Automating Operational Decisions in Real-time

Chris Sanden
Senior Analytics Engineer

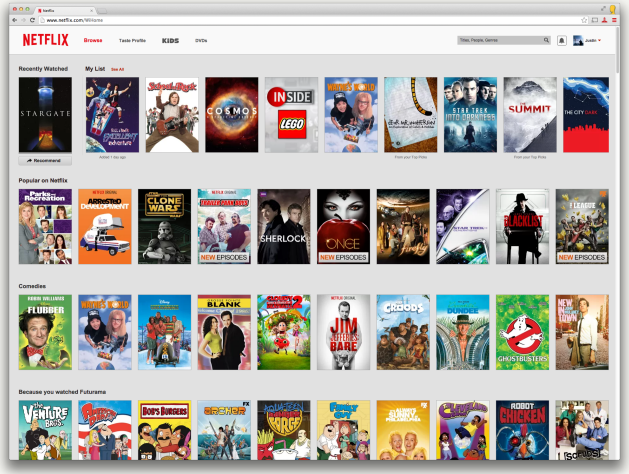
NETFLIX

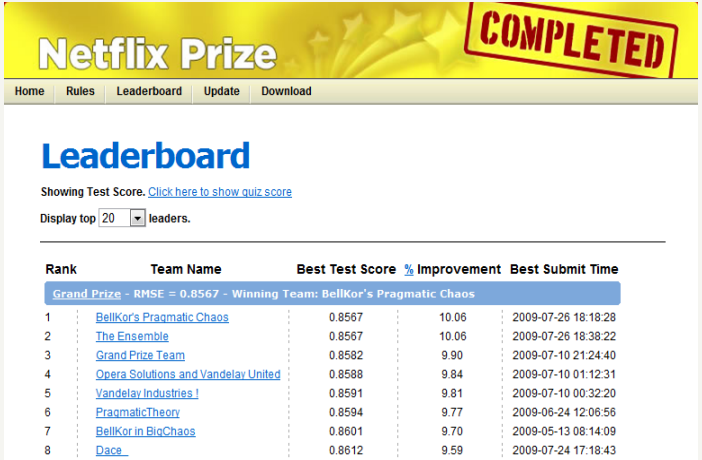
About Me.

- Senior Analytics Engineer at Netflix
- Working on Real-time Analytics
  - Part of the Insight Engineering Team
- Tenure at Netflix: 2 years
- Twitter: @chris_sanden

What is the first thing that comes to mind when think about Netflix and machine learning?

Recommendations & Netflix Prize





Automating Operational Decisions in Real-time

Supporting operational availability and reliability

The Goal.

Discuss how machine learning, and statistical analysis techniques can be used to automate decisions in real-time with the goal of supporting operational availability and reliability.

While Netflix's scale is larger than most other companies, we believe the approaches discussed are highly relevant to other environments.

Motivation

The importance of automating operational decisions

Netflix.

We strive to provide an amazing experience to each member, winning the "moments of truth" where they decide what entertainment to enjoy.

Key business metrics.

- 62 million members
- 50 countries
- 1000 devices supported
- 3 billion hours per month

Netflix.

Cloud based infrastructure leveraging Amazon Web Services (AWS).

- Service oriented architecture
- Running in three AWS regions
- Hundreds of services (>700)
- Thousands of server instances
- Millions of metrics

Squishy Decisions.

Humans cannot continuously monitor the status of all these services.

- Human effort does not scale.
- People make “squishy” decisions.
  - Repeatability of decisions is important.
- Difficult to evaluate a squishy decision.
  - “That metric looks wonky”

Automated Decisions.

- Need tools that automatically analyze our environments.
- Make intelligent operational decisions in real-time.
    - Reproducible decisions.

Case Studies

- Automated Canary Analysis
- Server Outlier Detection
- Smarter Alerting

Automated Canary Analysis

No canaries were harmed in this presentation

Canary Release.

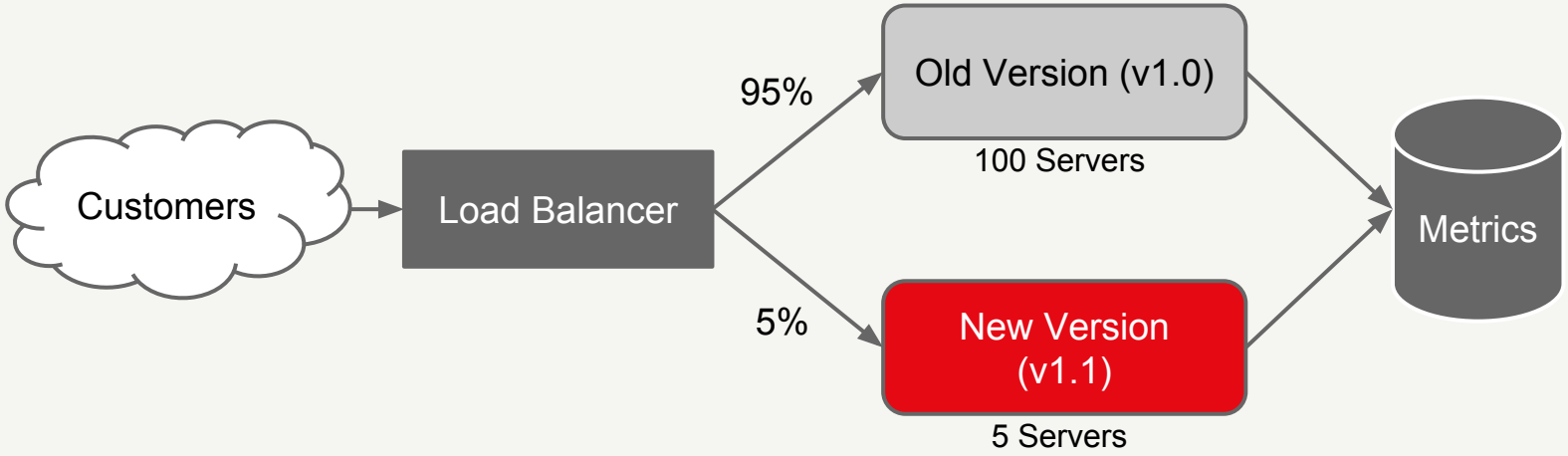
A deployment pattern where a new change is gradually rolled out to production.

- Checkpoints are performed to examine the new (canary) system.
- A go/no-go decision is made at each checkpoint.

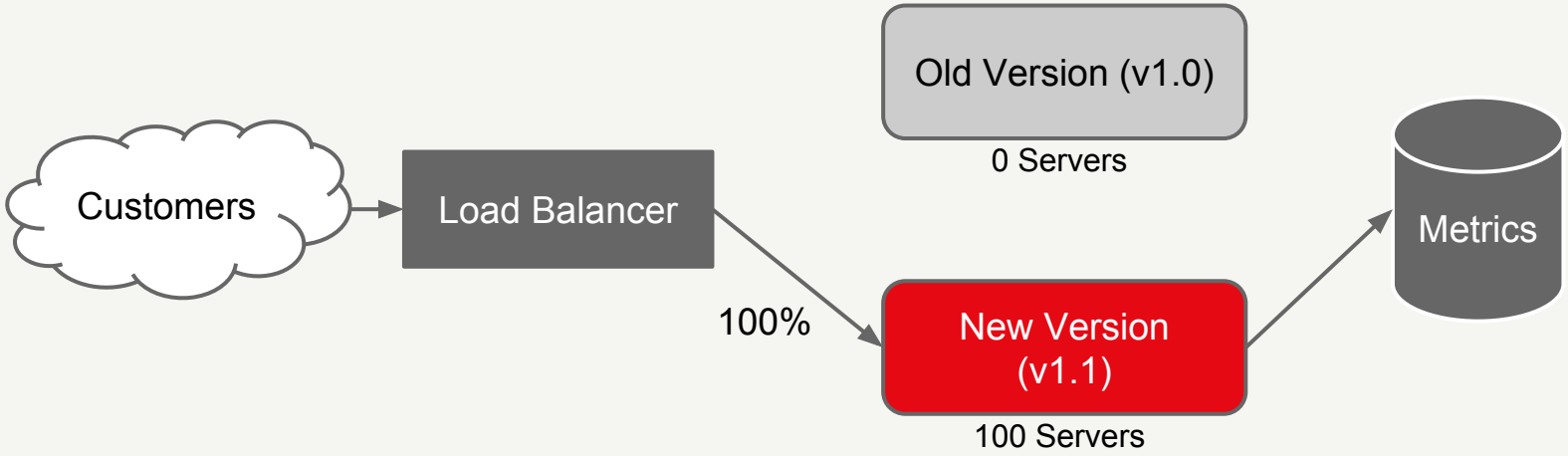
Canary Release is not:

- A replacement for any sort of software testing.
- Releasing 100% to production and hoping for the best.

Canary Release Process.



Canary Release Process.

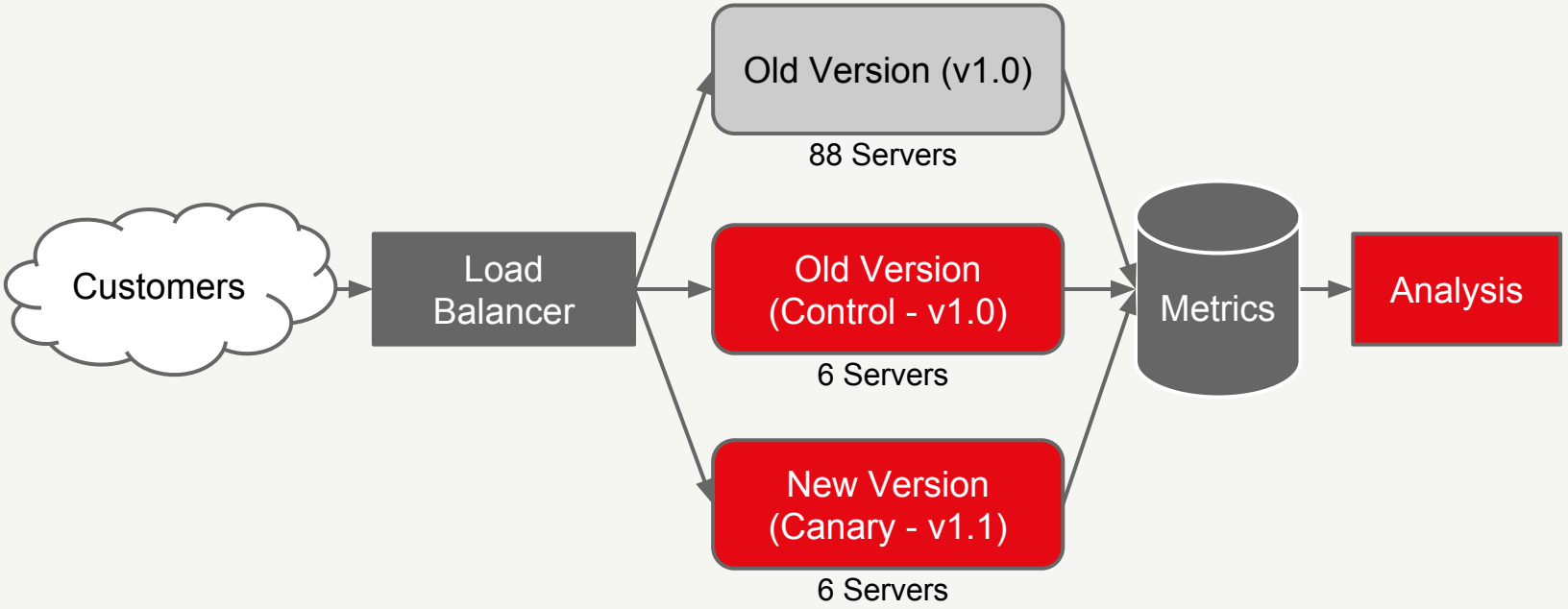


Canary Release.

Advantages of a canary release

- Better degree of trust and safety in deployments.
- Faster deployment cadence.
- Helps to identify issues with production ready code.
- Lower investment in simulation engineering.

Netflix Canary Release Process.



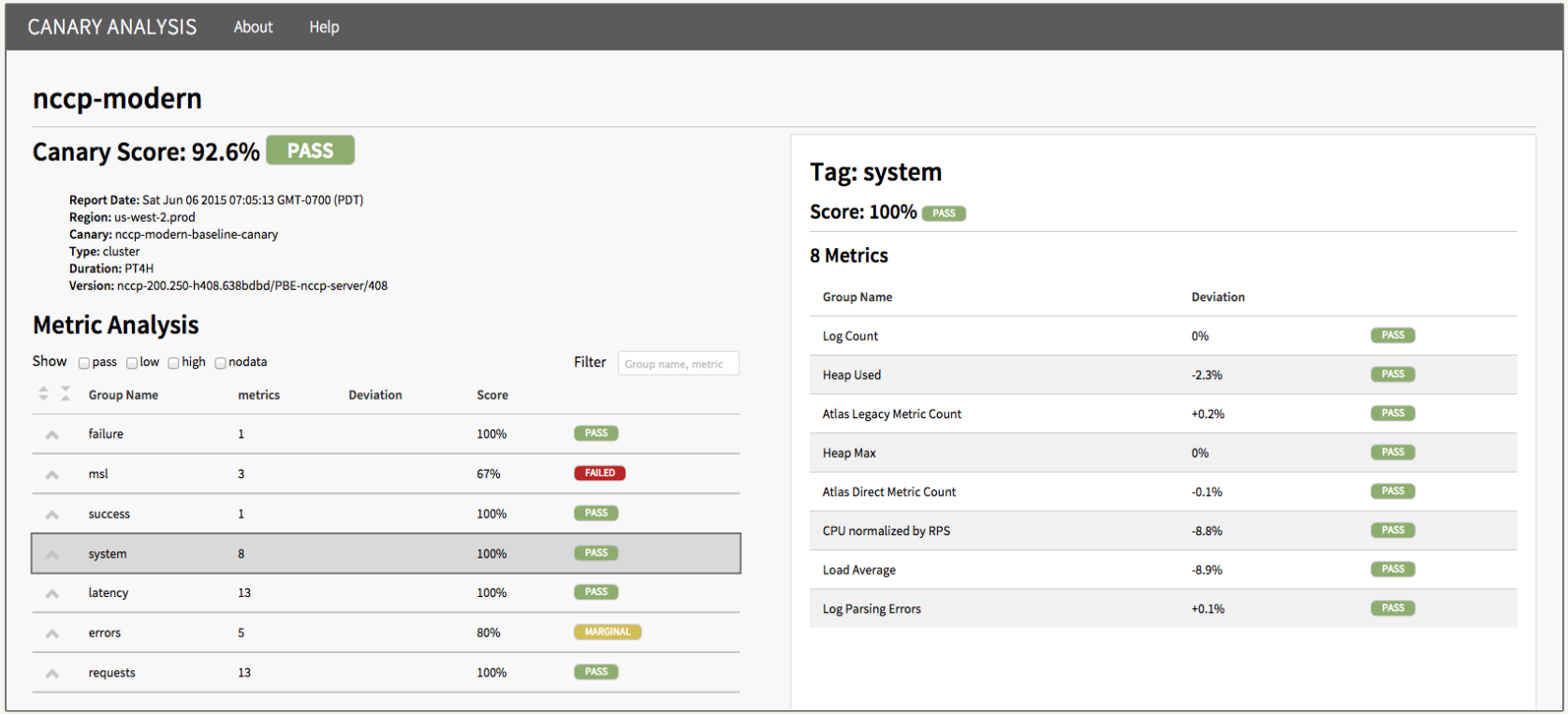
Automated Analysis.

- For a set of metrics compare the canary and control.
- Identify any canary metrics that deviate from the control.
- Generate a score that indicates the overall similarity.
- Associate a go/no-go decision based on the score.

Automated Analysis.

Every *n* minutes perform the following:

- For each metric:
    1. Compute the mean value for the canary and control.
    2. Calculate the ratio of the mean values.
    3. Classify the ratio as high, low, etc.
- Compute the final canary score.
    - Percentage of metrics that match in performance.
- Make go/no-go decision based on the score.
    - Continue with release if score is > 95%


CANARY ANALYSIS
About
Help
nccp-modern
Canary Score: 92.6%
PASS
Report Date: Sat Jun 06 2015 07:05:13 GMT-0700 (PDT)
Region: us-west-2.prod
Canary: nccp-modern-baseline-canary
Type: cluster
Duration: PT4H
Version: nccp-200.250-h408.638bdbd/PBE-nccp-server/408
Metric Analysis
Show pass low high nodata
Filter
Group name, metric
Group Name metrics Deviation Score
failure 1 100% PASS
msl 3 67% FAILED
success 1 100% PASS
system 8 100% PASS
latency 13 100% PASS
errors 5 80% MARGINAL
requests 13 100% PASS
Tag: system
Score: 100% PASS
8 Metrics
Group Name Deviation
Log Count 0% PASS
Heap Used -2.3% PASS
Atlas Legacy Metric Count +0.2% PASS
Heap Max 0% PASS
Atlas Direct Metric Count -0.1% PASS
CPU normalized by RPS -8.8% PASS
Load Average -8.9% PASS
Log Parsing Errors +0.1% PASS

The Numbers.

Duration	Num. ACA	Resources	Fail %
Past 7 Days	1557	142	19%
Past 4 Weeks	6309	432	16%
Past 8 Weeks	12482	823	16%

Considerations.

- Selecting the right application and performance metrics.
- Frequency of analysis.
- Amount of data for the analysis.
- A canary metric that deviates from the control might not indicate an issue.
- A single “outlier” server can skew the analysis.

Server Outlier Detection

Servers gone wild!

Somewhere out there a few unhealthy servers are among thousands of healthy ones.

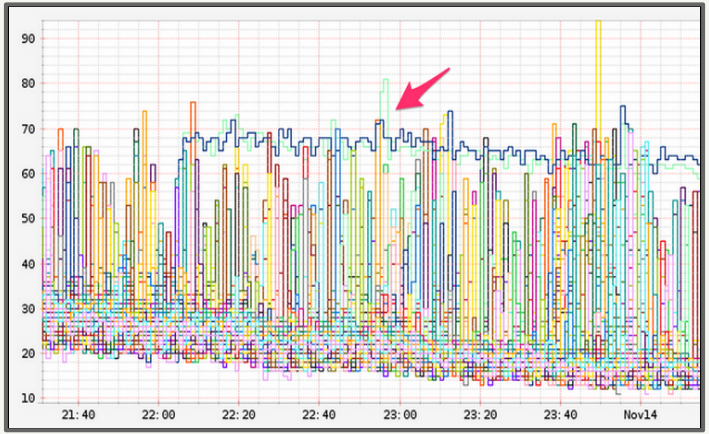
A Wolf Among Sheep.

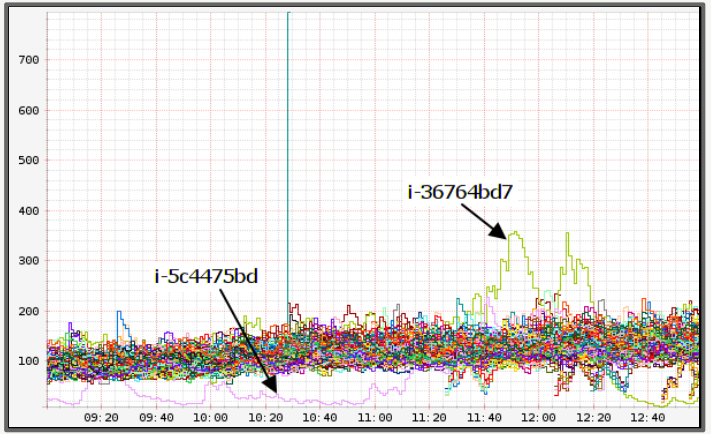
Netflix currently runs on thousands of servers

- We typically see a small percentage of those become unhealthy.
- Effects can be small enough to stay within the tolerances of our monitoring system.
- Time is wasted paging through graphs looking for evidence.
- Customer experience may be degraded.

We need a near-real time system for detecting server instances that are not behaving like their peers.

Examples.


90
80
70
60
50
40
30
20
10
21:40
22:00
22:20
22:40
23:00
23:20
23:40
Nov14


700
600
500
400
300
200
100
i-36764bd7
i-5c4475bd
09:20
09:40
10:00
10:20
10:40
11:00
11:20
11:40
12:00
12:20
12:40

Solution.

- We have lots of unlabelled data about each server.
- Servers running the same hardware and software should behave similar.

Cluster Analysis

- Task of grouping objects in such a way that objects in the same group are more similar to each other than those in other groups.
- Unsupervised machine learning.

DBSCAN.

- Density-Based Spatial Clustering of Applications with Noise.
- Iterate over a set of points and marks those in regions with many neighbors as clusters and mark those in lower density regions as outliers.

Conceptually

- If a point belongs to a cluster it should be near lots of other points as measured by some distance function.

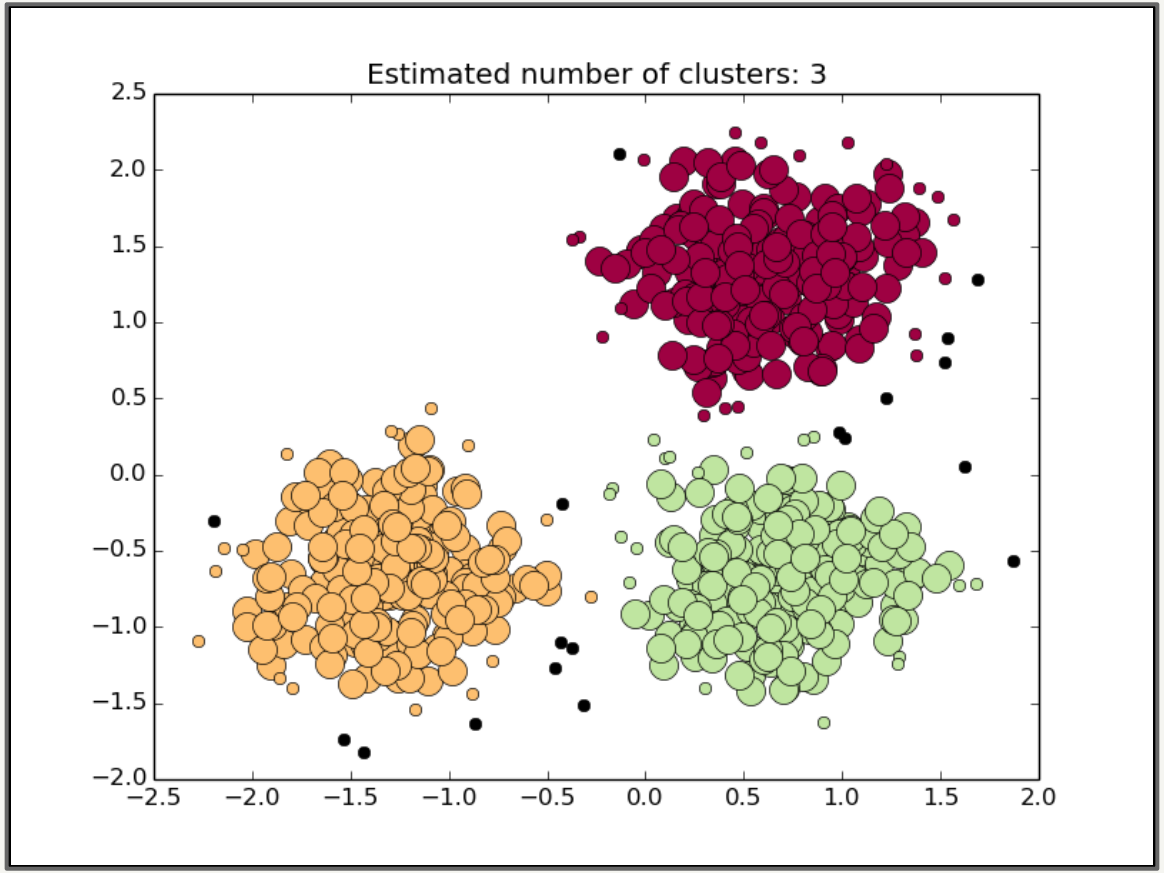


Image from the scikit-learn documentation

In Practice.

- Collect a window of data from our telemetry system.
- Run DBSCAN on the window of data.
- Process the results and apply rules defined by the server owner.
    - Ex. ignore servers which are out of service.
- Perform an action
    - Terminate
    - Remove from service

The Secret Sauce.

- DBSCAN has two input parameters which need to be selected.
- Service owners do not want to think about finding the right parameters.

Compromise

- Users define the number of outliers at configuration time.
- Based on this knowledge, the parameters are selected using simulated annealing.

The Numbers.

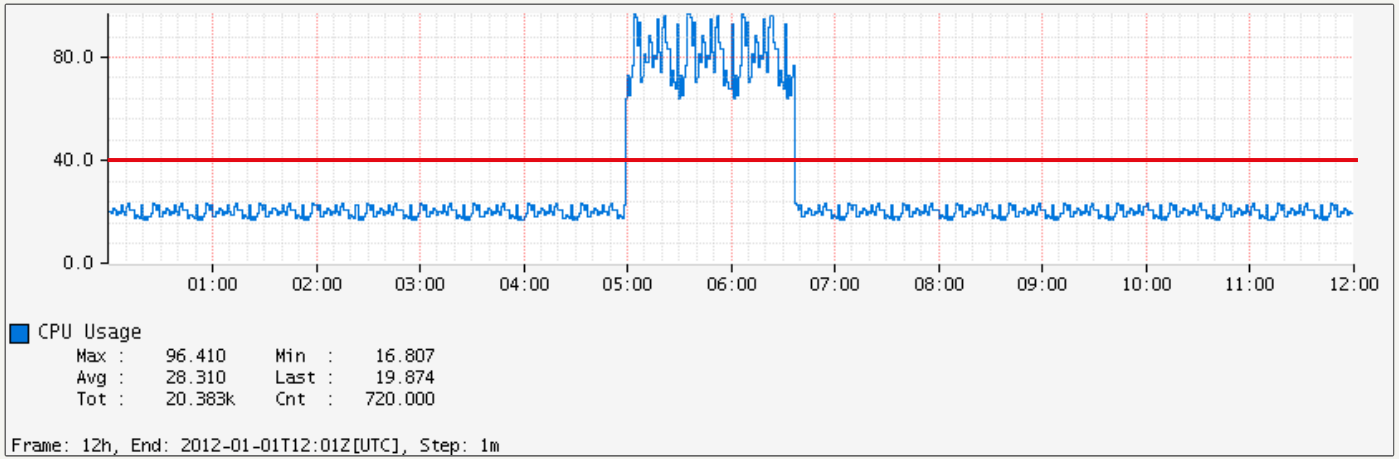
Duration	Remediations
Past 7 Days	225
Past 4 Weeks	739
Past 12 Weeks	1395

Smarter Alerting

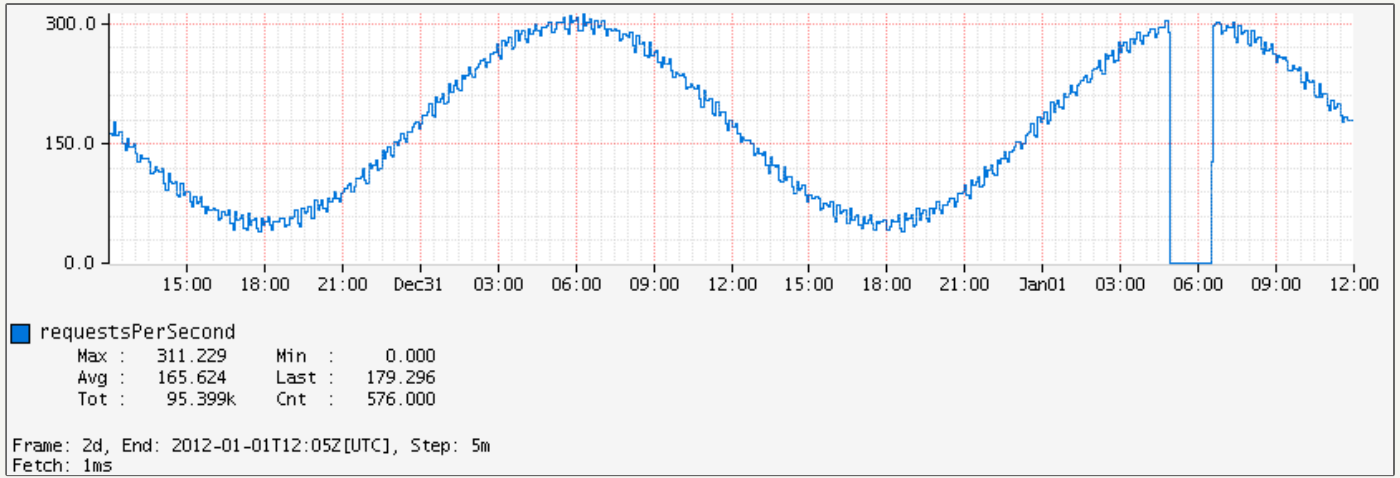
The boy who cried wolf

Alerting is an important part of any system to let you know when things go bump in the night.

Stationary Signals.



Periodic Signals.



Anomaly Detection.

Techniques and Libraries

- Robust Anomaly Detection (RAD) - Netflix
- Seasonal Hybrid ESD - Twitter
- Extendible Generic Anomaly Detection System (EGADS) - Yahoo
- Kale - Etsy

Books

- Outlier Analysis - Charu Aggarwal
- Robust Regression and Outlier Detection - Rousseeuw and Leroy

Waking up at 3AM due to a false alarm is not fun.

The Art of Drifting.

- Performance of systems drift over time.
- Models need to be updated to account for this drift.
- Expectations of users can change over time.
    - New models may need to be used to meet expectations.
- Manually updating models and parameters does not scale.

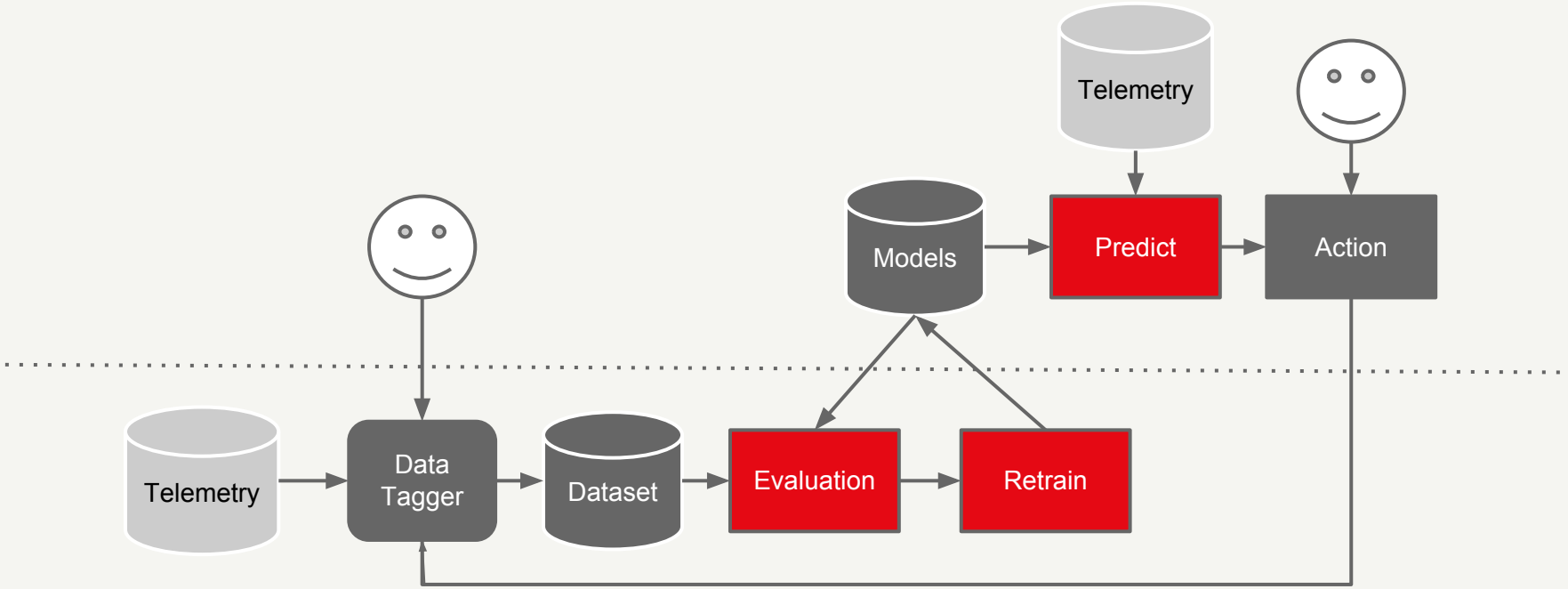
Models need to be constantly evaluated and updated.

Evaluation and Feedback.

Handling Drift

- Evaluate models against ground-truth periodically.
  - Evaluate each model against benchmark data nightly.
  - Retrain models when performance degrades.
  - Automatically switch to more accurate models.
- Capture when users think a model has drifted.
- Make it easy to capture and annotate new data for testing.

Online



Offline

Considerations.

- User feedback may not be accurate and inconsistent.
  - Accuracy of timestamps for an anomaly.
- How to prevent overfitting.

Bootstrap your data

- Generate synthetic data.
- Yahoo Anomaly Benchmark dataset.
- Open Source Time-series databases.

Lessons Learned

Lessons from production

Chance of Fog.

- Visibility into why a decision was made is important.
  - Reports
  - Graphs
  - Visualizations
- User may trust your system, but want to understand why a decision was made.
  - “Why was that instance terminated?”
- Debugging machine learning models can be time consuming.
  - Proper instrumentation can help mitigate this.

The Last Mile.

“Machine learning is really good at partially solving just about any problem.”

- Relatively easy to build a model that achieves 80% accuracy.
- After that, the returns on time, brainpower, and data diminish rapidly.
- You will spend a few months getting to 80%.
  - Last 20%: between a few years and eternity.
- Learn when good is good enough.
  - In some domains, you may only need to be 80% accurate.

Being Wrong.

Sometimes your systems is going to make the wrong decision.

- Don’t let Skynet out into the wild without a leash.
- Put in place reasonable safeguards.
  - Test those safeguards.
- The wrong decision is still a decision.
  - Learn from what went wrong.

The Future

The road ahead

Future Work.

- Measure the impact our decisions have on the environment.
- Frequent itemset mining.
  - Find combinations of events that are unusual.
- Meta alerting and alert correlation.
- Data stream mining and event stream processing.
  - Mantis - A reactive stream processor from Netflix.

Thank you.

NETFLIX