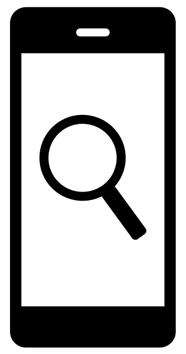
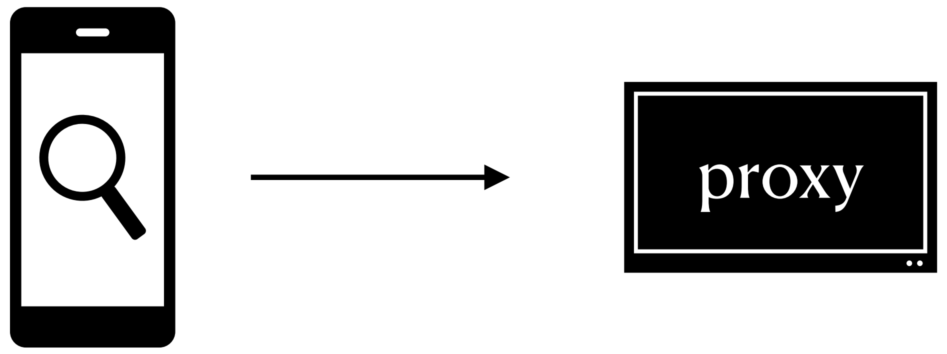
Cloud Telemetry Modeling via Residual Gauss-Markov Random Fields

Nick Landolfi, Dan O'Neill, Sanjay Lall

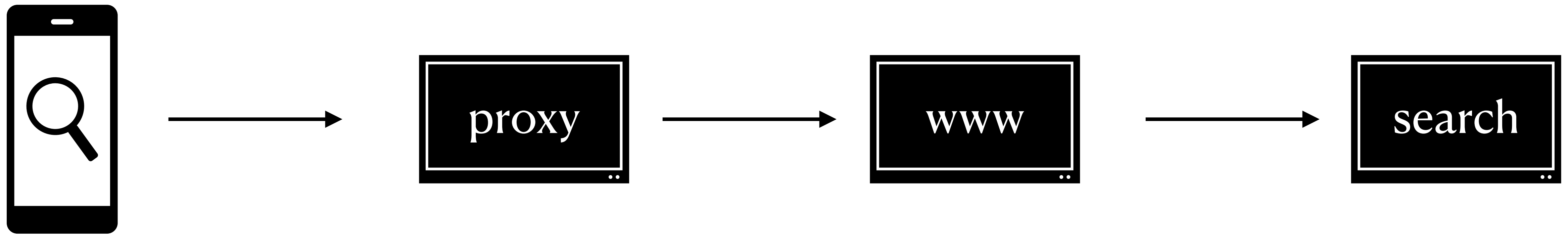
Stanford University

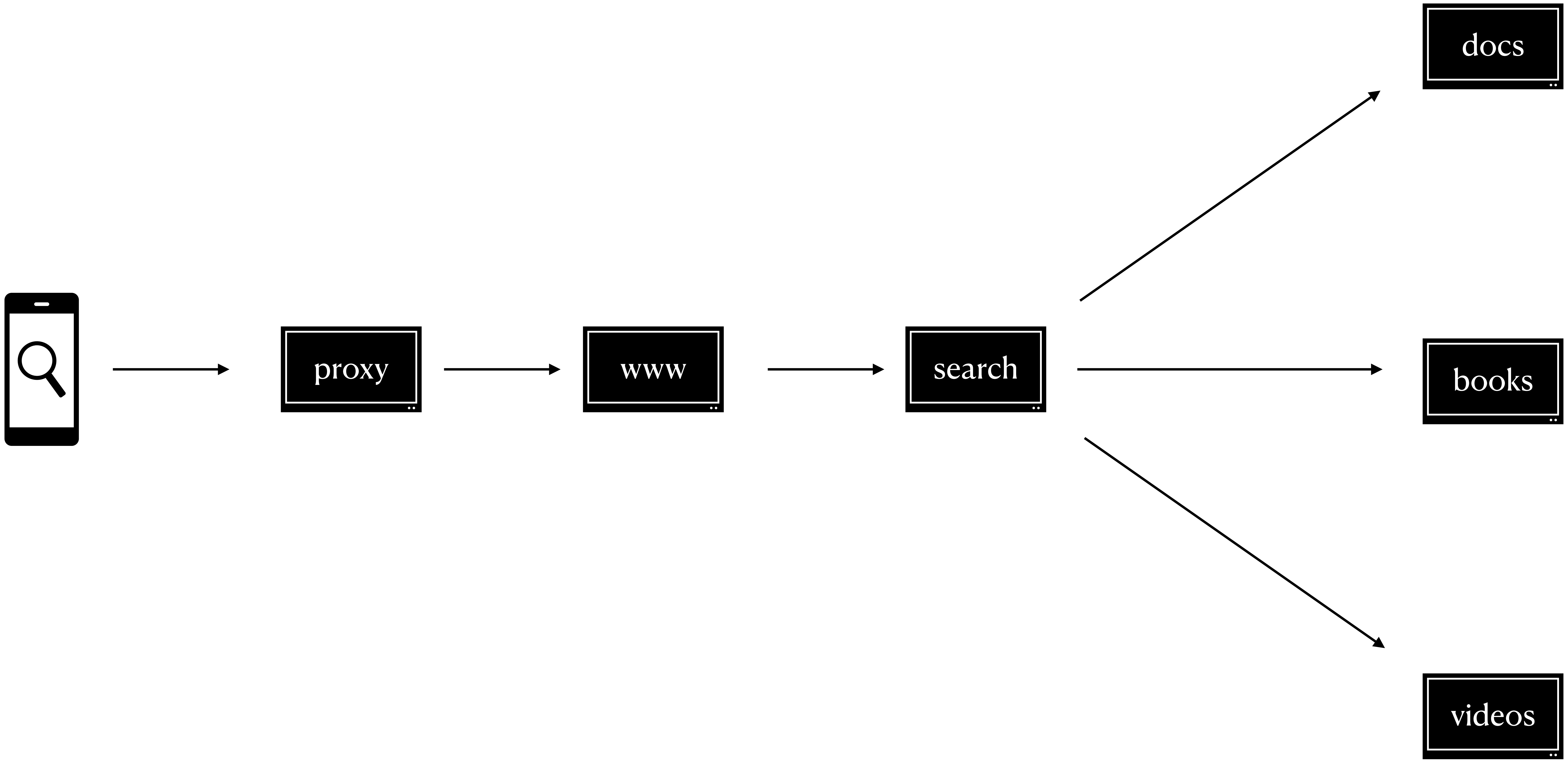

cloud systems are large

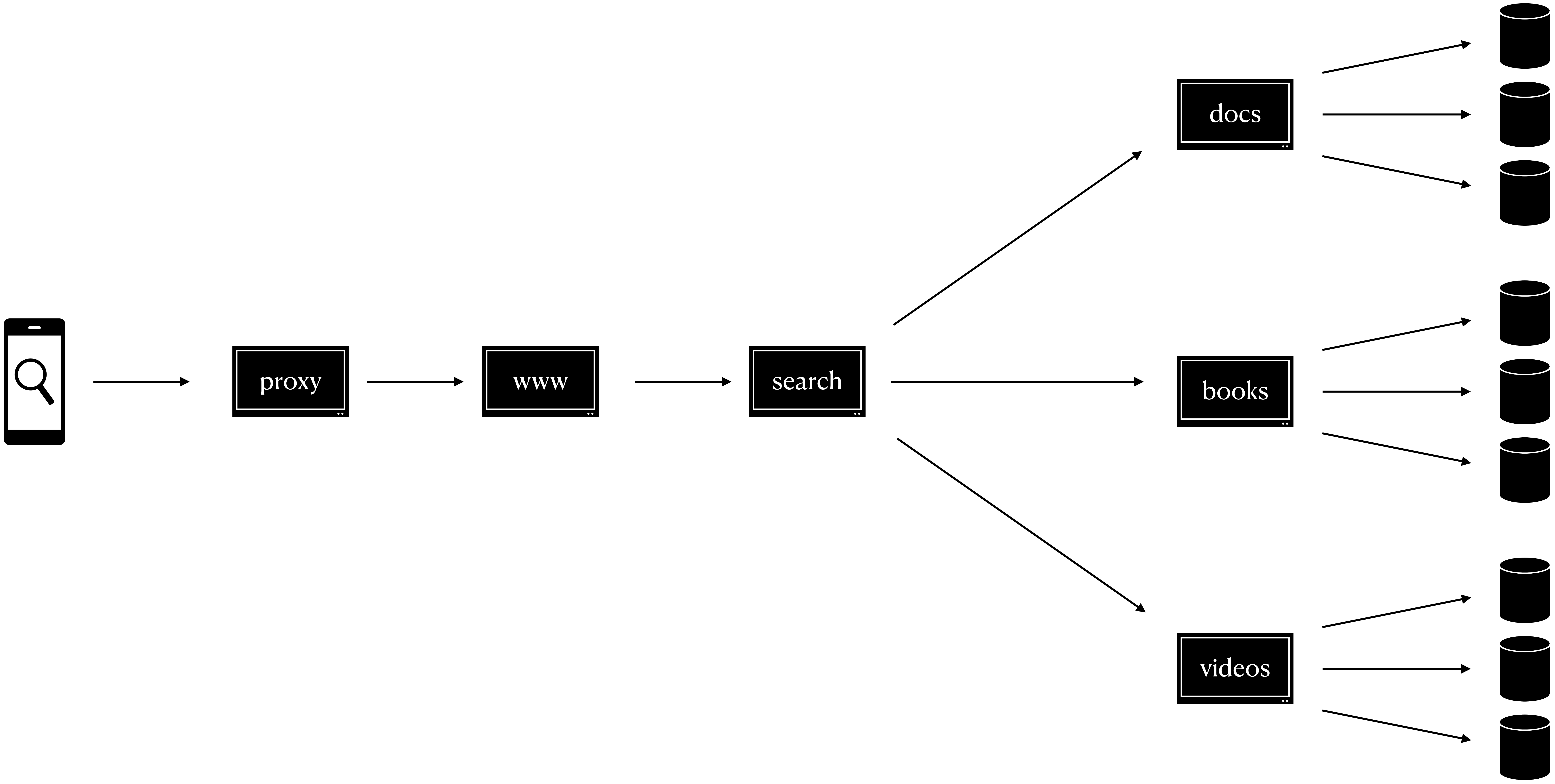


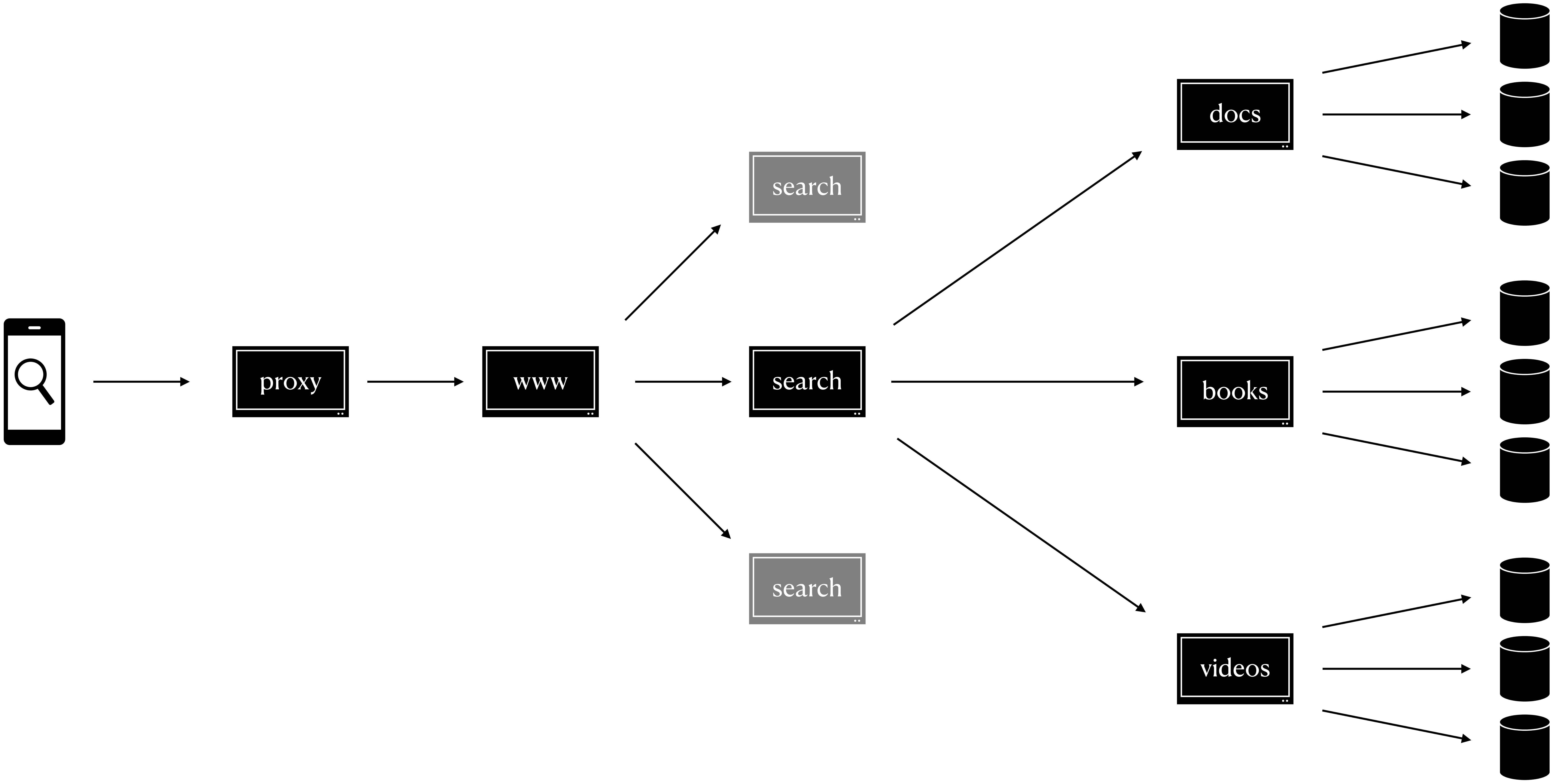

proxy

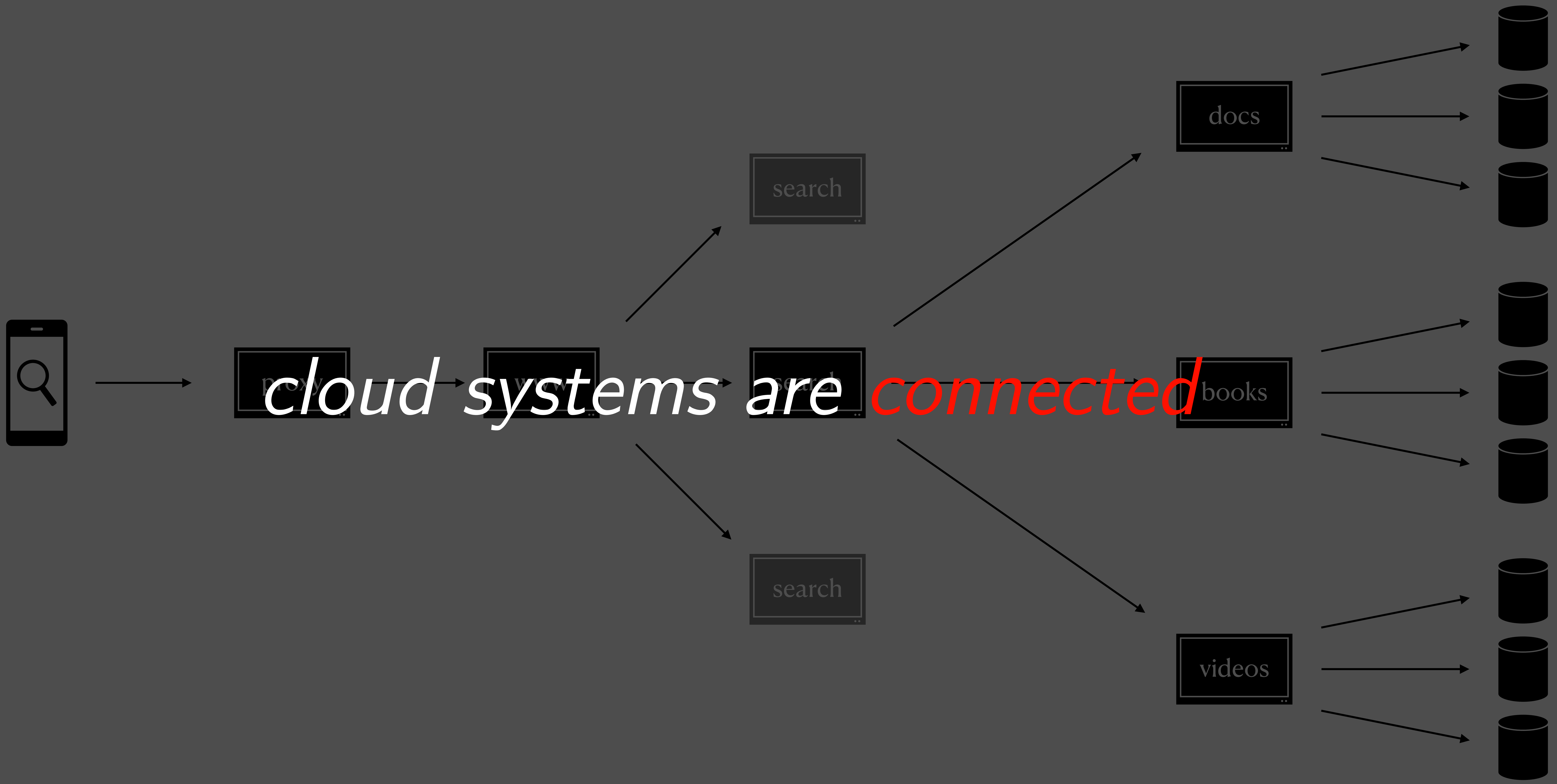

proxy
www


proxy
www
search

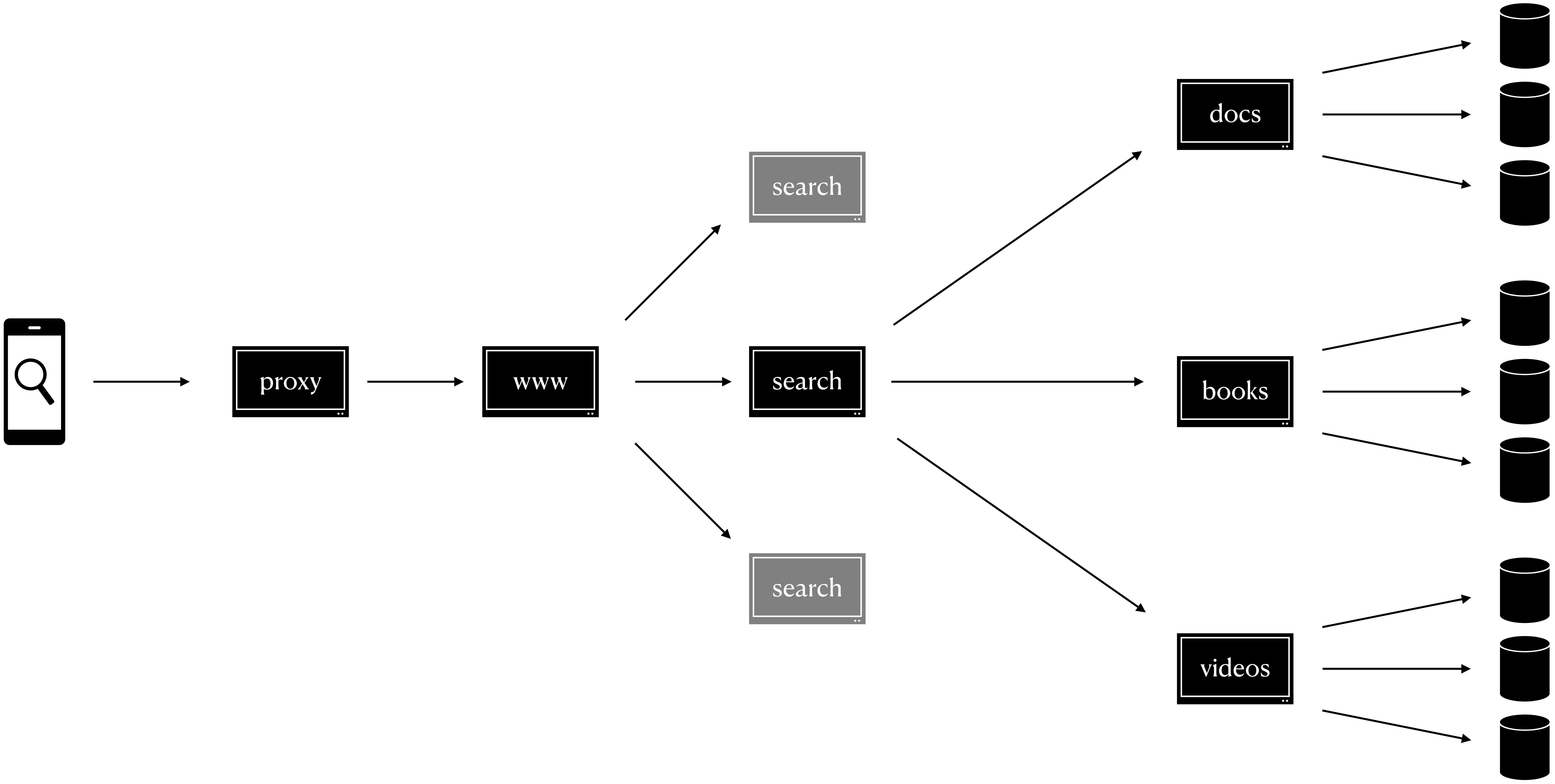

proxy
www
search
docs
books
videos


proxy
www
search
docs
books
videos

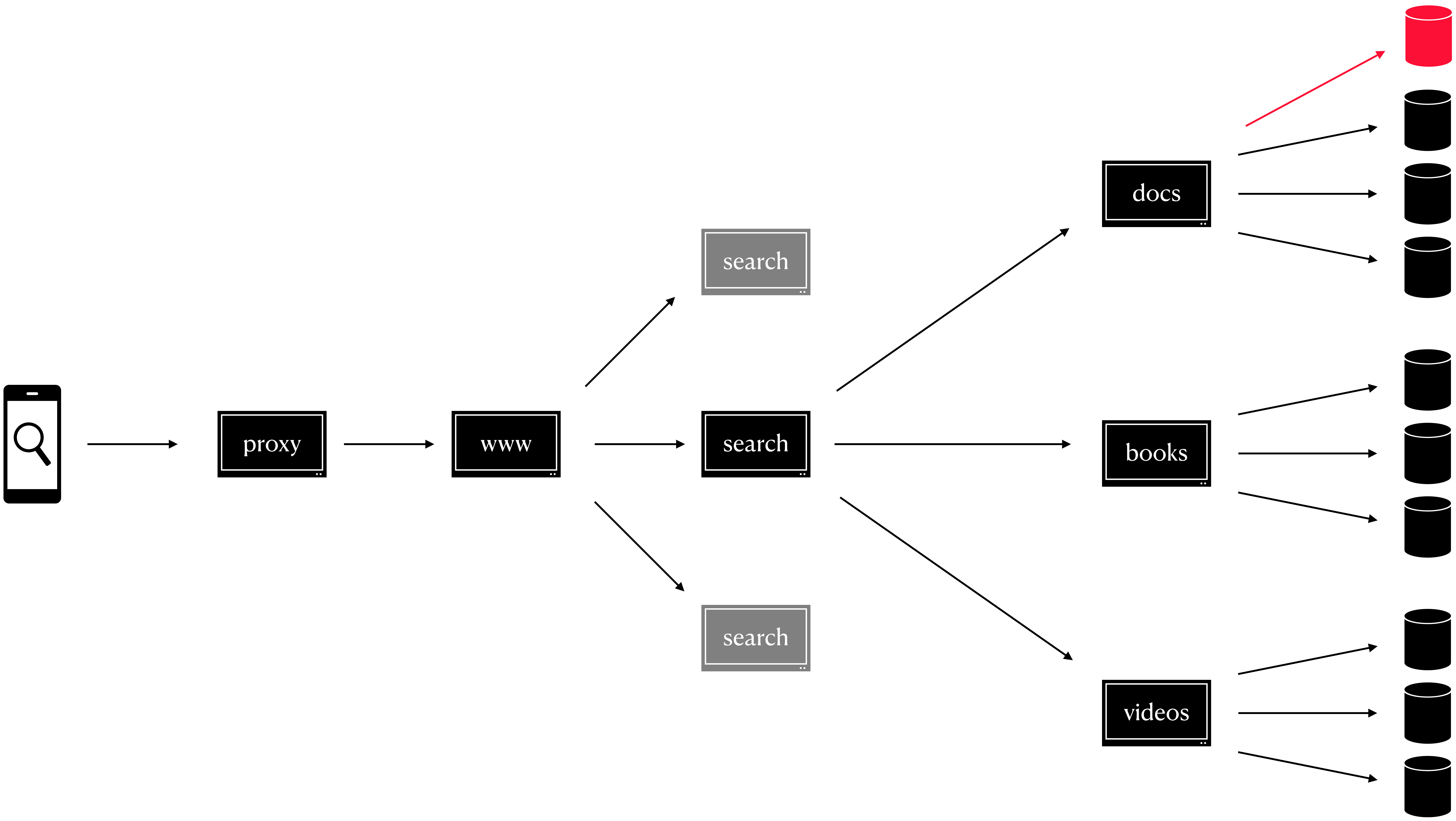

proxy
www
search
search
search
docs
books
videos

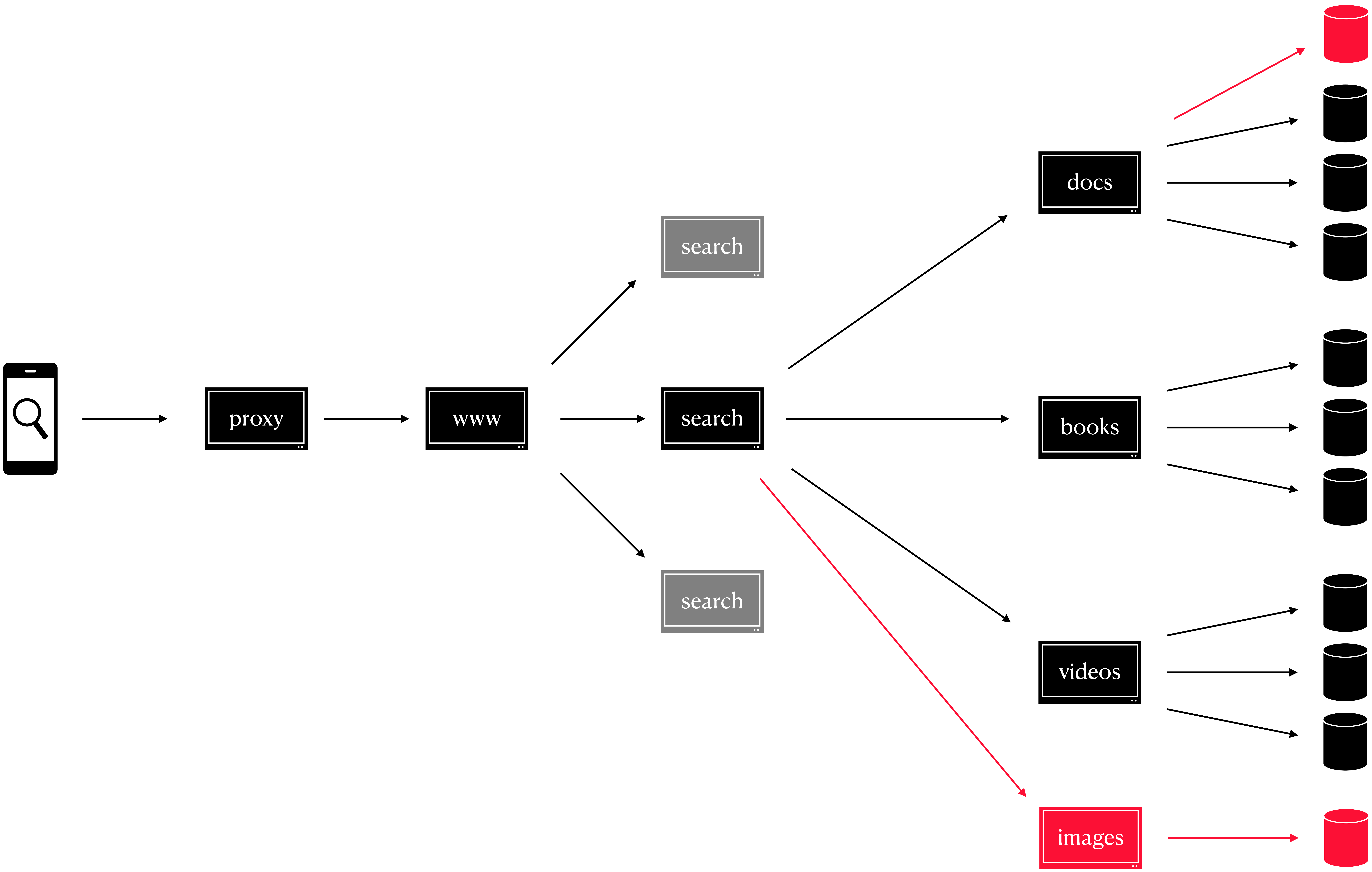


cloud systems are connected

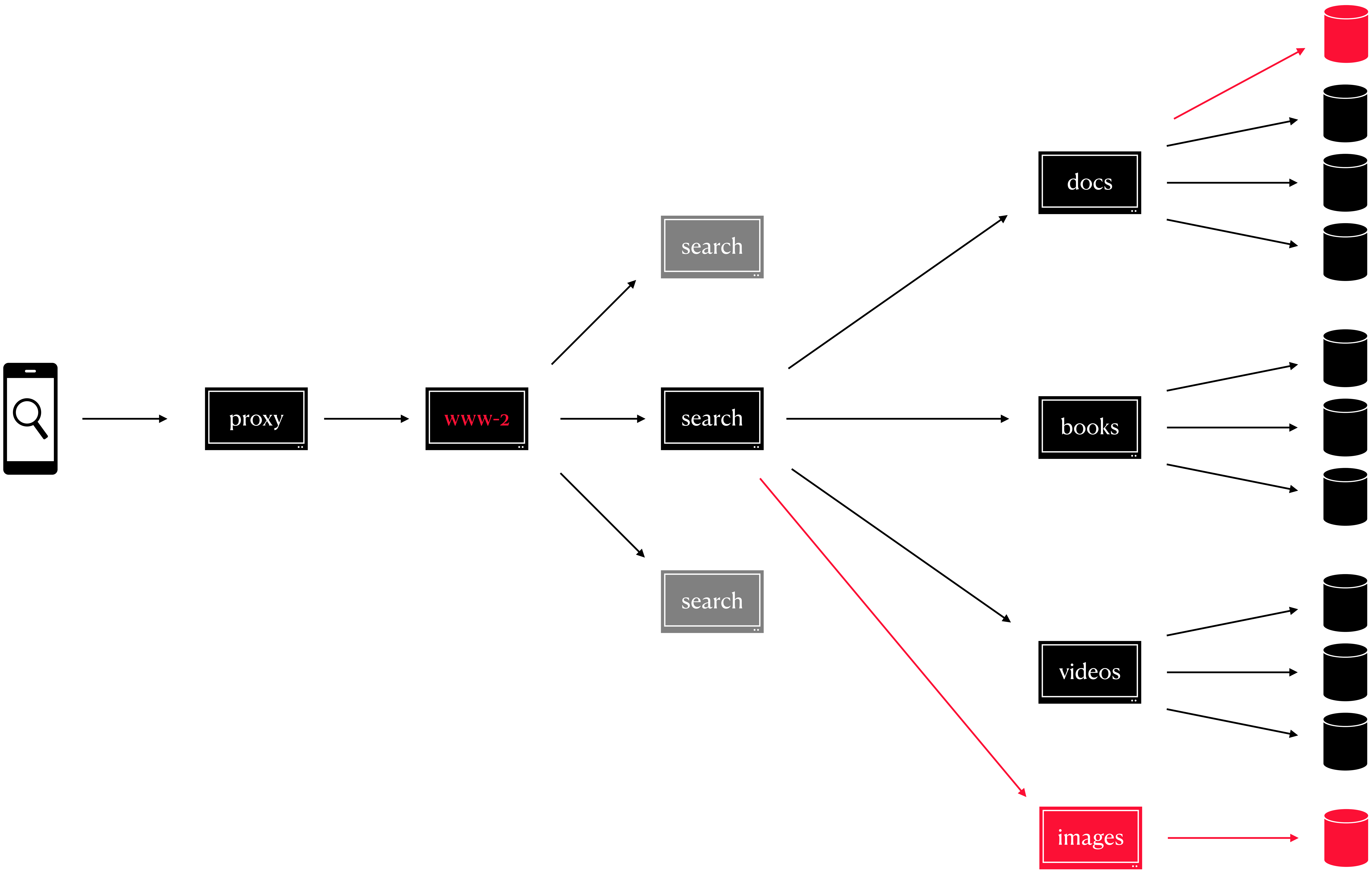


proxy
www
search
search
search
docs
books
videos

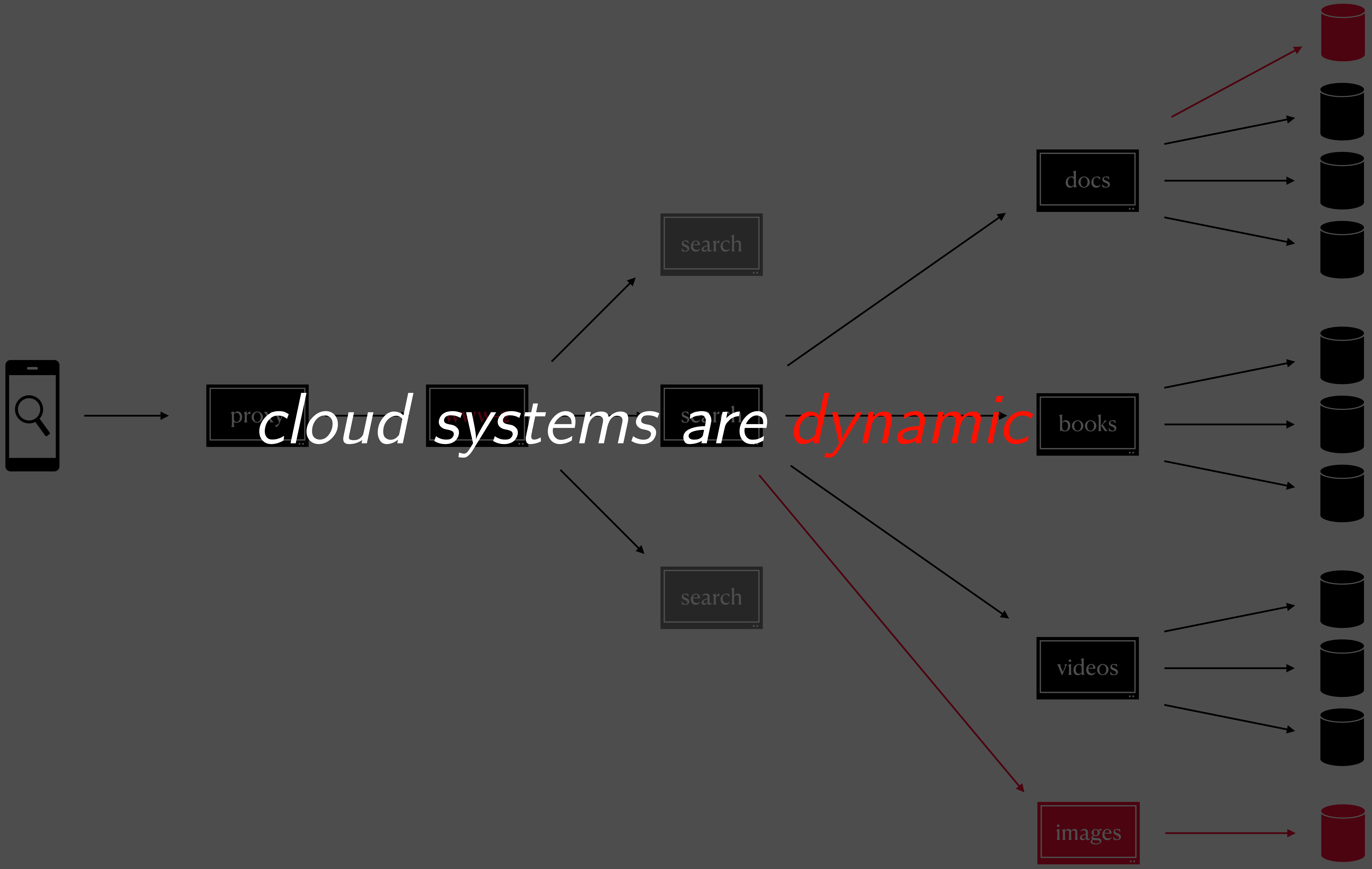

proxy
www
search
search
search
docs
books
videos



proxy
www
search
search
search
docs
books
videos
images


proxy
www-2
search
search
search
docs
books
videos
images

cloud systems are *dynamic*



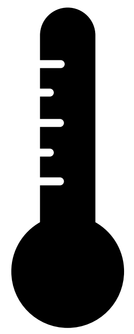
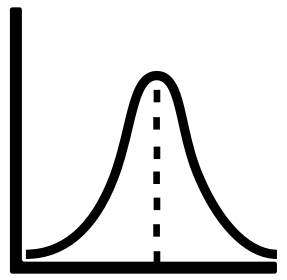
We want to characterize normal operating conditions and recognize changes

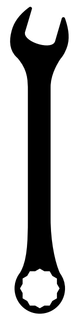
step 1: instrumentation

step 2: model construction

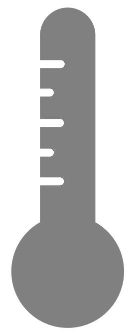
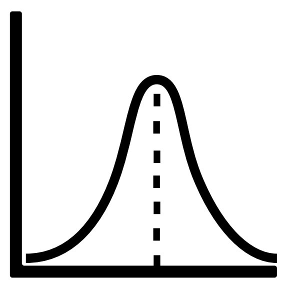
step 3: judge health

step 4: *problem remediation*



instrumentation, though difficult, largely addressed by prior work

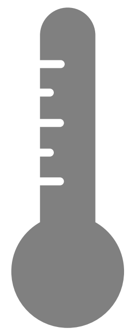
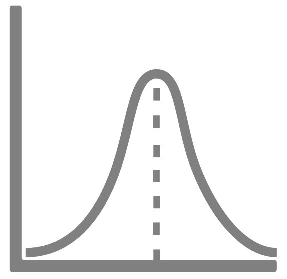
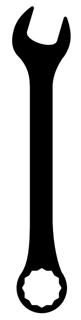



model construction is difficult, *our focus*

judge health via *anomaly detection*

can not experiment, only *unsupervised* data

remediate problems with help from *anomaly localization*

not all joint anomalies are marginal anomalies

need to model *spatial associations*

Our question

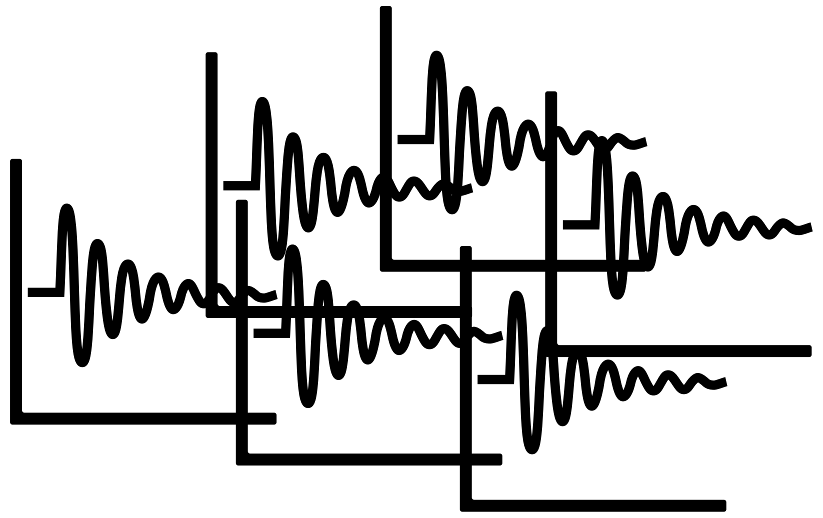
Can undirected graphical models characterize cloud telemetry?

Our question

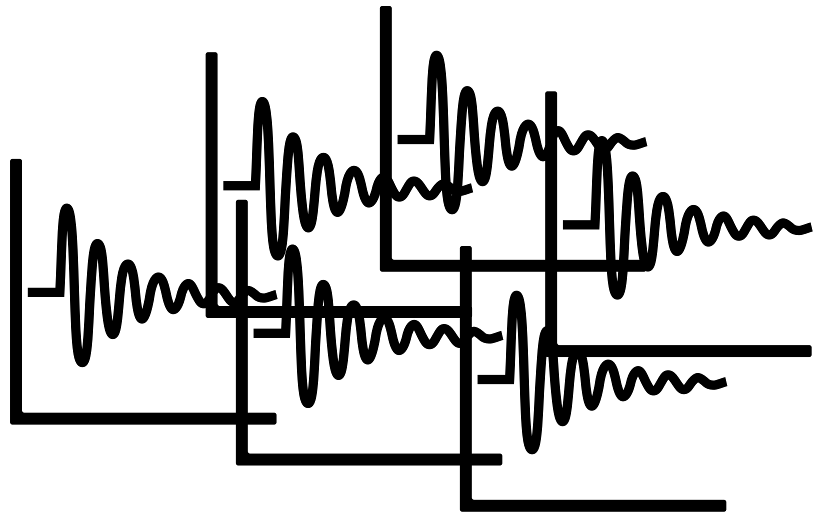
Can undirected graphical models characterize cloud telemetry?

we give a data model and positive preliminary results

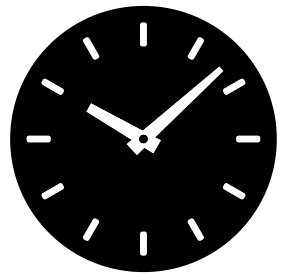
We have...

 unlabeled correlated signals

We have...

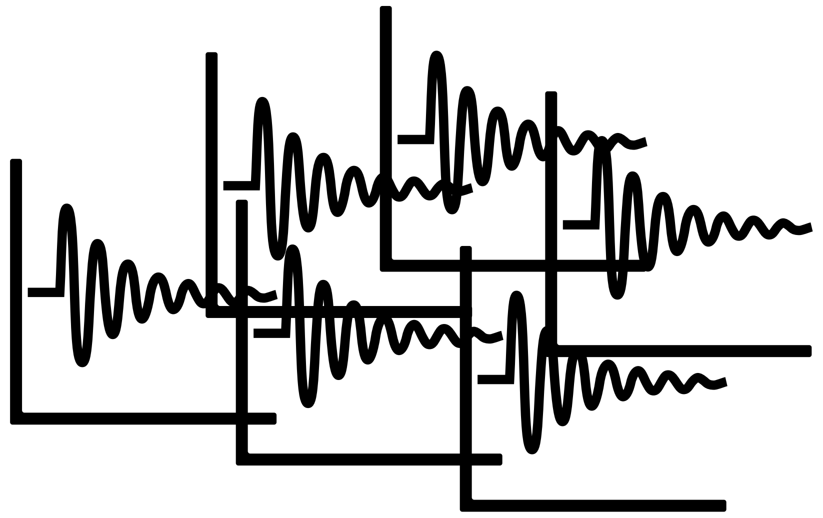


unlabeled correlated signals

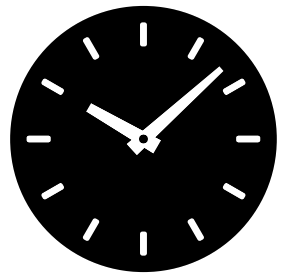


arriving in real-time

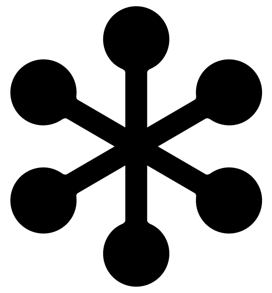
We have...



unlabeled correlated signals

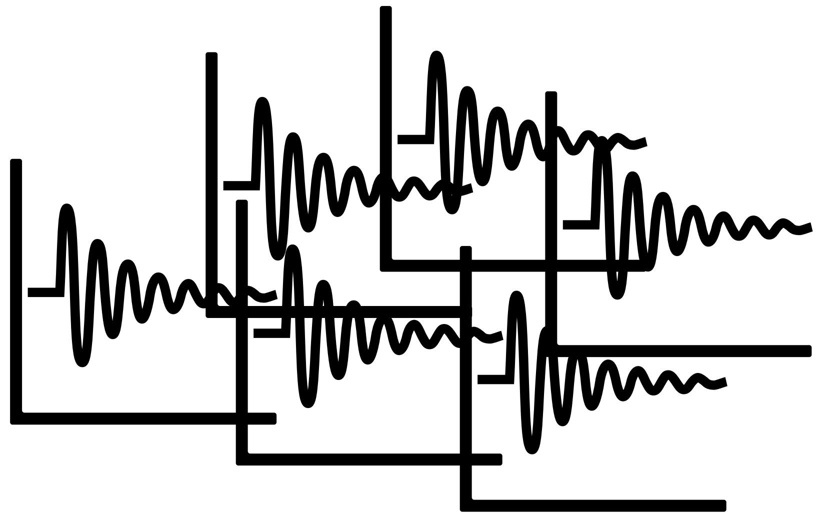


arriving in real-time

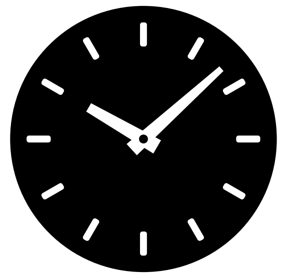


high-dimensional and spatial

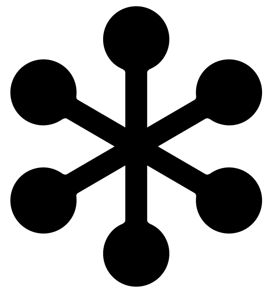
We have...



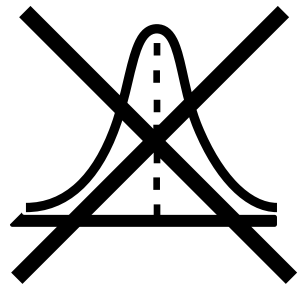
unlabeled correlated signals



arriving in real-time

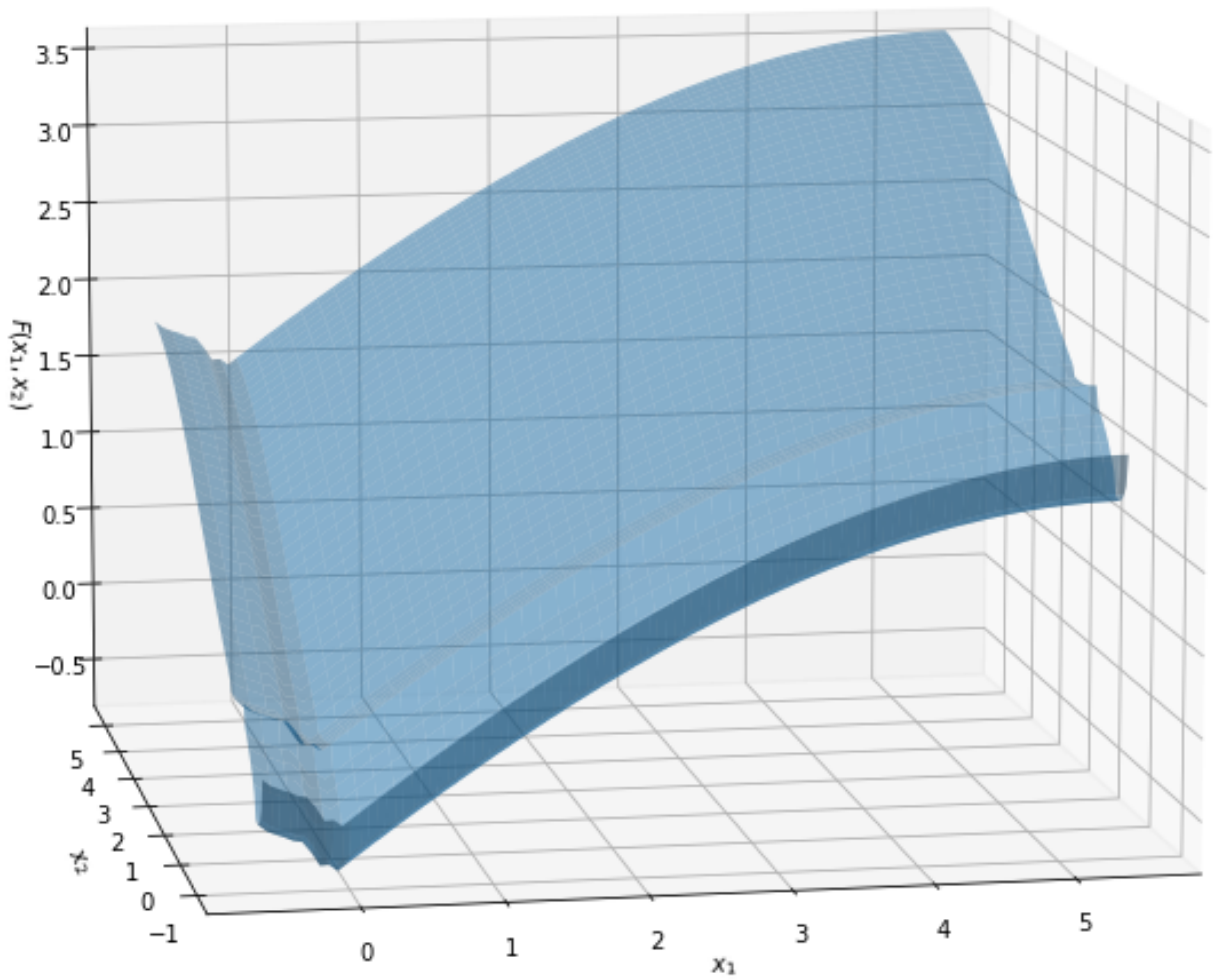


high-dimensional and spatial

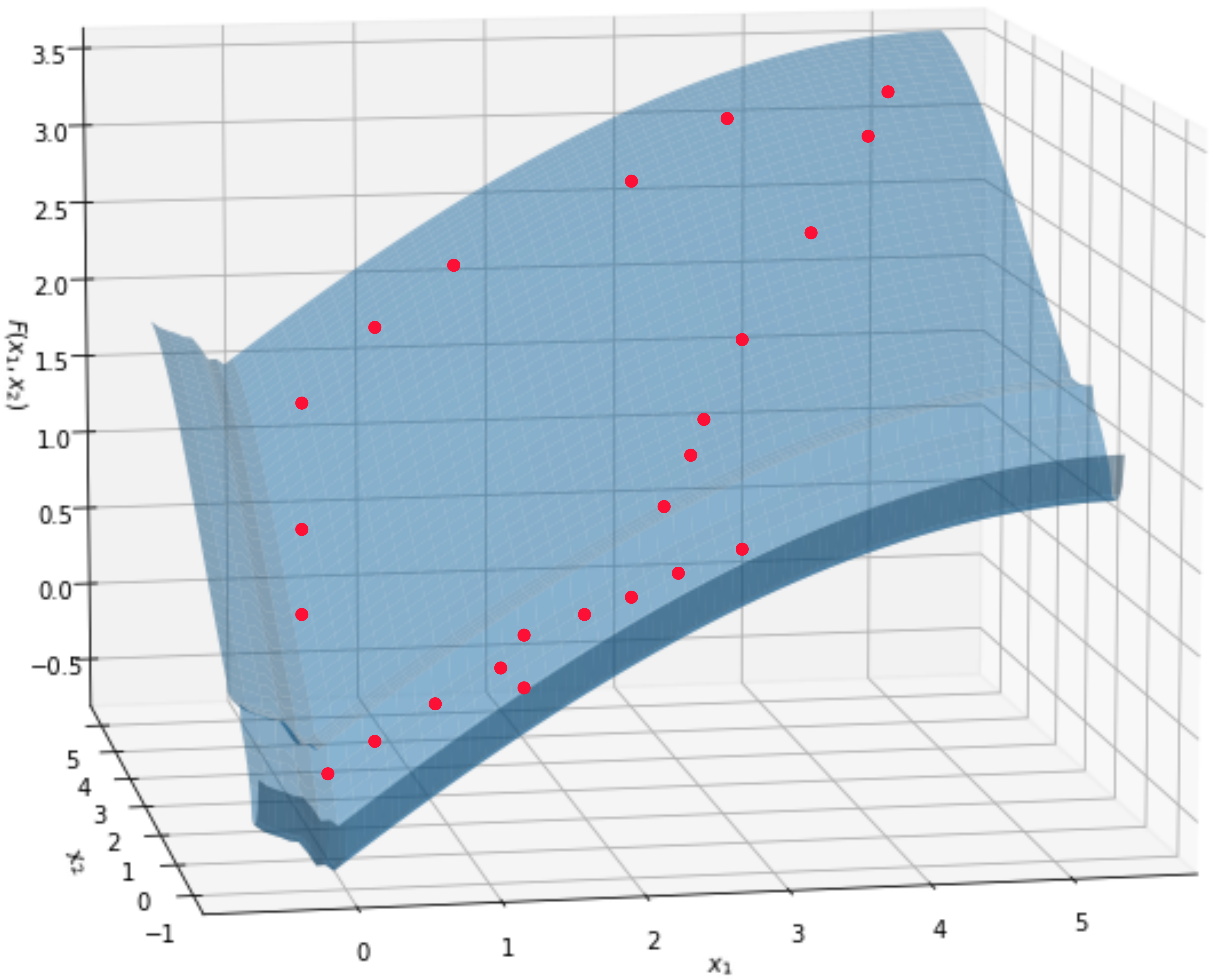


not normally distributed!

Data model: intution



Data model: intuition



Data model

$$\boldsymbol{x}$$

vector of measurements

Data model

$$\boldsymbol{x}_i$$

one component

Data model

$$\boldsymbol{x}_i \qquad \boldsymbol{x}_{-i}$$

the other components

Data model

$$x_i \qquad f_i(x_{-i})$$

assume exists

Data model

$$x_i \approx f_i(x_{-i})$$

can approximately *predict* each component from others

Data model

$$x_i - f_i(x_{-i})$$

take the difference

Data model

$$\delta_i = x_i - f_i(x_{-i})$$

call it the deviation

Data model

$$\delta_i = x_i - f_i(x_{-i})$$

model deviations as mean-zero *Gauss-Markov random field*

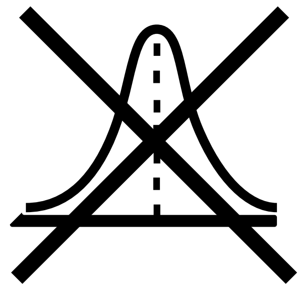
Data model

$$\delta_i = x_i - f_i(x_{-i})$$

✻ covariance of deviations accounts for *spatial* associations

Data model

$$\delta_i = x_i - f_i(x_{-i})$$

 normality an *approximation*, more practical for residuals

Data model: anomaly detection and localization

$$\delta_i = x_i - f_i(x_{-i})$$

joint anomalies — use a chi-squared test on vector of deviations

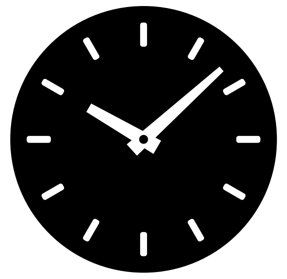
Data model: anomaly detection and localization

$$\delta_i = x_i - f_i(x_{-i})$$

individual anomalies — chi-squared test on conditional distribution

Data model: anomaly detection and localization

$$\delta_i = x_i - f_i(x_{-i})$$

detection and localization are simple and fast 

Data model: parameterization

need predictor parameters

$$\delta_i = x_i - f_i(x_{-i})$$

Data model: parameterization

need predictor parameters

$$\delta_i = x_i - f_i(x_{-i})$$

and covariance of residuals

Data model: parameterization

need predictor parameters

$$\delta_i = x_i - f_i(x_{-i})$$

and covariance of residuals

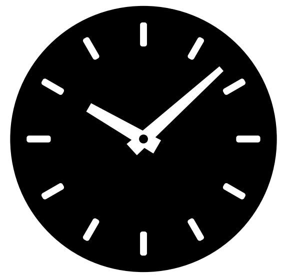
estimate by approximate *maximum likelihood*

Data model: parameterization

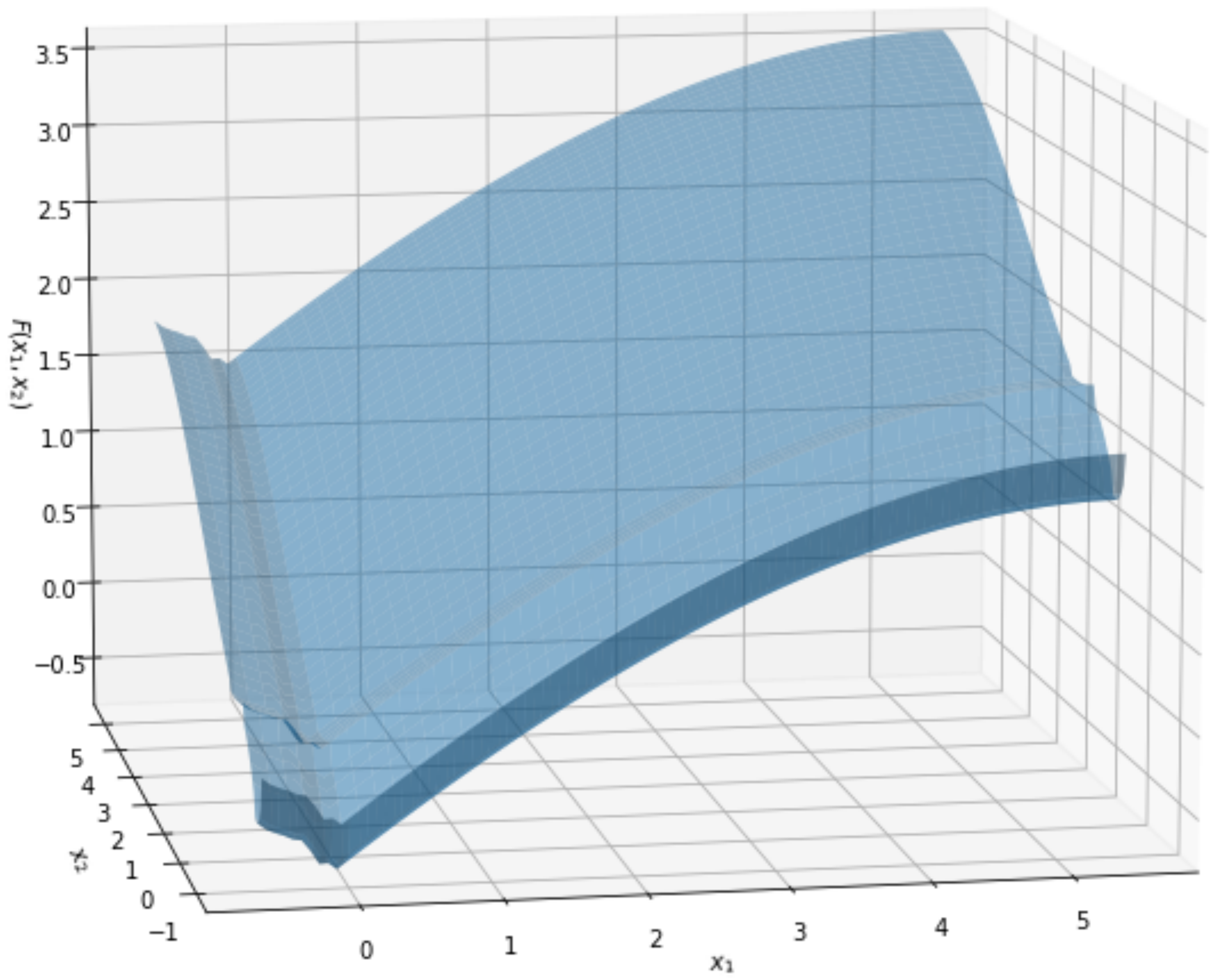
need predictor parameters

$$\delta_i = x_i - f_i(x_{-i})$$

and covariance of residuals

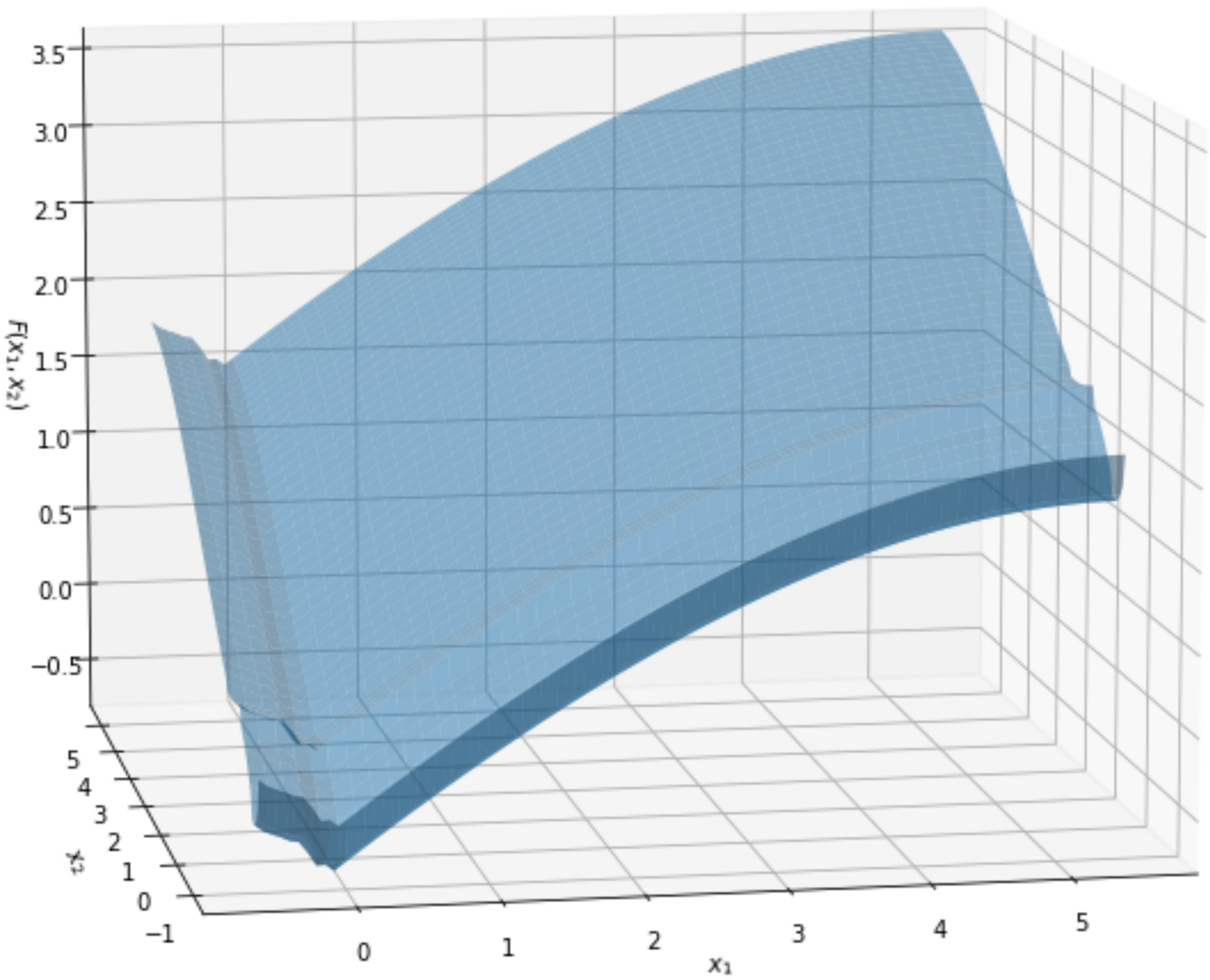
we solve two *convex programs*, which is fast 

Model interpretation

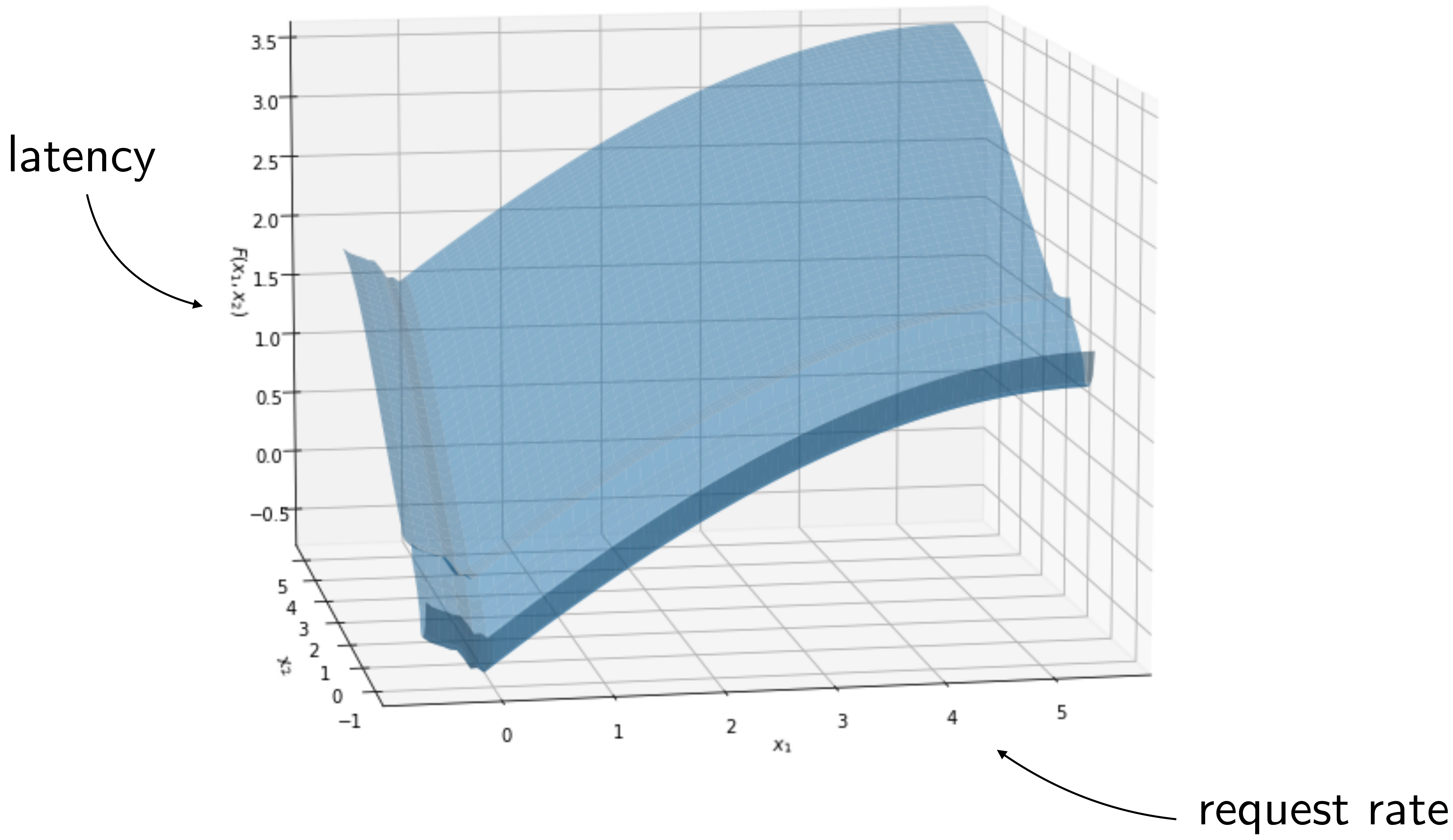


Model interpretation

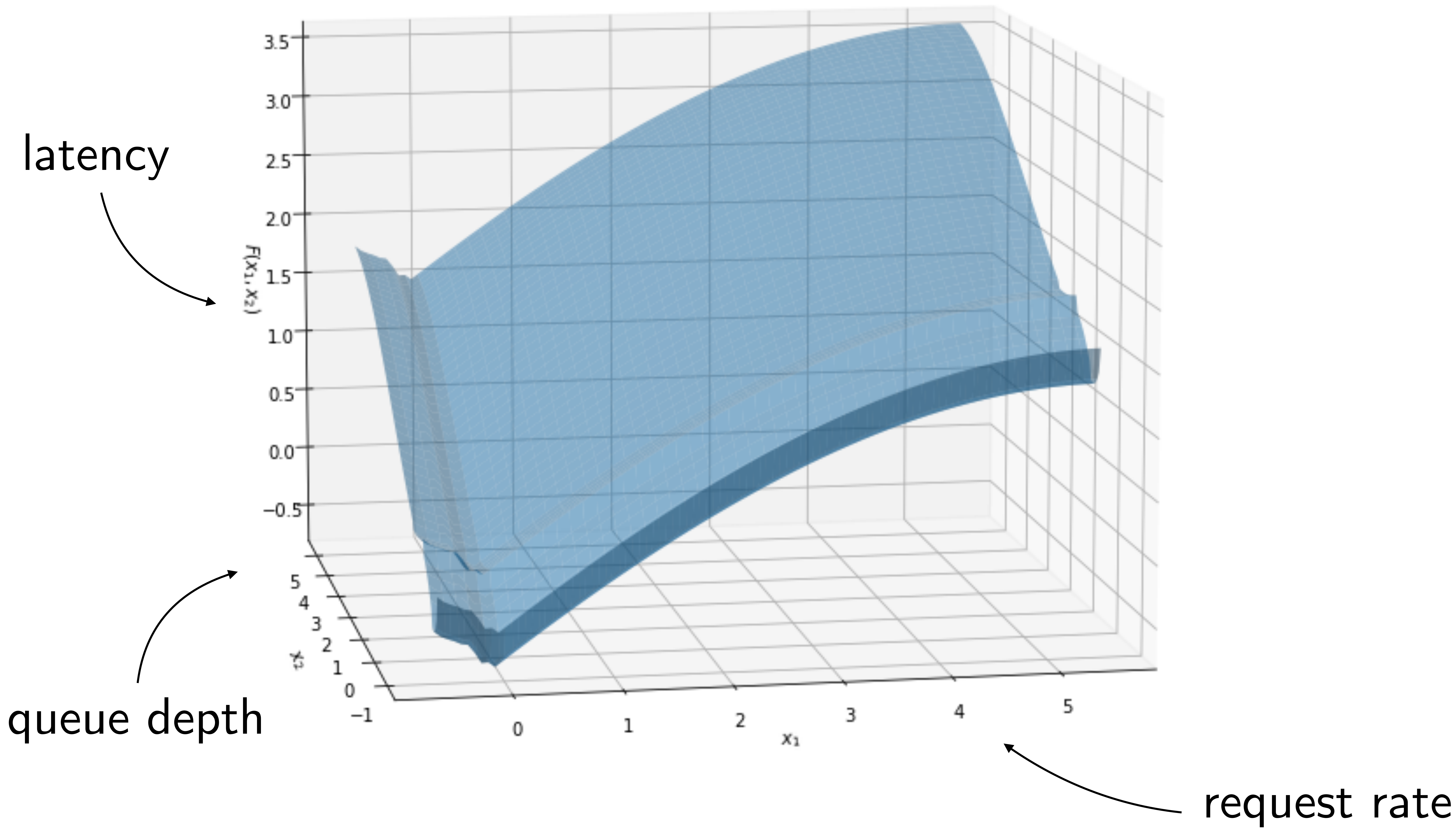
latency



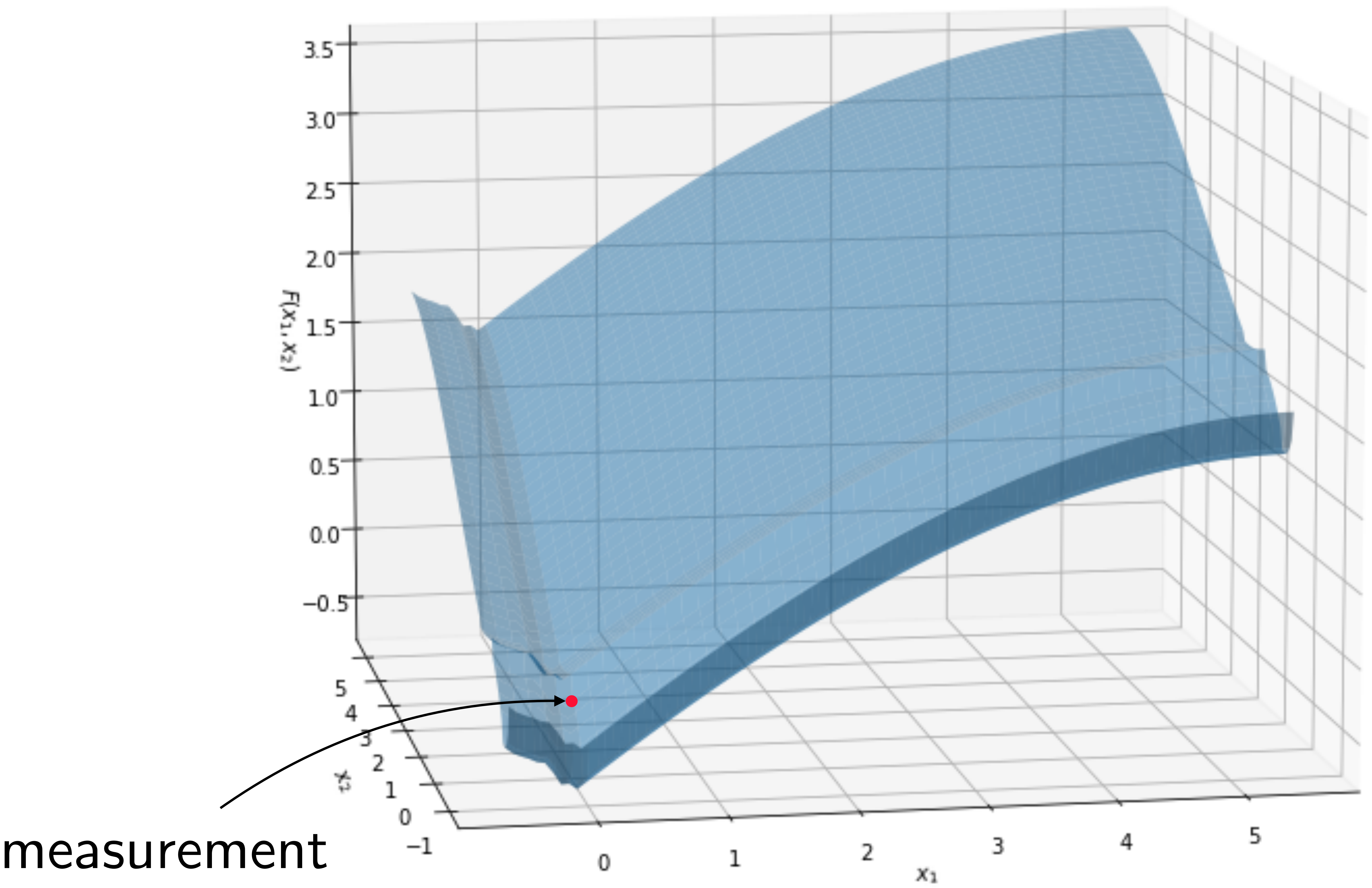
Model interpretation



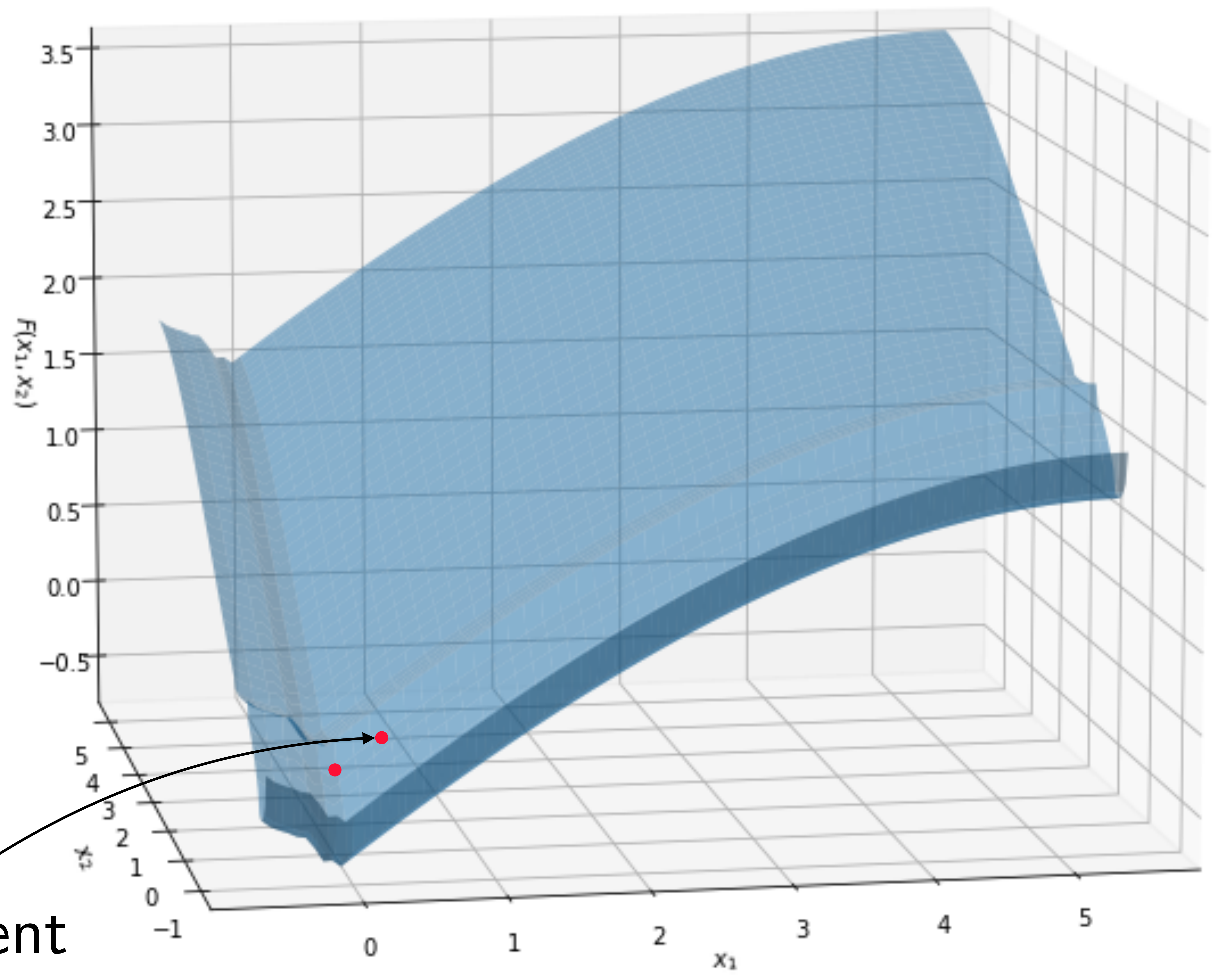
Model interpretation



Model interpretation

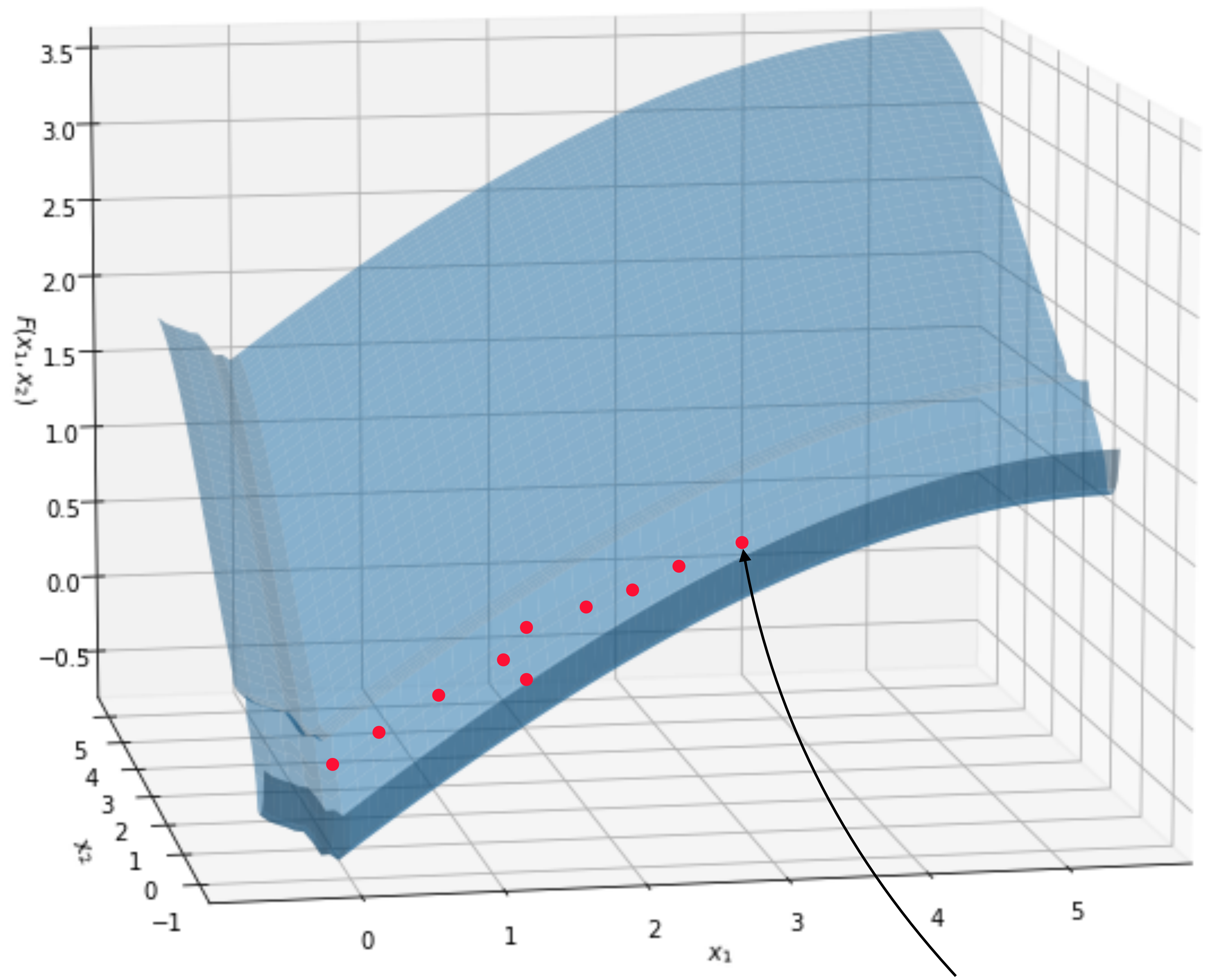


Model interpretation



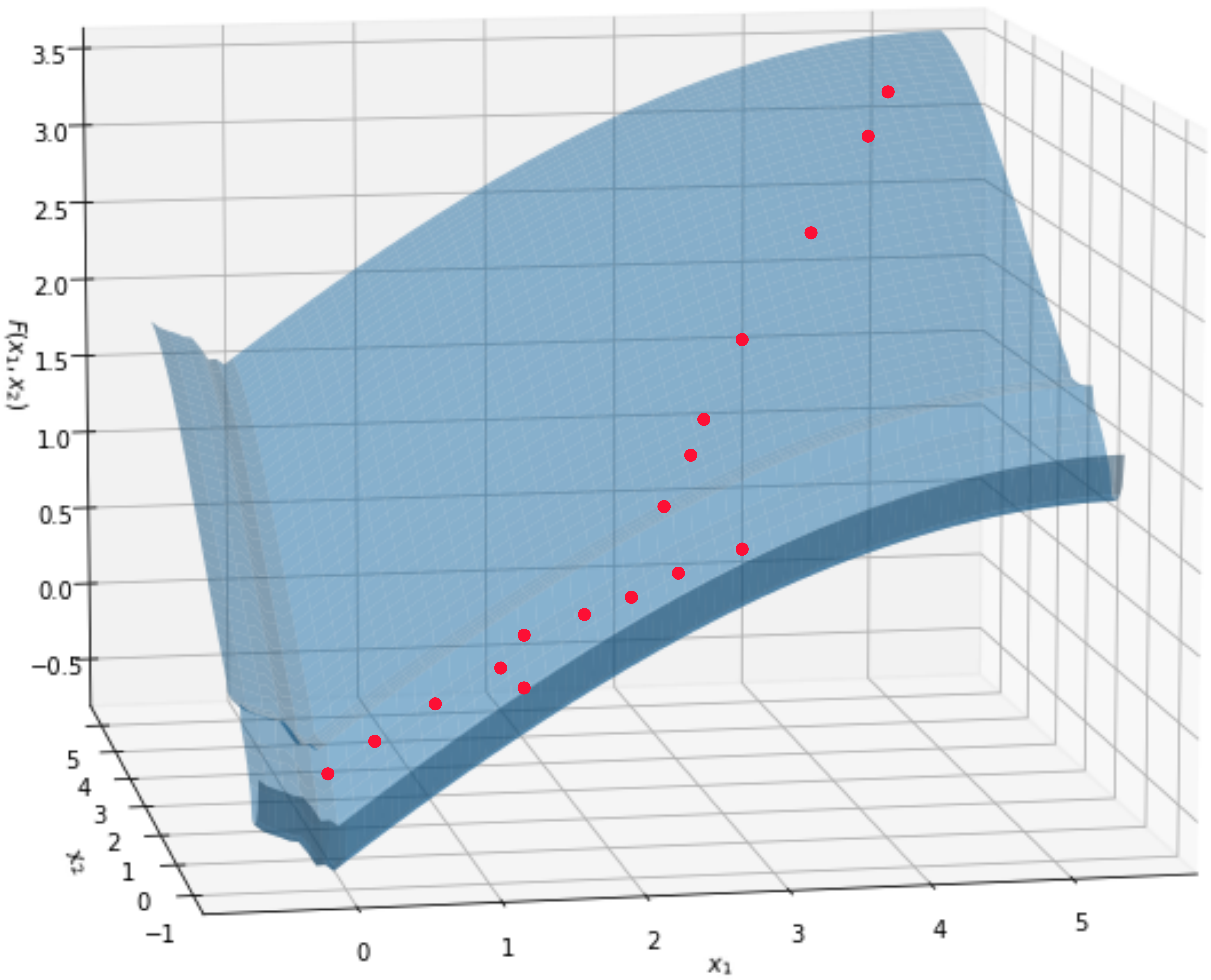
measurement

Model interpretation



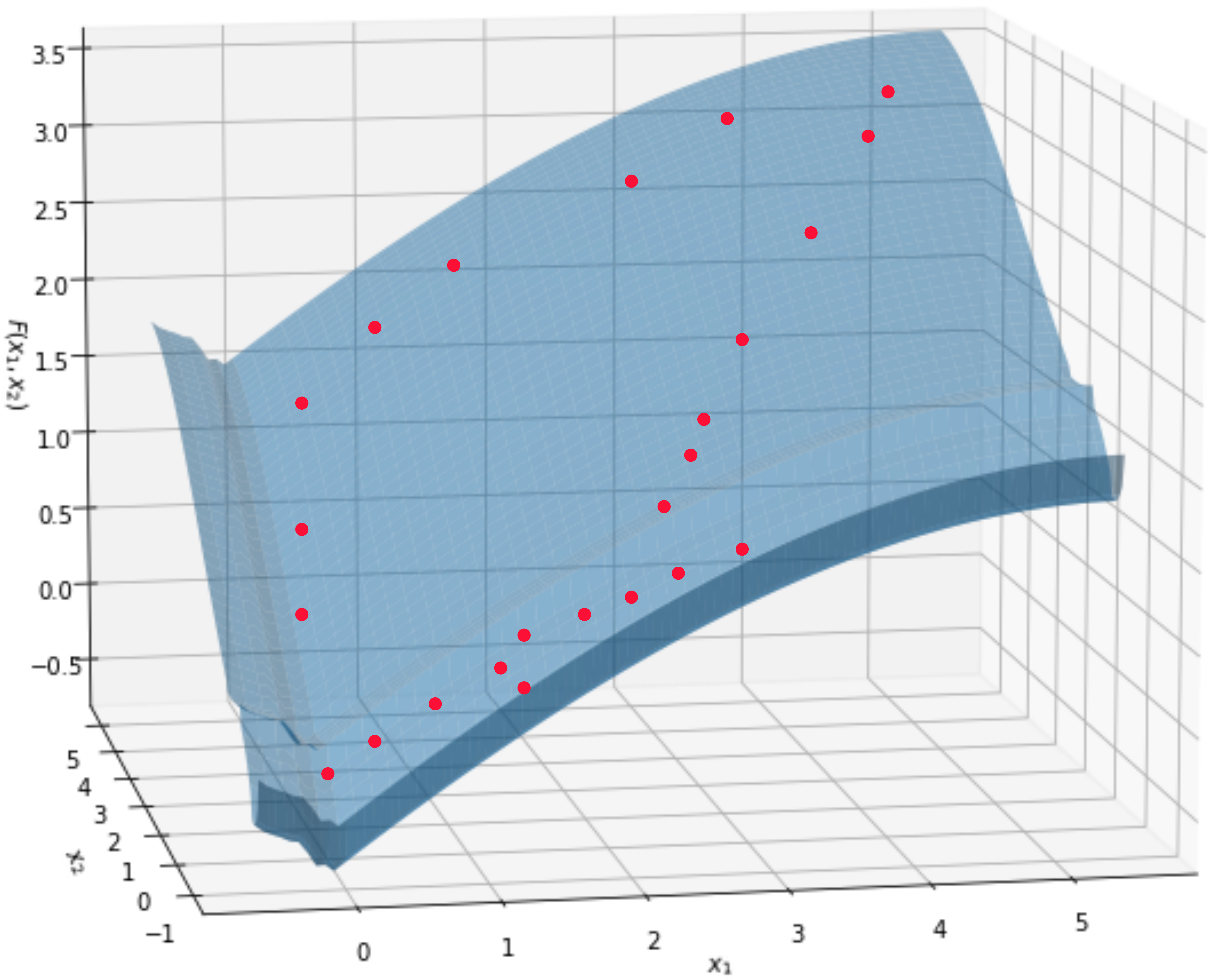
request rate increases so too does latency

Model interpretation



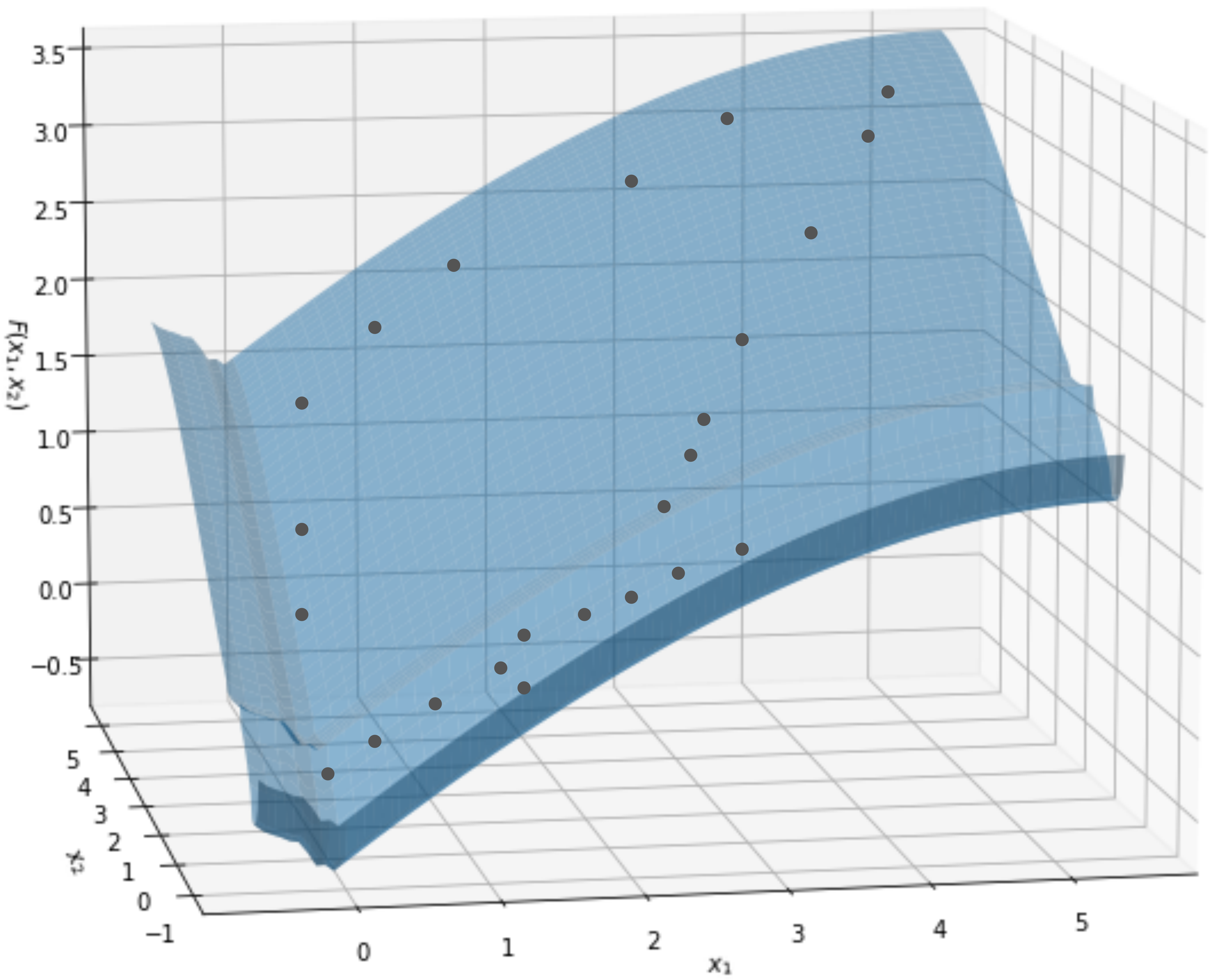
request rate increases so too does queue

Model interpretation



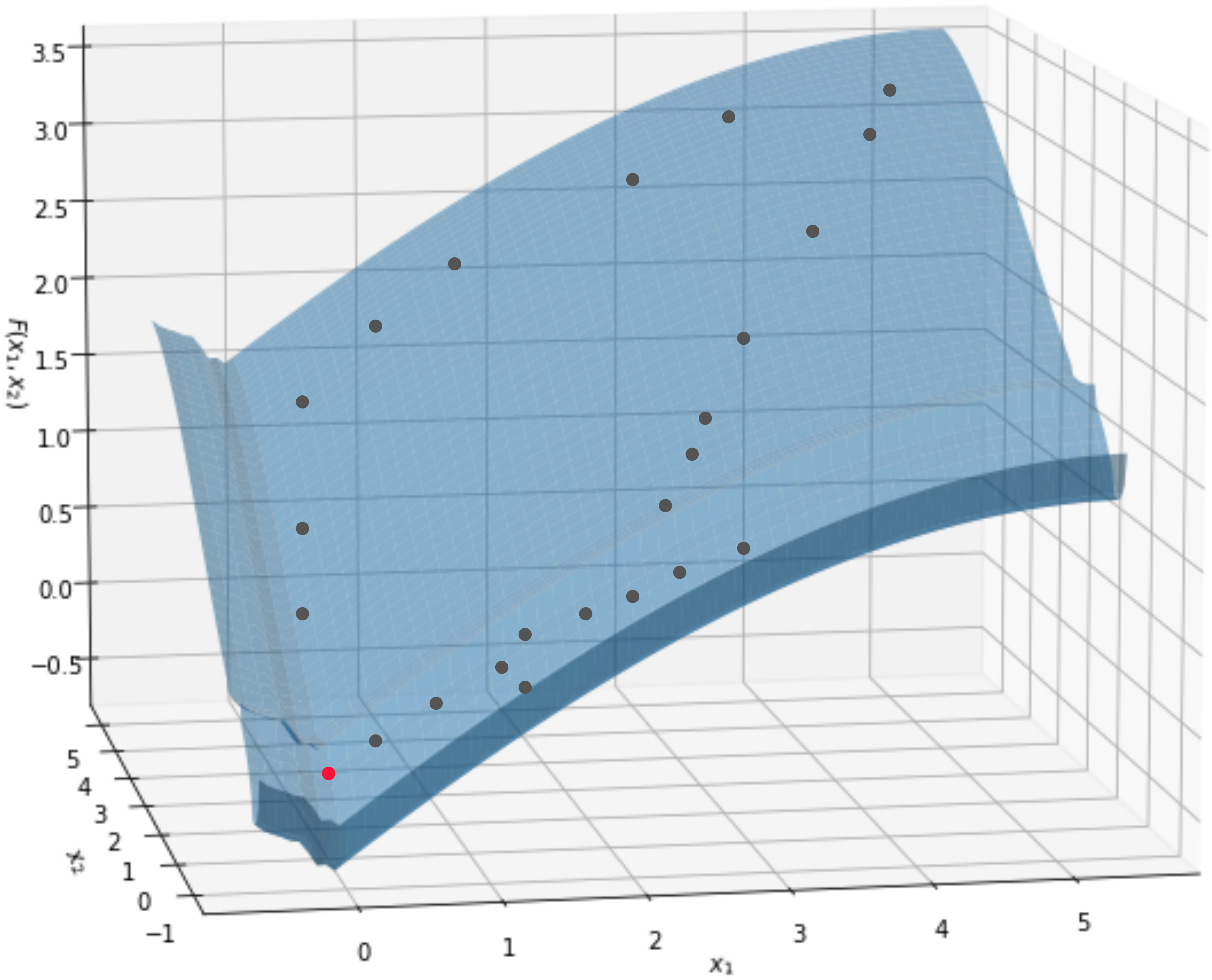
system sheds load

Model interpretation



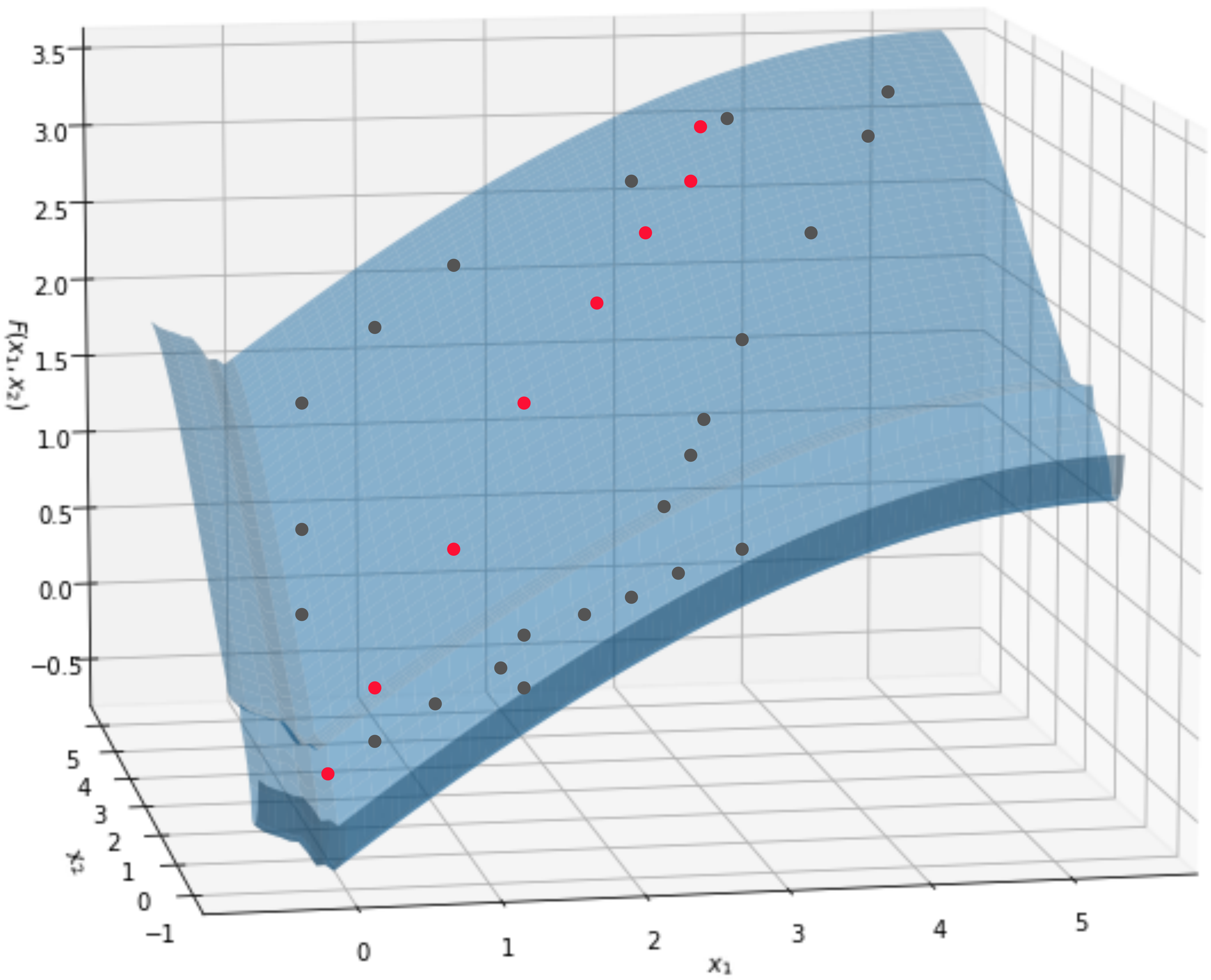
but suppose we (poorly) change configuration

Model interpretation



and increase load again

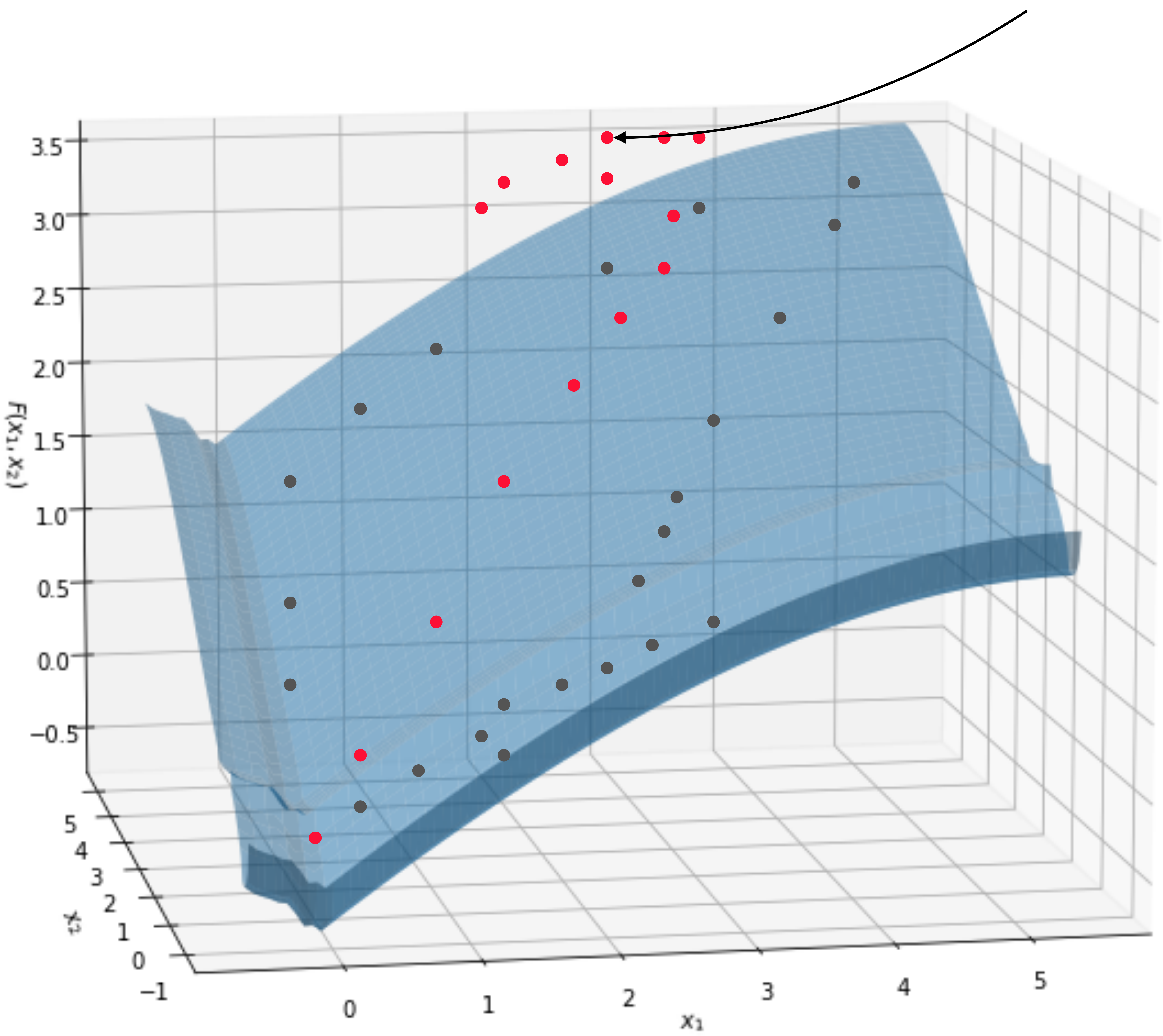
Model interpretation



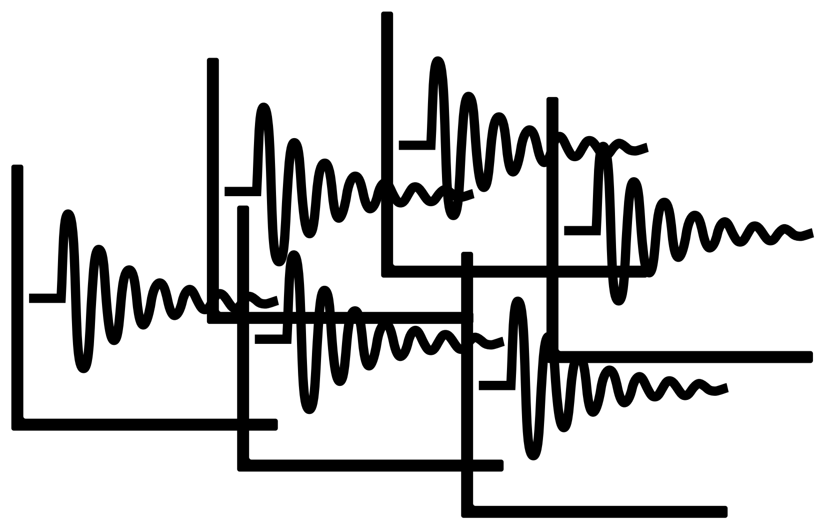
latency and queue length increase

Model interpretation

large deviations from predictor

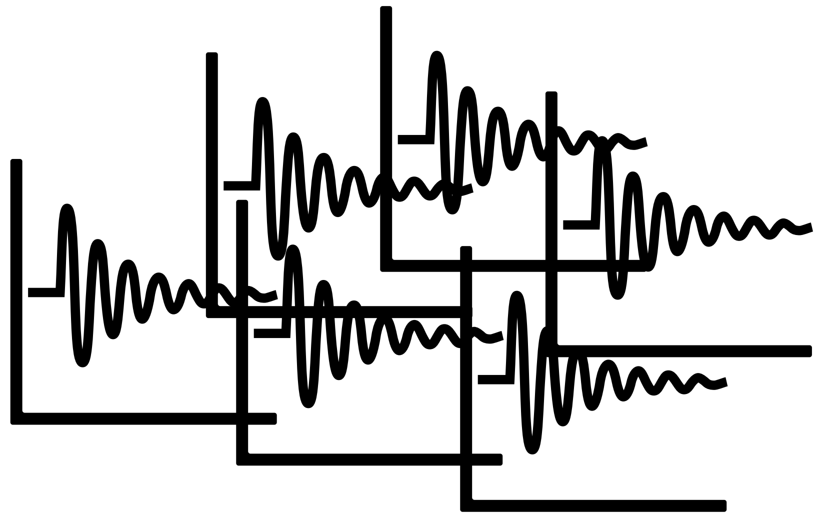


Model construction



collect recent data

Model construction

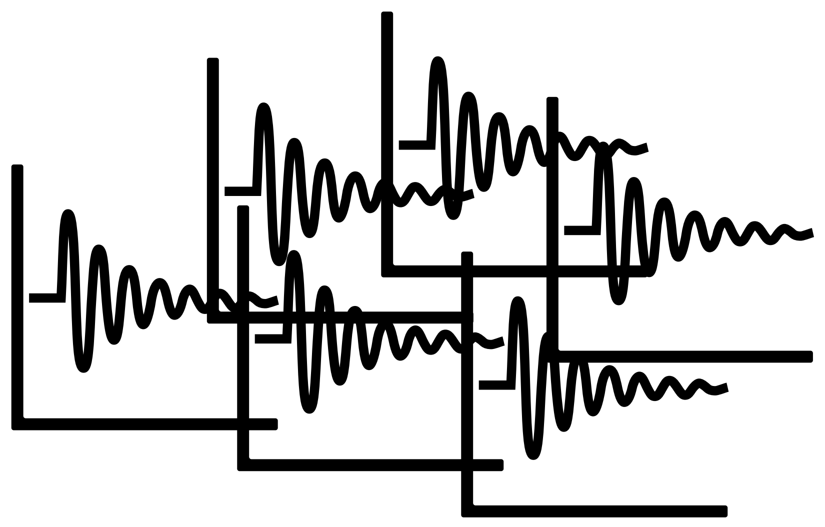


$$f_i$$

collect recent data

build predictors

Model construction



collect recent data

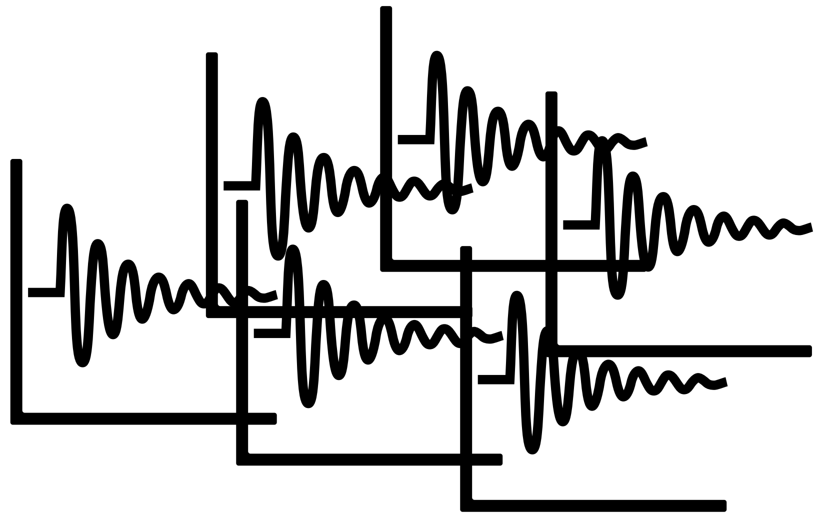
f_i

build predictors

δ

estimate covariance

Model construction



f_i

δ

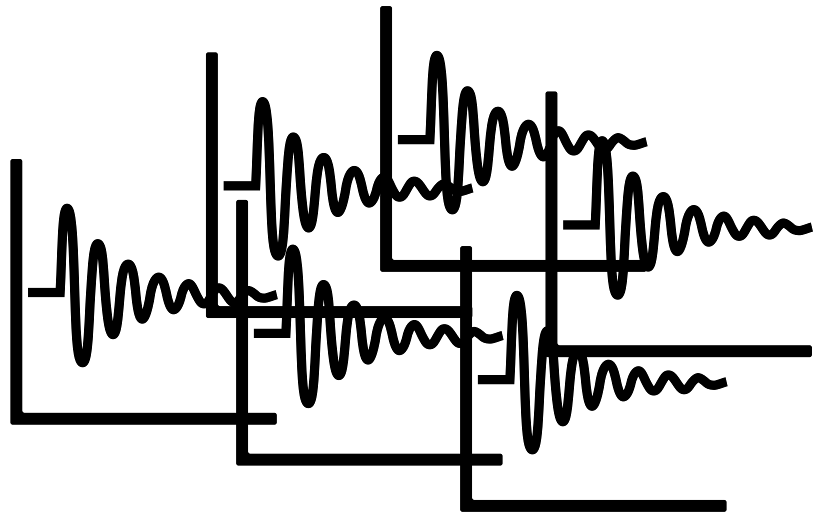
collect recent data

build predictors

estimate covariance

we use *general additive models* with *splines*

Model construction



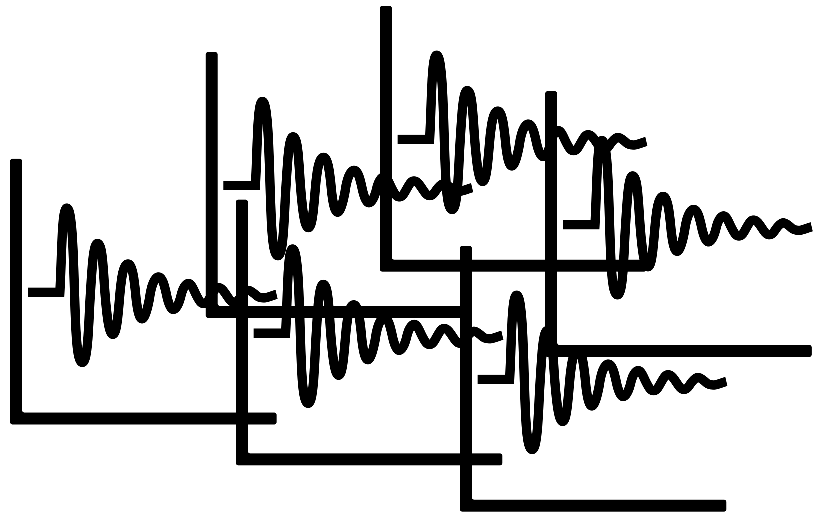
f_i

δ

collect recent data build predictors estimate covariance

impose *sparsity* since we know which processes communicate

Model construction



f_i

δ_i

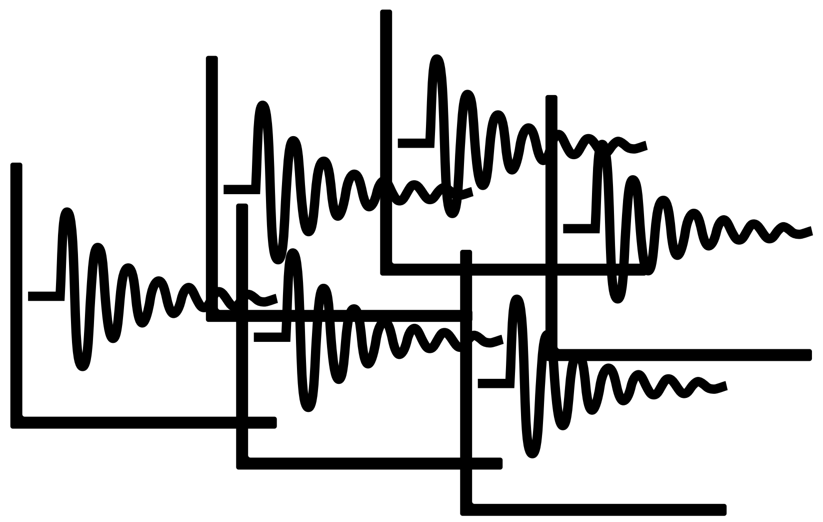
collect recent data

build predictors

estimate covariance

impose *conditional independence* on deviation covariance

Model construction



f_i

δ

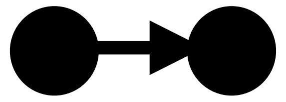
collect recent data

build predictors

estimate covariance

impose *conditional independence* on deviation covariance

Numerical experiments



synthetic two-process model

Numerical experiments

synthetic two-process model

mongo database instance

Numerical experiments

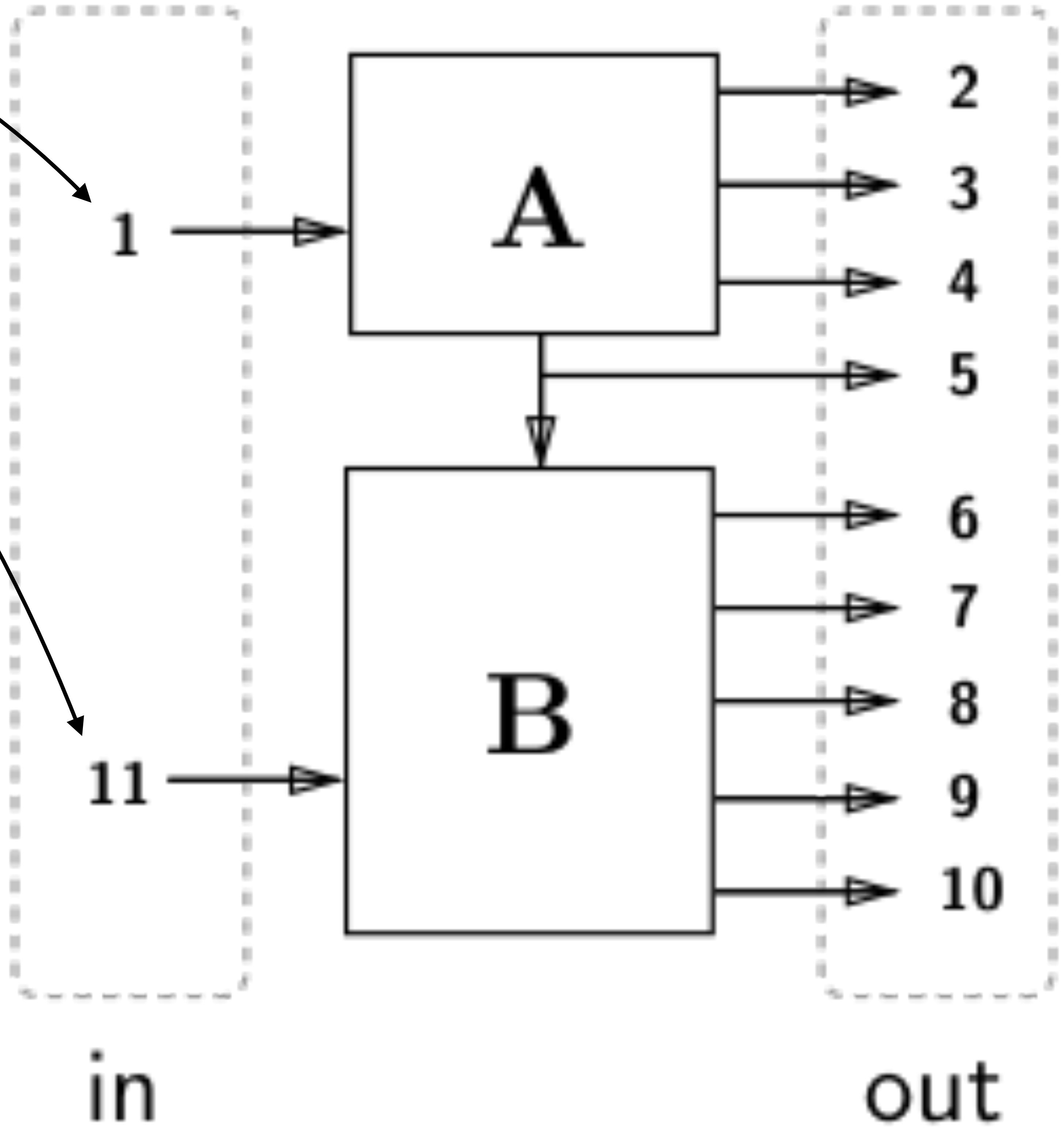
synthetic two-process model

mongo database instance

lacking full-scale cloud experiment; future work

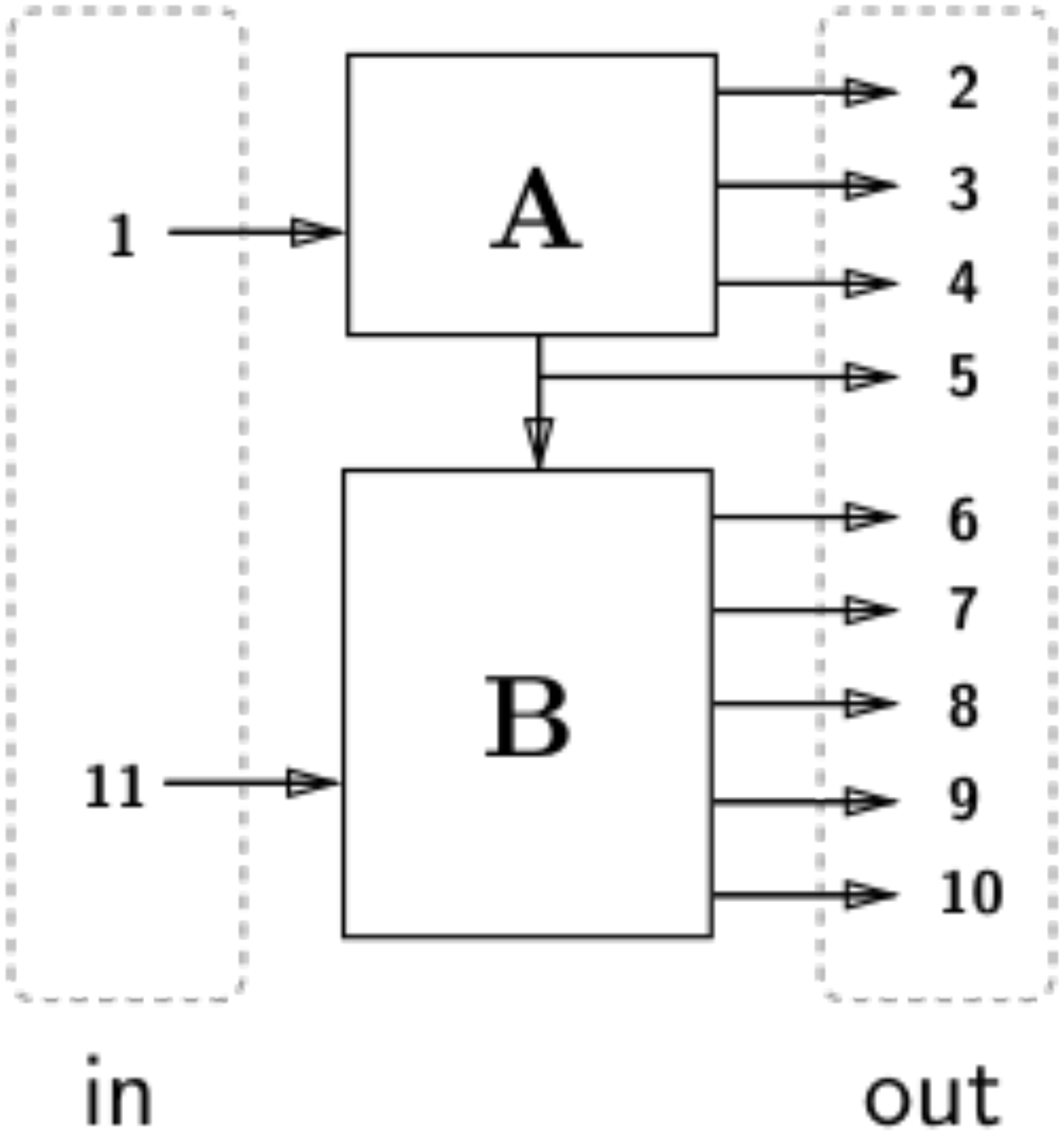
Numerical experiments: synthetic environment

two *input* stochastic processes

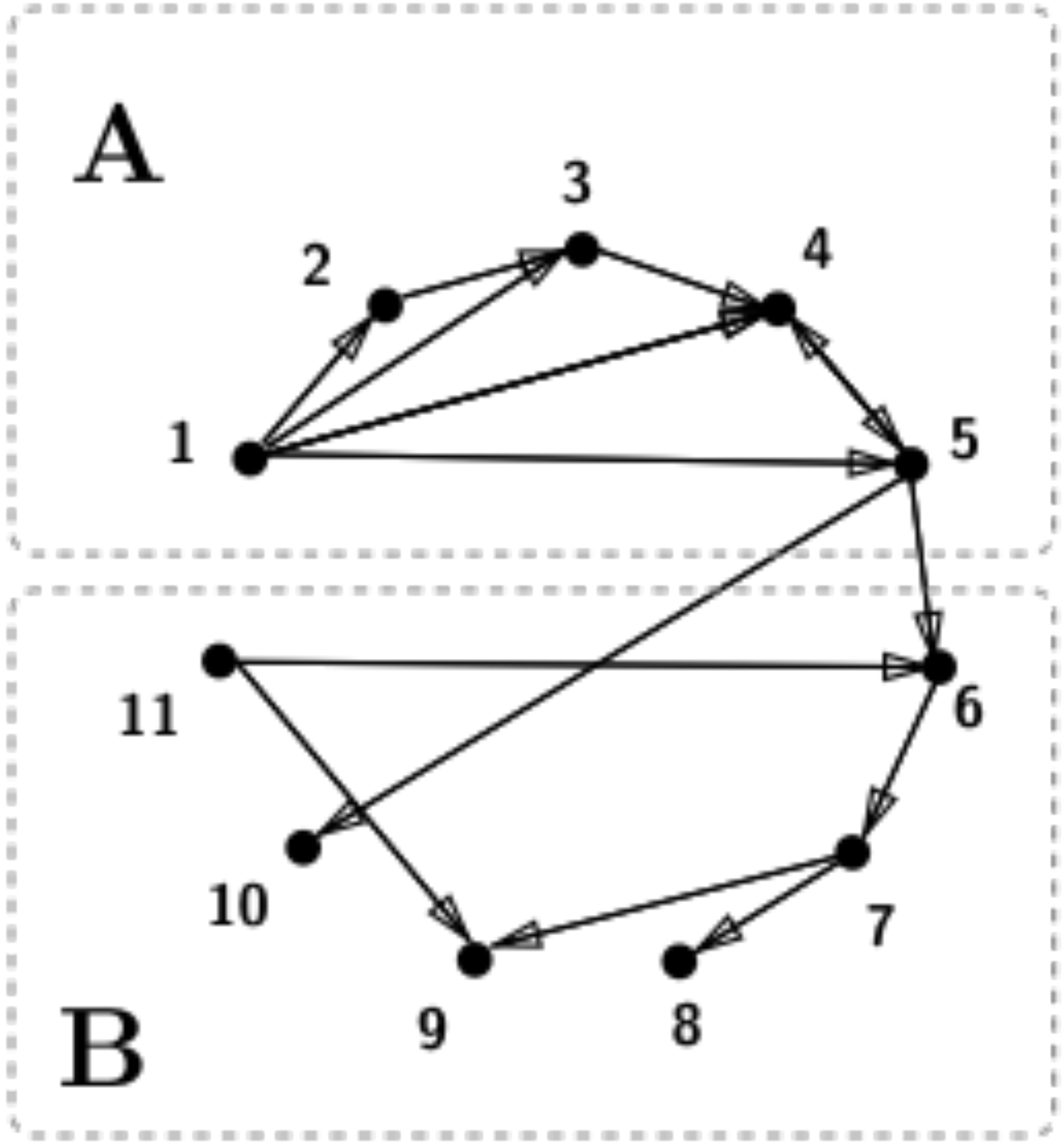


Numerical experiments: synthetic environment

nine *output* stochastic processes

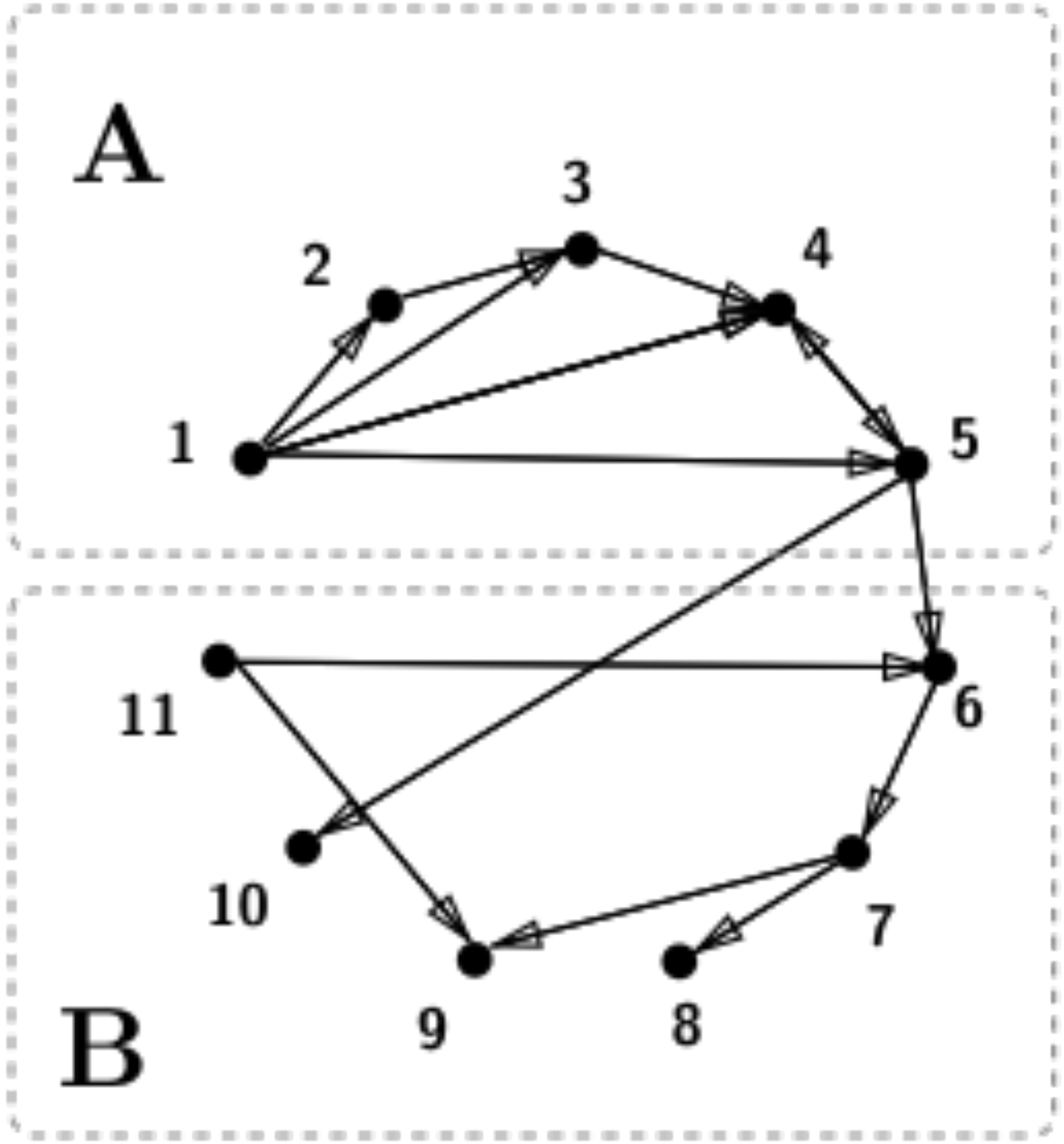


Numerical experiments: synthetic environment



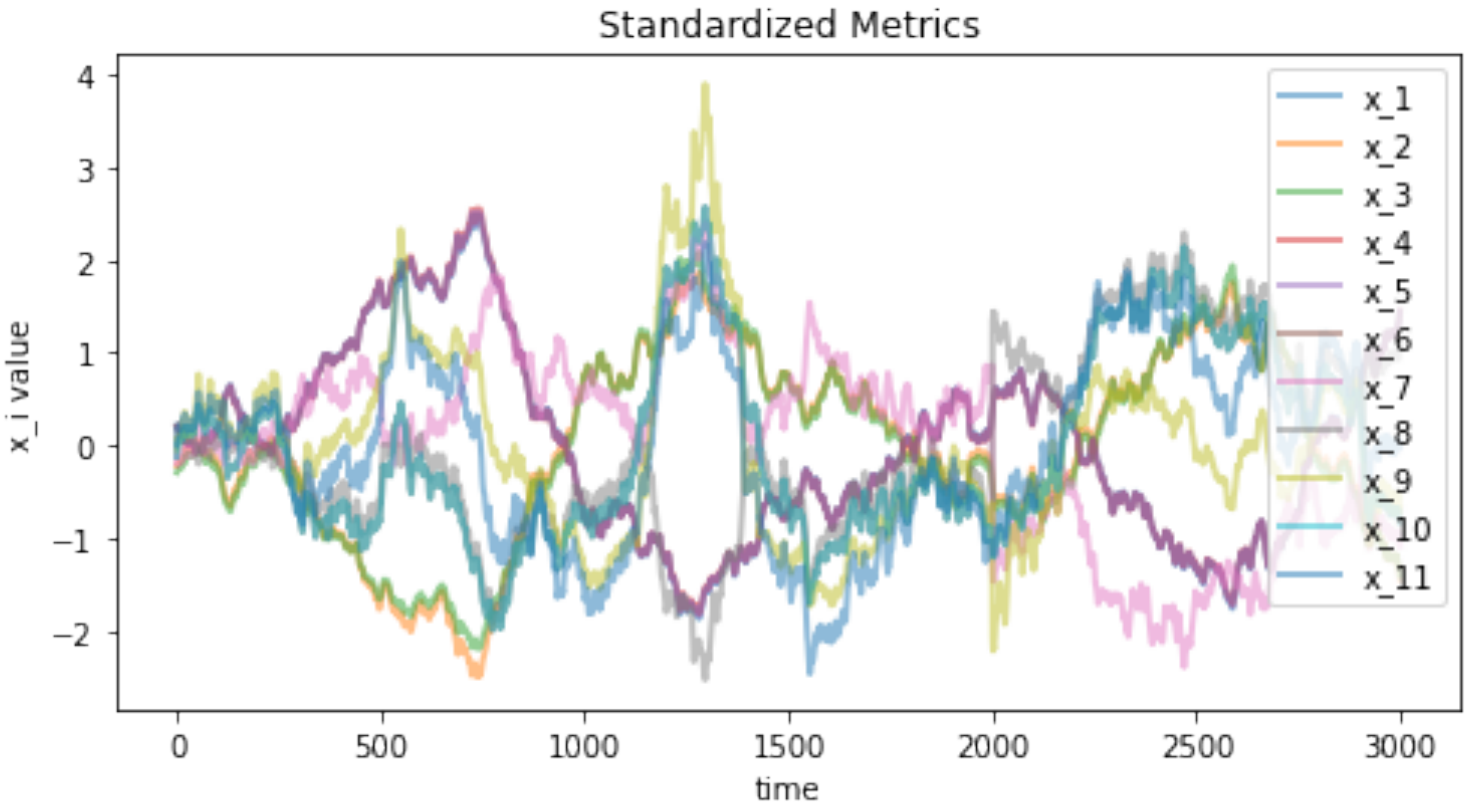
metrics *related functionally* to those with arrows

Numerical experiments: synthetic environment

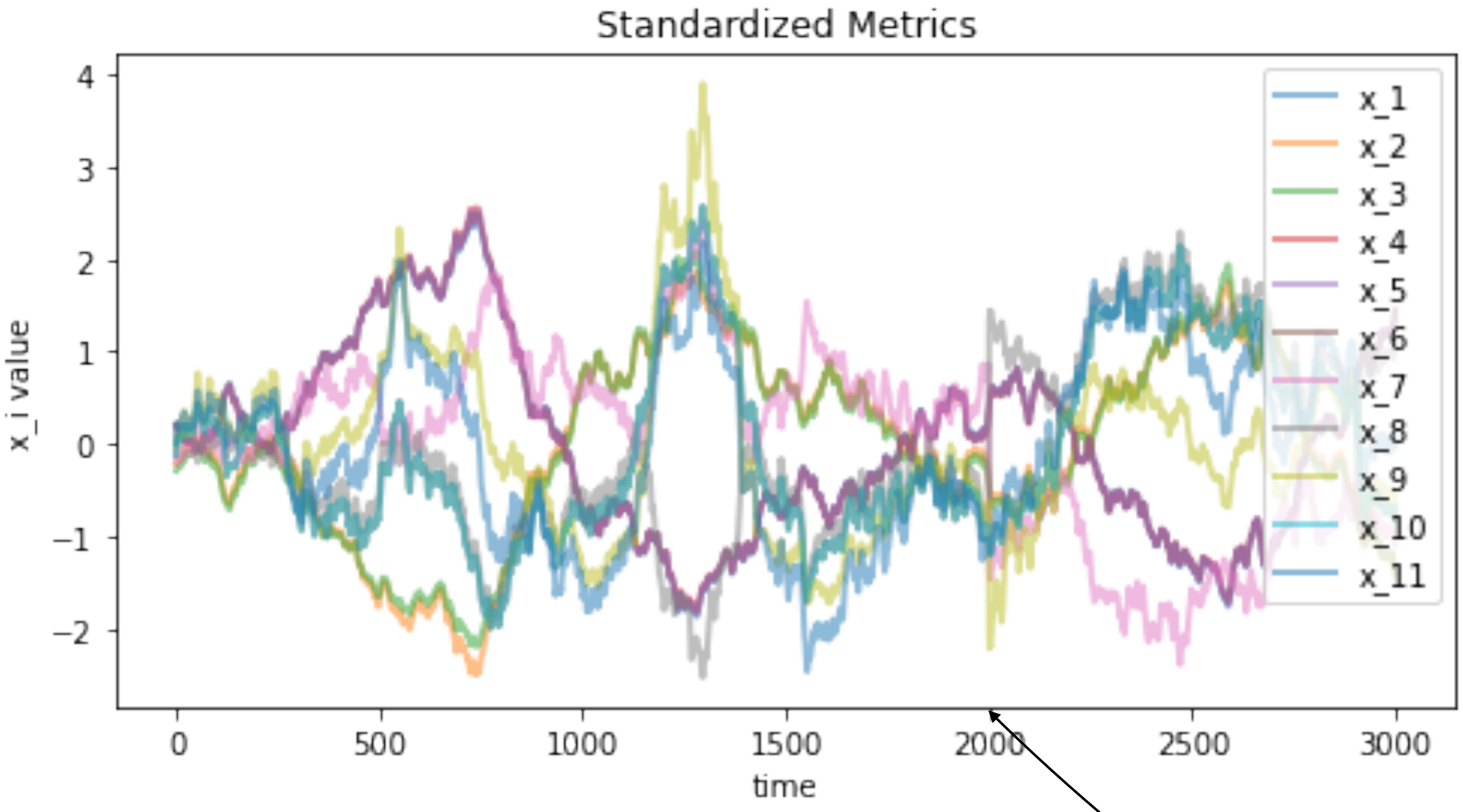


we collect normal data, then *change parameter on edge 6 to 7*

Numerical experiments: synthetic environment

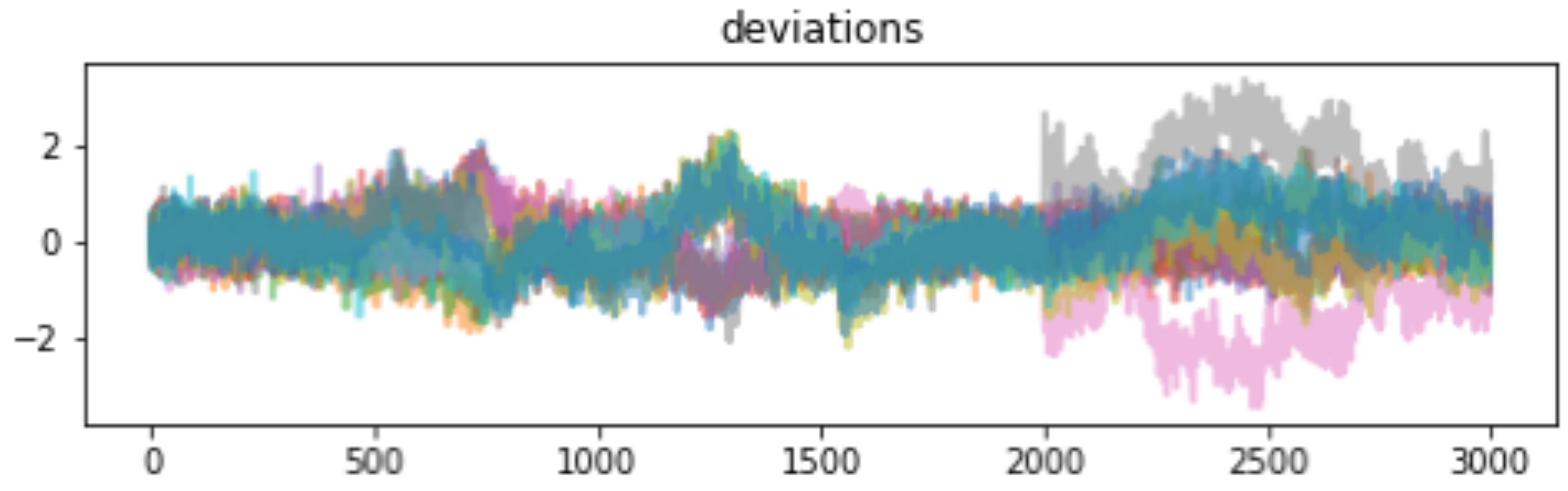


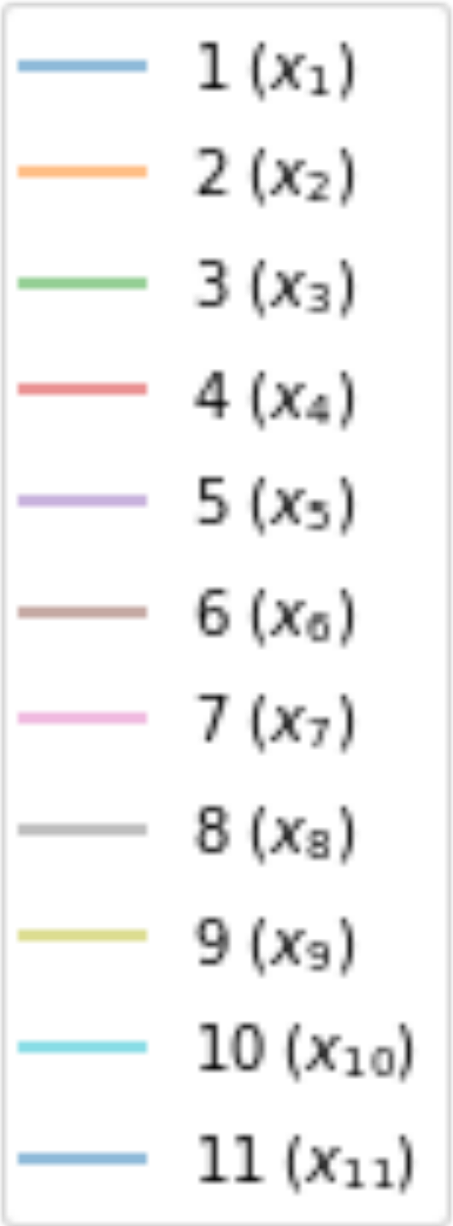
Numerical experiments: synthetic environment



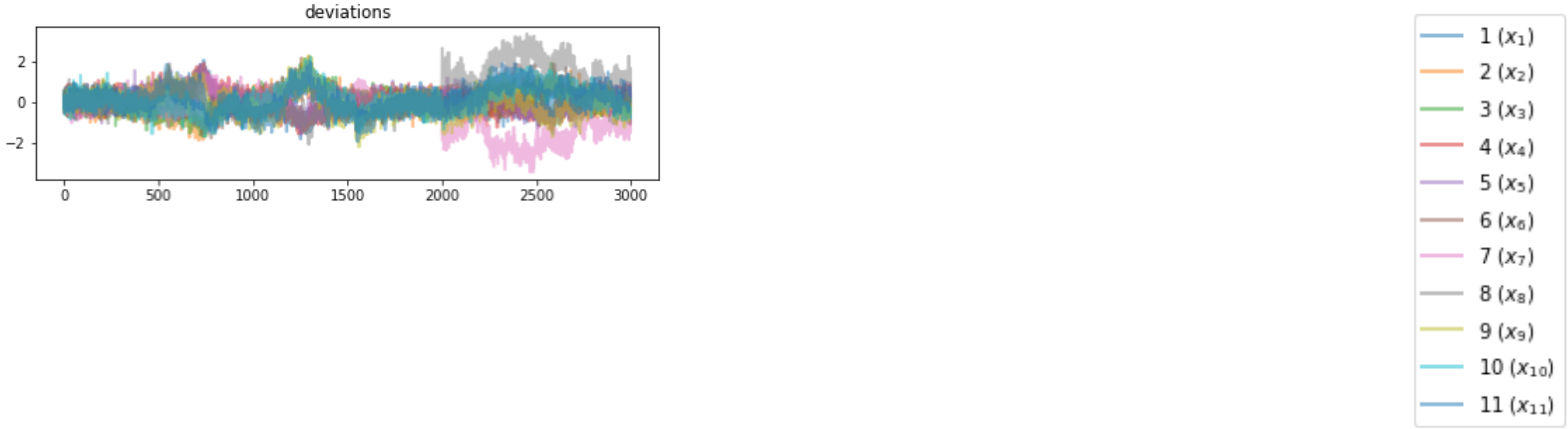
system changes at time 2000

Numerical experiments: synthetic environment



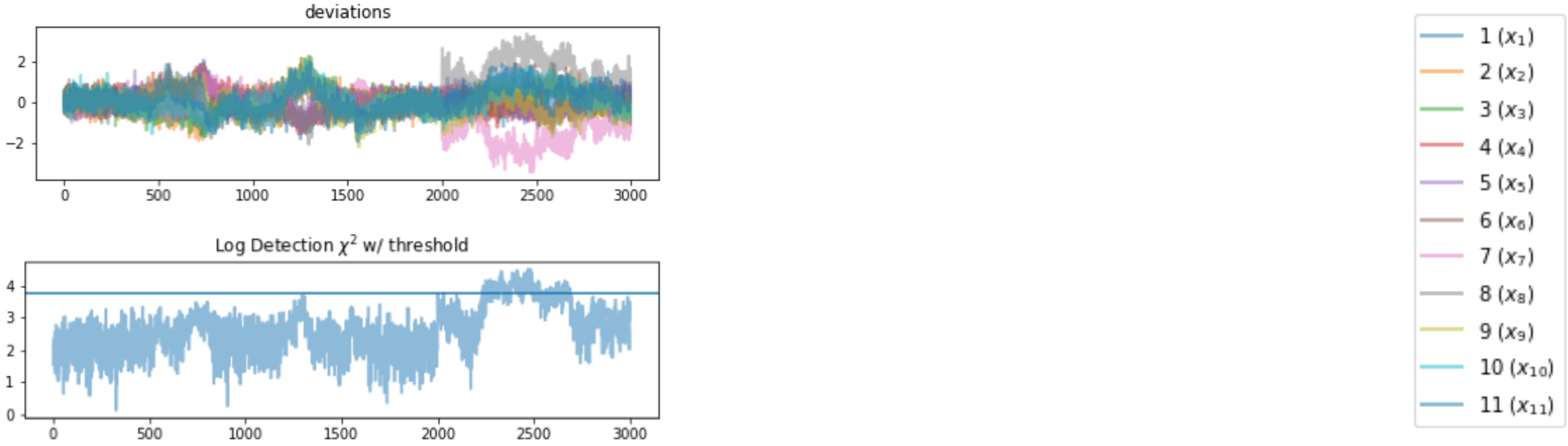


Numerical experiments: synthetic environment

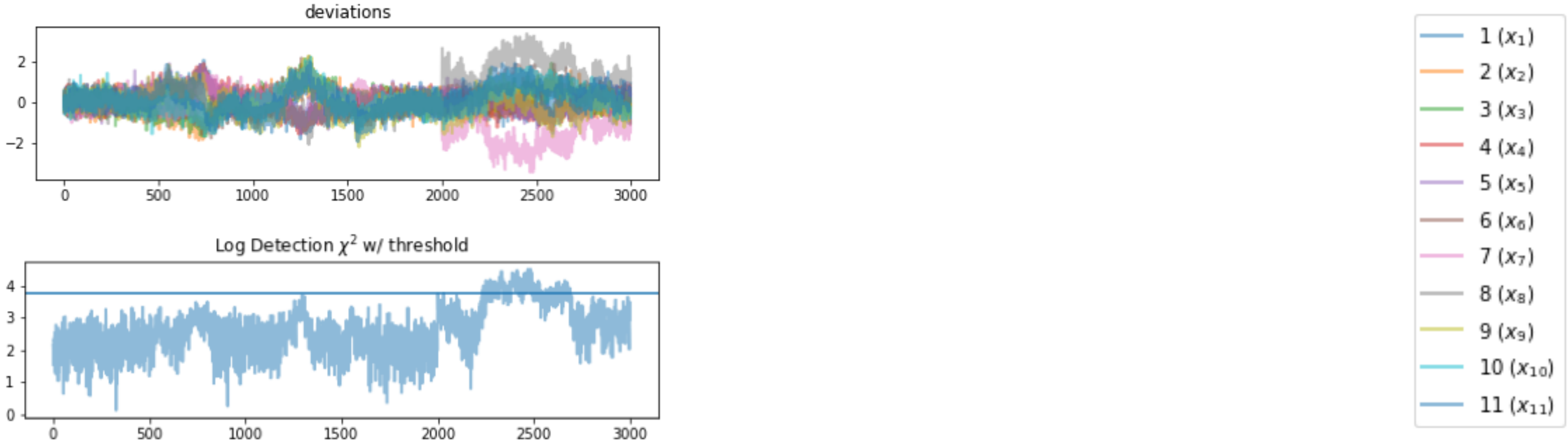


deviations get bigger at time step 2000

Numerical experiments: synthetic environment

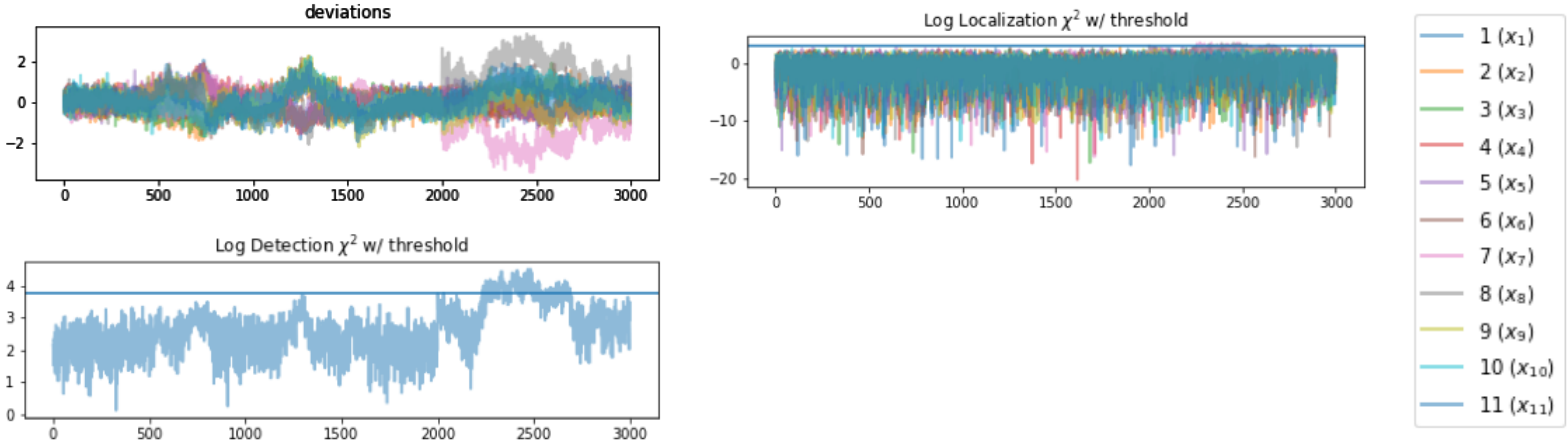


Numerical experiments: synthetic environment

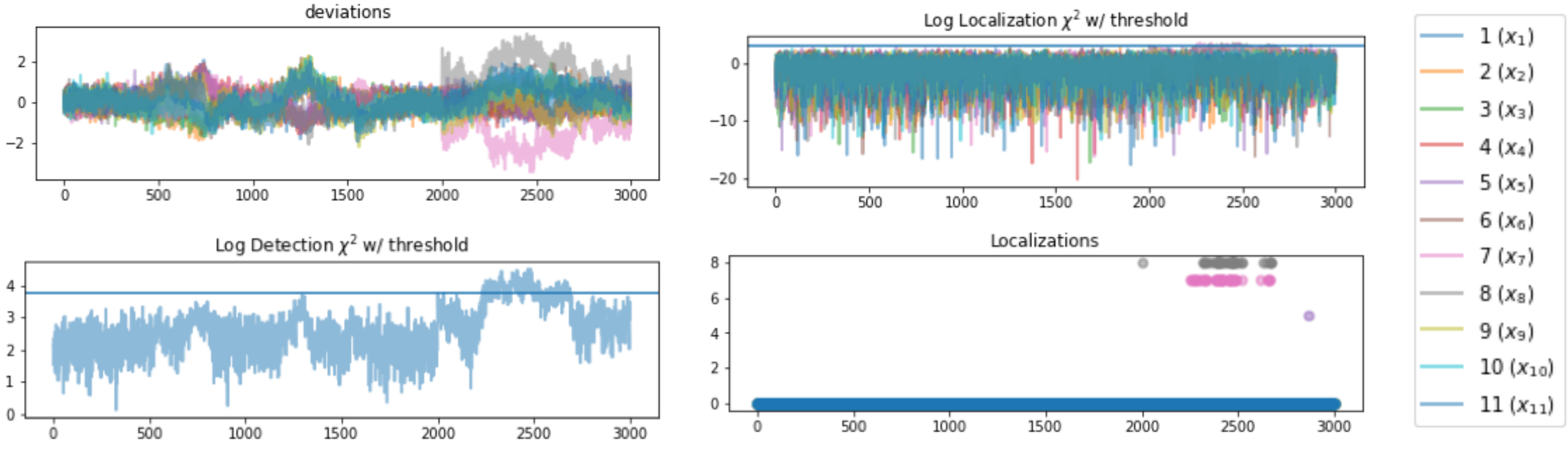


the system begins to look jointly anomalous

Numerical experiments: synthetic environment

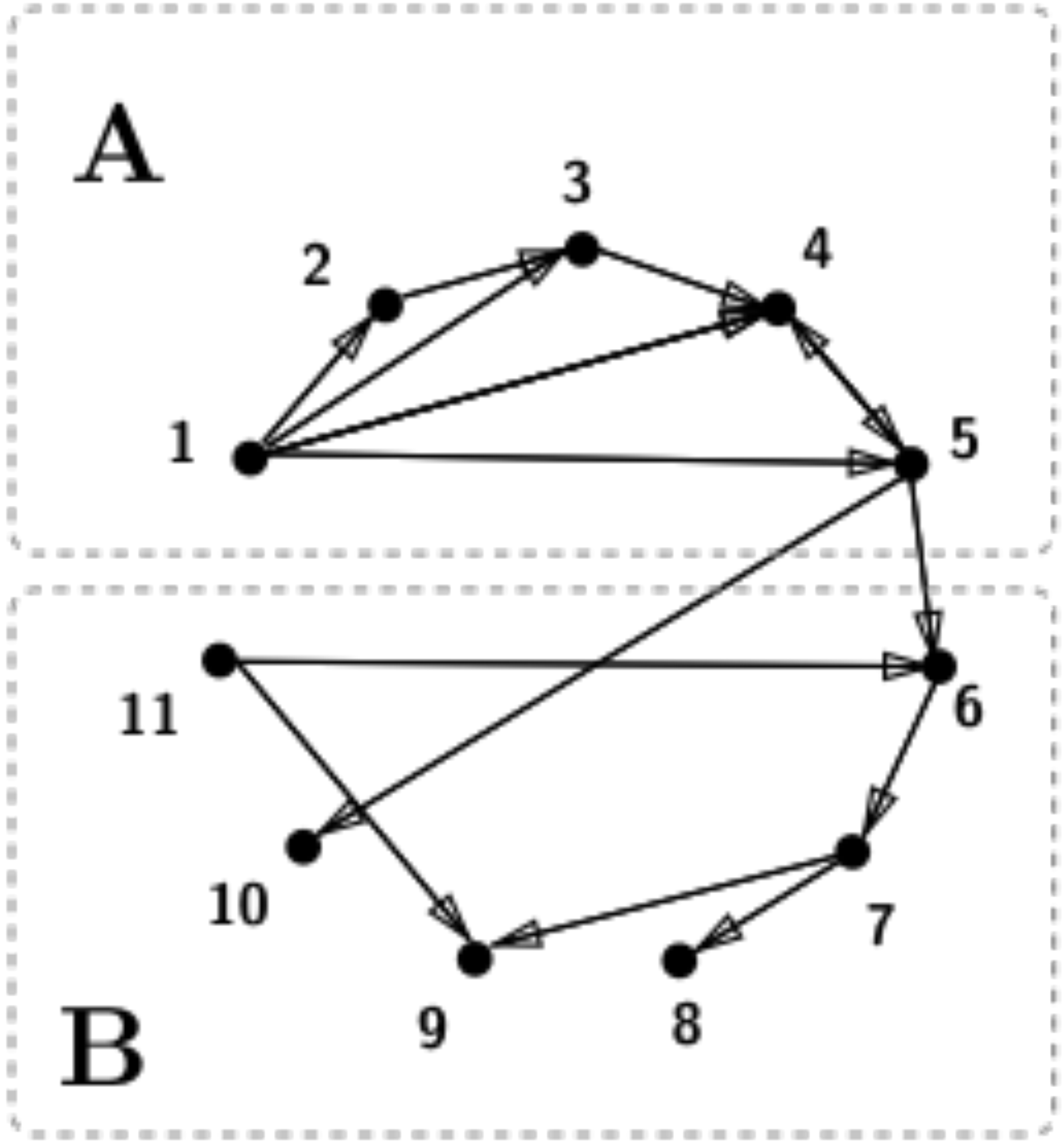


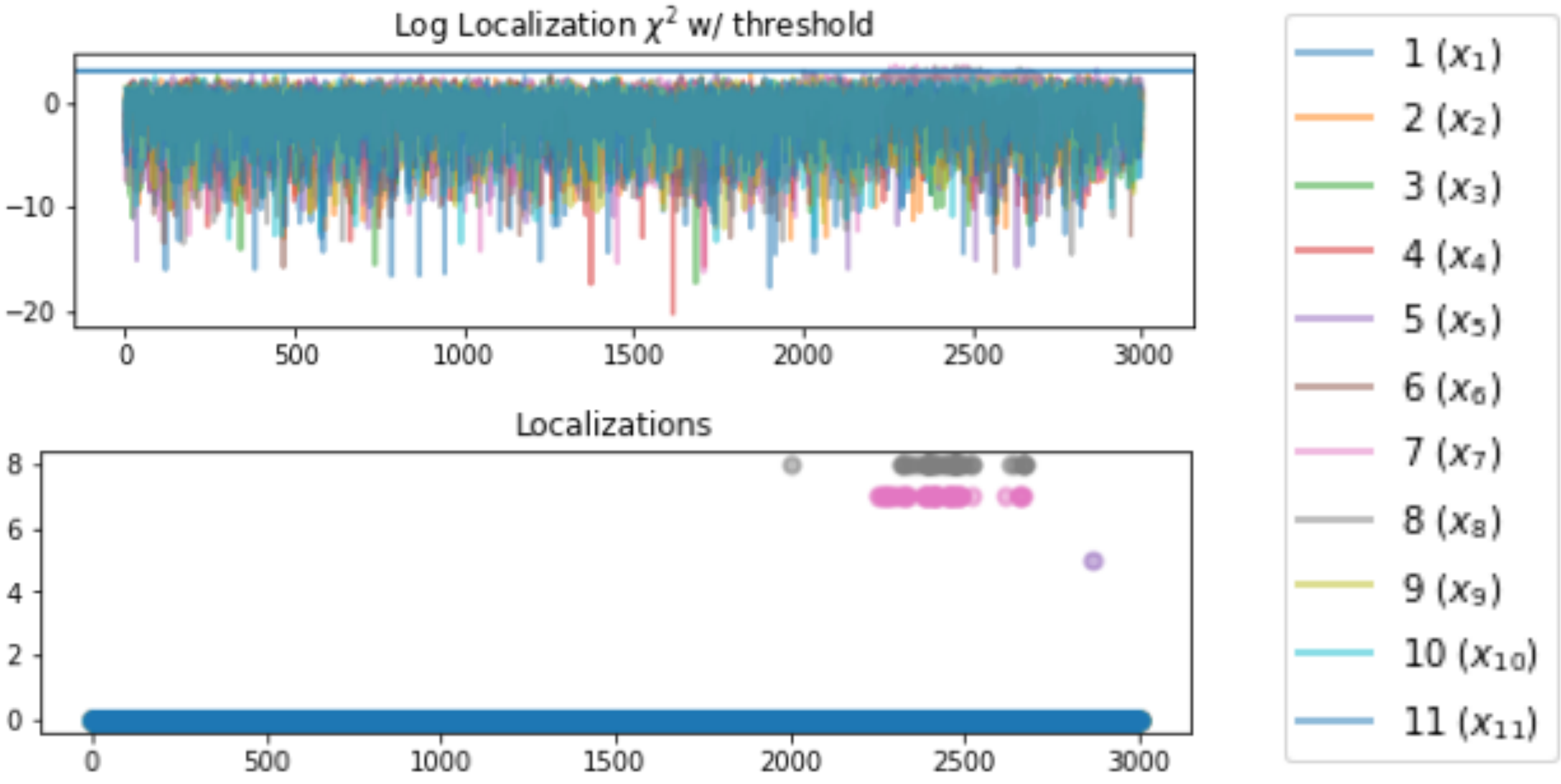
Numerical experiments: synthetic environment



localize to metric 7 and 8

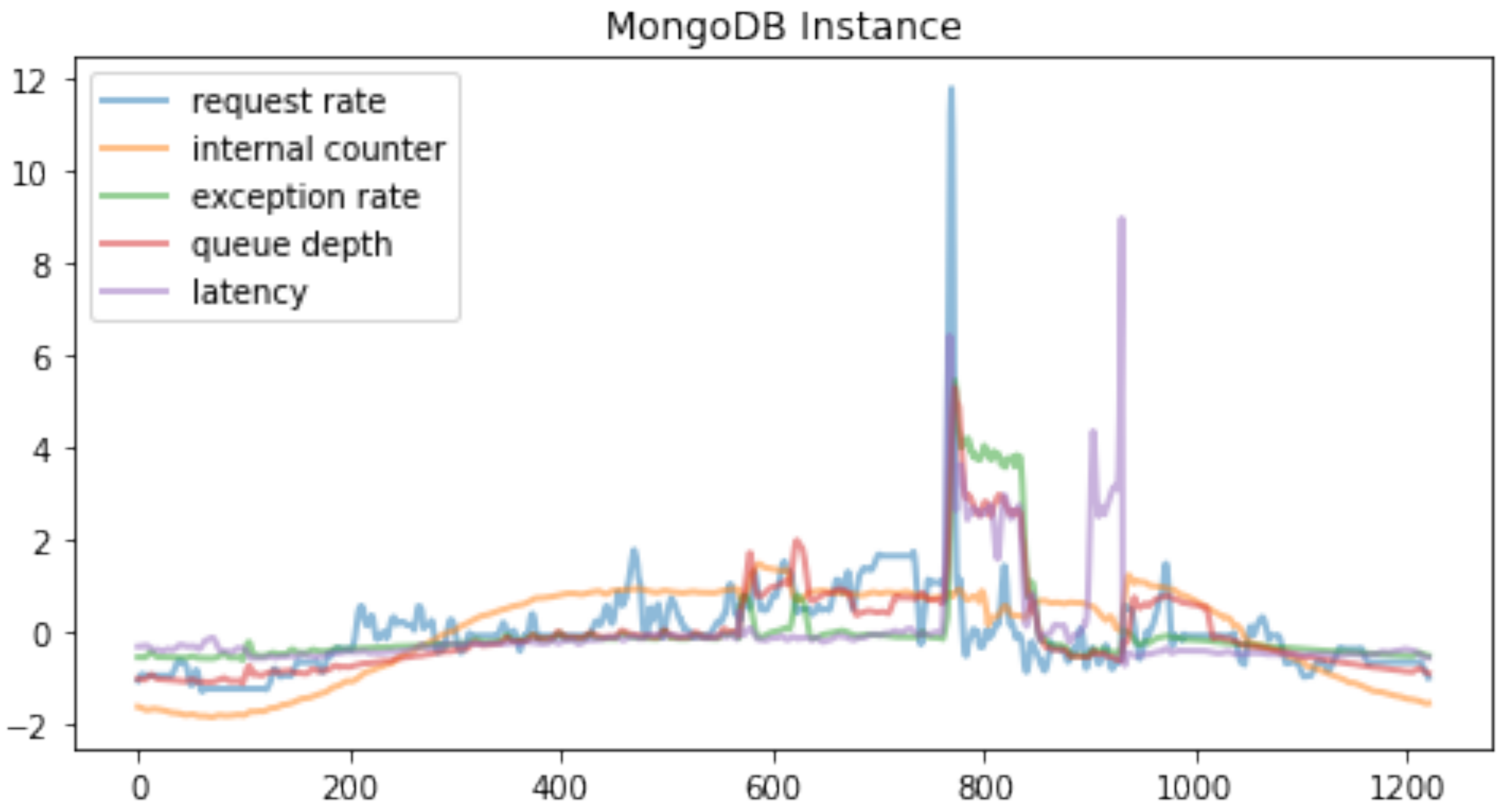
Numerical experiments: synthetic two-process environment



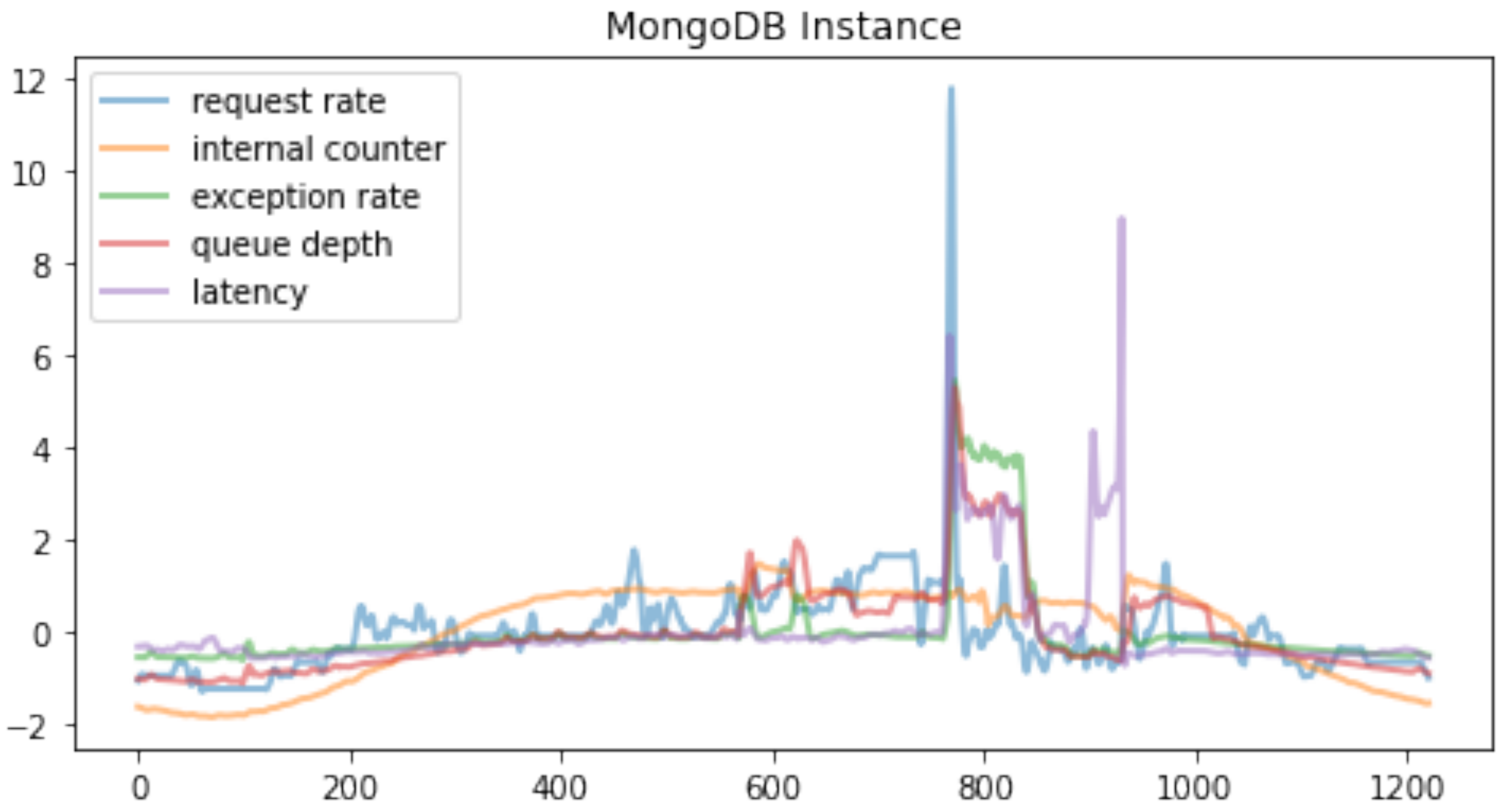


recall we changed 7, and 7 affects 8

Numerical experiments: mongo database instance

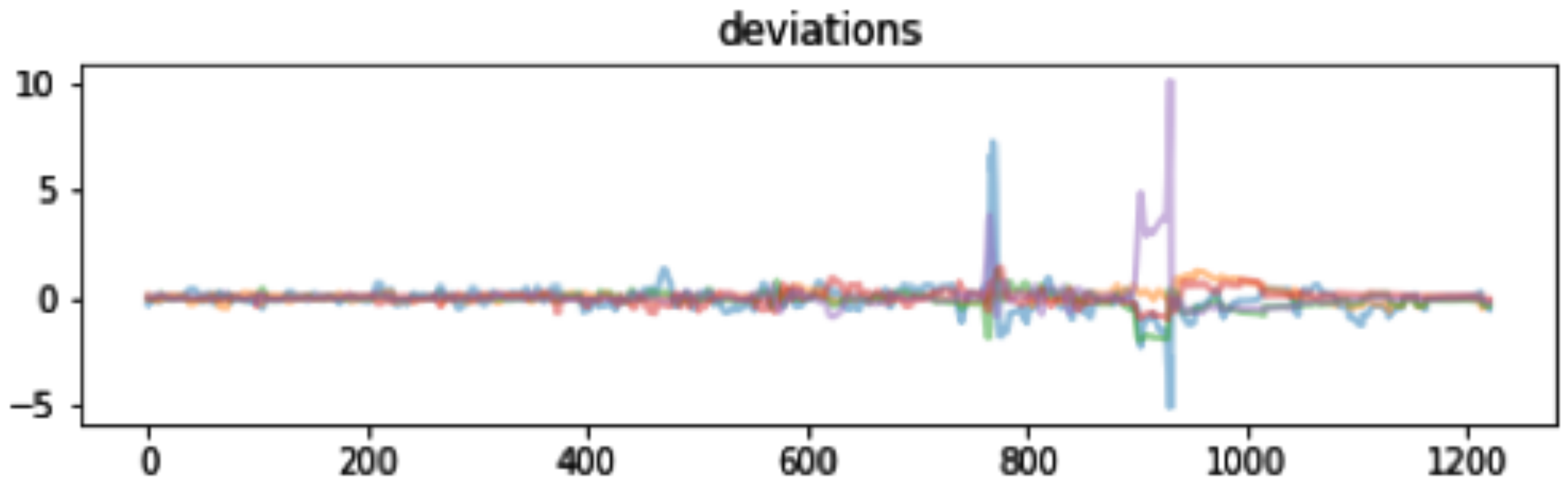


Numerical experiments: mongo database instance

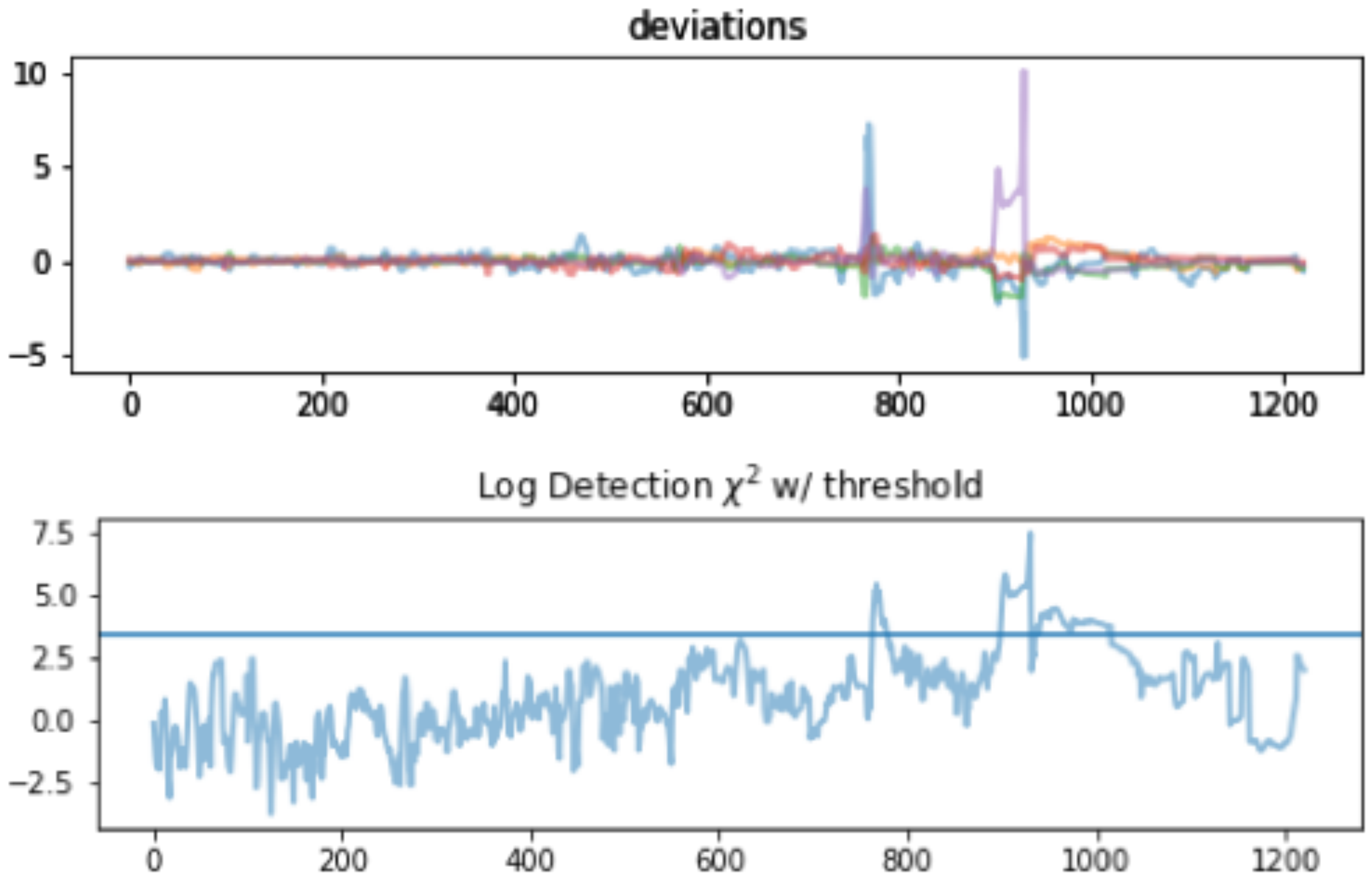


increase latency around sample 750, change configuration around sample 900

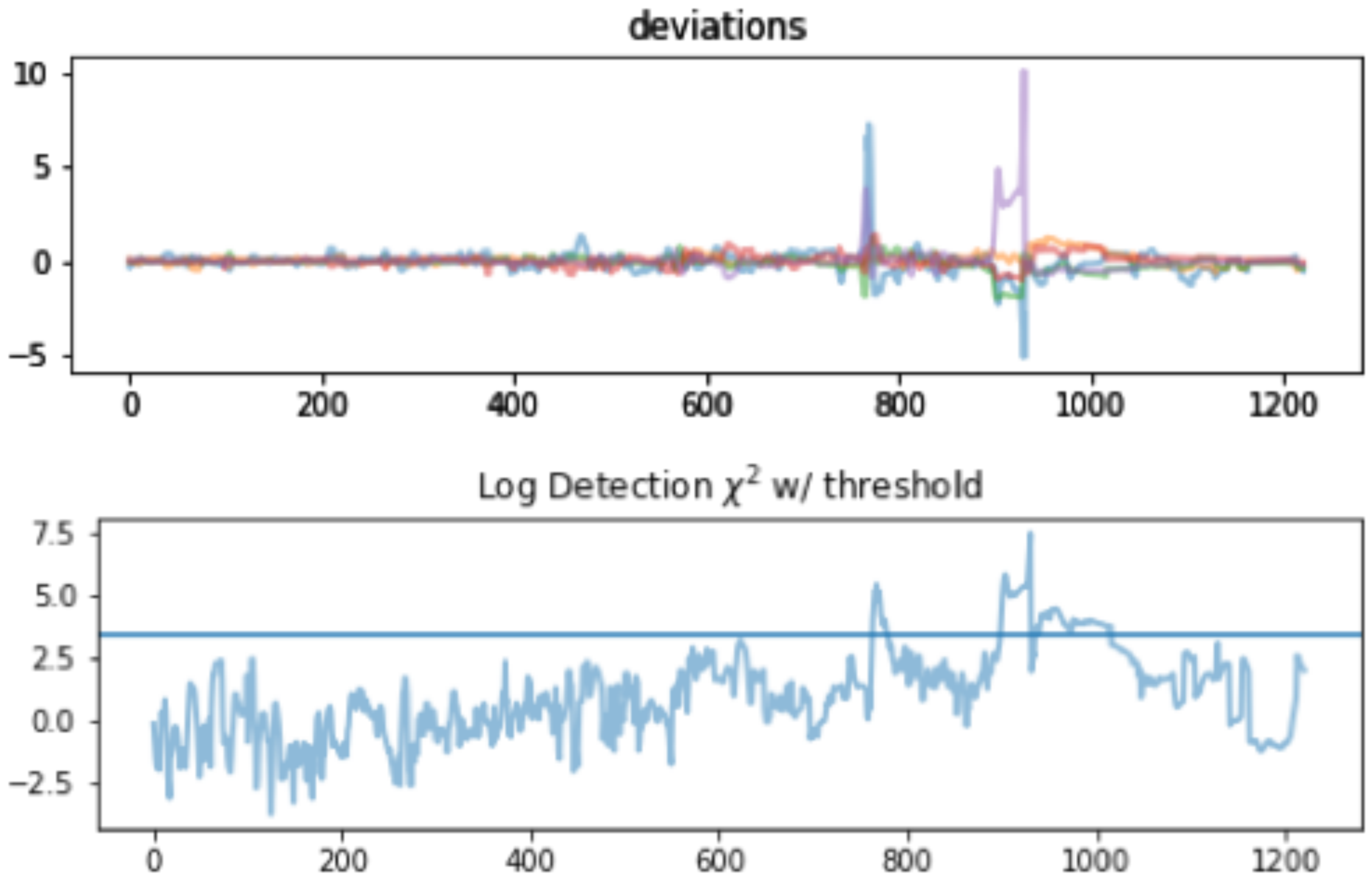
Numerical experiments: mongo database instance



Numerical experiments: mongo database instance

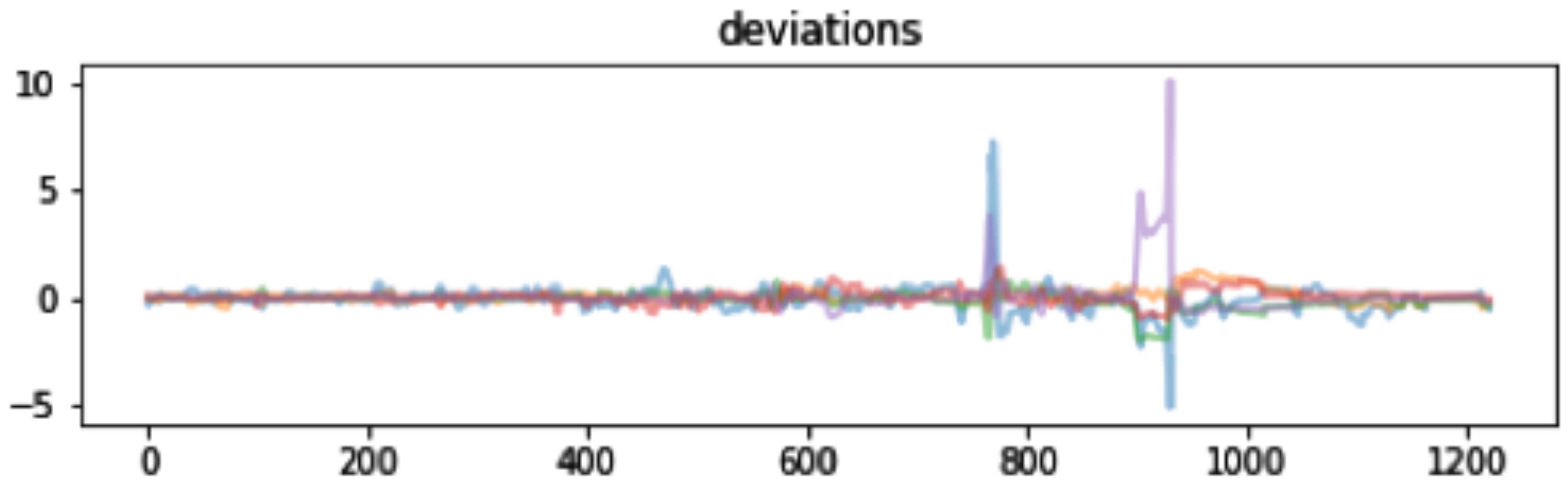


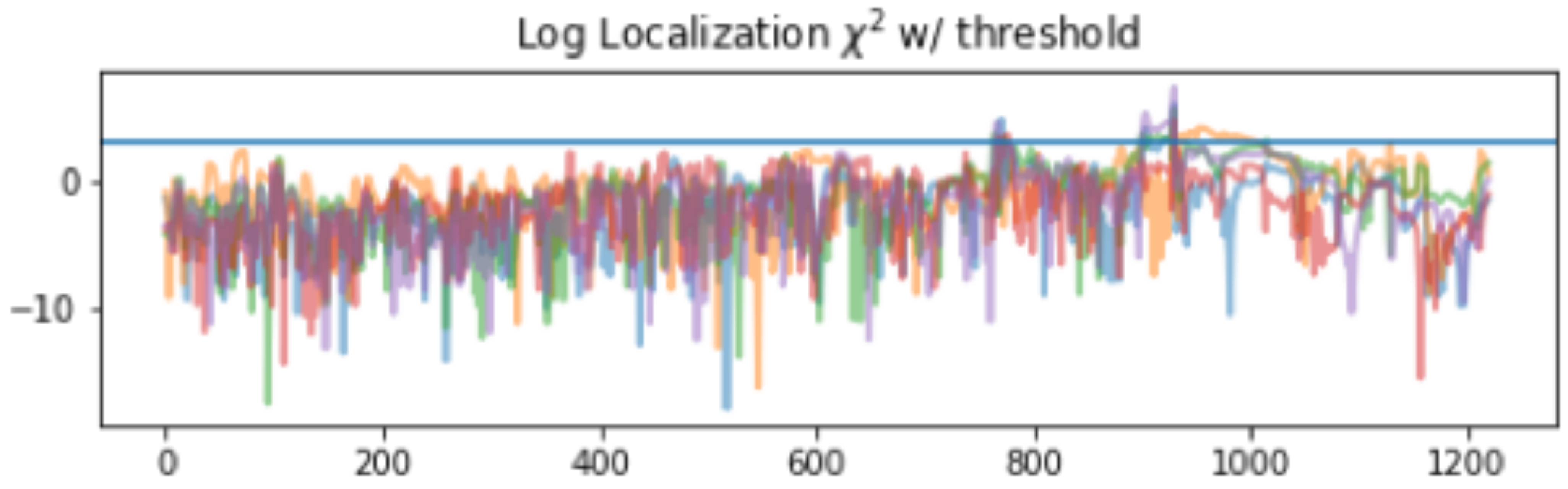
Numerical experiments: mongo database instance

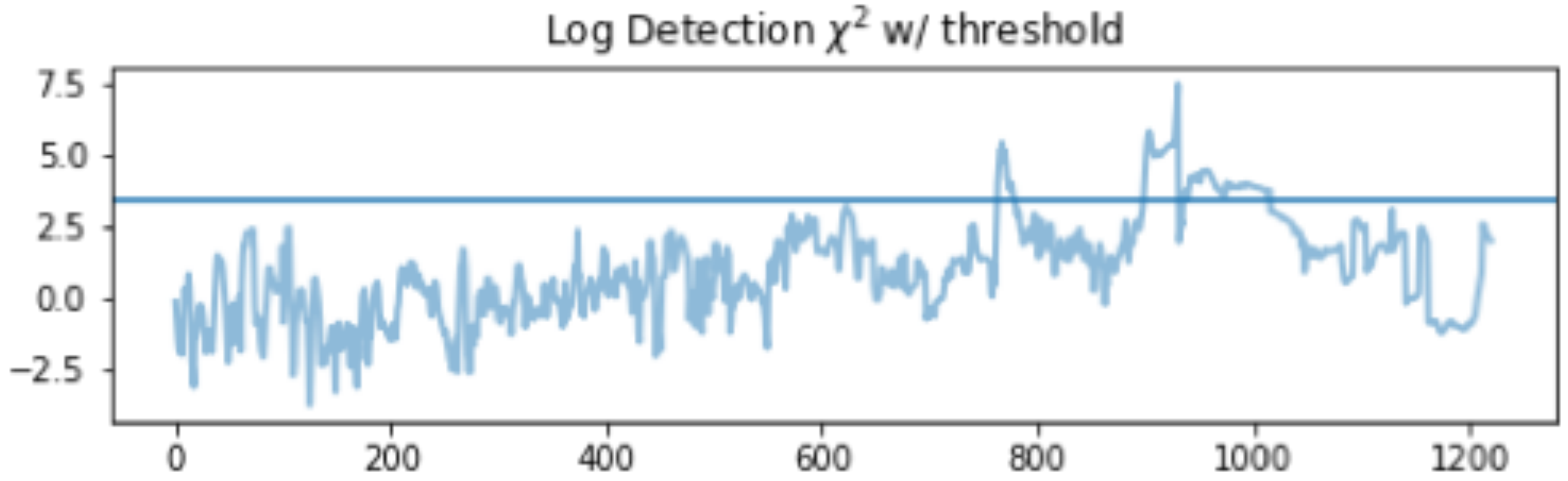


we see anomalies

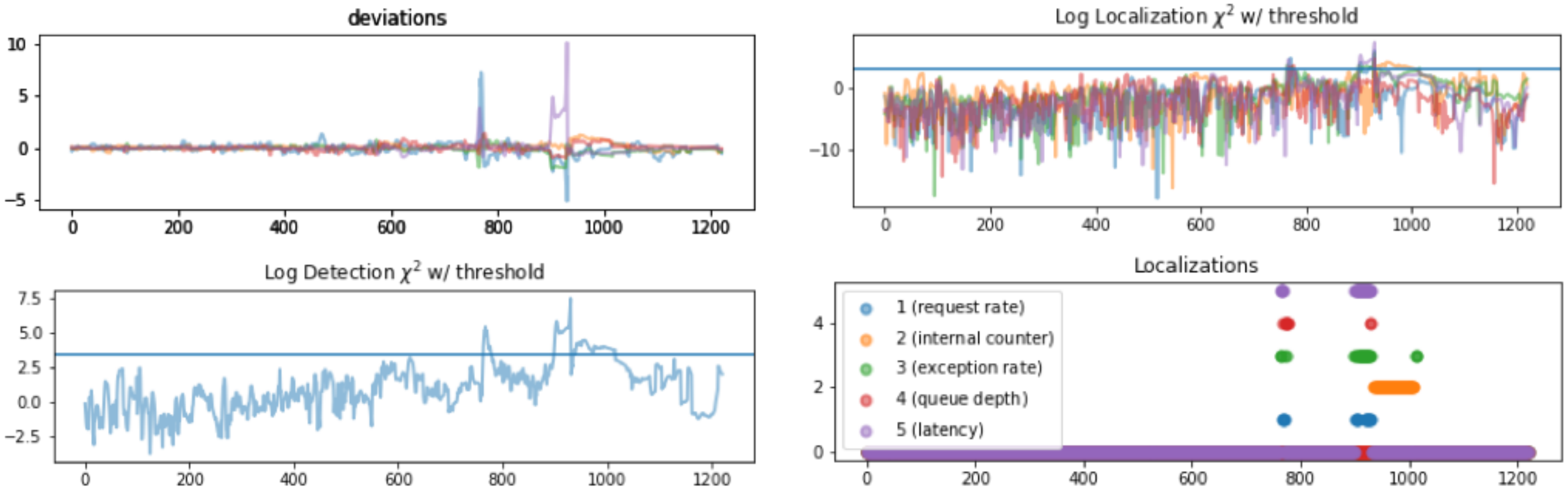
Numerical experiments: mongo database instance



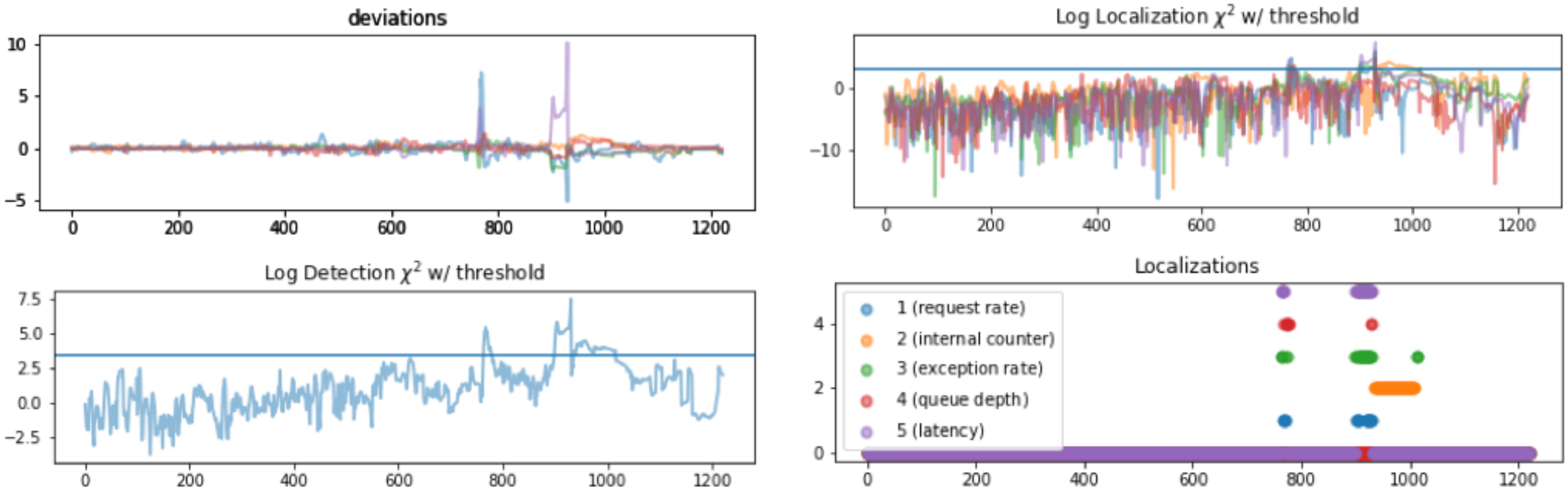




Numerical experiments: mongo database instance

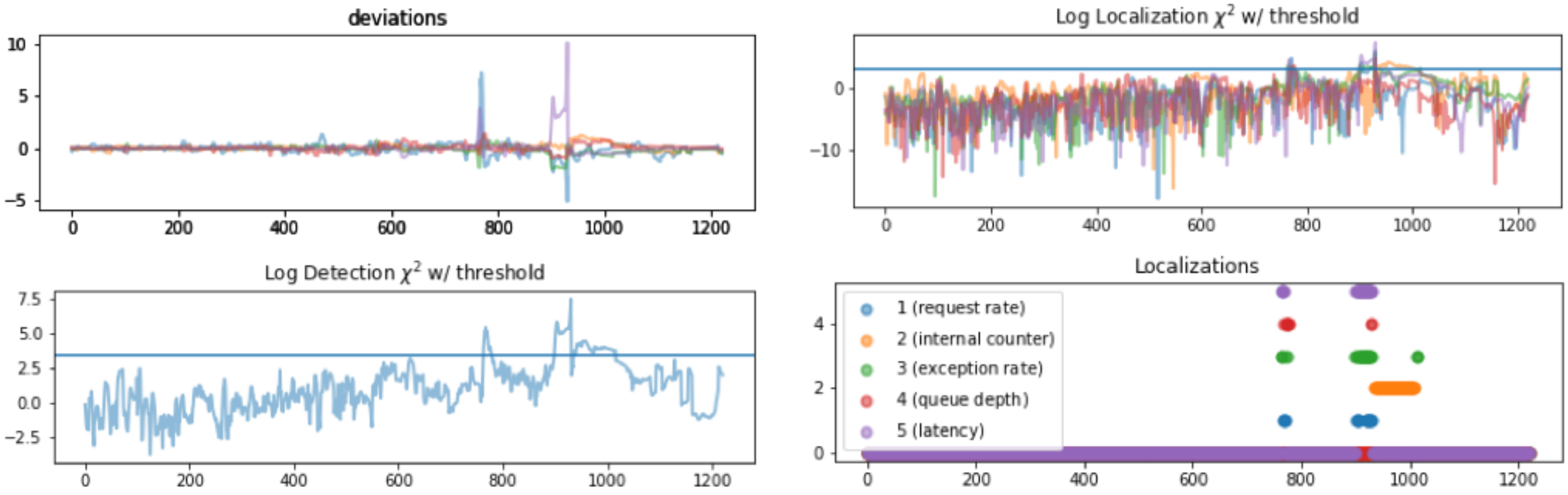


Numerical experiments: mongo database instance

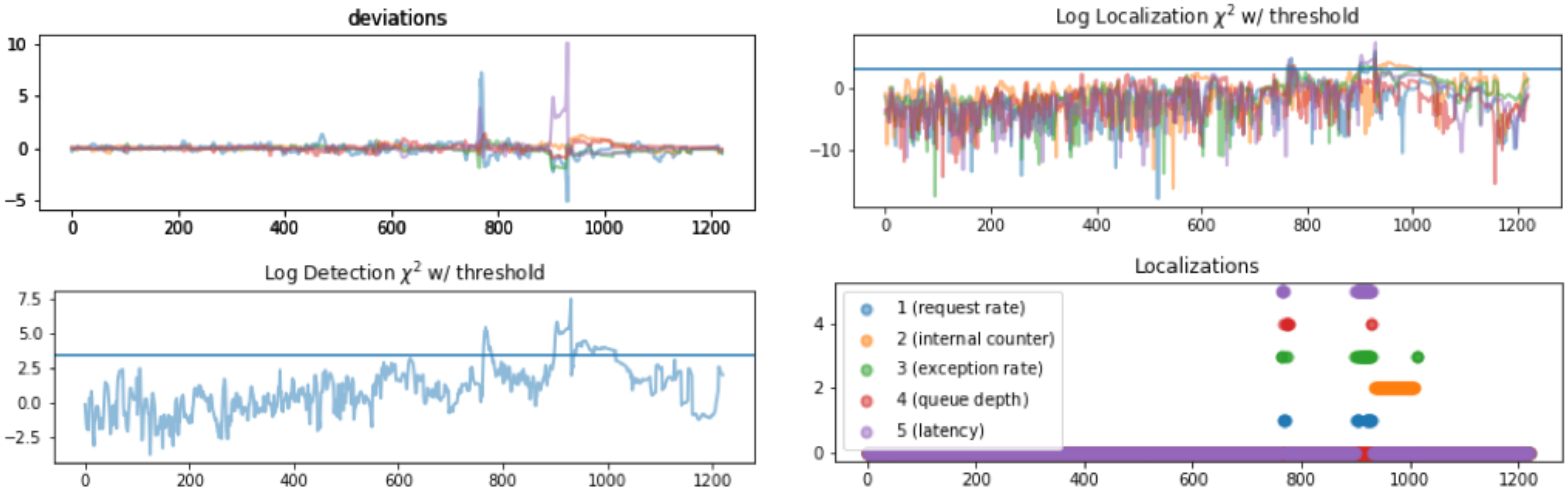


first detection excludes internal counter, may indicate *input change*

Numerical experiments: mongo database instance



Numerical experiments: mongo database instance



second detection includes counter, *internal change*

Our data model localizes potential problem sources

Our data model localizes potential problem sources

Thank you!