MACHINE LEARNING IN ZABBIX 6.0 LTS: ANOMALY DETECTION AND BASELINES

ALEKSANDRS KALIMULINS
C Developer, Zabbix, Latvia
MODERN MONITORING CHALLENGES

- More devices, VMs, servers and applications
- More monitored entities means more metrics
- IT environments are changing rapidly
- New concepts emerge frequently
MODERN MONITORING CHALLENGES

- Less time to keep track of what is normal
- Hard to get right signal-to-noise ratio
MACHINE LEARNING: ZABBIX APPROACH

"Field of study that gives computers the ability to learn without being explicitly programmed”

- Arthur Samuel (computer scientist, machine learning pioneer)
MACHINE LEARNING: ZABBIX APPROACH

EASY AND TRANSPARENT:
WHAT IS MACHINE LEARNING?

EASY AND TRANSPARENT:

- Simple configuration
- Easy to understand
- Easy to verify
FOCUS AREAS

Smart triggers

• Anomaly detection
  • Analyse historical data
  • Find outliers in analysis results
FOCUS AREAS

Smart triggers

- Anomaly detection
  - Analyse historical data
  - Find outliers in analysis results

- Baselines
  - Calculate averages in past calendar periods
  - Find how far are current values
ANOMALY DETECTION
ANOMALY DETECTION

- Works when the majority is normal data
- Long-term analytics, works with trends
- Zabbix uses STL decomposition
STL DECOMPOSITION
STL DECOMPOSITION

\[ Y_t = T_t + S_t + R_t \]

1. Apply smoothing to the original curve, get \( T_t \)
2. Subtract result from the original curve, split into seasons
3. Apply averaging to seasons, get seasonal curve \( S_t \)
4. Subtract \( T_t \) and \( S_t \), get residue \( R_t \)
STL DECOMPOSITION
DEVIATIONS

- Deviation is a measure of data variability

- How “far” values are from average?
DEVIATIONS

- Standard and median deviations in Zabbix
  - stddevpop(), stddevsamp(), mad()

- Also supported in anomaly function
ANOMALY DETECTION ALGORITHM

- Get trend values for the period
- Decompose values, get remainder
- Calculate deviation for values in remainder
- Select values with deviations > threshold
ANOMALY DETECTION FUNCTION

trendstl(/host/key, period: time shift, detection period, season, deviations, dev algorithm)

- Returns $0 \leq number \leq 1$ (ratio anomaly count / value count)
ANOMALY DETECTION FUNCTION

trendstl(/host/key,period:time shift,detection period,season,deviations,dev algorithm)

Parameters

- /host/key - item
- period:time shift - evaluation period (for decomposition)
- detection period – report anomalies in this period
- season – season's length for decomposition
- deviations, dev algorithm
ANOMALY DETECTION FUNCTION

- trendsl(/Web/net.if.out[en0],30d:now/d,7d,12h,3,"mad") > 0.1
  - Decompose last 30 days
  - Report anomalies within last 7 days
  - Use season 12 hours
  - Count points > 3 median deviations
- Same as:
  - trendsl(/Web/net.if.out[en0],30d:now/d,7d,12h) > 0.1
CAVEATS

trendstl()

- Long term analytics, works only with trends
- Usable only if data has seasonality
- Season parameter is seconds
WHAT IS BASELINE?

“BASELINE IS A VALUE DERIVED FROM AN AVERAGE OVER MULTIPLE CALENDAR PERIODS OF THE SAME LENGTH”

– Zabbix (best monitoring solution)
PERIODS AND SEASONS

- Periods and seasons
- Average from past calendar periods
  - E.g., every Monday of the past 4 weeks
  - Monday is a period, week is a season
- Periods linked to current time
  - If today is Wednesday, then periods are Tuesdays
PERIODS VS SEASONS

- Period 1 day
- Tuesday
- Season 1 week
- Current week
- Seasons 4 weeks
**BASELINE FUNCTIONS**

`baselinewma(/host/key,period<:time shift>,seasons)`

- Returns baseline by averaging data periods in seasons
- Uses **Weighted Moving Average** algorithm (WMA)
baselinewma(/host/key,period<:time shift>,seasons)

- Returns baseline by averaging data periods in seasons
- Uses Weighted Moving Average algorithm (WMA)
BASELINE FUNCTIONS

baselinedev(/host/key,period<:time shift>,seasons)

⊙ Returns number of standard deviations

⊙ Compares last period to periods before within seasons
**BASELINE FUNCTIONS**

baselinedev(/host/key, period<:time shift>, seasons)

- Returns number of standard deviations
- Compares last period to periods before within seasons
**BASELINE FUNCTIONS**

baselinedev(/Zabbix server/system.cpu.load,1h,10d)>3

- Check if load for last hour > 3 deviations away from mean
- Use 10 one-hour periods over last 10 days
BASELINE FUNCTIONS

baselinewma(/Zabbix server/nginx.requests.total.rate,1d,12w)*2 < trendavg(/Zabbix server/nginx.requests.total.rate,1d:now/d)

- Check if web traffic yesterday is > 2x higher than WMA on the same weekdays over last 12 weeks
CAVEATS

Baselines “remember” problems

- Abnormal values included in calculations
- Time units are not interchangeable
- $7d \neq 1w$
TECHNICAL CONSIDERATIONS

- Maintain trend storage intervals
- Set reasonable TrendCacheSize
- Set reasonable intervals for calculated items
WHAT TO CHOOSE?

- Suitable only for long term analytics
- trendstl() heavier on resources
- Calendar periods in baselinewma/dev()
- trendstl() works best with few anomalous points
Thank you!

www.zabbix.com