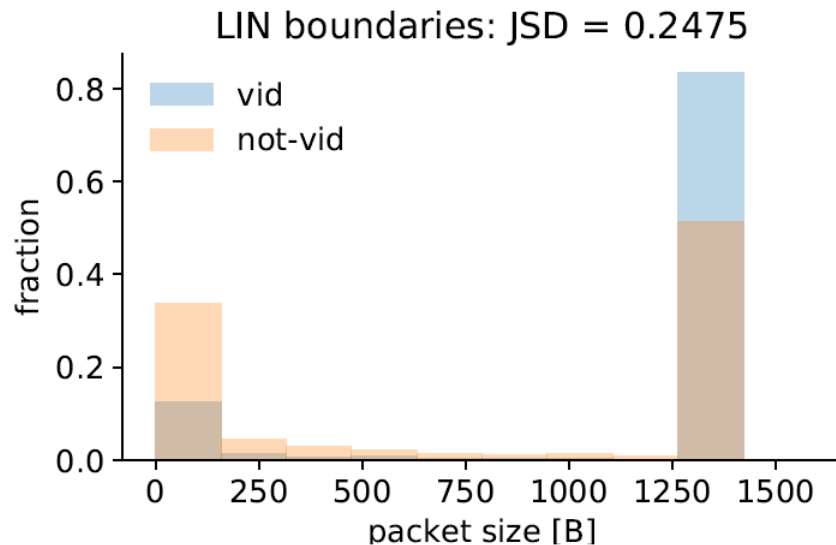
1:  $bins \leftarrow [a, b]$ 
2: while  $len(bins) < nbins + 1$  do
3:    $maxDif \leftarrow 0$ 
4:   for  $i$  in  $range(len(bins)-1)$  do
5:      $x_{min} \leftarrow bins[i]$ 
6:      $x_{max} \leftarrow bins[i + 1]$ 
7:      $o^c \leftarrow x_{min} \leq o < x_{max}$ 
8:      $\hat{F}^c(t) \leftarrow \frac{1}{n^c} \sum_{j=1}^{n^c} I(o_j^c < t)$ 
9:      $t_s \leftarrow \arg \max_t |\hat{F}_1^c(t) - \hat{F}_2^c(t)|$ 
10:     $f_l \leftarrow \left| \hat{F}_1^c(t_s) \frac{n_1^c}{N_1} - \hat{F}_2^c(t_s) \frac{n_2^c}{N_2} \right|$ 
11:     $f_r \leftarrow \left| (1 - \hat{F}_1^c(t_s)) \frac{n_1^c}{N_1} - (1 - \hat{F}_2^c(t_s)) \frac{n_2^c}{N_2} \right|$ 
12:    if  $maxDif < \max(f_l, f_r)$  then
13:       $maxDif \leftarrow \max(f_l, f_r)$ 
14:       $x_{add} \leftarrow t_s$ 
15:    end if
16:  end for
17:   $bins.append(x_{add})$ 
18:   $sort(bins)$ 
19: end while

```

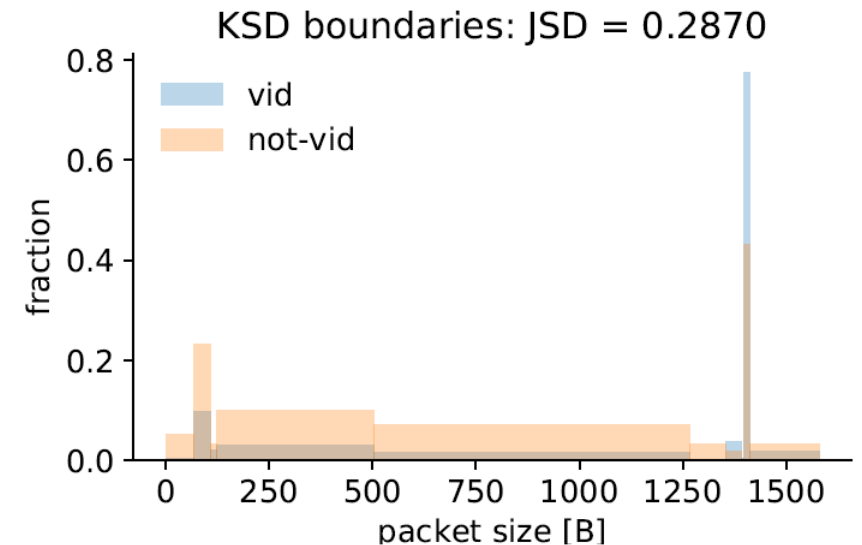


LINEAR VS KSD BINNING OF PACKET SIZE DISTRIBUTIONS

Linear binning of packet sizes:



KSD binning of packet sizes:



Synthetic evaluation



SYNTHETIC EVALUATION APPROACH

- Discretization: Linear, probabilistic, MDLP, KSD, KSD_NMDLP
- Distribution separation evaluation metric:
Jensen-Shannon distance, ~~Chi2, Kullback-Leibler divergence~~
- Random forest classification evaluation metric: ROC-AUC
- Number of runs for JSD (Random forest) evaluation:
1000 (200) Realizations of distribution mixtures
12 (5) instantiation of different nr of samples 12-5000 (10-100)

- Mixing:
 $M(x) = \sum_{i=1}^K w_i P_i(x)$ and $w = 1/K$ and $K = 2..5$

Gaussian distributions:

$$P(x) = \mathcal{N}(x|\mu, \sigma) \text{ with } \mu = U(0, 10) \text{ and } \sigma = U(0.1, 3)$$

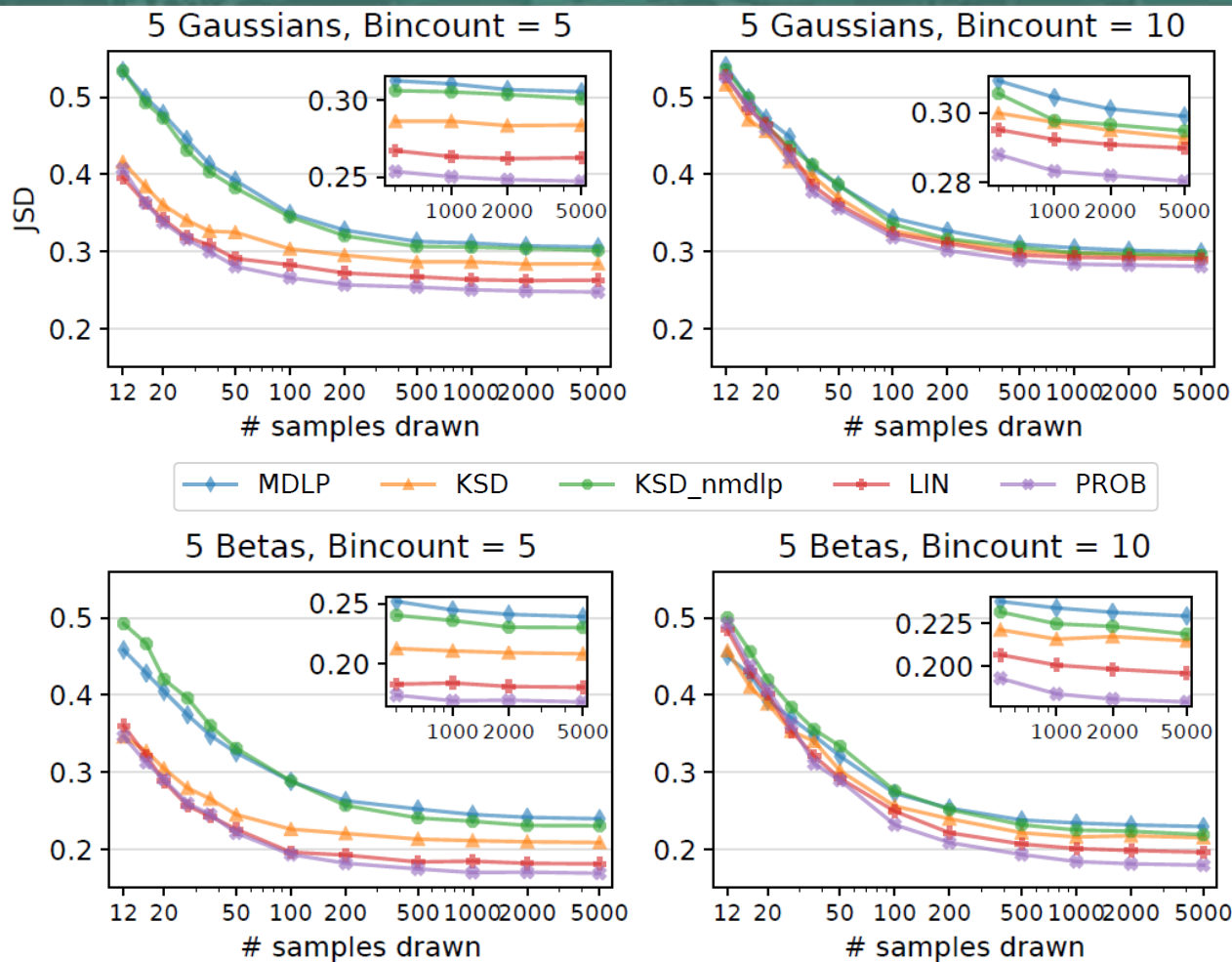
Beta distributions:

$$P(x) = \text{Beta}(x|\alpha, \beta, t, s) \text{ with shape } \alpha = \beta = U(0.5, 1.5), \\ \text{scaling } s = U(3, 10) \text{ and translation } t = U(0, 10 - s).$$



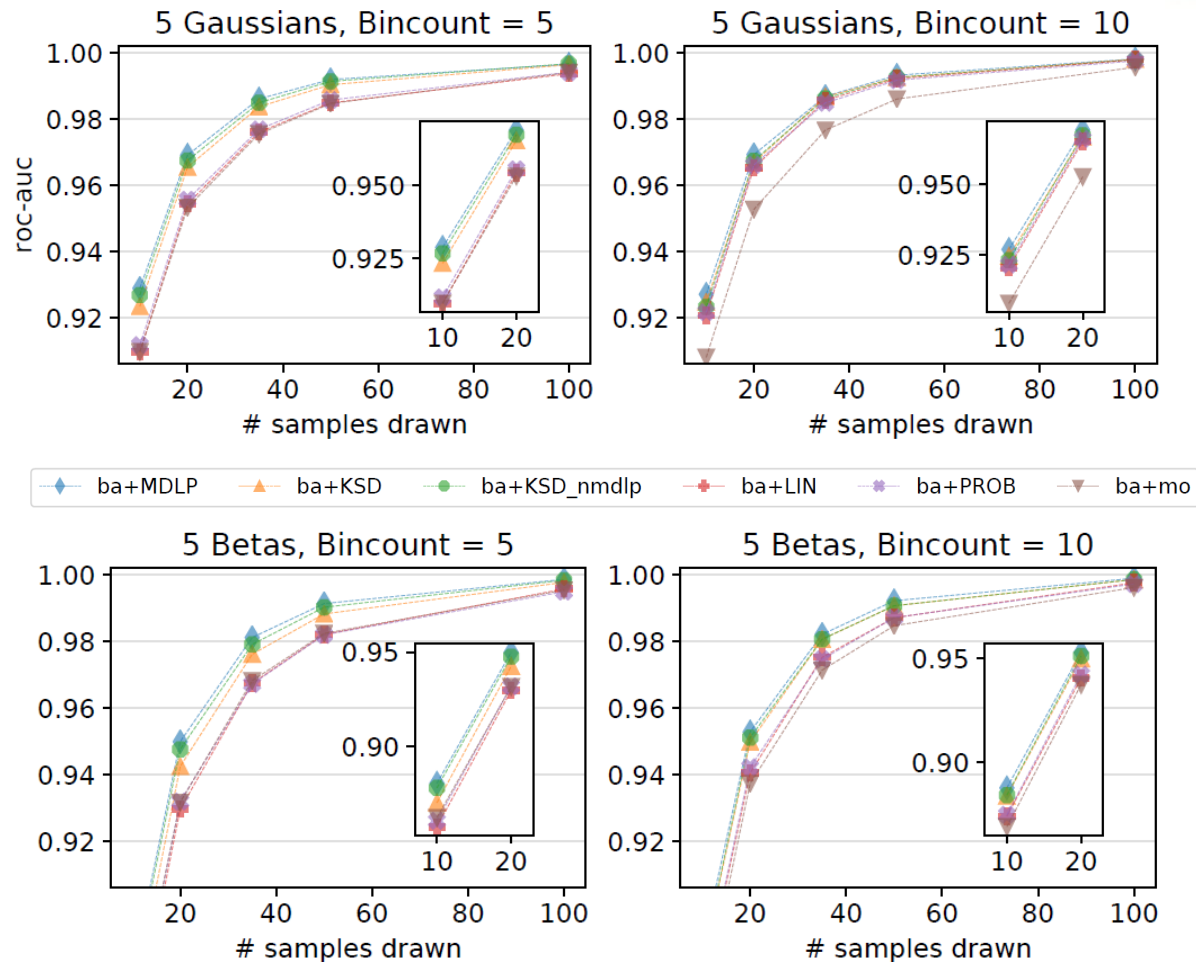
JENSEN-SHANNON DISTANCE OF DISCRETIZERS

- MDLP & KSD_NMDLP best (but have more bins)
- KSD better than LIN and PROB in most cases for same bin nr
- The more complex distribution (i.e Beta mixtures) gives larger difference



RANDOM FOREST CLASSIFICATION ON SYNTHETIC DATA

- More samples (packets) give better performance
- Ba+mo (moments) consistently bad
- More complex distributions give worse performance



Empirical evaluation



DATA COLLECTION

- Data collected by specially instrumented commercial DPI HW inside live cellular network during Feb 2017
- Per-packet data and flow classification labels (i.e ground-truth) collected for first 60 seconds of each flow
- 2.1B packets / 834M packets after filtering / 10M flows
- Set of Video and VoIP application labels provided by DPI vendor
- Per-flow features were computed based on this per-packet data



FEATURES USED IN EVALUATION

Feature label	Dir	Description	Group
nb	u/d	Total amount of Bytes in pkts	fa
np	d	Number of downlink packets	fa
fd		Time between first & last pkt	fa
nnfp		Nr of nonfull pkts, i.e., < 1400 B	fa
mean_	u/d	Mean of packet sizes/IAT	ba
min/max_	t/u/d	Min/Max of packet sizes/IAT	ba
std_/var_	u/d	Std dev/variance of pkt sizes/IAT	mo
skew_/kurt_	u/d	Skew/kurtosis of pkt sizes/IAT	mo
bn_M_N	u/d	N bins placed with method M	bn

- Four feature groups:

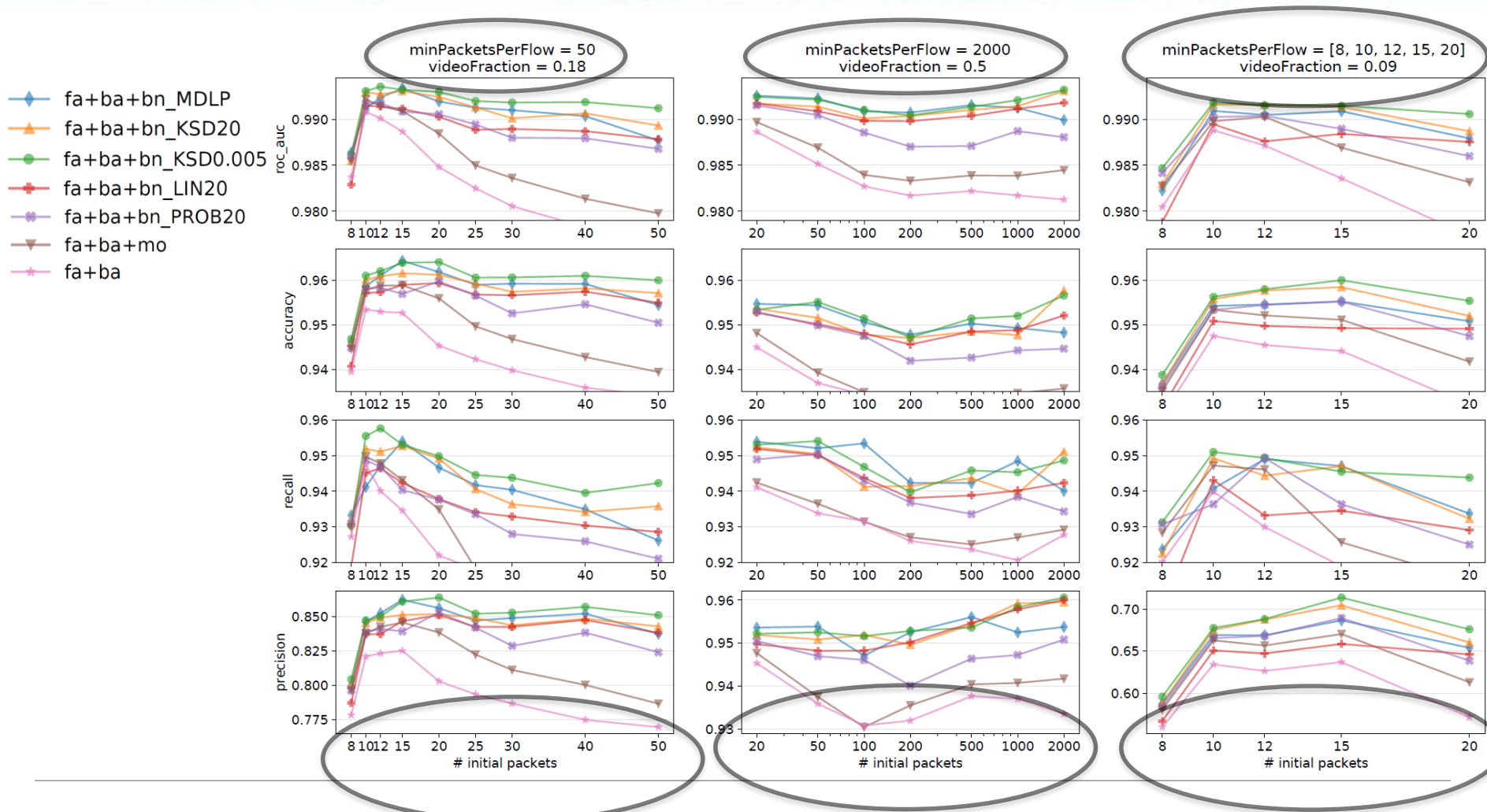
fa: Flow attributes – Non-distributional flow features

ba: Basic statistics – Basic distribution-derived features

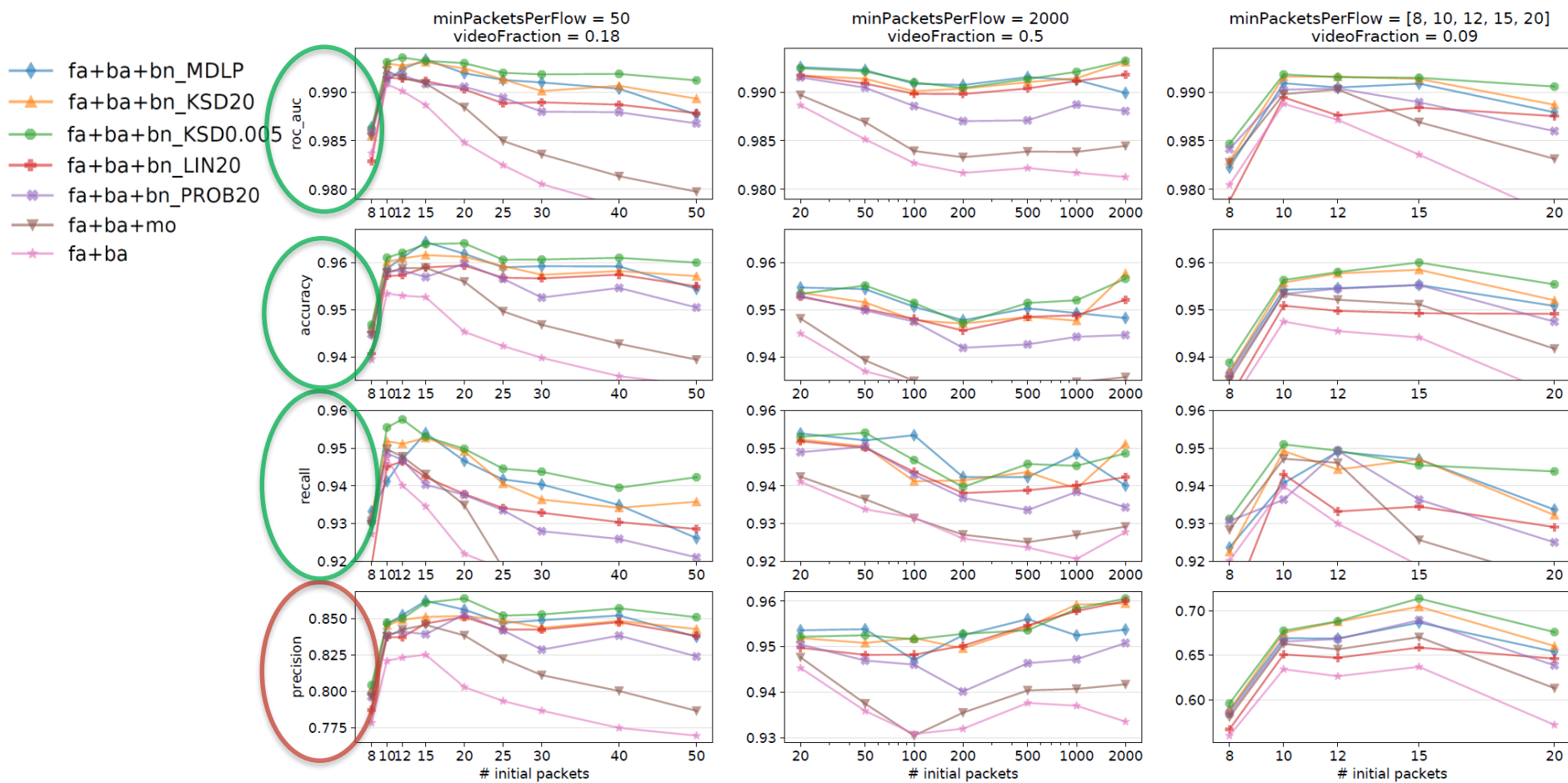
mo: Statistical moments – Extended distribution-derived features

bn: Histogram-based features – using a specific discretization method

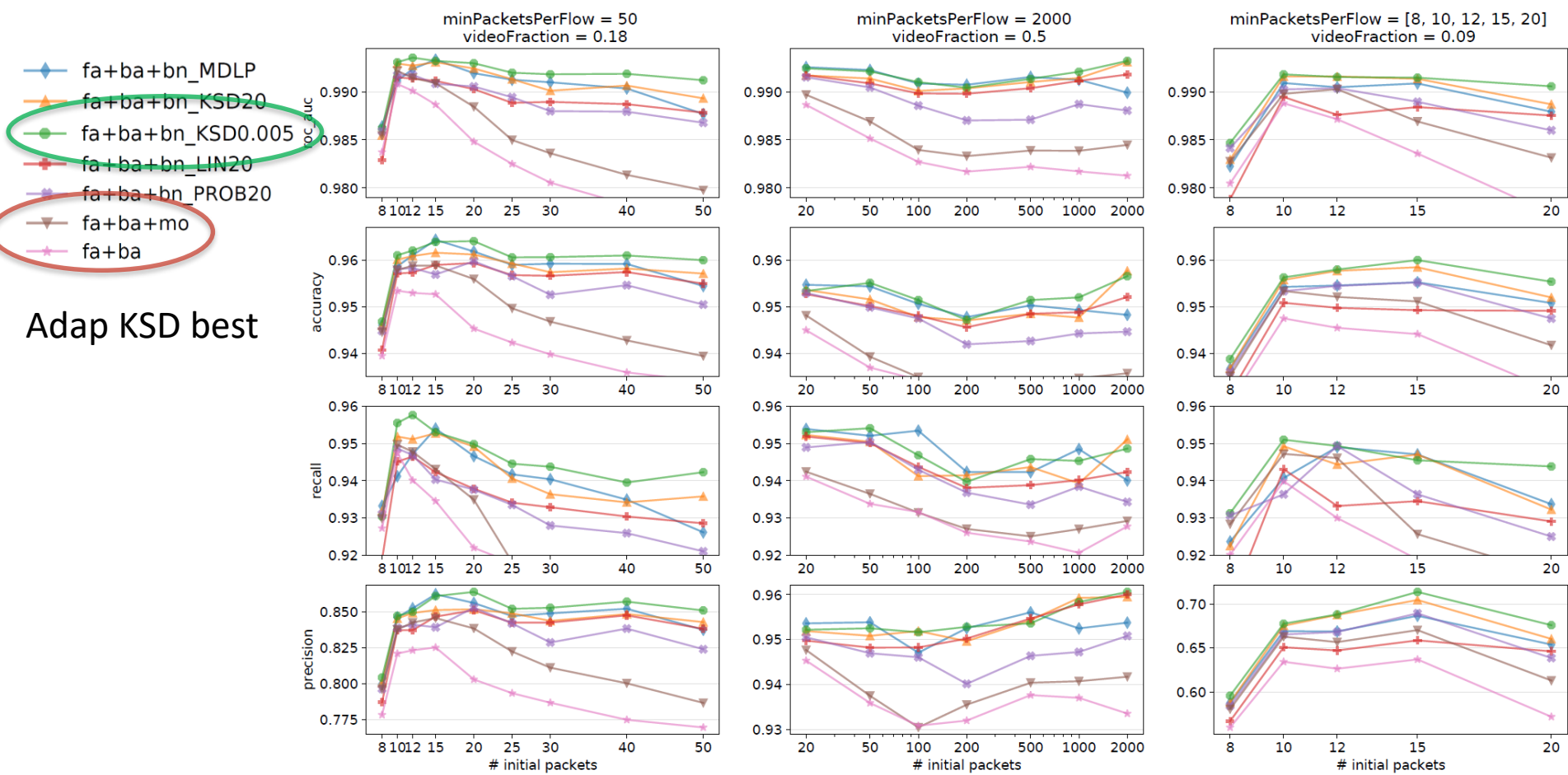
ACCURACY RESULTS



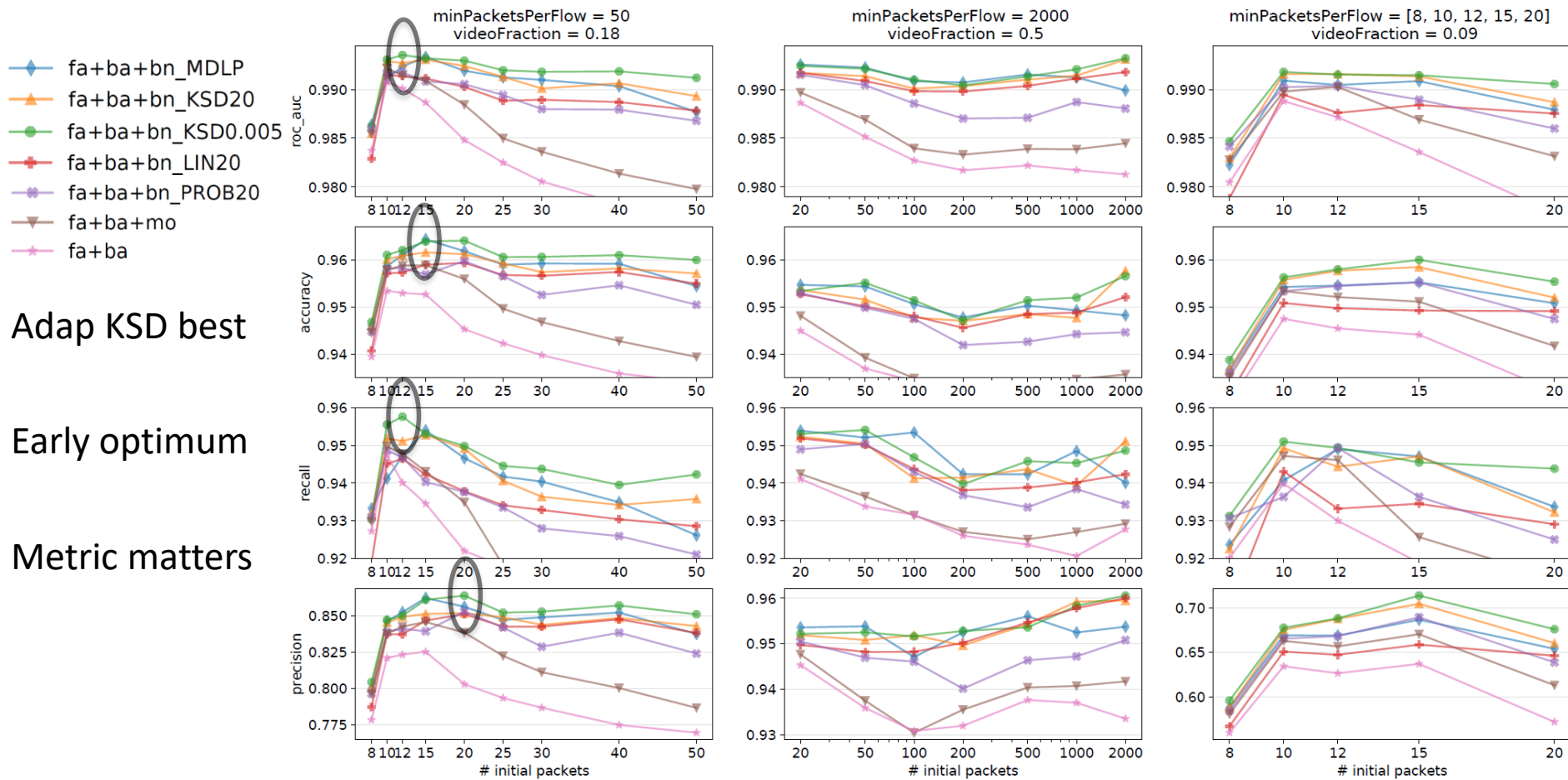
ACCURACY RESULTS



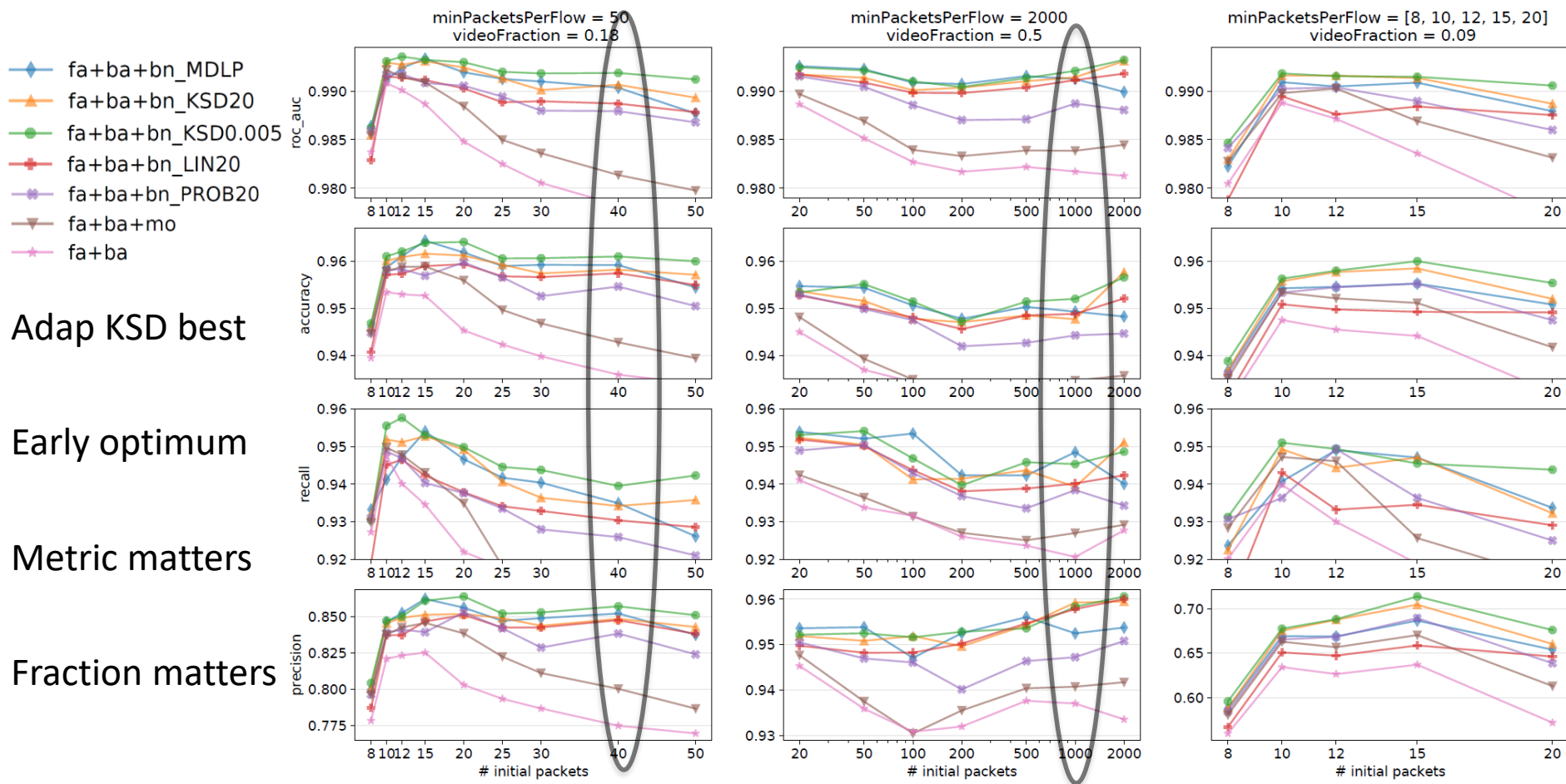
ACCURACY RESULTS



ACCURACY RESULTS



ACCURACY RESULTS



CONCLUSIONS AND OBSERVATIONS

- Histogram-based distribution-derived features improves on statistical moments by achieving:
 - Better classification performance
 - Better run-time performance, i.e. lower computational complexity
 - Allows for a flexible choice in the number of feature descriptors
- Among the evaluated histogram discretization approaches:
 - Adaptive KSD performs best with MDLP quite close
 - KSD is designed to allow a flexible number of bins, and has lower (offline) computational complexity
 - Linear and probabilistic discretization falter.
- Nr of initial packets have a noticeable impact on classification performance.
- JSD distance, simulated RForest, and empirical RForest differ (un)expectedly

