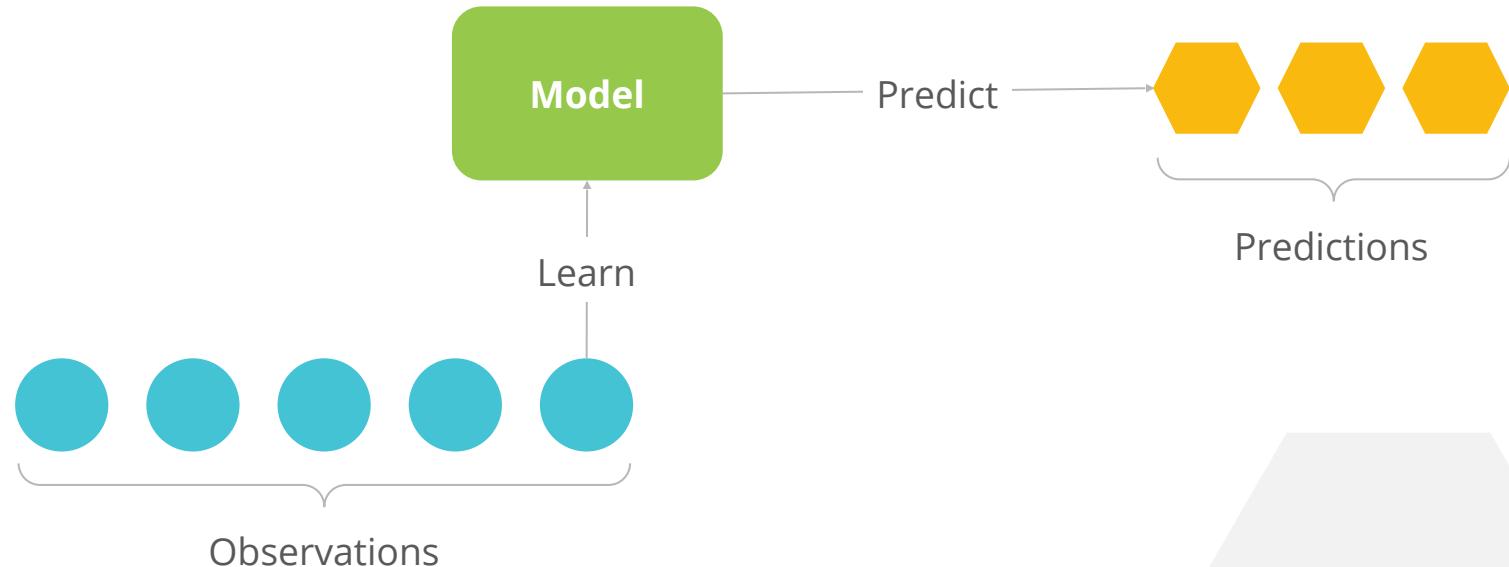




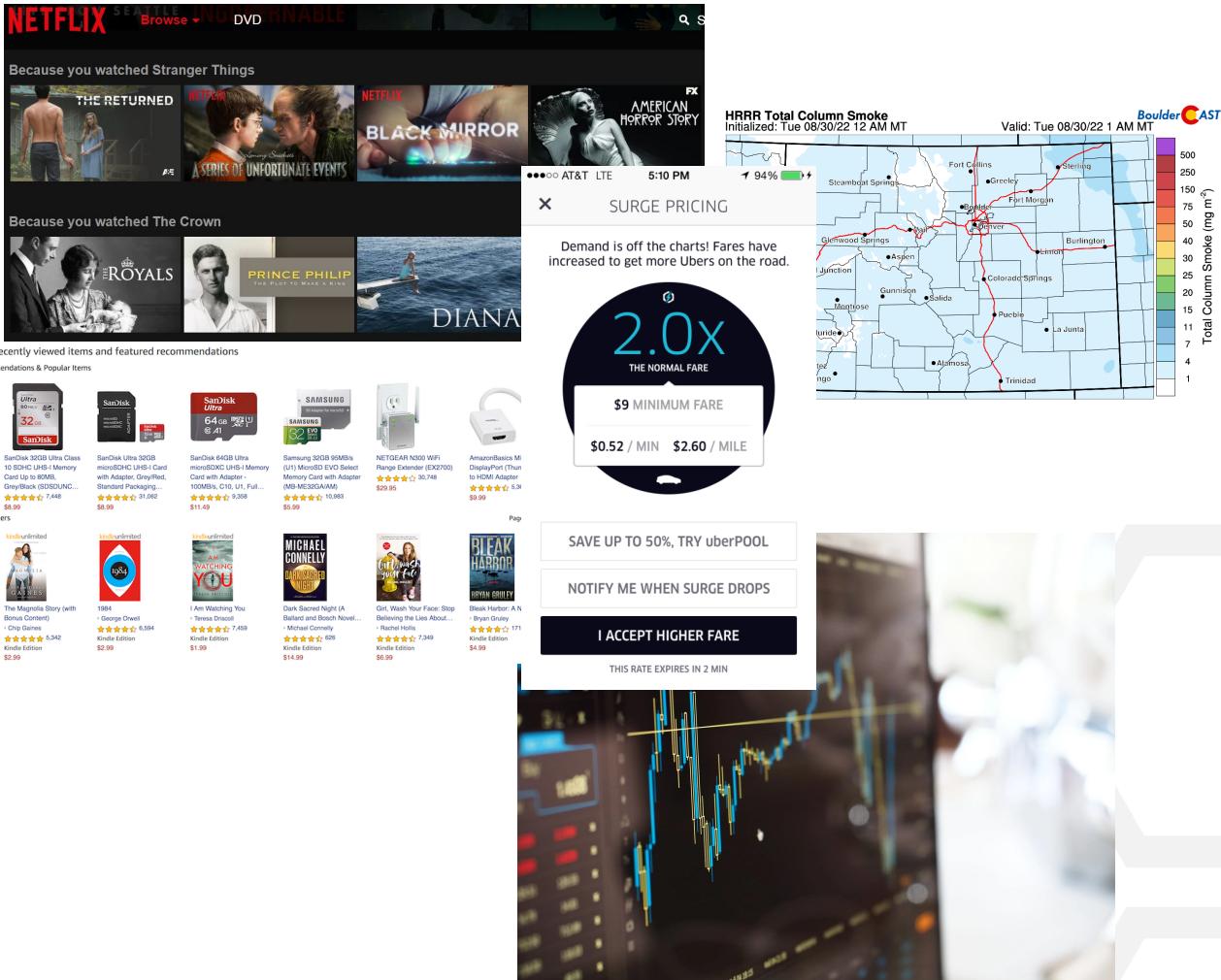
# Online Machine Learning

Bytewax + RiverML

# Online Machine Learning



- Content Personalization
- Recommendations
- Nowcasting
- Fraud Detection
- Algorithmic Trading
- Manufacturing Defects
- Dynamic Pricing





**River** is a Python library with algorithms and tools for online learning.



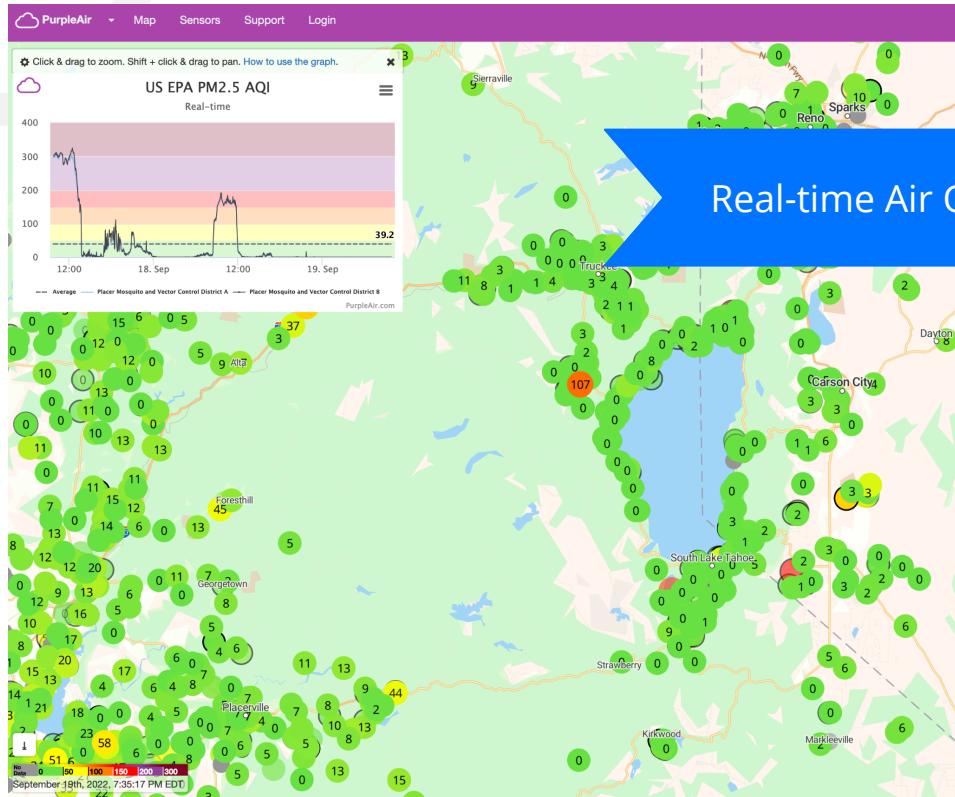
**Bytewax** is a Python library for stateful stream processing in conjunction with streaming platforms



**Apache Kafka** streaming platform.



# Detecting Fires in Real-time with Air Quality Data

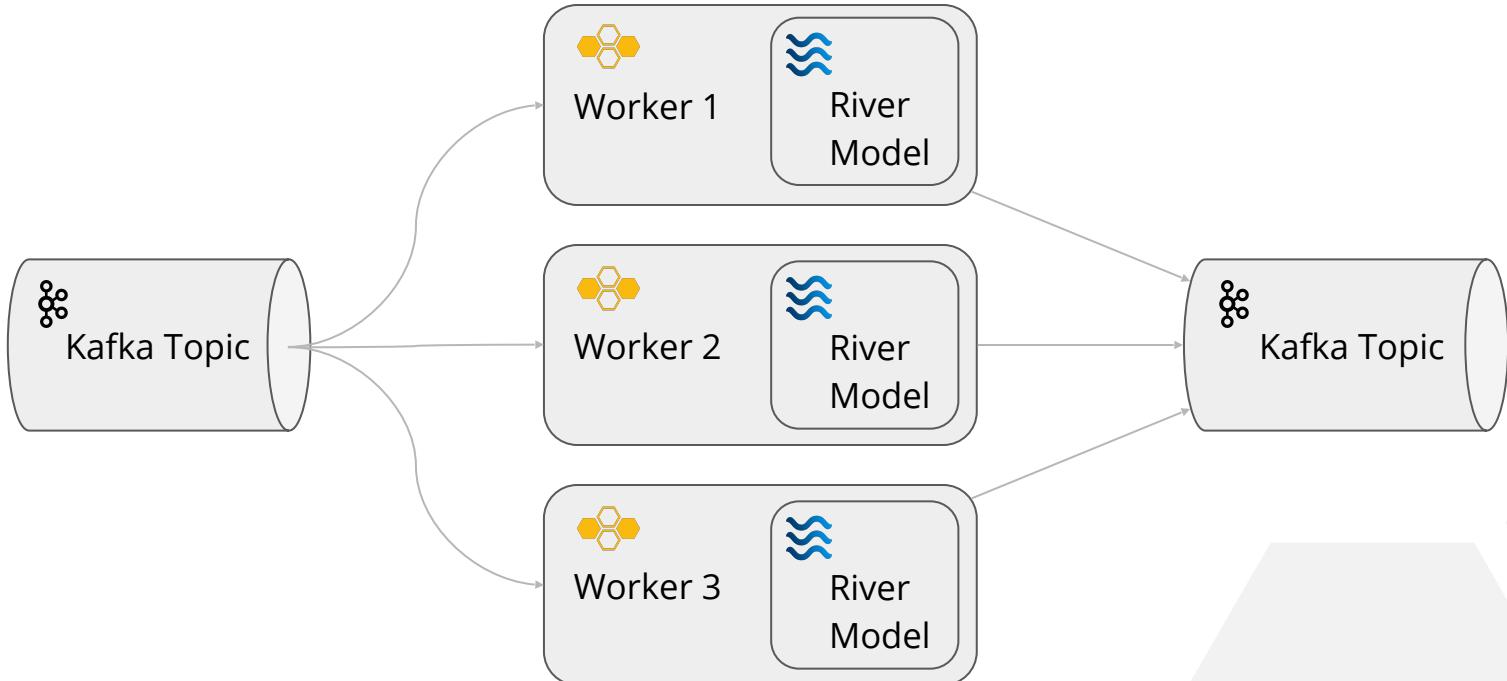


Real-time Air Quality Data

Bytewax Dataflow



Fires/Anomalies



# Bytewax Dataflow

```
● ● ●

flow = Dataflow()
flow.input(
    "aqi_state", KafkaInputConfig(
        brokers=["localhost:9092"],
        topic="sensor_data",
        starting_offset = "end",
        tail = True
    )
)

flow.map(deserialize)
flow.stateful_map(
    step_id = "anomaly_detector",
    builder = lambda: AnomalyDetector(n_trees=4, height=3, window_size=50, seed=11),
    mapper = AnomalyDetector.update,
)
flow.capture(
    KafkaOutputConfig(
        brokers=["localhost:9092"],
        topic="sensor_data",
    )
)
```

# River Stateful Anomaly Detector

```
● ● ●

class AnomalyDetector:

    def __init__(self, n_trees=10, height=8, window_size=72, seed=11):
        self.scaler = preprocessing.MinMaxScaler()
        self.detector = anomaly.HalfSpaceTrees(
            n_trees=n_trees,
            height=height,
            window_size=window_size,
            limits={'x': (0.0, 500.0)},
            seed=seed
        )

    def update(self, data):
        data['score'] = self.detector.score_one({'x': float(data['PM2.5_CF1_ug/m3'])})
        self.detector.learn_one({'x': float(data['PM2.5_CF1_ug/m3'])})
        return self, data
```



# The End



@MathesonZander,



@awmatheson

**Repo** [github.com/awmatheson/current22](https://github.com/awmatheson/current22)