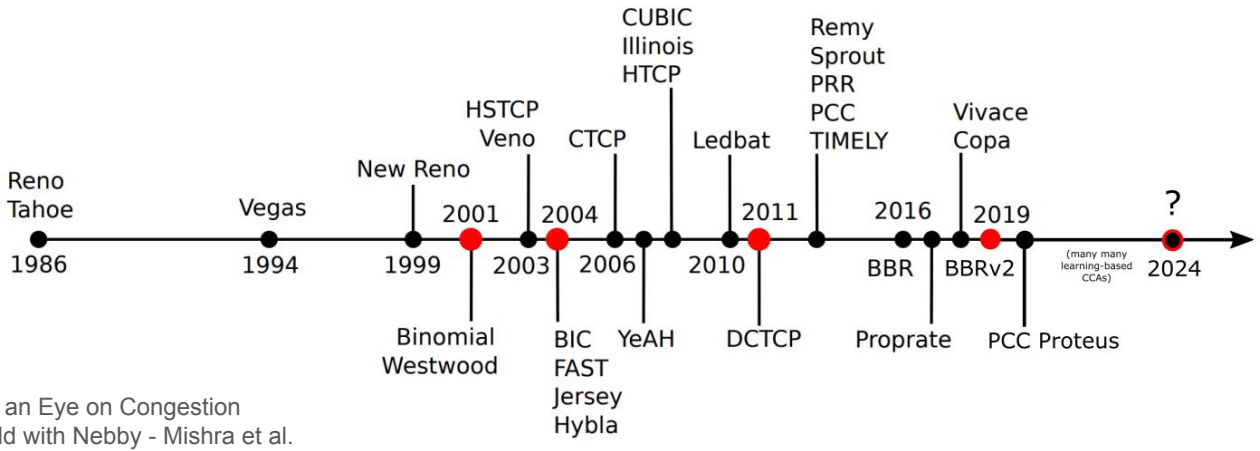
Time-series Clustering for CCA Classification

Michael Bryant, Manos Giannopoulos, Majed AlMunefi, David Zhao

Motivation - People Problem

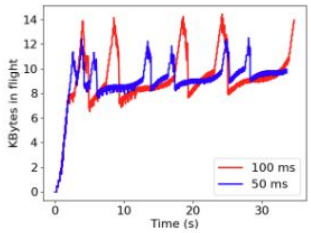
- Internet is rapidly evolving, new Congestion Control Algorithms (CCAs) are being deployed frequently.
- Easier than ever to deploy custom-made CCAs through QUIC.
- Need to capture the landscape of CCAs to ensure proper congestion control, fairness.



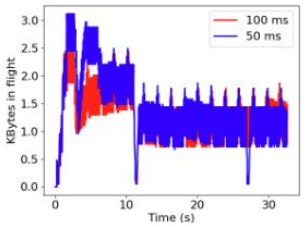
Source: Keeping an Eye on Congestion Control in the Wild with Nebby - Mishra et al.

Previous Work - Nebby

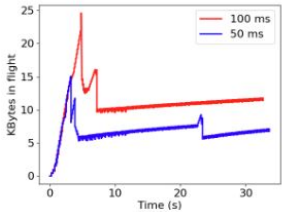
- Estimates the “Bytes in Flight” and creates a waveform.
- 2-step decision tree - checks for BBR, if not, fit a polynomial and classify as loss-based.
- Can plug custom-made classifiers for new CCAs



(a) CUBIC



(c) BBRv1



(g) New Reno

Source: Keeping an Eye on Congestion Control in the Wild with Nebby - Mishra et al.

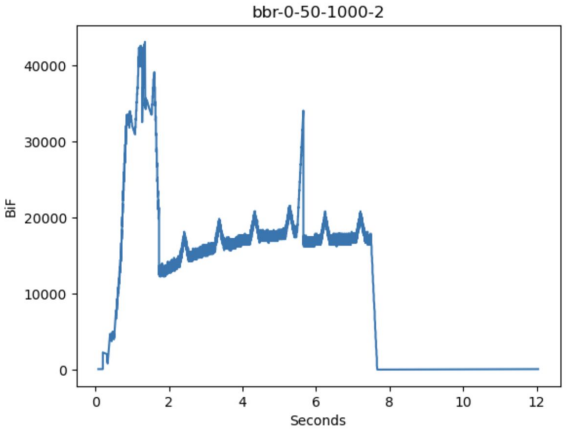
Technical Problems - Nebby

- Training on control vs deployment on real data:
  - Control data is used for training and accuracy testing.
  - Deployment on real data for inference, assuming that the classifier is correct. Distribution shift!
  - Small-scale verification by asking the website owners or crawling tech posts.

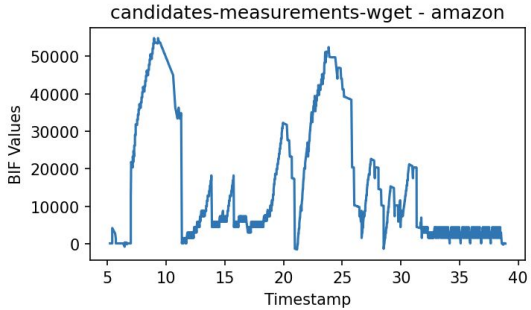
Authors need to implement the algorithms themselves to get labeled data.

We cannot evaluate the performance of Nebby on the actual data.

Source: Keeping an Eye on Congestion Control in the Wild with Nebby - Mishra et al.



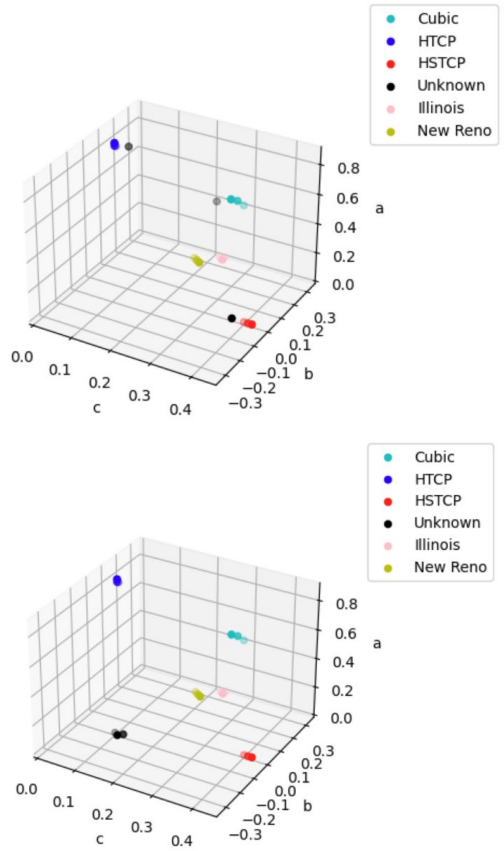
Control BBR



BBR in the wild

Technical Problems - Nebby

- Inability to characterize unknown variants:
  - Nebby will classify each BiF flow as one of the known algorithms or “Unknown”.
  - It does not provide any information about unknown flows.



Authors need to manually inspect unknown flows to realize if they need to plug-in a new classifier. Not robust to small deviations.

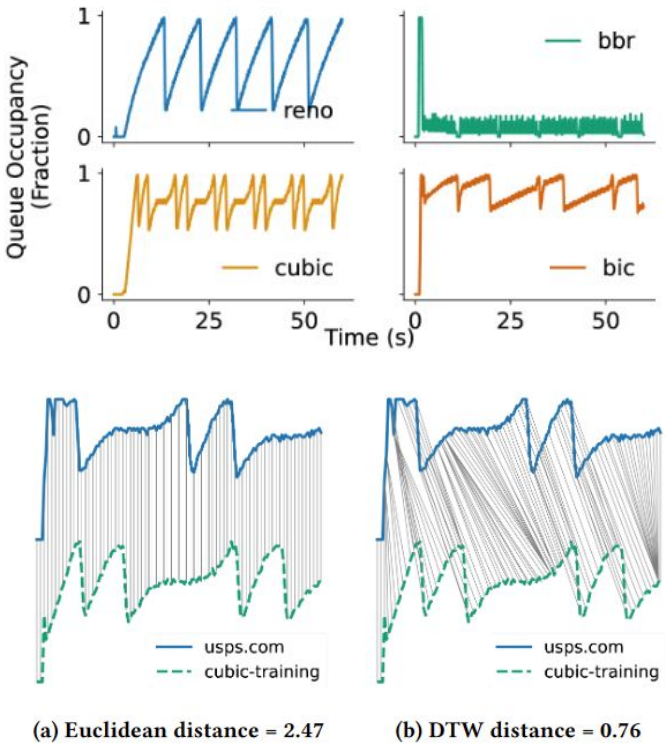
Source: Keeping an Eye on Congestion Control in the Wild with Nebby - Mishra et al.

Previous Work - CCAnalyzer

- Uses queue occupancy measurements to construct a time-series.
- Uses Distance-Based Clustering based on DTW measure.

\+ Can handle unknown CCAs and cluster them efficiently

but…



Source: CCAnalyzer: An Efficient and Nearly-Passive Congestion Control Classifier - Ware et al.

Technical Problems - CCAnalyzer

- Distance-based Clustering on Raw time-series.
- Each time series is essentially a 200-dimensional vector - “Curse of Dimensionality”

CCAnalyzer fails to classify ~70% of the top 10k websites.

CDN	BBR	BIC	CDG	CUBIC	Highspeed	HTCP	NV	Reno	Vegas	Westwood	Yeah	Unknown	All Invalid	RTT > 85ms	Unresponsive	Total
Akamai	470/491	0	3/4	4	0	0	0	0	0	0	0	115/91	189	36	233	1050
Cloudflare	1233/1595	0	6/7	5/6	0	0	0	1	0	0	0	824/460	394	55	989	3507
Cloudfront	530/545	0	9/10	7/10	0	0	3/2	0	0	0	0	74/56	78	10	121	832
Fastly	21/25	1/13	3	25/130	0	0	0/1	0	0	0	0	174/52	26	3	30	283
Google	29	0	1	2	0	0	2	0	0	0	1	230	37	18	66	386
Other CDN	28/32	2/0	1	53/92	0	0	1	0	0	0	0	72/31	54	41	226	478
No CDN	116/122	3/0	8/9	89/116	3/5	2	5	4/3	1/0	3	0	146/115	127	1205	1752	3464
Total	2427/2839	6/13	31/35	185/360	3/5	2	11	5/4	1/0	3	1	1635/1035	905	1368	3417	10000

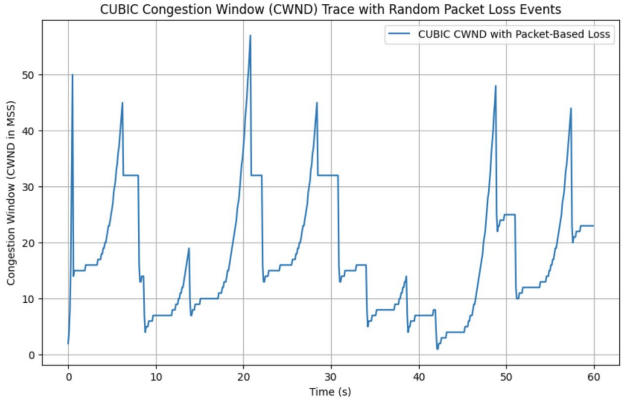
Source: CCAnalyzer: An Efficient and Nearly-Passive Congestion Control Classifier - Ware et al.

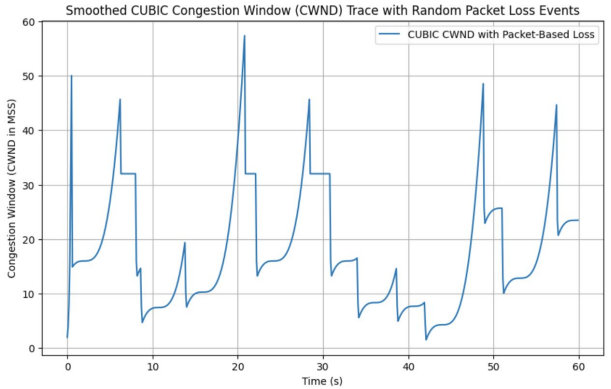
Proposal

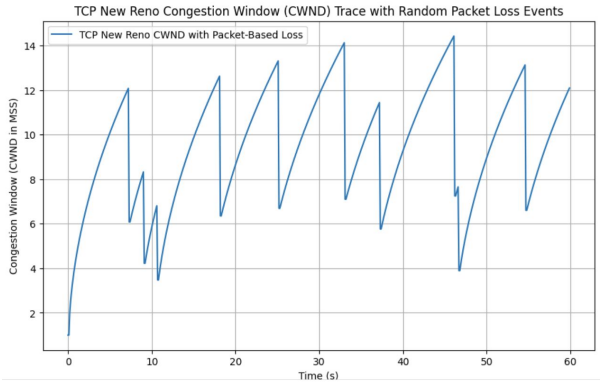
- Treat CCA identification as an unsupervised clustering problem.
- Use time-series clustering with Dynamic Time Warping (DTW):
  - Aligns time-series data with varying lengths and timing.
  - Groups CCAs based on similarity.
  - Able to group unknown and known CCAs without hard assignment of an algorithmic label.
- Introduce a latent space encoder: [Future Work - Work in Progress]
  - Maps Bytes-in-Flight (BiF) waveforms into a feature-rich low-dimensional latent space.
  - Visualizes and preserves differences between similar CCAs (e.g., customized variants).

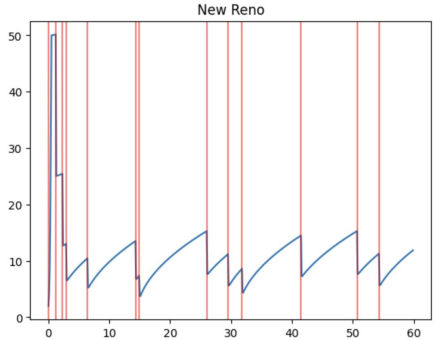
Methodology - Control Data Collection

- Simple ns-3 simulations for supported algorithms with stochastic fixed rate packet loss.
- Smoothing to prepare the data for polynomial fitting methods.
- 500 60-second length simulations for each algorithm under different Bandwidth, Delay and Packet Loss Percentage






TCP New Reno Congestion Window (CWND) Trace with Random Packet Loss Events
TCP New Reno CWND with Packet-Based Loss
Congestion Window (CWND in MSS)
Time (s)
0
10
20
30
40
50
60
2
4
6
8
10
12
14

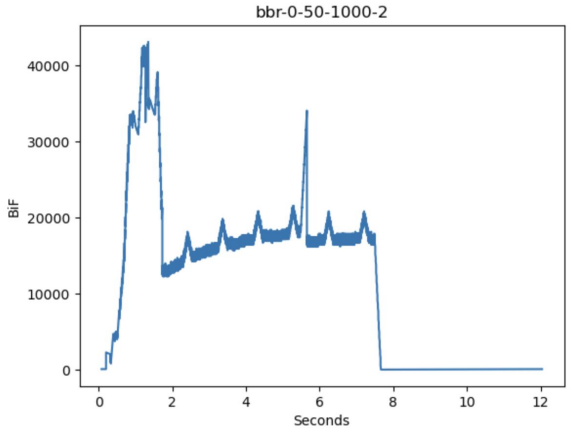

New Reno
0
10
20
30
40
50
60

Clustering: Loss Based vs Non Loss Based CCAs

Loss Based CCAs: congestion detection occurs when packets are dropped, therefore packet drops can be used an an indicator for congestion
Non-loss Based (BBR): Uses bandwidth probing and RTT_min estimation to infer congestion

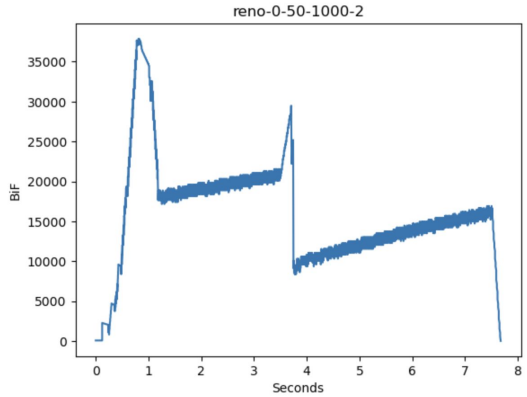
Non-Loss Based

BBR



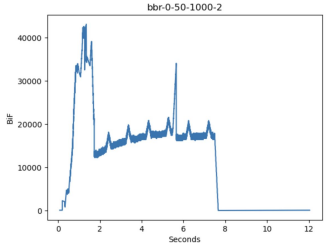
Loss Based

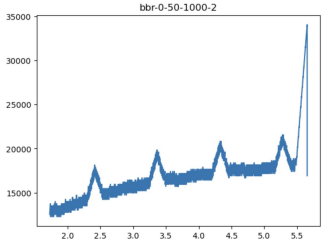
CUBIC, Reno, Vegas, Westwood, YeAH, etc.

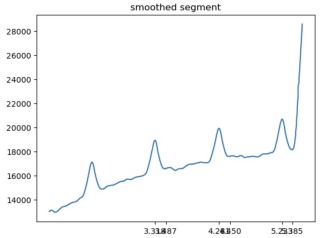


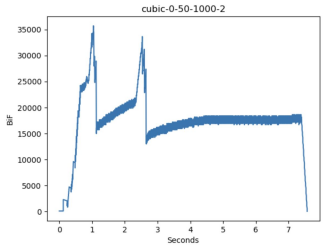
Clustering: Feature Engineering

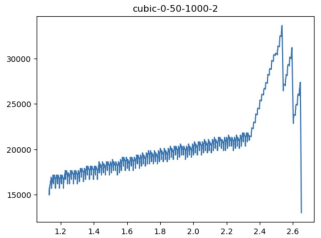
Segmentation/Smoothing

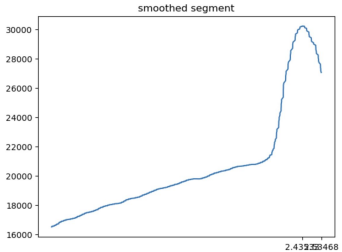


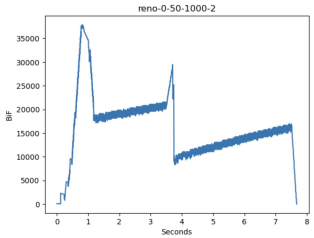


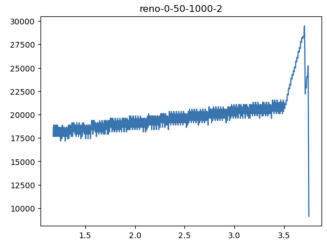


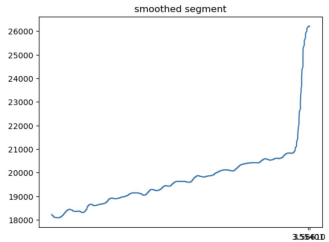






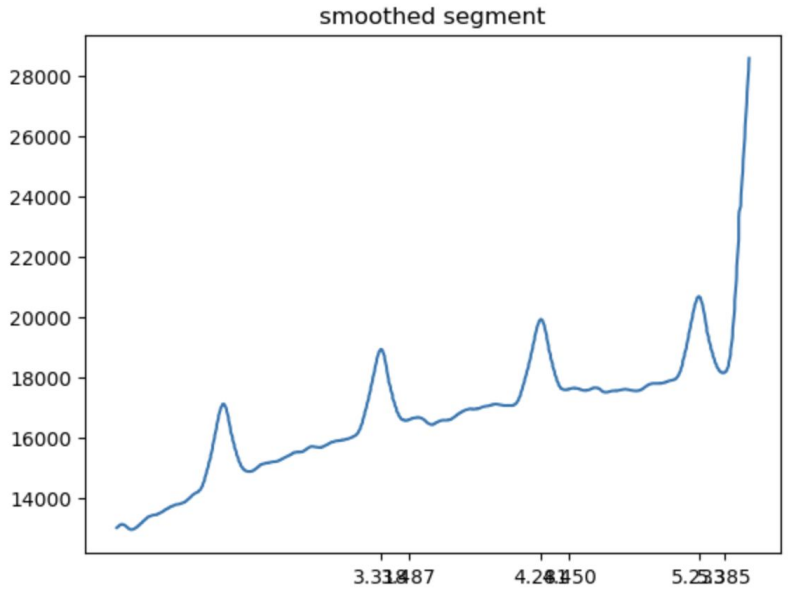




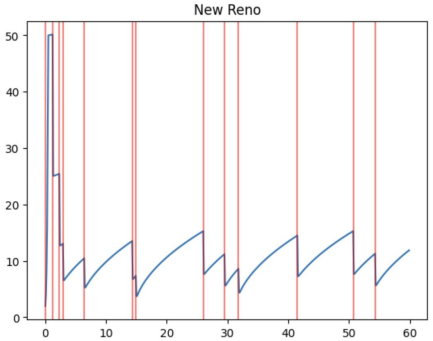


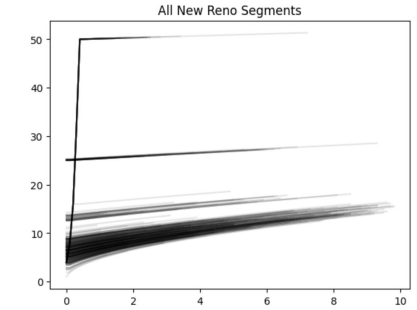
BBR Classification

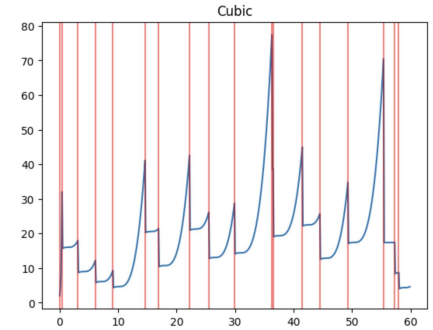
- Low variance for extended period of time
- Periodicity within segment
- (Nebby 2024)-back off every 10 seconds, increases sending rate by 25% every 8 RTTs (BBR v1)
- BBR v2: back off every 5 seconds
- **Results**
- Classified 37/77 websites as BBR (akamai-hulu, amazon, prime video, youtube, tiktok)

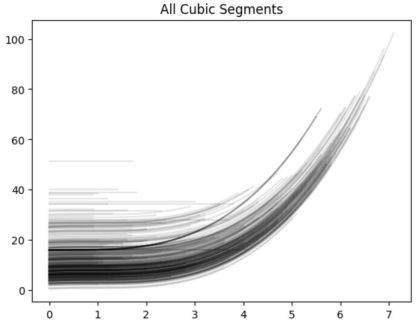


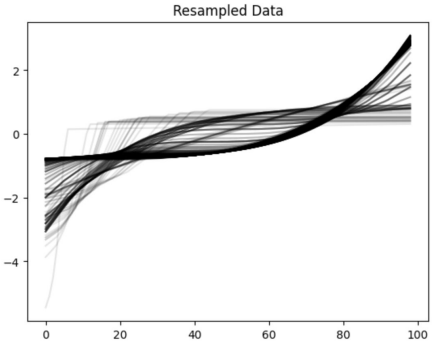
Time Series Clustering with DTW & Time Series KMeans

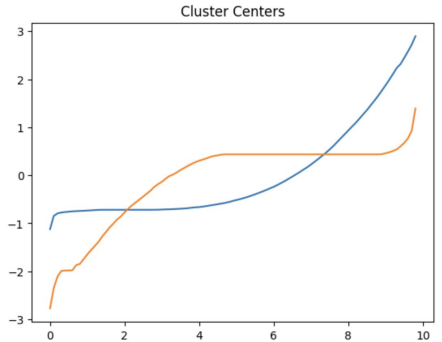




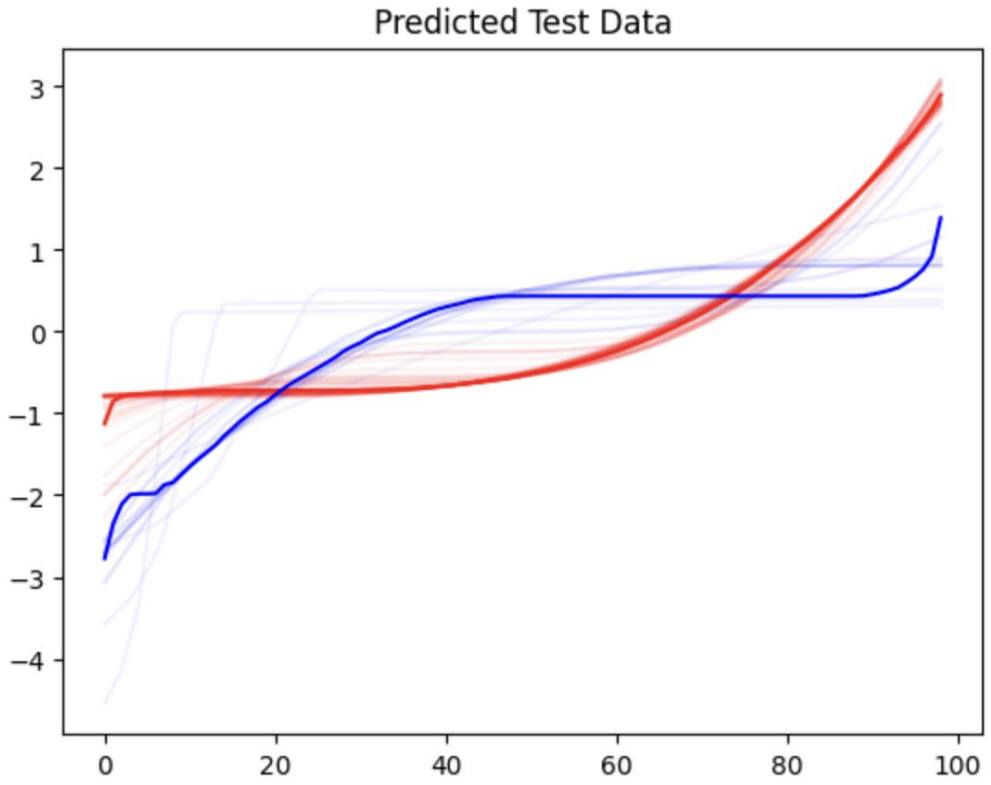






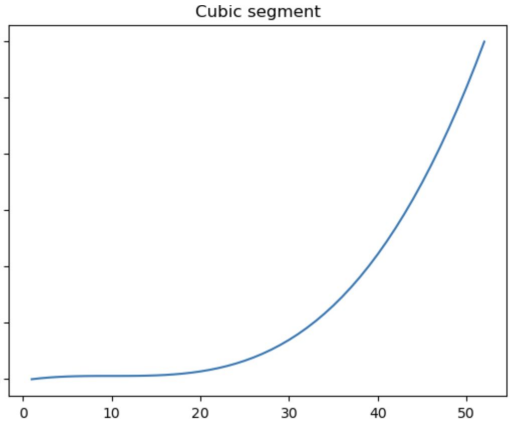


Time Series Clustering with DTW & Time Series KMeans

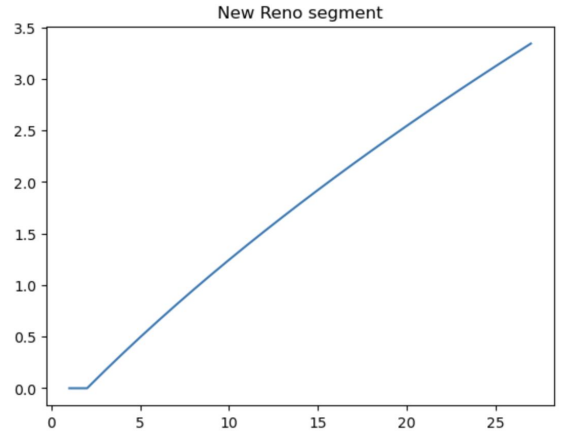



```
Accuracy: 0.7014925373134329
```

Clustering with Polynomial Fitting



[0.0004 -0.012 0.12 -0.1084]



[-1.46555840e-05 -3.69284022e-04 1.53081252e-01 -2.45174914e-01]

Correct classifications: 683
False Cubic: 19
False Reno: 86
Number of segments: 788
Accuracy: 0.866751269035533

Segments -> Flow

- If all segments are unknown or one segment is incorrect, the classification is unknown for the flow
- If the classifications for all segments are unknown or a single classification, and there is at least one non-unknown classification, the flow is classified as the non-unknown classification

[Future Work -WIP]

Challenges

- Difficult to Run Nebby's Source Code:
  - Unclear Documentation
  - Dependencies and difficulties to run locally
- Incomplete and Low Quality Data:
  - Hard to navigate and understand
  - Lack of certainty of CCA
  - CCAs have similar shapes
- No concrete problem definition:
  - Do we want to hard-label each CCA?
  - What happens with custom ones (e.g. Copa, PCC Vivace)? How about unlabeled ones in the wild?
  - If we want to characterize instead of hard-label, what information do we want?

References

- Ayush Mishra, Lakshay Rastogi, Raj Joshi, and Ben Leong. 2024. Keeping an Eye on Congestion Control in the Wild with Nebby . In ACM SIGCOMM 2024 Conference (ACM SIGCOMM '24), August 4–8, 2024, Sydney, NSW, Australia. ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3651890.3672223

- A. Mishra, X. Sun, A. Jain, S. Pande, R. Joshi, and B. Leong, "The great internet tcp congestion control census," Proc. ACM Meas. Anal. Comput. Syst., vol. 3, no. 3, Dec. 2019. [Online]. Available: https://doi.org/10.1145/3366693

- N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "Bbr:Congestion-based congestion control," ACM Queue, vol. 14, September-October, pp. 20– 53, 2016. [Online]. Available: http://queue.acm.org/detail.cfm?id=3022184

- Ranysha Ware, Adithya Abraham Philip, Nicholas Hungria, Yash Kothari, Justine Sherry, and Srinivasan Seshan. 2024. CCAnalyzer: An Efficient and Nearly-Passive Congestion Control Classifier. In Proceedings of the ACM SIGCOMM 2024 Conference (ACM SIGCOMM '24). Association for Computing Machinery, New York, NY, USA, 181–196. https://doi.org/10.1145/3651890.3672255