

Tencent

# Game Data Analysis

Tencent Data Science Intern  
Take Home Challenge

Jiawei Xia | Mar 25

# Content

**01**

Anomaly Detection

**02**

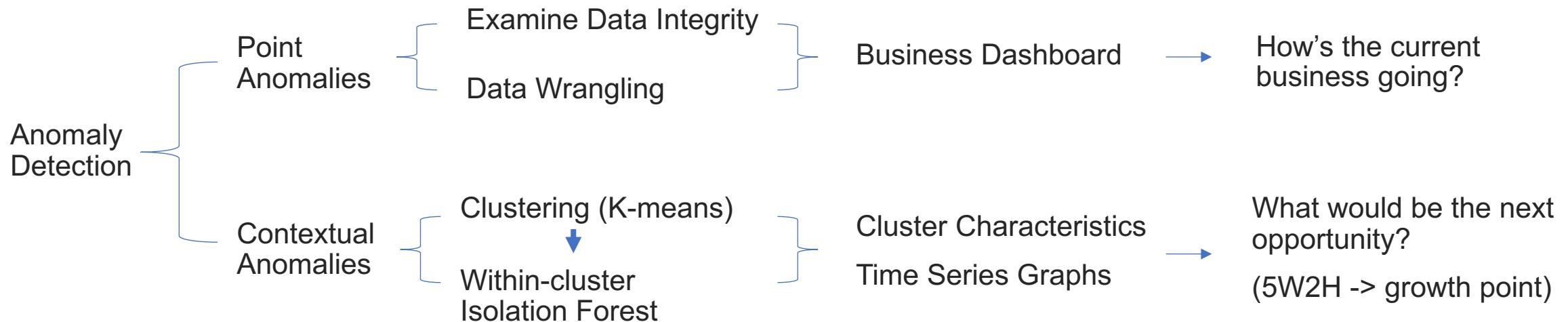
Data Science  
(Clustering/Prediction)

**03**

Visualization/  
Time Series

**04**

Questions Answered



# Point Anomalies

Examine Data Integrity 

## Data Wrangling

### Data Frame

- › Head sample data
- › Shape (748264 x 14)
- › Data types
- › Missing values (0)

### Columns

- › Date range (from 2018-03-15 to 2020-03-09)
- › Countries (239)
- › Platforms (3)
- › Key stats (min, max, etc)
- › Duplicated columns (1)

### Rows

- › Duplicated rows (~50%)
- › Conflicting rows (in country 'Dominica')
- › Inconsistent country format

### Pairwise

- › Diagonal: concentrated data
- › Line shape: possible linear relationship
- › L shape & U shape: 2 possible clusters
- › E shape: 3 possible clusters

#### Actions:

- › Transform str object into datetime for "Date"

#### Actions:

- › Remove anomalies disobeying business rules
- › Drop the second "Date"

#### Actions:

- › Drop repeated rows
- › Drop conflicting rows
- › Translate zh-cn to en

#### Actions:

- › Assumption for clustering k=2 or 3
- › Linear correlation

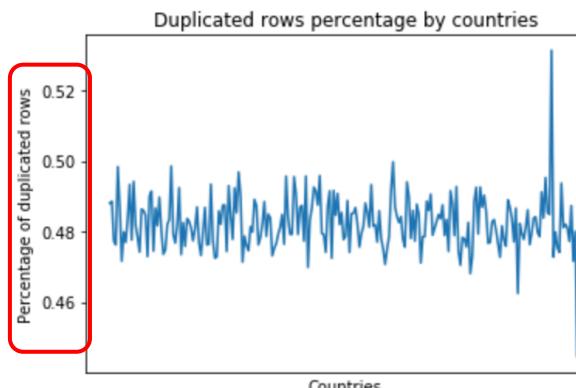
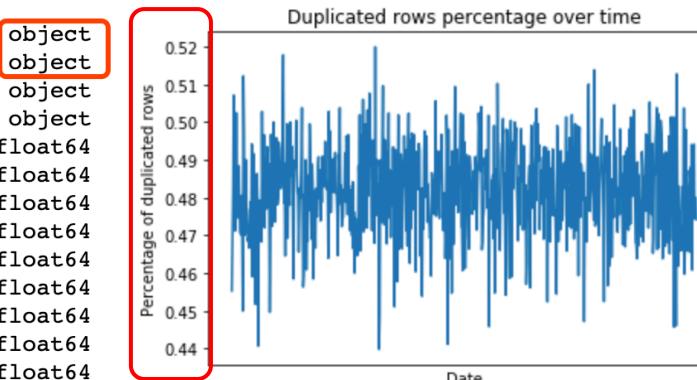
# Point Anomalies

Examine Data Integrity 

## Data Wrangling

```
Date          object
Date.1        object
Platform      object
Country       object
TRU           float64
DAU           float64
Items          float64
Trans          float64
Items Per Trans float64
Items per DAU  float64
Conversion     float64
Cash Flow      float64
Return Customer float64
Time Spend Per Day(seconds) float64
dtype: object
```

Iran(伊朗)	3441
Australia(澳大利亚)	3434
圣卢西亚	3366
Israel(以色列)	3365
Iraq(伊拉克)	3363
...	
厄立特里亚	529
托克劳	496
圣巴泰勒米	239
unknown(未知)	31
蒙塞拉特岛	18



	TRU	DAU	Items	Trans	Items Per Trans	Items per DAU	Conversion	Cash Flow	Return Customer	Time Spend Per Day(seconds)
count	7.482640e+05	7.482640e+05	7.482640e+05	748264.000000	748264.000000	748264.000000	748264.000000	748264.000000	748264.000000	748264.000000
mean	7.399568e+05	1.816949e+04	5.089120e+04	3958.49364	31.569433	1.414330	2.902036	5093.453512	39.795054	122.605716
std	6.776595e+05	1.801799e+04	4.610241e+04	3655.62504	23.629177	1.208879	2.591666	4712.790645	24.796009	43.331318
min	6.286478e+05	1.529217e+04	4.342911e+04	3428.09000	17.580000	1.300000	2.300000	4527.010000	12.210000	-9.990000
25%	6.288102e+05	1.529677e+04	4.342911e+04	3428.09000	17.580000	1.300000	2.300000	4527.010000	28.660000	98.950000
50%	6.314452e+05	1.537393e+04	4.352124e+04	3433.85000	28.090000	1.320000	2.560000	4532.770000	39.430000	118.180000
75%	6.574735e+05	1.617663e+04	4.466021e+04	3522.53000	36.080000	1.390000	3.000000	4611.080000	48.920000	139.370000
max	2.677675e+07	2.141935e+06	4.387412e+06	434846.58000	1744.870000	444.240000	232.630000	819825.280000	3912.090000	1390.060000



# Point Anomalies

Examine Data Integrity 

Data Wrangling 

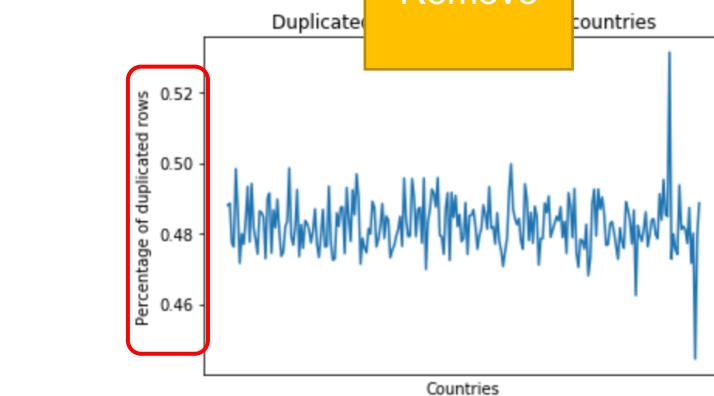
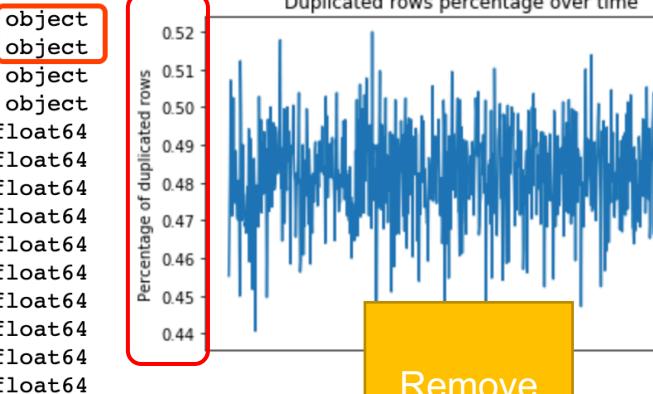
```
Date          object
Date.1        object
Platform      object
Country       object
TRU           float64
DAU           float64
Items          float64
Trans          float64
Items Per Trans float64
Items per DAU  float64
Conversion     float64
Cash Flow      float64
Return Customer float64
Time Spend Per Day(seconds) float64
dtype: object
```

**Drop**

**Transform**

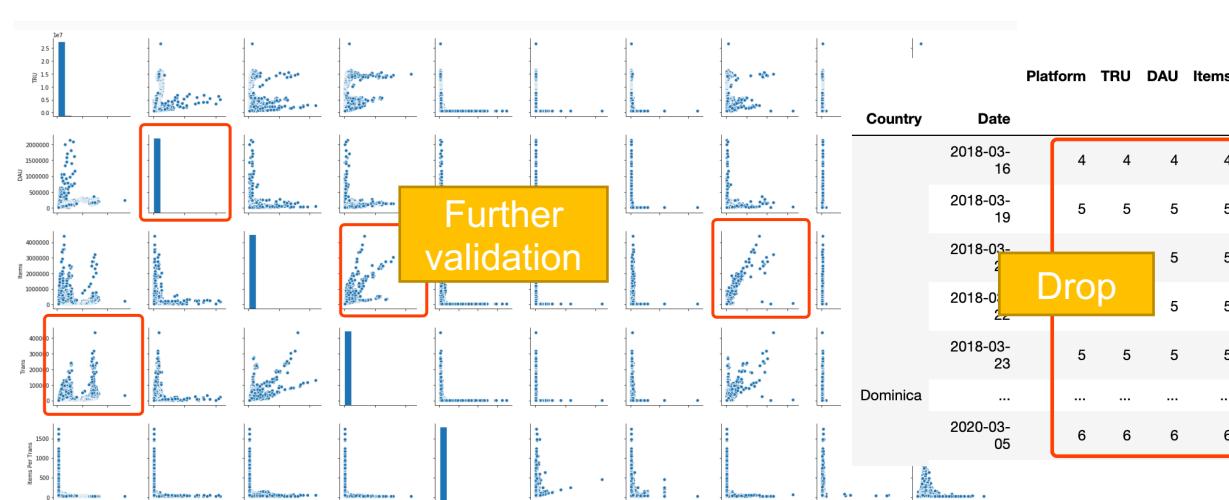
Iran(伊朗)	3441
Australia(澳大利亚)	3434
圣卢西亚	3366
Israel(以色列)	
Iraq(伊拉克)	
厄立特里亚	529
托克劳	496
圣巴泰勒米	239
unknown(未知)	31
蒙塞拉特岛	18

**Translate**



	TRU	DAU	Items	Trans	Items Per Trans	Items per DAU	Conversion	Cash Flow	Return Customer	Time Spend Per Day(seconds)
count	7.482640e+05	7.482640e+05	7.482640e+05	748264.000000	748264.000000	748264.000000	748264.000000	748264.000000	748264.000000	748264.000000
mean	7.399568e+05	1.816949e+04	5.089120e+04	3958.49364	31.569433	1.414330	2.902036	5093.453512	39.795054	122.605716
std	6.776595e+05	1.801799e+04	4.610241e+04	3655.62504	23.629177	1.208879	2.591666	4712.790645	24.796009	43.331318
min	6.286478e+05	1.529217e+04	4.342911e+04	3428.09000	17.580000	1.300000	2.300000	4527.010000	12.210000	-9.990000
25%	6.288102e+05	1.529677e+04	4.342911e+04	3428.09000	17.580000	1.300000	2.300000	4532.770000	39.430000	98.950000
50%	6.314452e+05	1.537393e+04	4.352124e+04	3433.85000	28.090000	1.320000	2.560000	4611.080000	48.920000	118.180000
75%	6.574735e+05	1.537393e+04	4.352124e+04	3522.53000	36.080000	1.390000	3.000000	4611.080000	48.920000	139.370000
max	2.677675e+07	2.000000e+07	2.000000e+07	34846.58000	1744.870000	444.240000	232.630000	819825.280000	3912.090000	1390.060000

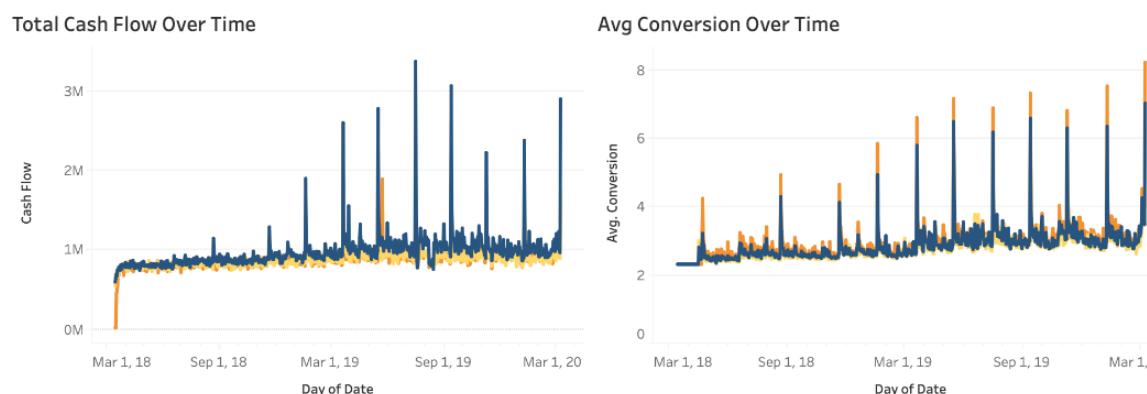
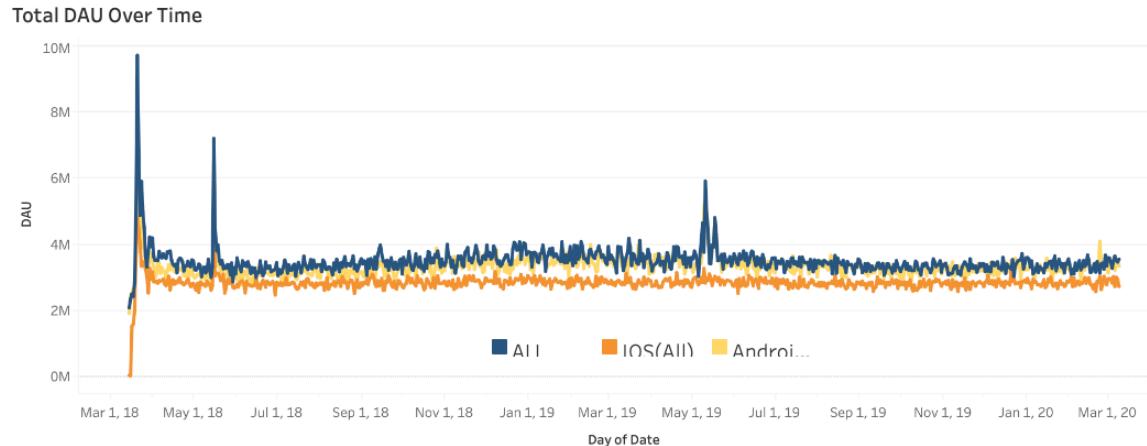
**Further validation**



**Further validation**

# Business Dashboard

## Overview of Key Performance Metrics



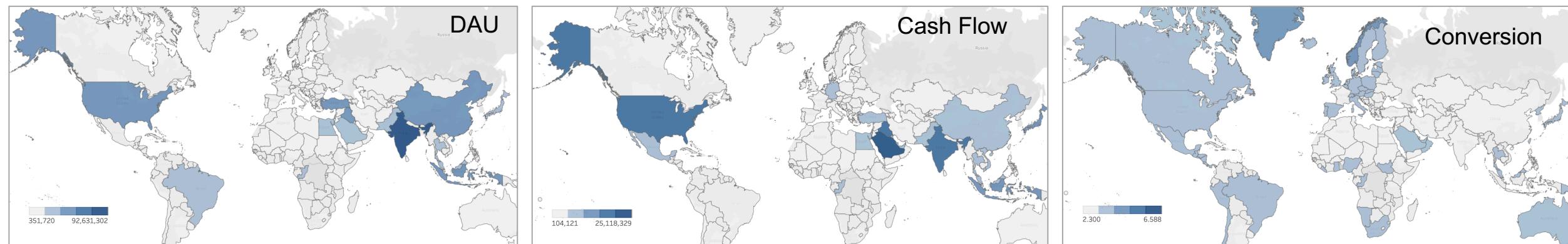
## Observations

Total DAU Today	Avg Conversion % Today	Total Cash Flow Today
9,569,579	7.147	6,245,403

- › Key Business Metrics
  - › Normal level DAU for a half year
  - › Excellent Cash Flow & Conversion
  - › CF & Conversion peaks occur bi-monthly
- › Platform Comparison
  - › Three platforms shared similar **patterns** and **metrics**
  - › All != iOS + Android (may be other platforms)
  - › Cash Flow ~ Conversion starting Nov 2018
  - › DAU: All>Android>iOS
  - › Cash Flow: All>Android~=iOS
  - › Conversion: iOS>All>~Andorid

# Business Dashboard

## Overview of Key Performance Metrics



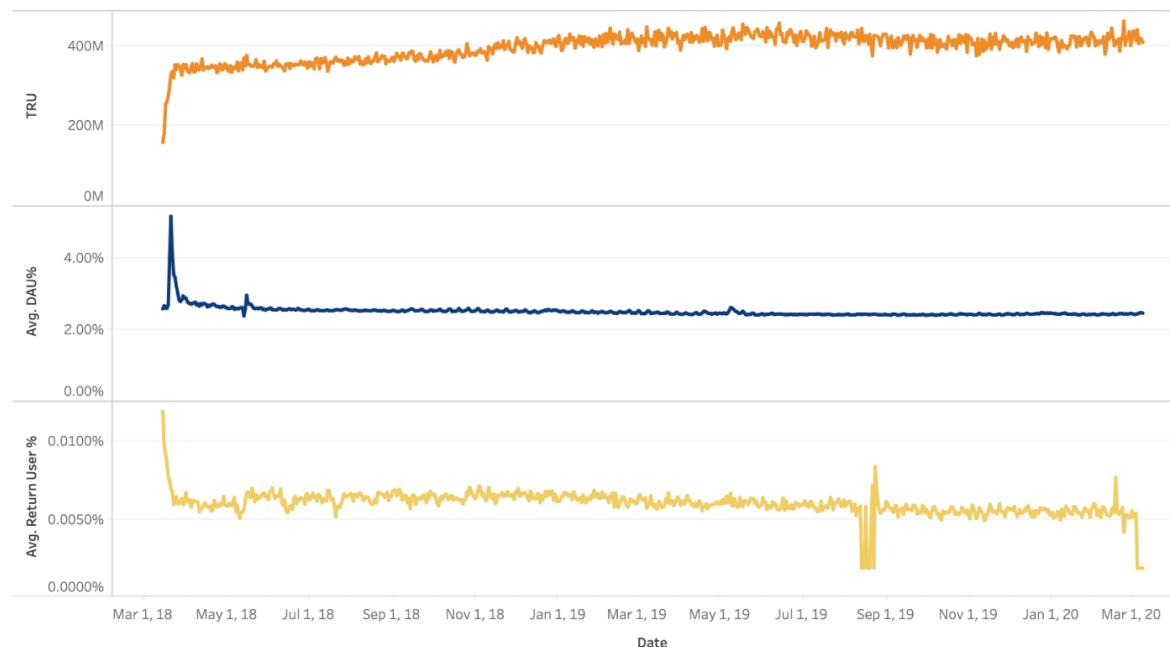
## Observations

- › Distribution: DAU~Cash Flow
- › High DAU/CF countries: along roughly the same latitude
- › High conversion countries: distributed by continents:  
North/South America, Europe, Oceania

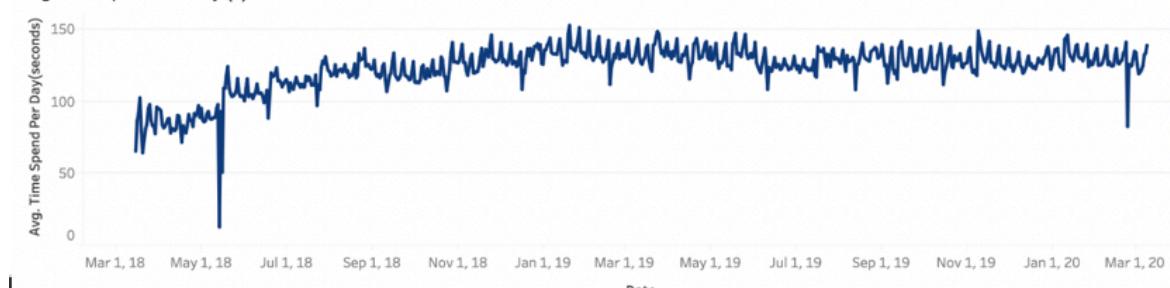
# Business Dashboard

## Customer Life Cycle

Customer Life Cycle Over Time



Avg Time Spend Per Day (s)



## Observations

- › Starting point: Highest return user, lowest TRU. When DAU increase, return user decrease.
- › Growth: No obvious user acquisition after first launch
- › Steady DAU over time
- › Drop of time spend per day on around 15th of every month

## Future work

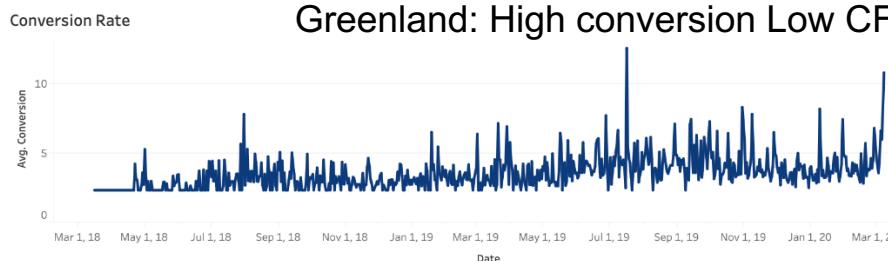
- › Analyze the plot by countries using the map



# Business Dashboard

## In-App Purchase

Conversion Rate



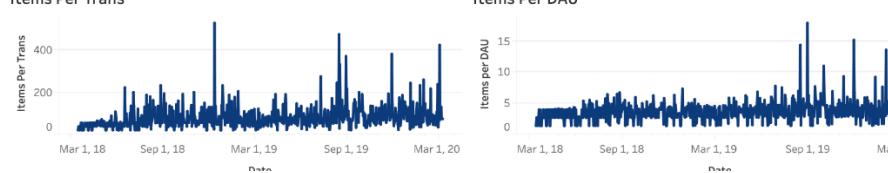
Cash Flow



Transactions



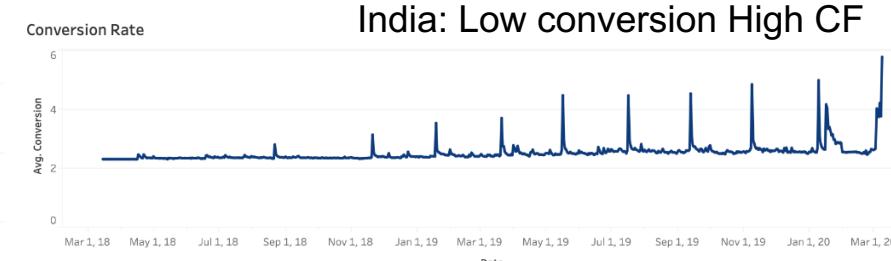
Items Per Trans



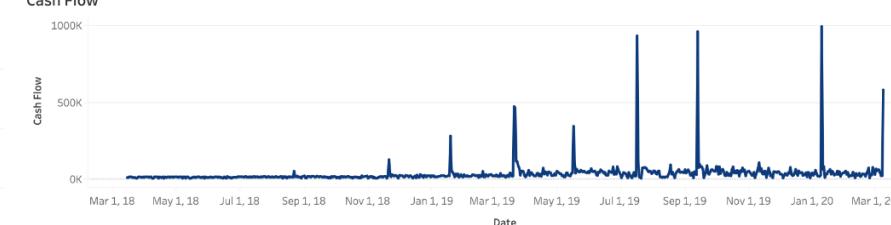
Items Purchased



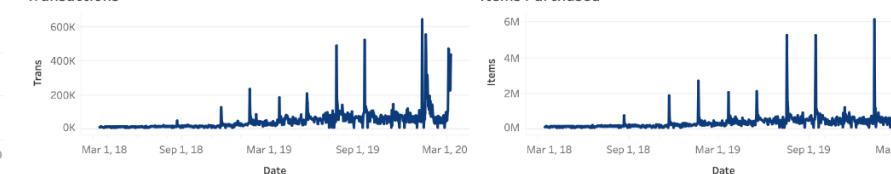
Conversion Rate



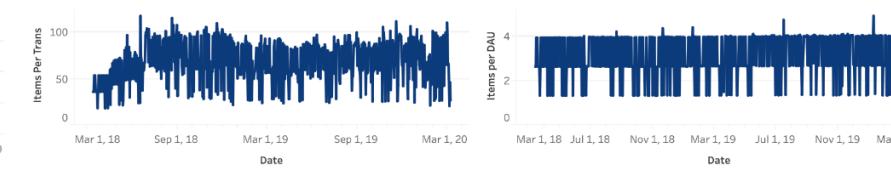
Cash Flow



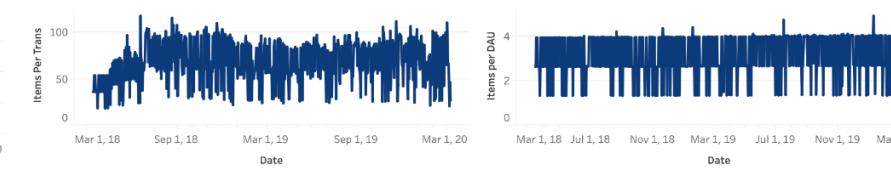
Transactions



Items Per Trans



Items Purchased



## Observations

India:

- › Conversion bimonthly
- › Large cash flow when converse
- › Less items bought

Greenland:

- › Low but continuous cash
- › More items bought

## Future Work

- › More country comparisons

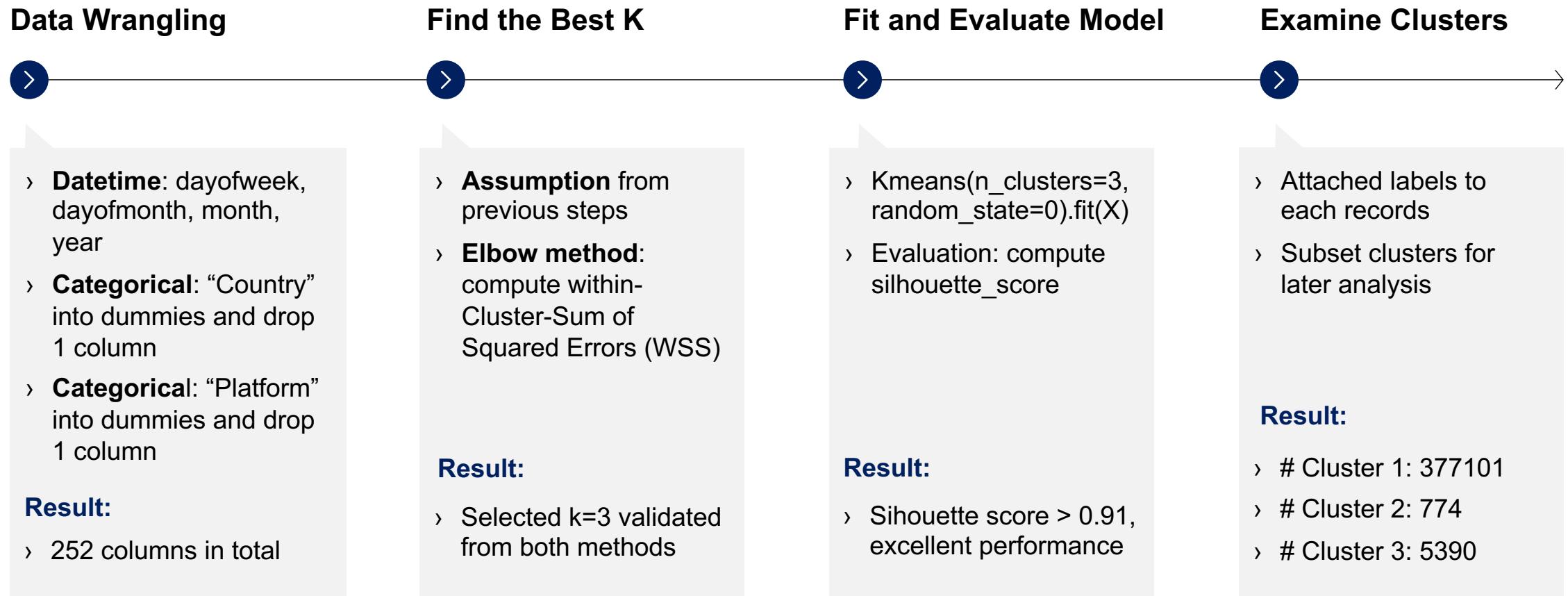
# How's the Current Business?

- **Periodicity:**
  - There might be some campaigns/promotions that occur once every two months, which are very effective in conversion and attract people to purchase.
  - However, this campaign does little on attracting new customers/retention customers/daily active users.
- **Users:** High(low) conversion low(high) cash flow countries may relate to the per capita income.
  - In western countries where per capita income is high, on average people are willing to spend money on games and purchase more items
  - In countries where per capita income is relatively low but having a small group of wealthy residents, on average people may purchase less but a small number of people would like to purchase "luxurious items" in game as a reflection of their lifestyle
- **Game:**
  - The game may convert from an old version as it started from higher return customer
  - The game has remained the same registration user for over a year. It might have been taken off from the app store, and DAU are very sticky loyal users.

# Contextual Anomalies

K-Means Clustering ✓

Within-cluster Isolation Forest



# Contextual Anomalies

K-Means Clustering ✓

Within-cluster Isolation Forest

Data Wrangling



```
game_data.shape
```

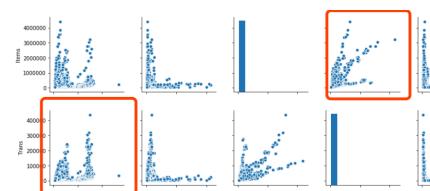
```
(383265, 13)
```



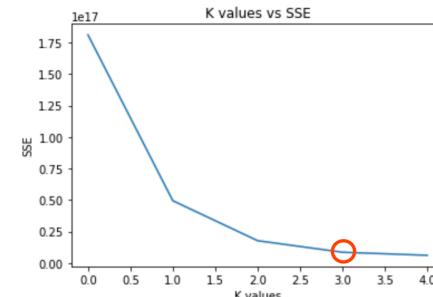
```
X.shape
```

```
(383265, 252)
```

Find the Best K



k=2/3



k=3

Fit and Evaluate Model



```
silhouette_score(X,
```

```
0.9194291375101088
```

Examine Clusters



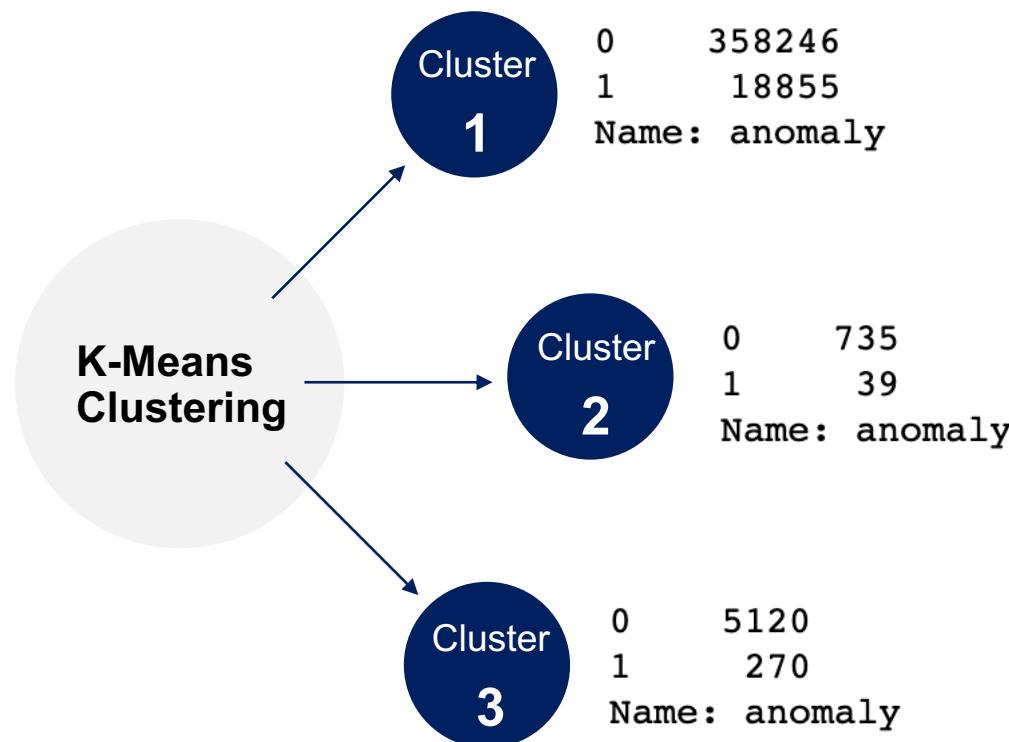
```
0      377101  
2      5390  
1      774  
Name: label, dtype: int64
```

```
cluster1  
cluster2  
cluster3
```

# Contextual Anomalies

K-Means Clustering ✓

Within-cluster Isolation Forest ✓



## Within-cluster Isolation Forest

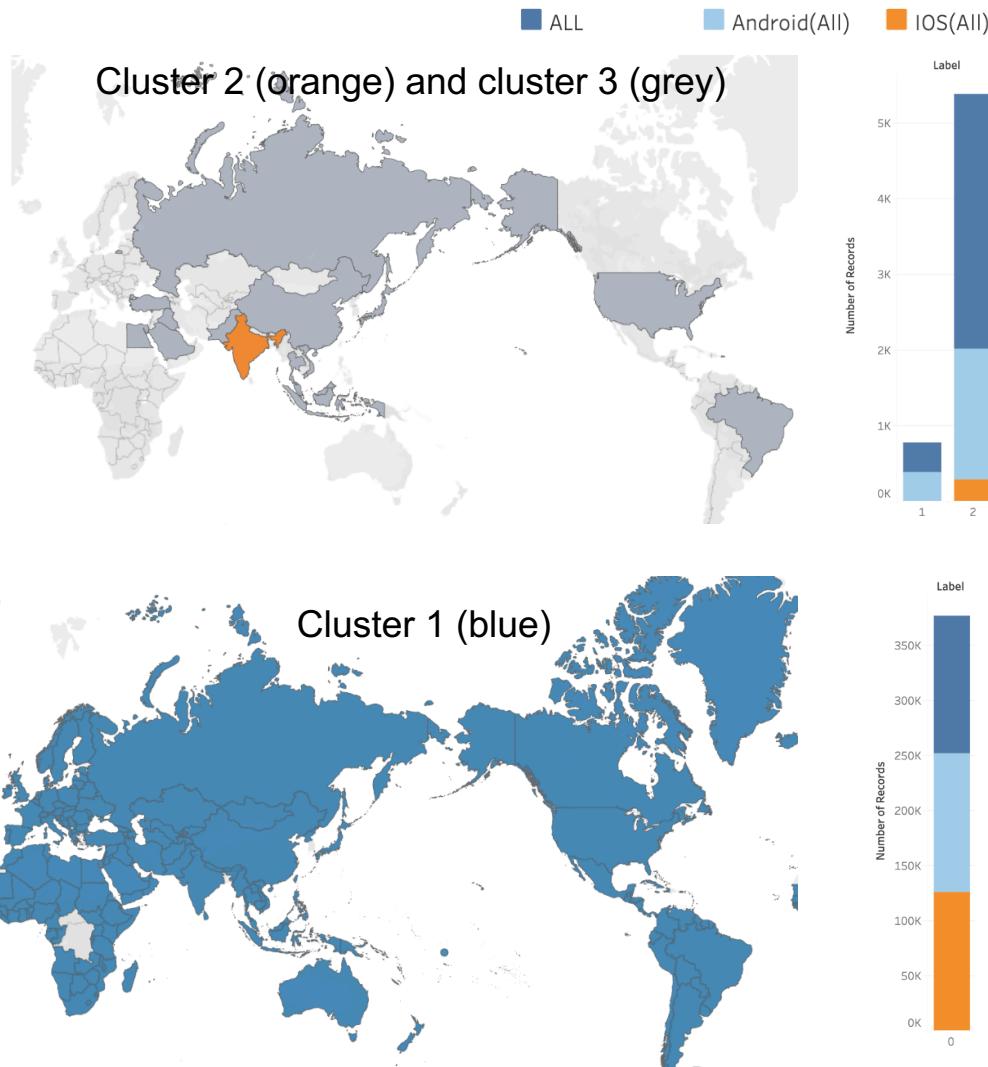
```
# input: 3 parameters
def isolation_forest(df, columns, outliers_fraction):
    data = df[columns] # extract data
    scaler = StandardScaler()
    np_scaled = scaler.fit_transform(data)
    data = pd.DataFrame(np_scaled) # scaled data

    model = IsolationForest(contamination=outliers_fraction)
    model.fit(data) # train isolation forest

    # attach anomaly label
    # 0: normal, 1: anomaly
    df['anomaly'] = np.array(pd.Series(model.predict(data)))
    df['anomaly'] = df['anomaly'].map( {1: 0, -1: 1} )

    # return dataframe with anomaly label
    return df
```

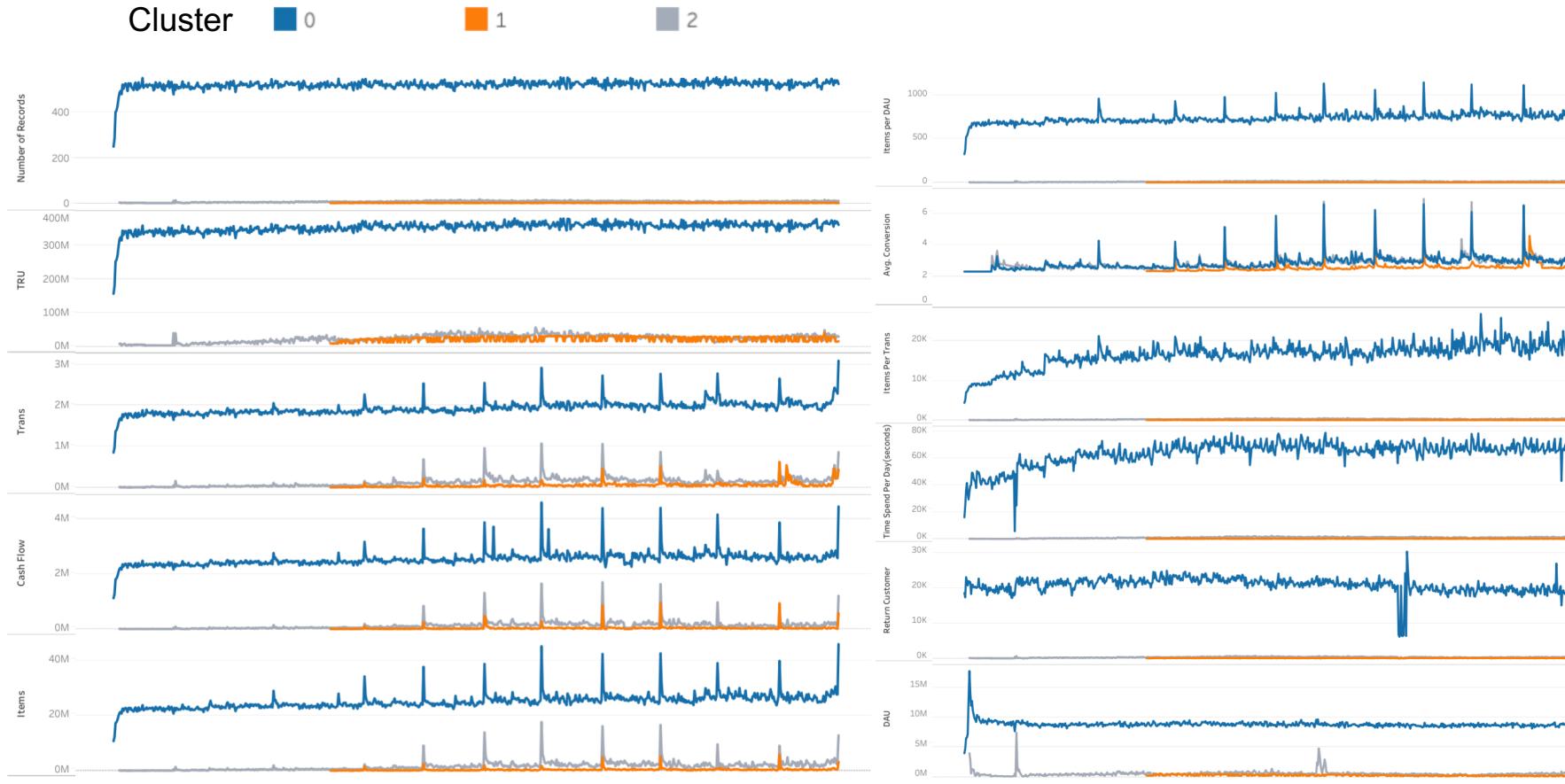
# Cluster Characteristics



## Observations

- › Cluster1:
  - › Worldwide
  - › Evenly divided platforms
- › Cluster2:
  - › India
  - › No iOS, evenly divided android/all
- › Cluster3:
  - › High DAU/Cash Flow country plus Russia
  - › Platform All>Android>iOS

# Cluster Time Series Analysis

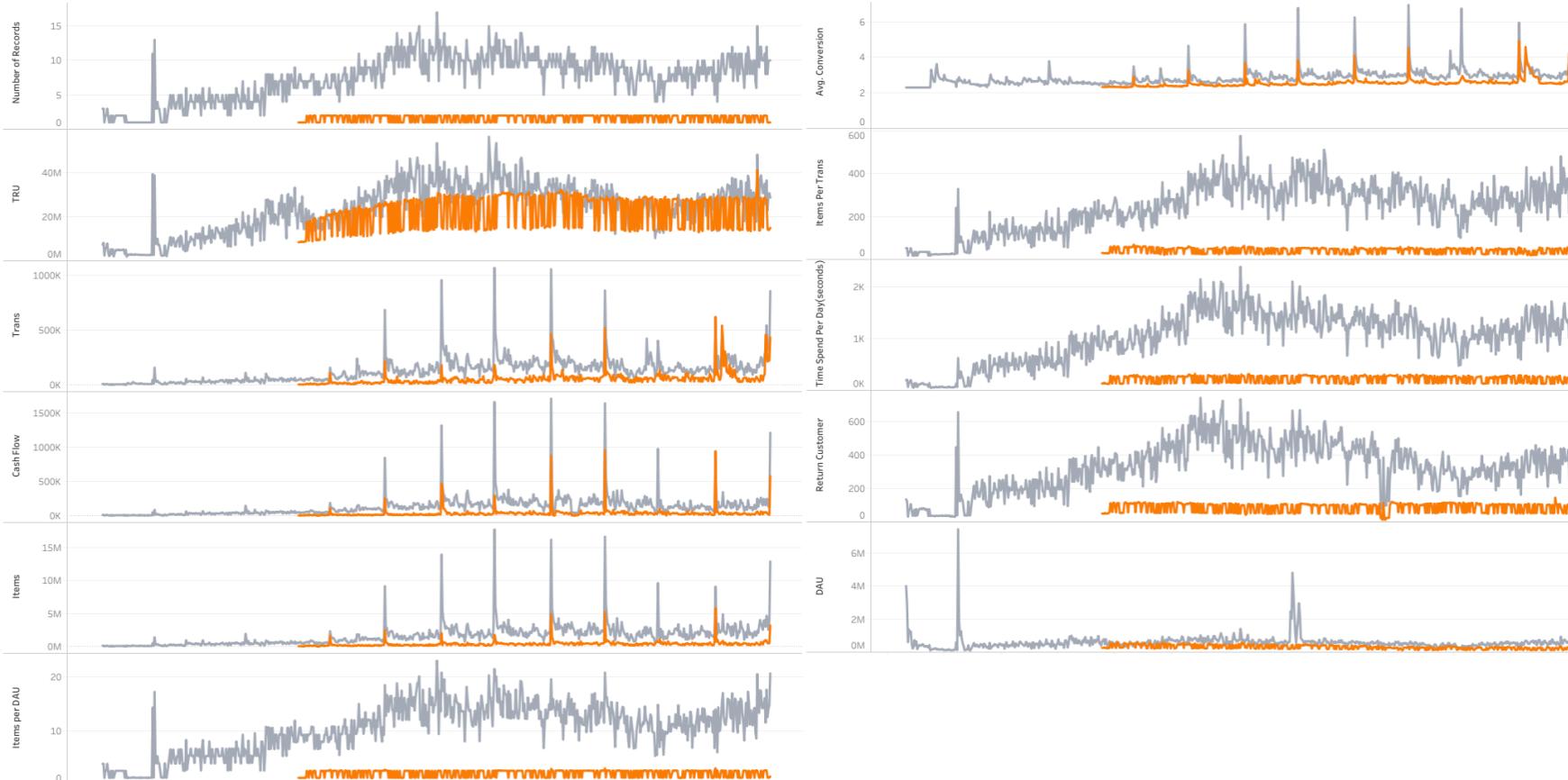


# Cluster Time Series Analysis

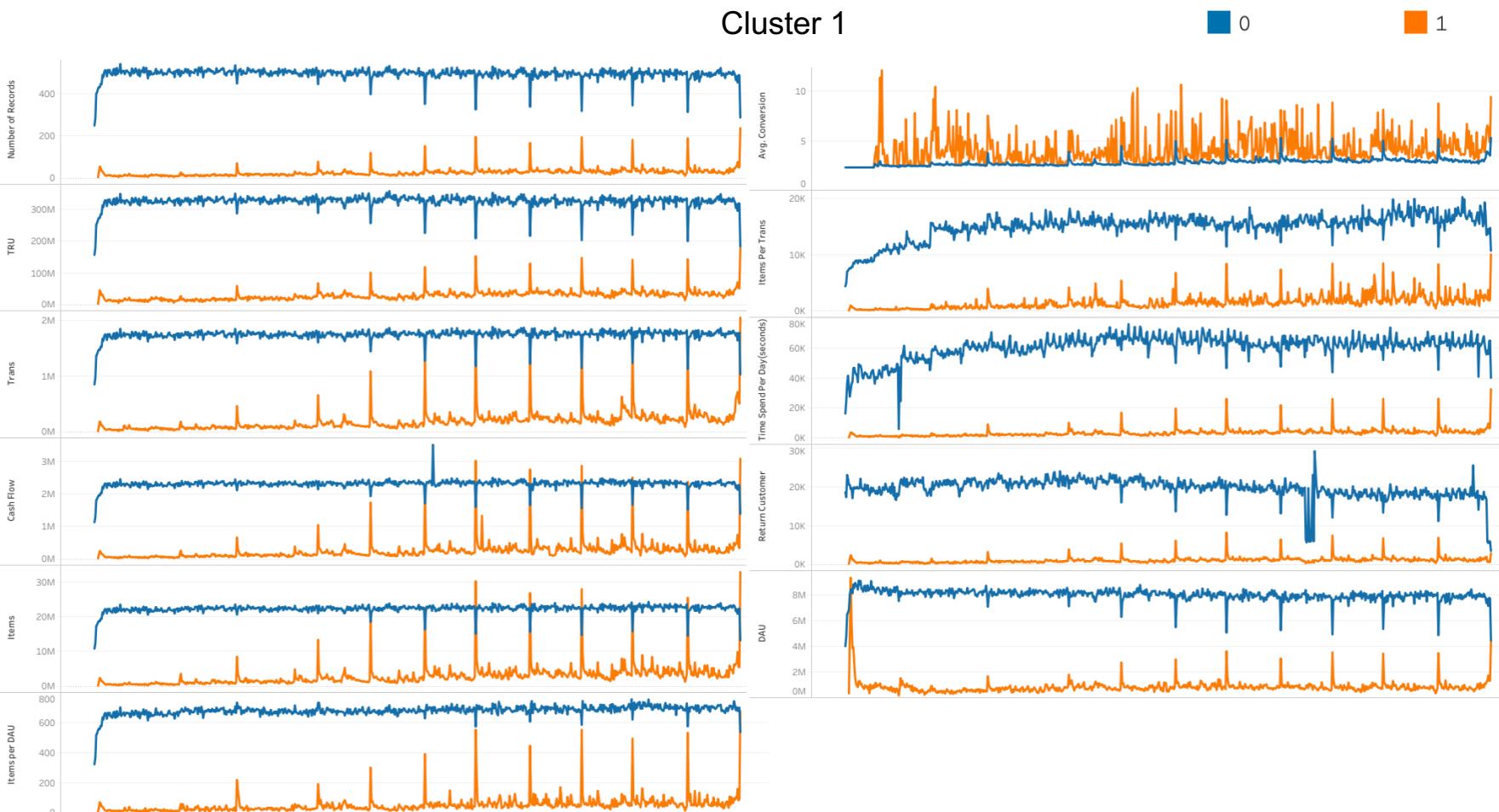
Cluster

1

2

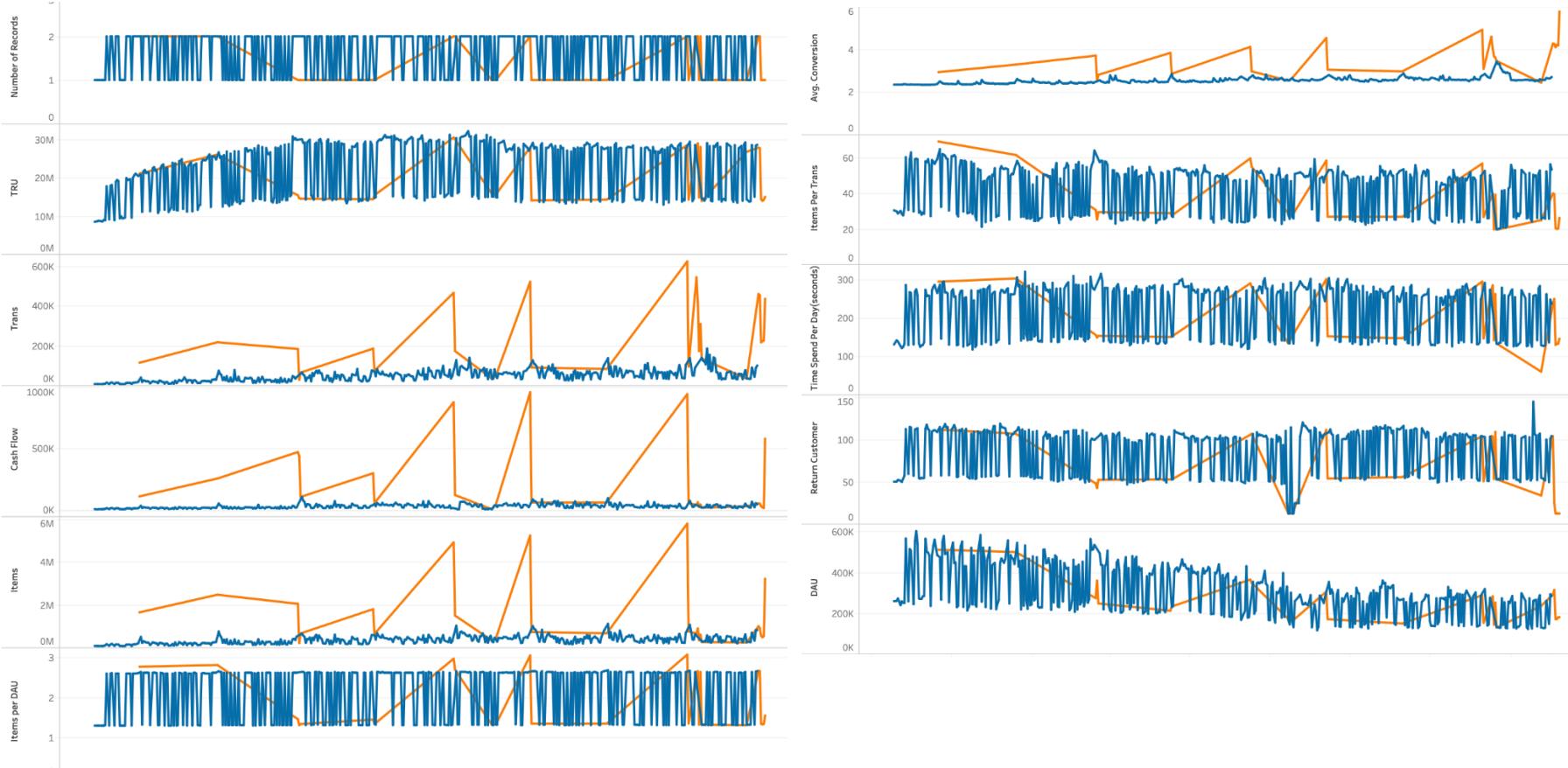


# Anomaly Time Series Analysis

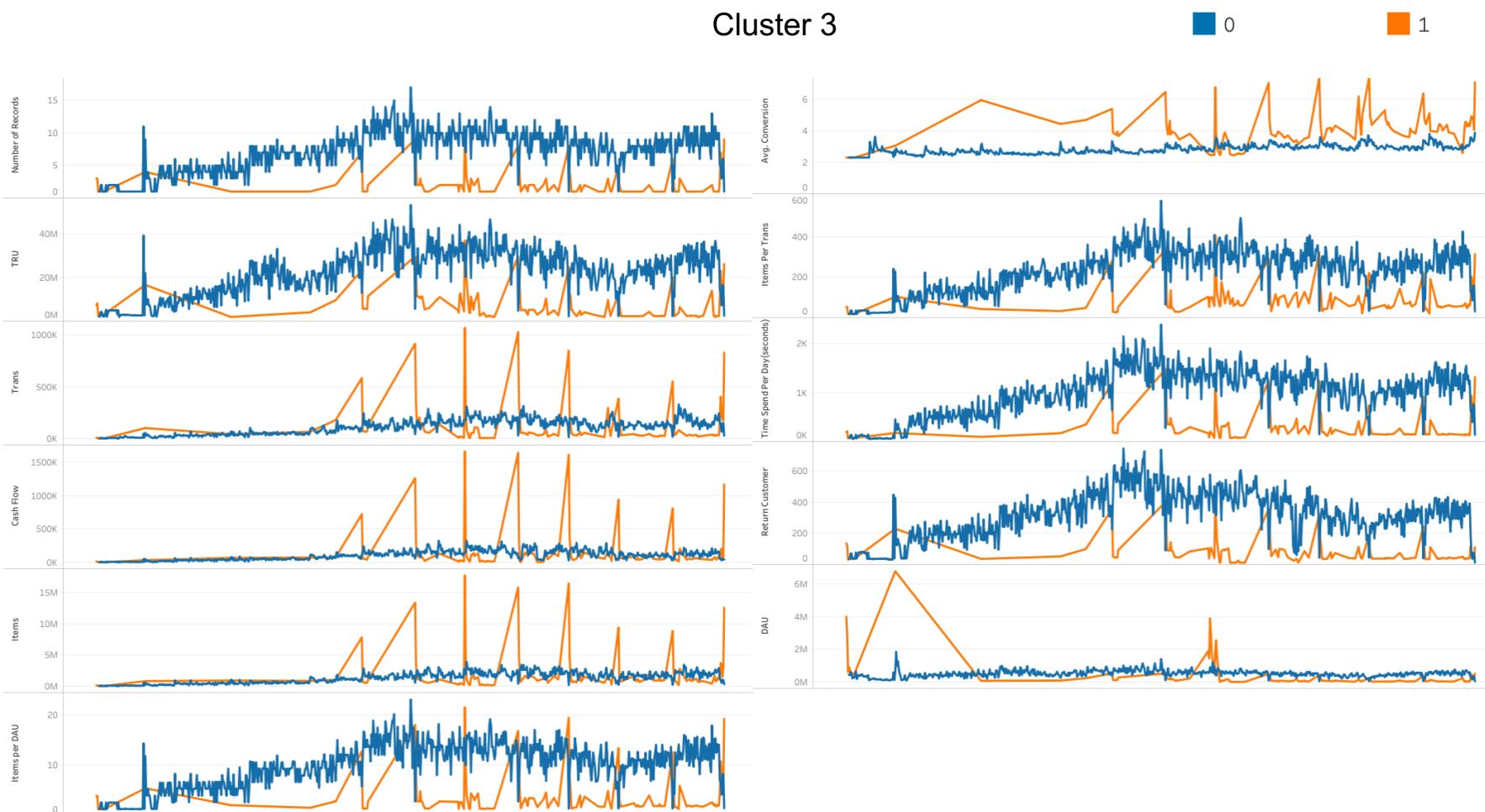


# Anomaly Time Series Analysis

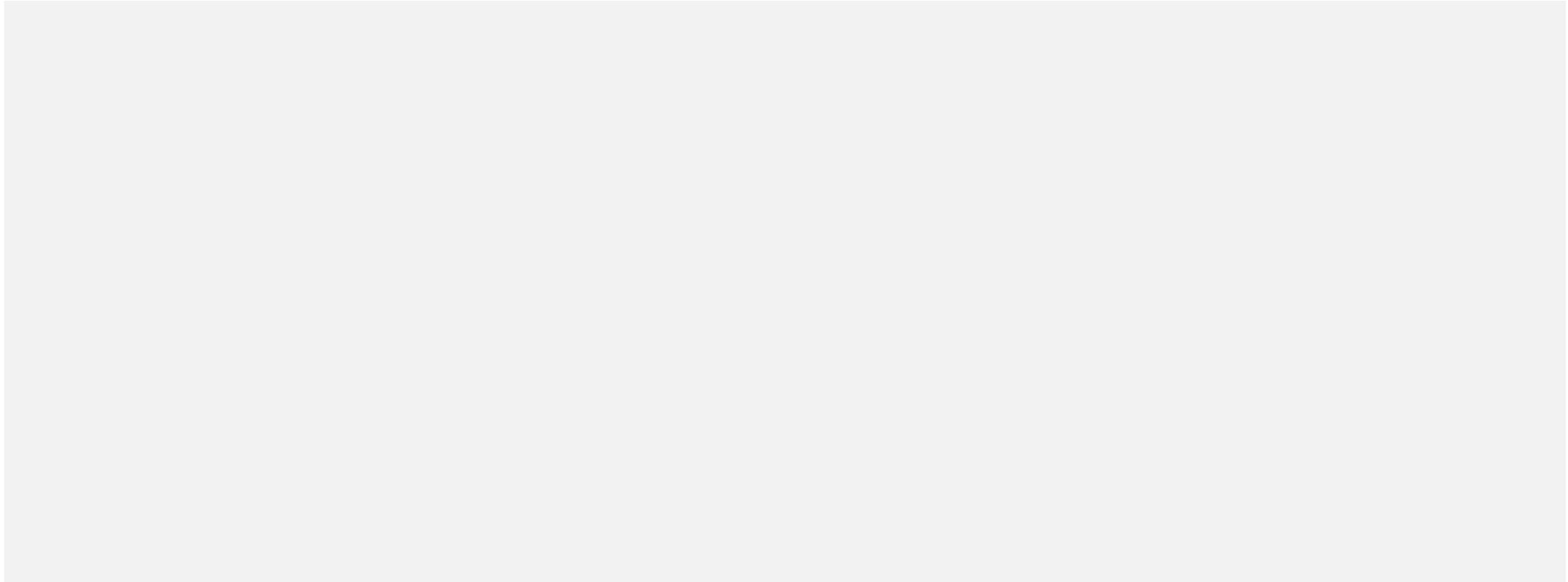
Cluster 2



# Anomaly Time Series Analysis



# Next Opportunity?





THANK YOU