PCA DCRNN FDD OSCAE ANN CVA-FDA SVM EDBN FL DAGCN SS ERBM BN GAT RL CBAM GA CEEMDAN RF MRF-GCN TL DCA-BiGRU KPCA VQ-VAE-2 DiPCA MSA FDI MKECA MLP SSAE SVR KFDA FIS MSPCA JITL WKPCA DBN GMKL DRL DPLVM PSO PLVM GBM AAE DA CAE MPCA LSTM-AE DiPLS MVU FDIR RVFL-AE RBFNN STPCA LS-SVM DGCCA ANFIS CNN-SAE AJITL LWPLS-JITL GMM GCFA DQN DCCA ACO DPCA-SVM XGBoost RRVS OL F-DINAMITE RPCA PCA-CD-BN DiCCA TDLN-trees RCA MRF-GCN RNN CVA RVM KECA TS-FS SVAE LWR GDA PPCA LDA DDPG

A process optimality graph for prescriptive analytics generated using established machine learning methods

Tobias M Louw,

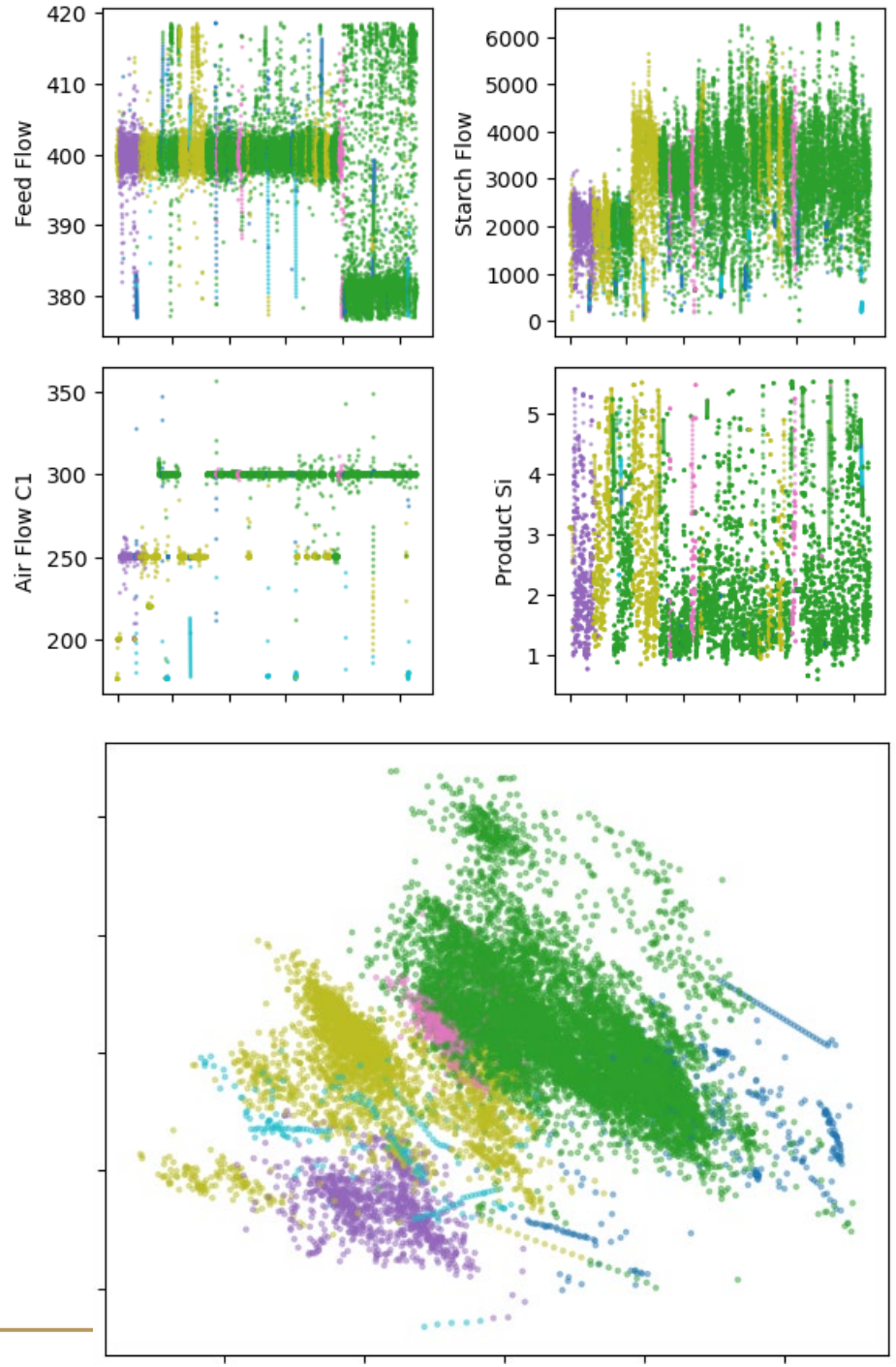
Alexander Schulze-Hulbe, Steven M Bradshaw

Department of Chemical Engineering, Stellenbosch University, Stellenbosch, 7600, South Africa (tmlouw@sun.ac.za)

STFT-CNN HHT HHT-SVM CWT-CNN MFCC MFCC-SVM PCA-KNN LDA-SVM ICA-SVM FDA-SVM CVA-SVM PLS-SVM PCR PCR-SVM RR RR-PLS LASSO ENet GLASSO SGL ALVEN SPLS SIMPLS NIPALS OSC O-PLS DOSC EPO GLS-W MSC SNV RNV EMSC OPLEC AUC-ROC KDE LOF iForest OC-SVM SVDD KPCA-SVDD DPCA-SVDD RSVDD YAMLA

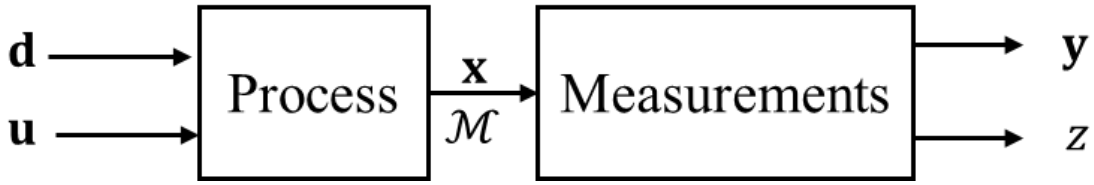
Prescriptive analytics

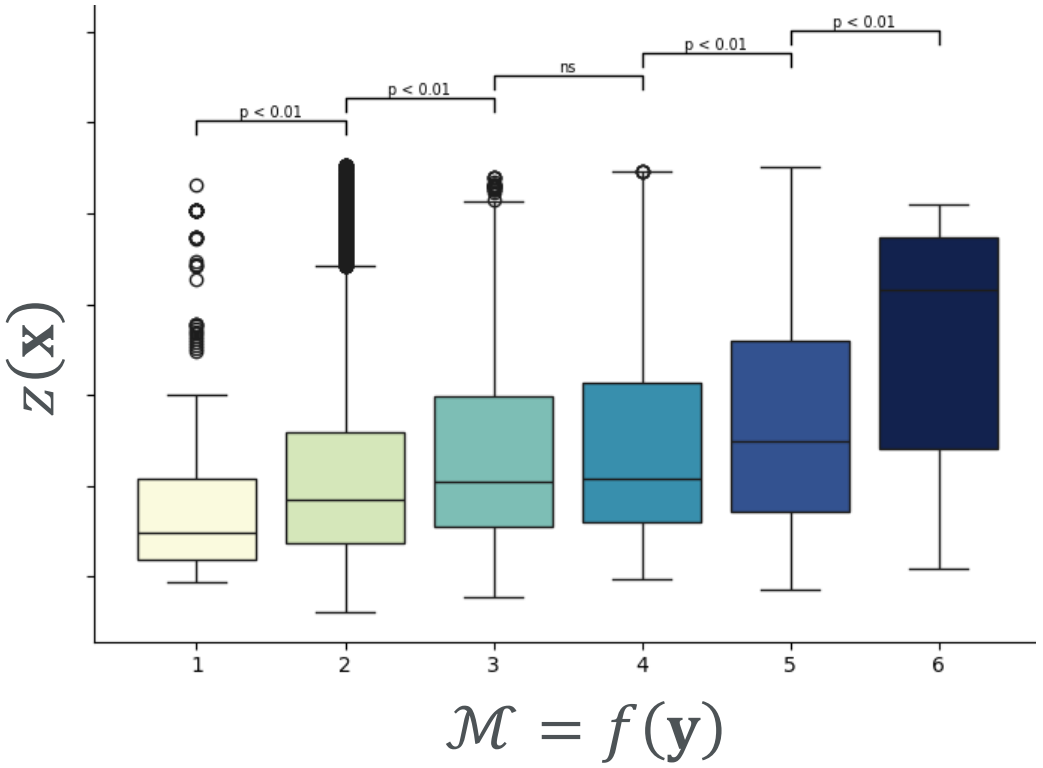
- Prescriptive analytics: Leverage data to recommend optimal course of actions (actionable advisories)
- Assume industrial operations can be classified into discrete process modes
- Plant performance may be improved by:
    1. Comparing current data to past operating conditions
    2. Identifying beneficial operator actions from history
    3. Recommending similar actions based on current process mode



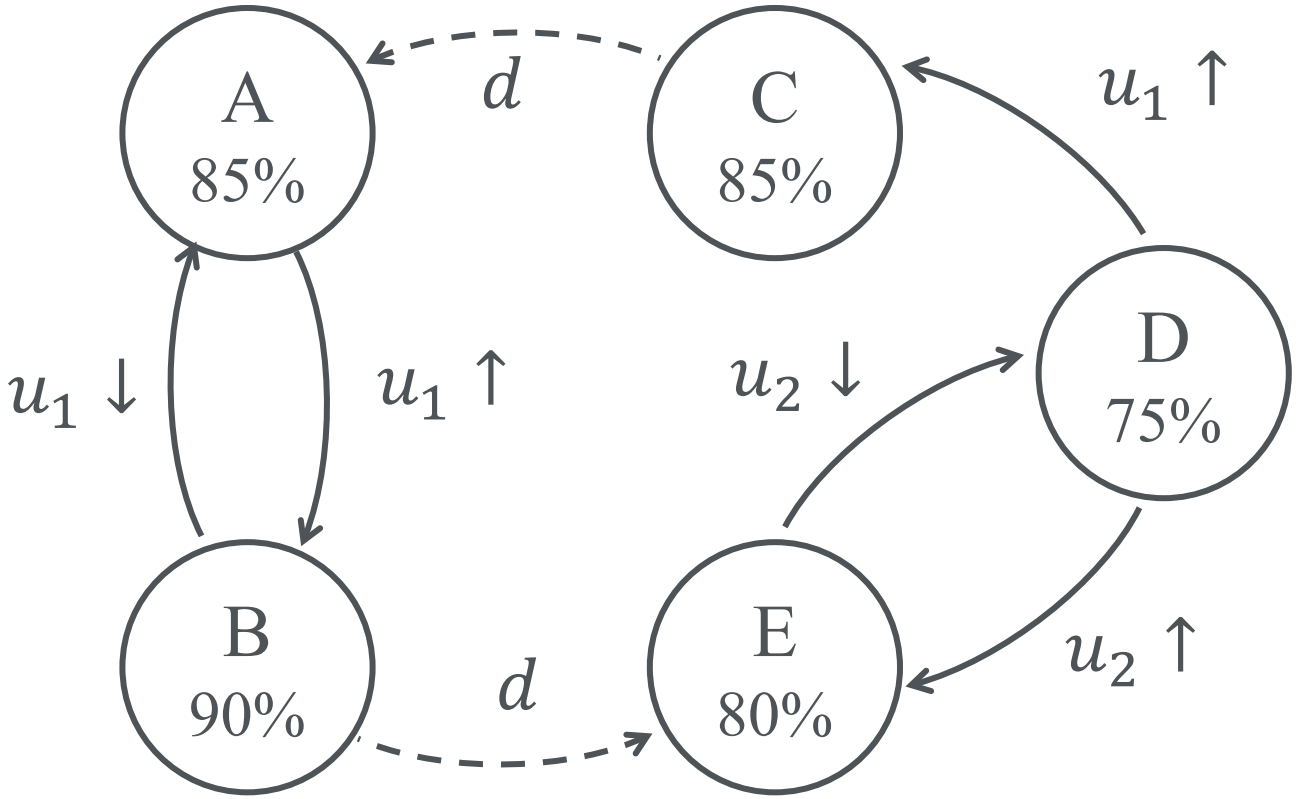
Optimality graph: definition and example

- Process defined by state $\mathbf{x}$,
- Subject to operator actions $\mathbf{u}$ and disturbances $\mathbf{d}$
- Measurements $\mathbf{y} = h(\mathbf{x})$
- Scalar performance indicator (KPI) $z(\mathbf{x})$ may be assigned retrospectively
- Classified into discrete modes $\mathcal{M} = f(\mathbf{x})$
- KPI varies between modes with reasonable statistical significance

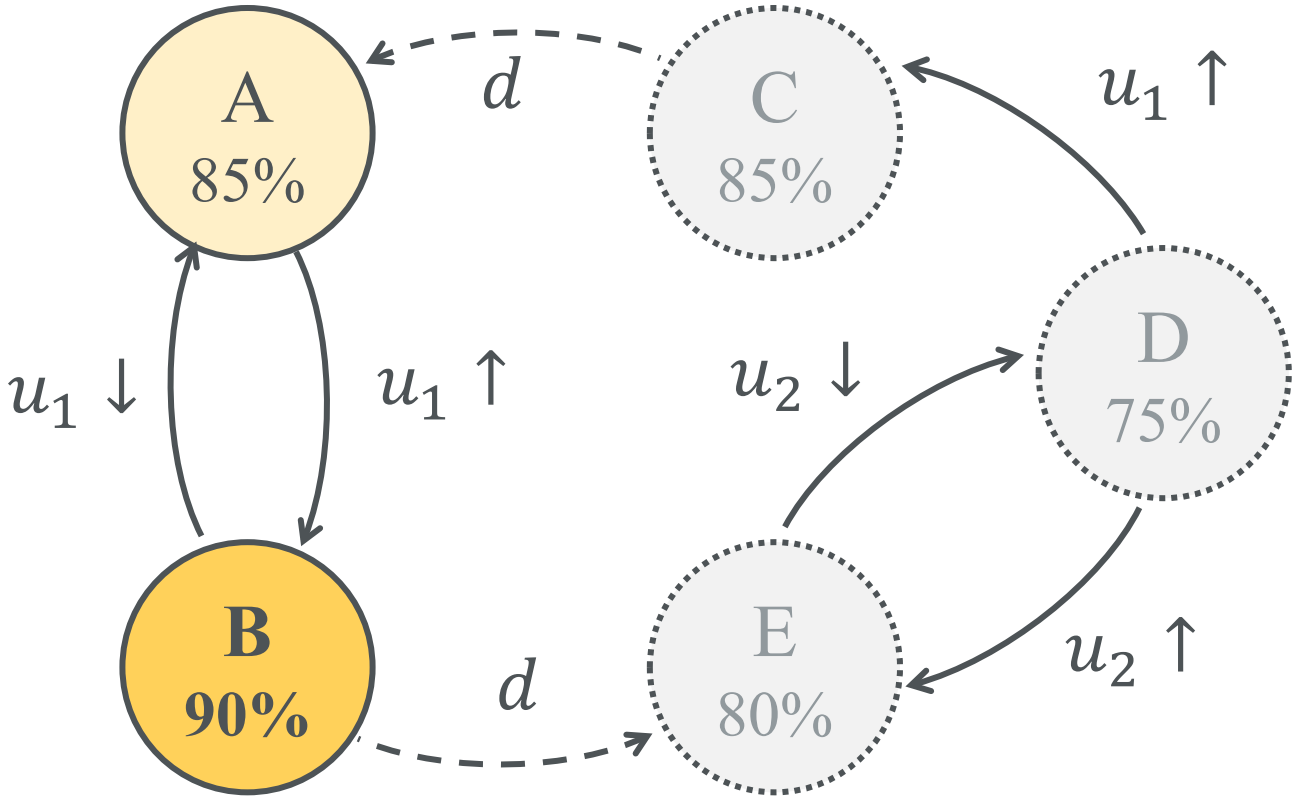




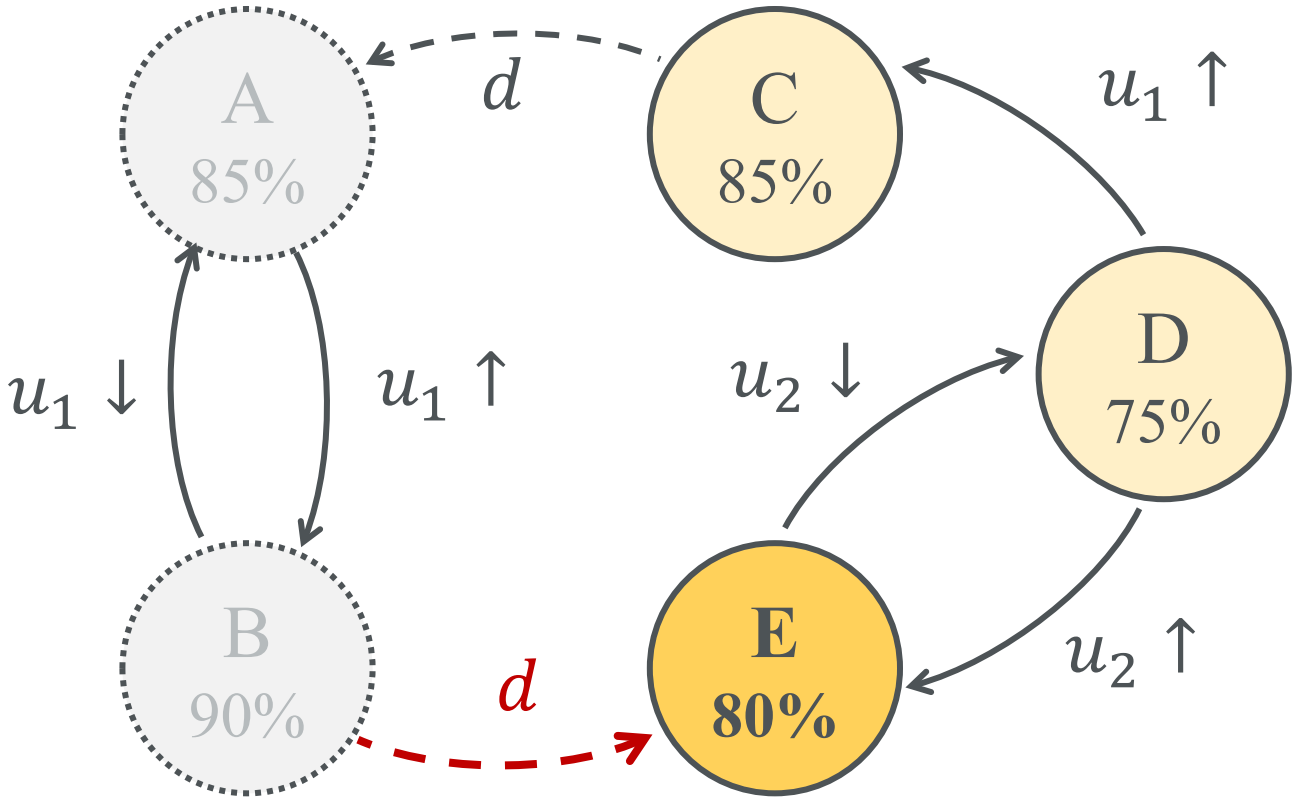
Optimality graph: definition and example



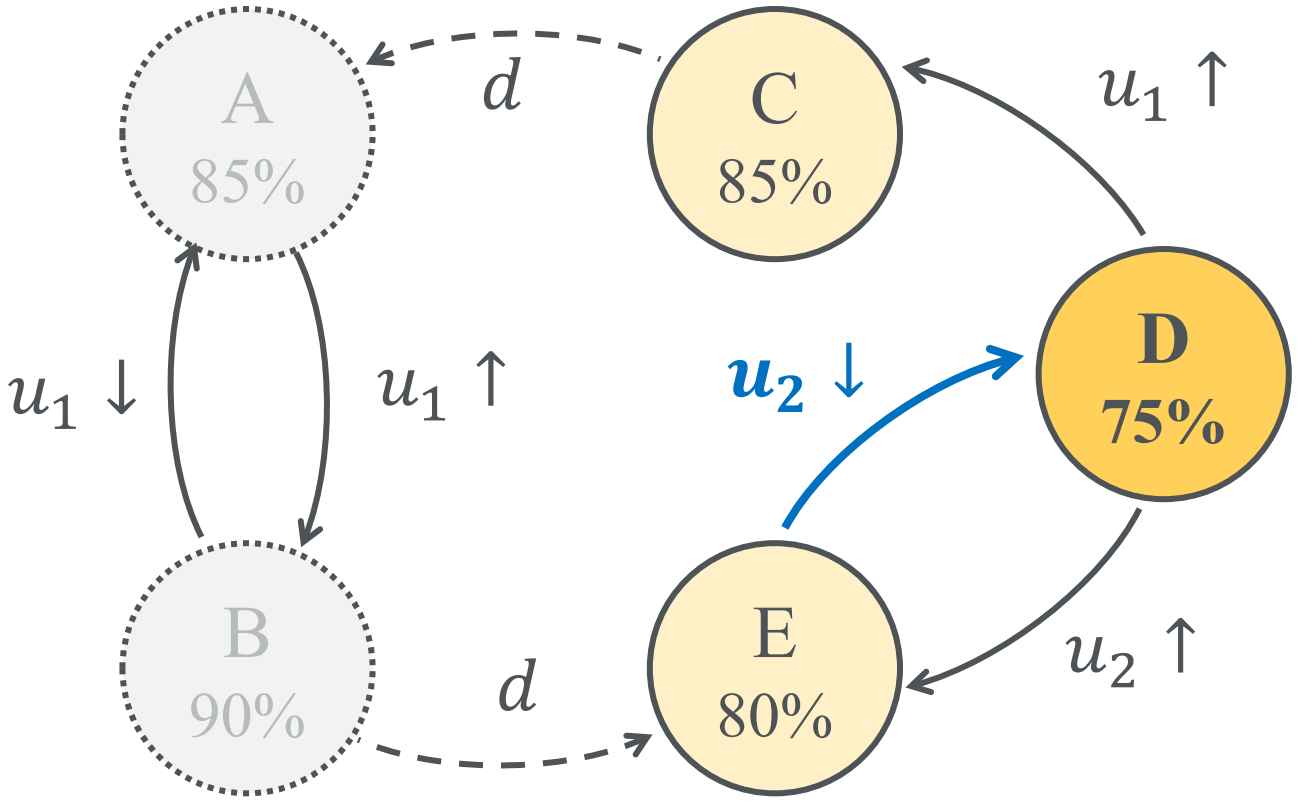
Optimality graph: definition and example



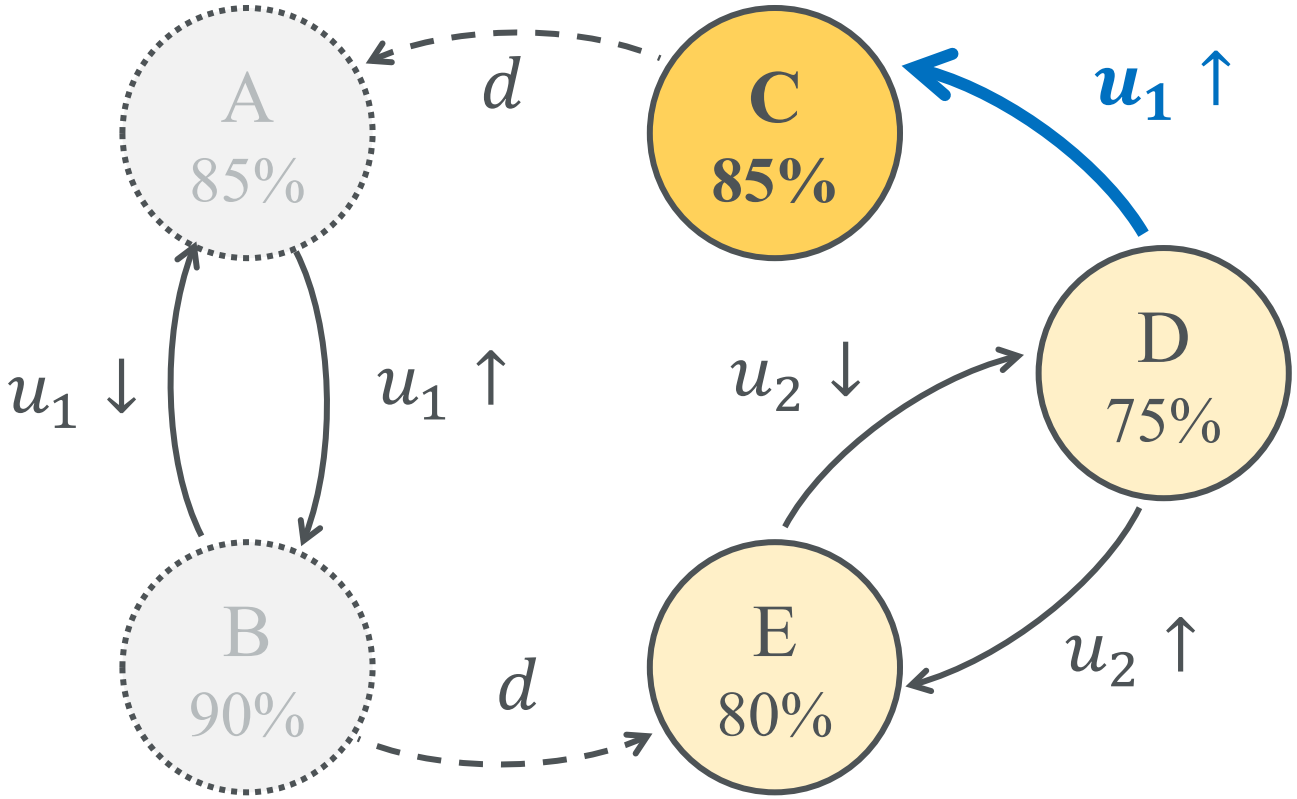
Optimality graph: definition and example



Optimality graph: definition and example



Optimality graph: definition and example



No YAMLA

- Process monitoring literature overflows with “new machine learning methods”
- Seldom useful to plant engineers
- Avoid Yet Another Machine Learning Method (YAMLA)
- Leverage basics available in well-established libraries (PID for Prescriptive Analytics)
- Careful prompt gets you 95% of the way
- Enable rapid data evaluation

```
# Import packages
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from scipy import stats
from sklearn ... import ...
import seaborn as sns
import lightgbm as lgb
import shap
```

Data wrangling challenges

- Assumes $z = z[\mathcal{M}] = z[f(\mathbf{x})] = z\left[f\left(h^{-1}(\mathbf{y})\right)\right]$ i.e., **assumes observability**
  - Lack of observability must be addressed with instrumentation, not machine learning

Data wrangling challenges

- Assumes $z = z[\mathcal{M}] = z[f(\mathbf{x})] = z\left[f\left(h^{-1}(\mathbf{y})\right)\right]$ i.e., **assumes observability**
  - Lack of observability must be addressed with instrumentation, not machine learning

- May require **extensive feature engineering** to ensure cluster separability
  - Time series alignment
  - Dynamic embedding
  - Increased features → curse of dimensionality

Data wrangling challenges

- Assumes $z = z[\mathcal{M}] = z[f(\mathbf{x})] = z\left[f\left(h^{-1}(\mathbf{y})\right)\right]$ i.e., **assumes observability**
  - Lack of observability must be addressed with instrumentation, not machine learning

- May require **extensive feature engineering** to ensure cluster separability
  - Time series alignment
  - Dynamic embedding
  - Increased features → curse of dimensionality

- May require multiple KPIs, i.e., $\mathbf{z} \in \mathbb{R}^k$
  - Requires additional process insight, multi-objective optimisation, etc.

Approach

0. Data wrangling
1. Dimensionality reduction
2. Clustering and cluster refinement
3. Mode labelling
4. Transition labelling
5. Online prescriptive analytics

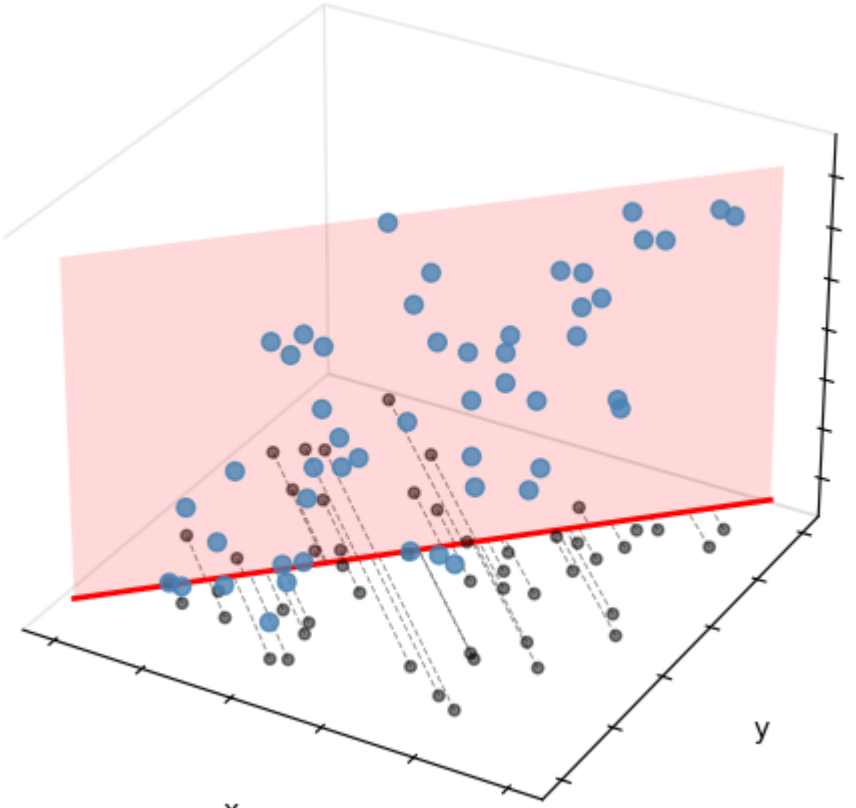
Dimensionality reduction

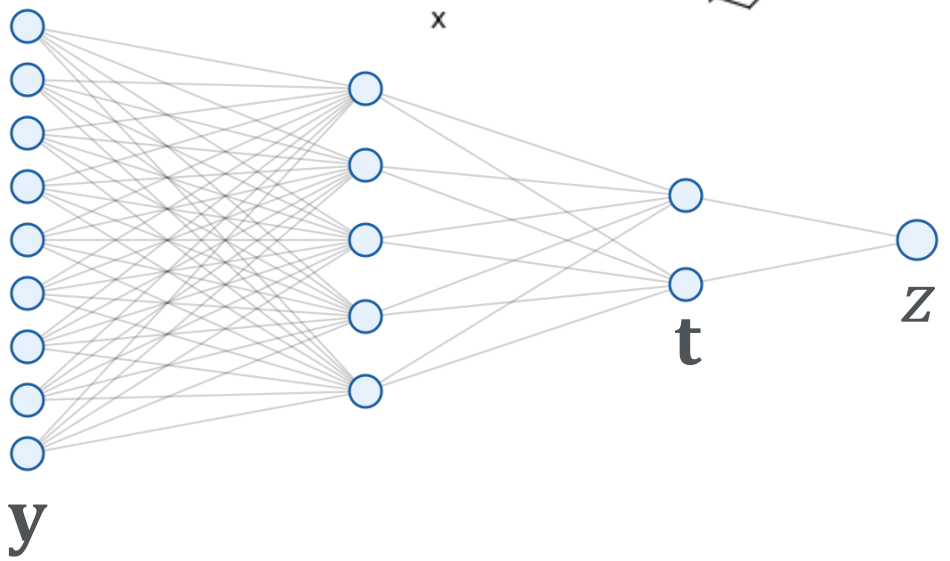
- Dimensionality reduction
  ~ *partially* unsupervised regularisation
- Reduces overfitting by combining covarying or mutually informative features into single components (method dependent)
- Latent space should capture variance in KPI
- Projection:

$$\mathbf{t} = \pi(\mathbf{y}), \qquad \mathbf{t} \in \mathbb{R}^d, \qquad \mathbf{y} \in \mathbb{R}^q, \qquad d < q$$

- Regression:

$$z = g(\mathbf{t})$$





Dimensionality reduction

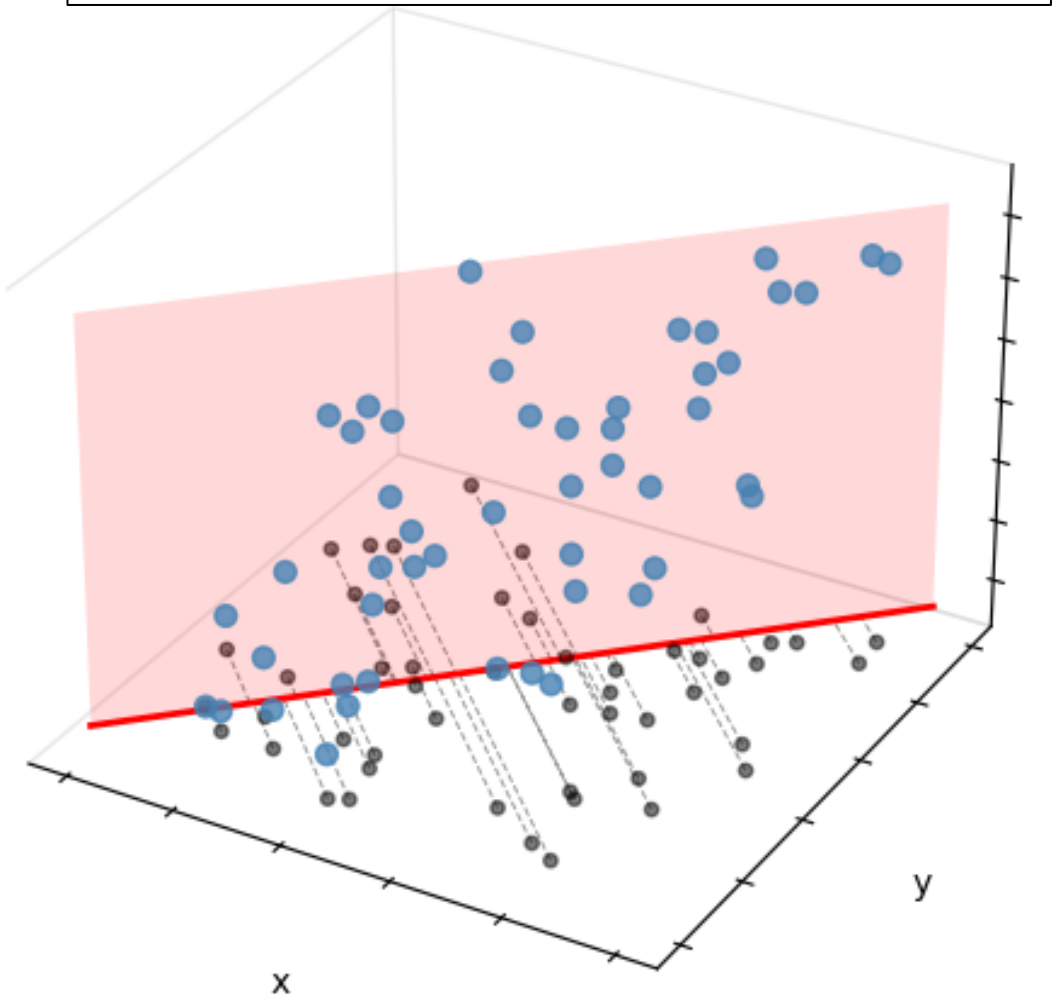
Partial least squares (PLS):
$\mathbf{t} = X\mathbf{p}$ where
$p = \arg\max[\mathrm{cov}^2(\mathbf{z}, X\mathbf{p})\mathrm{var}(X\mathbf{p})]$

- Dimensionality reduction
  ~ *partially* unsupervised regularisation
- Reduces overfitting by combining covarying or mutually informative features into single components (method dependent)
- Latent space should capture variance in KPI
- Projection:

$$\mathbf{t} = \pi(\mathbf{y}), \qquad \mathbf{t} \in \mathbb{R}^d, \qquad \mathbf{y} \in \mathbb{R}^q, \qquad d < q$$

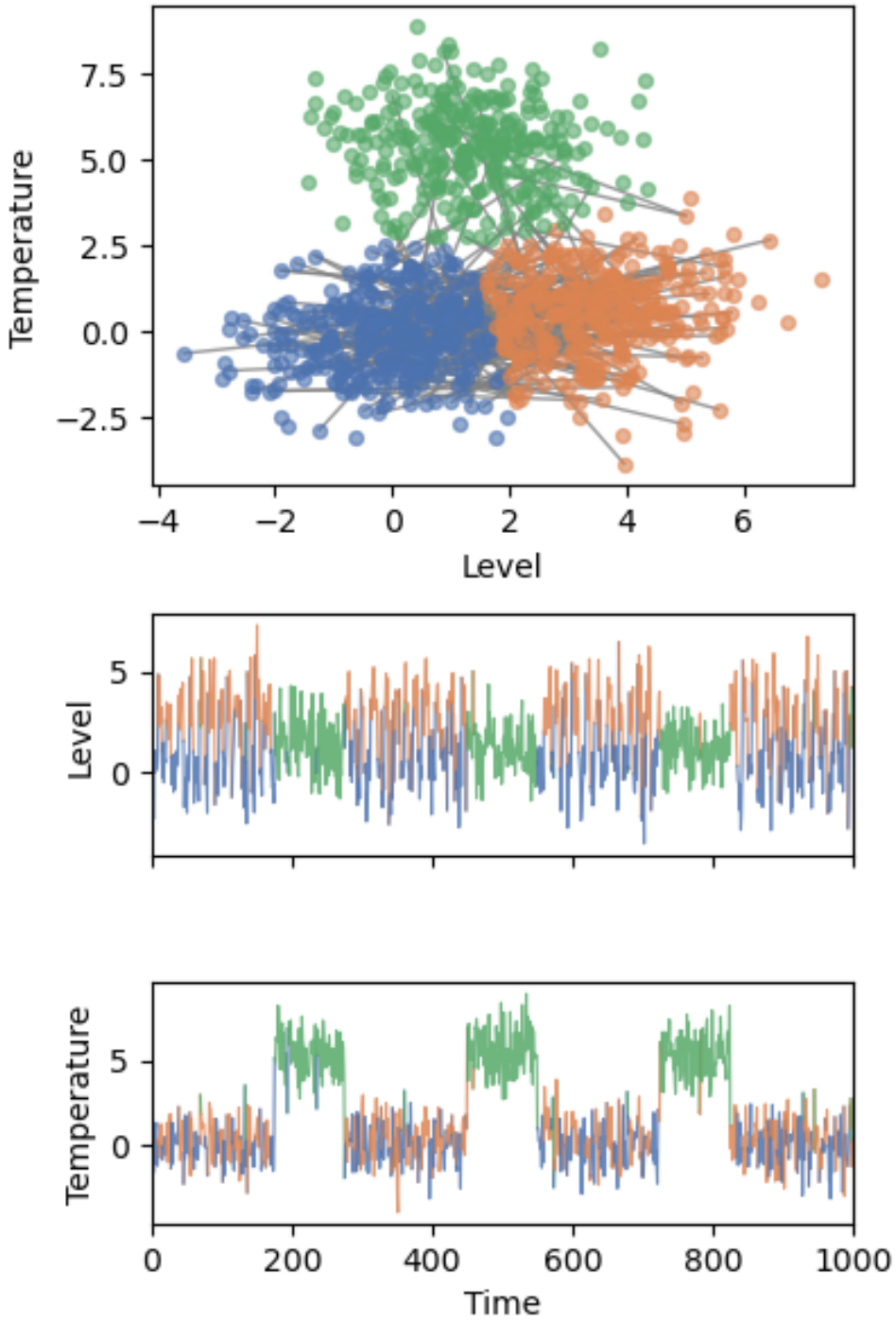
- Regression:

$$z = g(\mathbf{t})$$



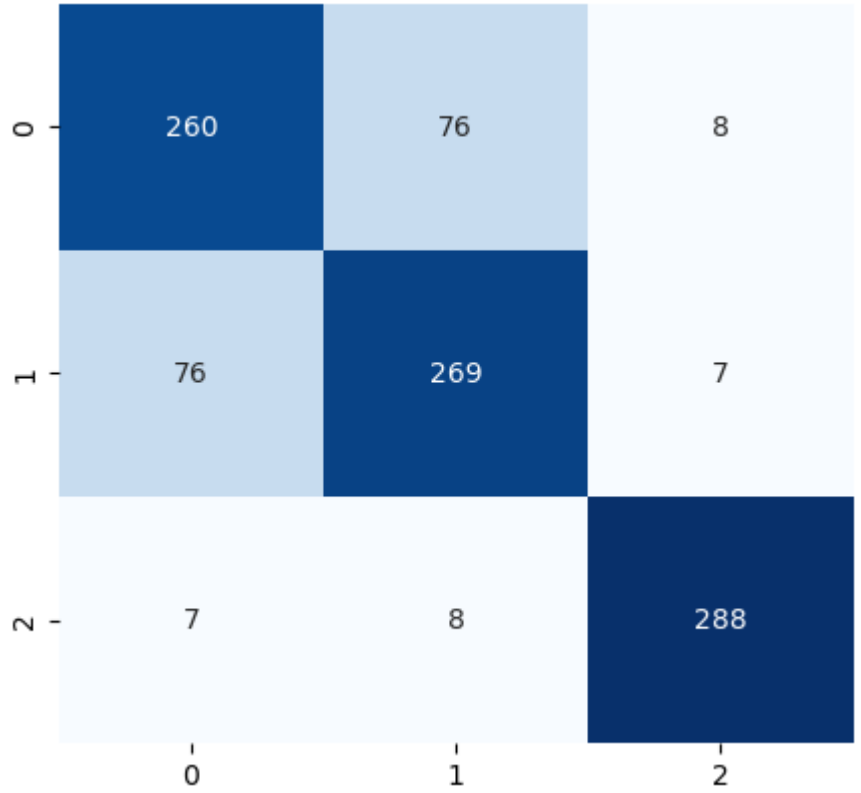
Clustering and cluster refinement

- Operating modes represent a specific set of states that the process occupies for a significant period of time
- Operating modes identified by clustering data in latent space $\mathbf{t} \in \mathbb{R}^d$ (unsupervised)
  - k-means is a reasonable first choice
- Clustering ignores time series nature of data: no requirement for contiguous cluster segments
- Clusters with rapid switching should be merged into a single cluster (i.e., operating mode)



Clustering and cluster refinement

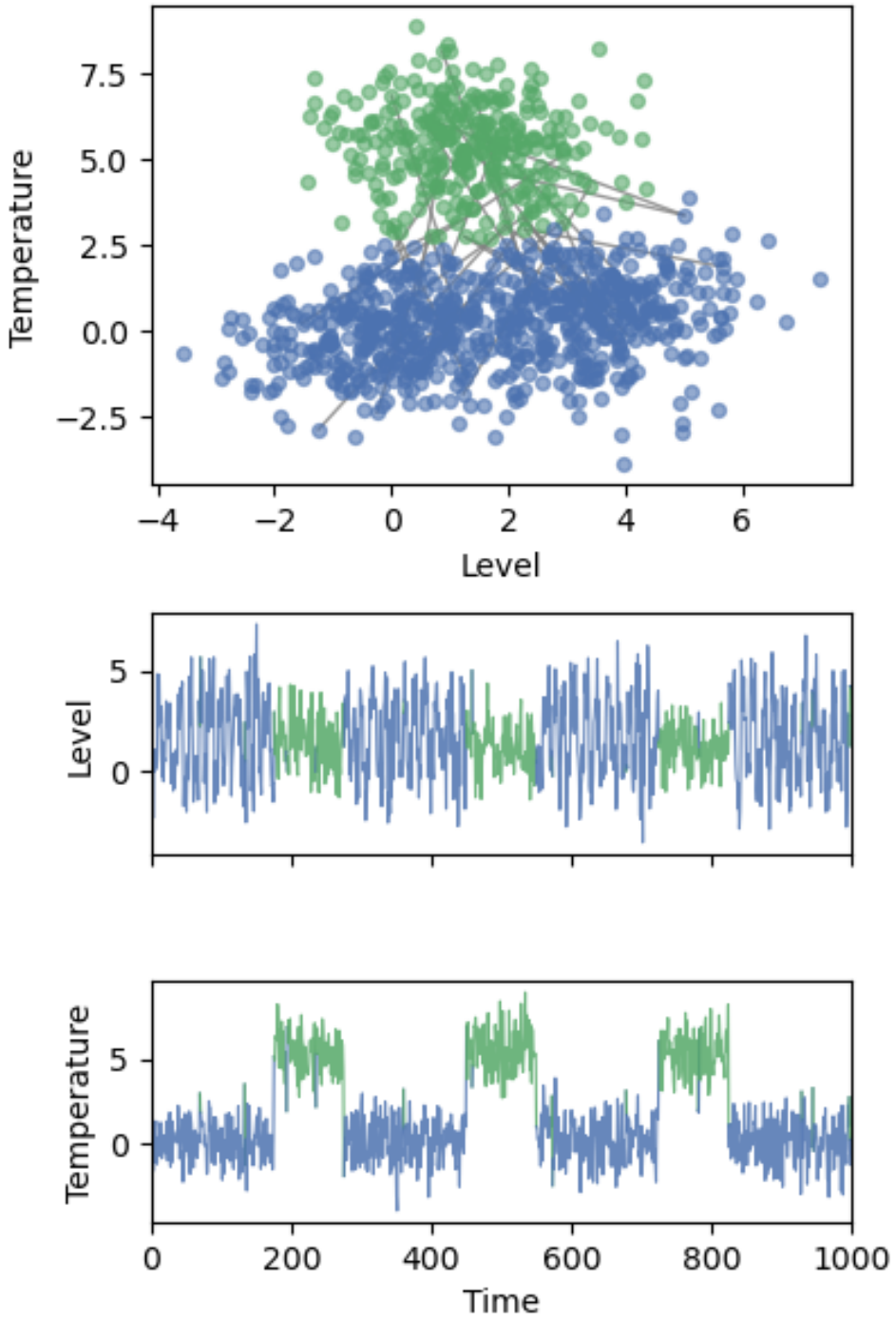
- Identify rapid mode switching using a confusion matrix between $\mathcal{M}(t_k)$ and $\mathcal{M}(t_{k+1})$
  - Matrix shows number of transitions from mode i to mode j
  - Merge clusters with many transitions into one



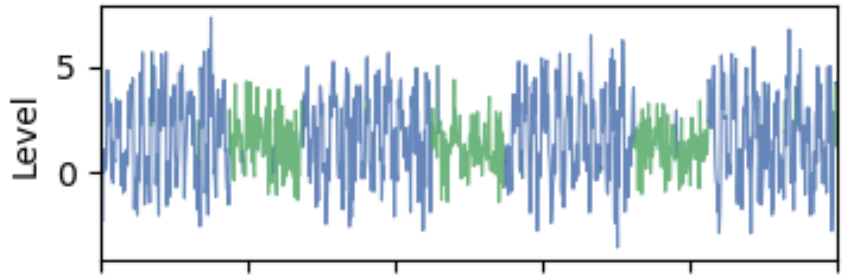

```
cm = confusion_matrix(df['cluster'][:-1], df['cluster'][1:])
sns.heatmap(cm, annot=True, fmt='.0f', cmap='Blues')
```

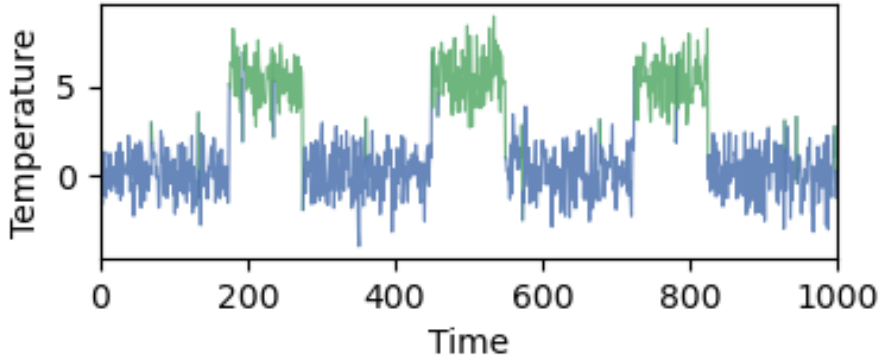
Clustering and cluster refinement

- Identify rapid mode switching using a confusion matrix between $\mathcal{M}(t_k)$ and $\mathcal{M}(t_{k+1})$
  - Matrix shows number of transitions from mode i to mode j
  - Merge clusters with many transitions into one

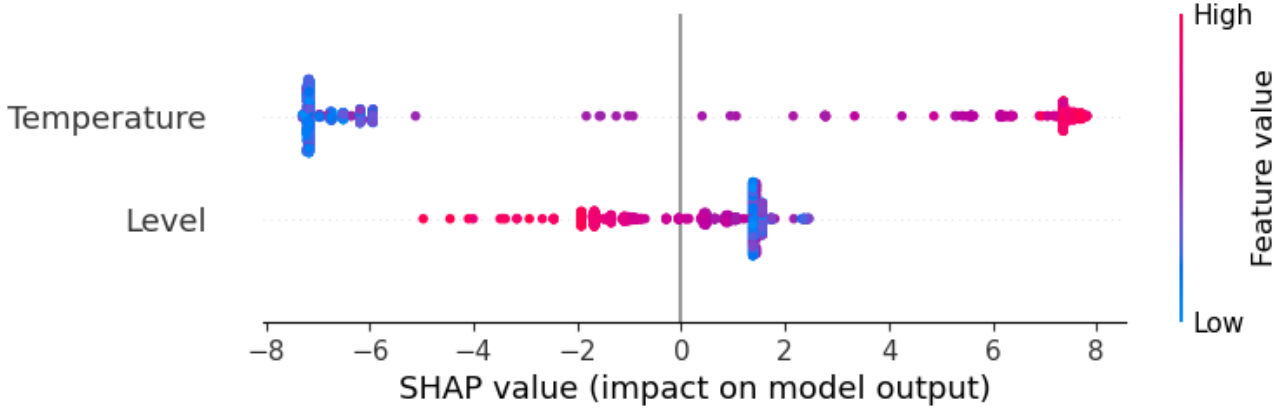


Mode labelling and transitions





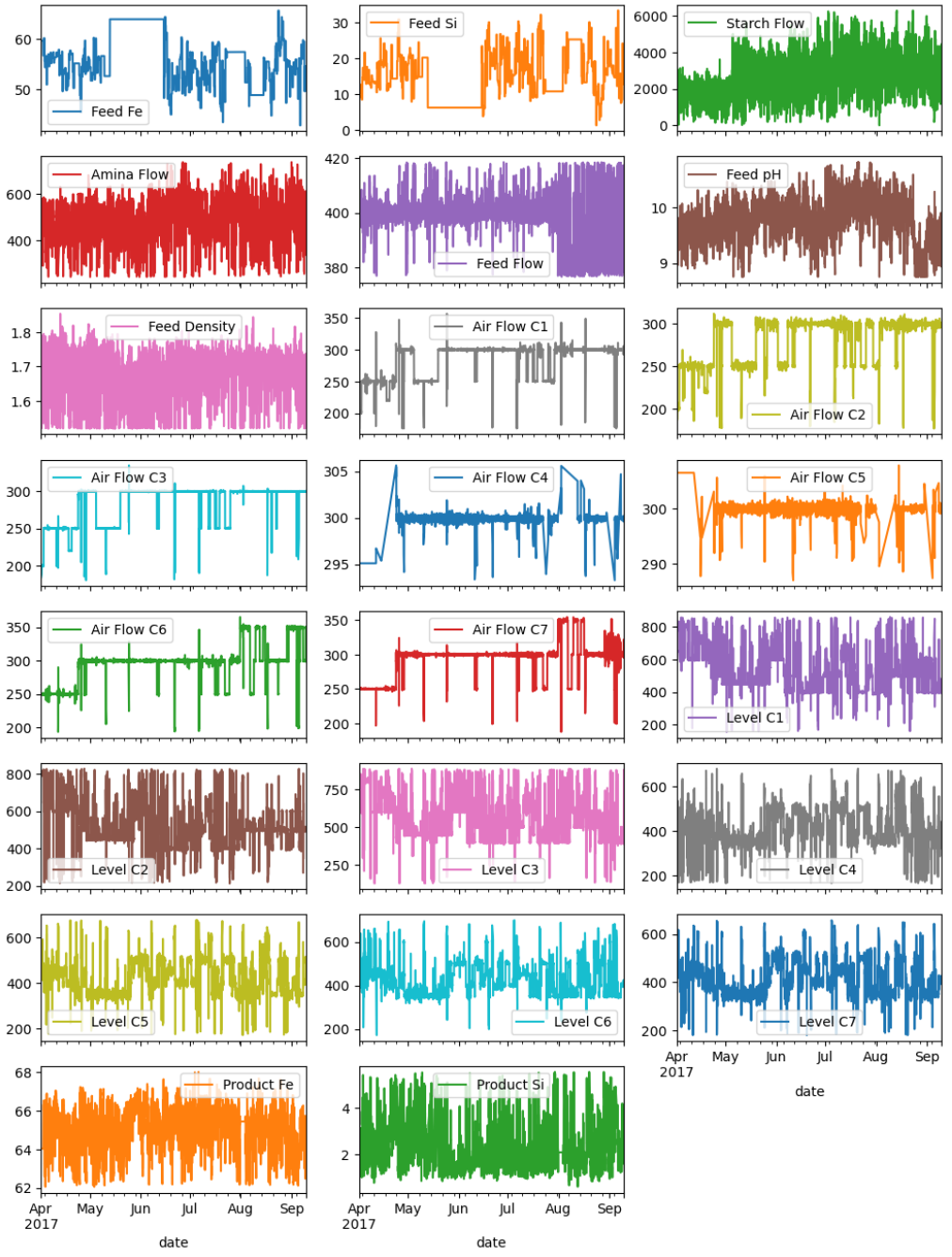
- Descriptive mode labels are far more useful than arbitrary numbers
- Use SHAP to generate descriptive labels according to features that significantly affect clustering results
- Inspect transitions in time series data and determine whether the transition was caused by operator action or not
- Optimality graph can be constructed manually or using appropriate software



Case study: iron ore flotation data

https://www.kaggle.com/datasets/edumagalhaes/quality-prediction-in-a-mining-process

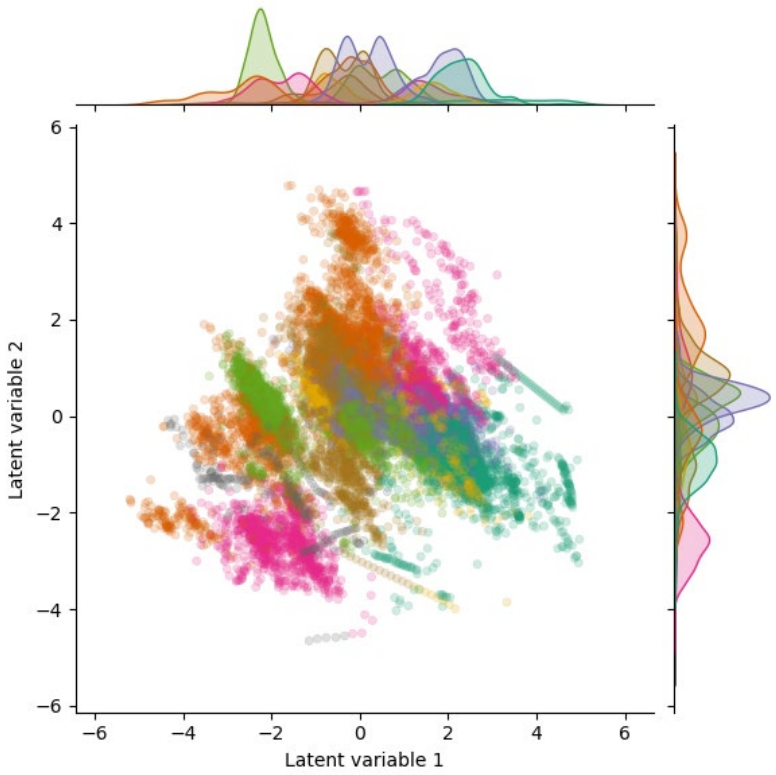
- Dataset contains 23 features (incl. KPI = % silica)
- Downsampled to 16 000 observations every 15 min

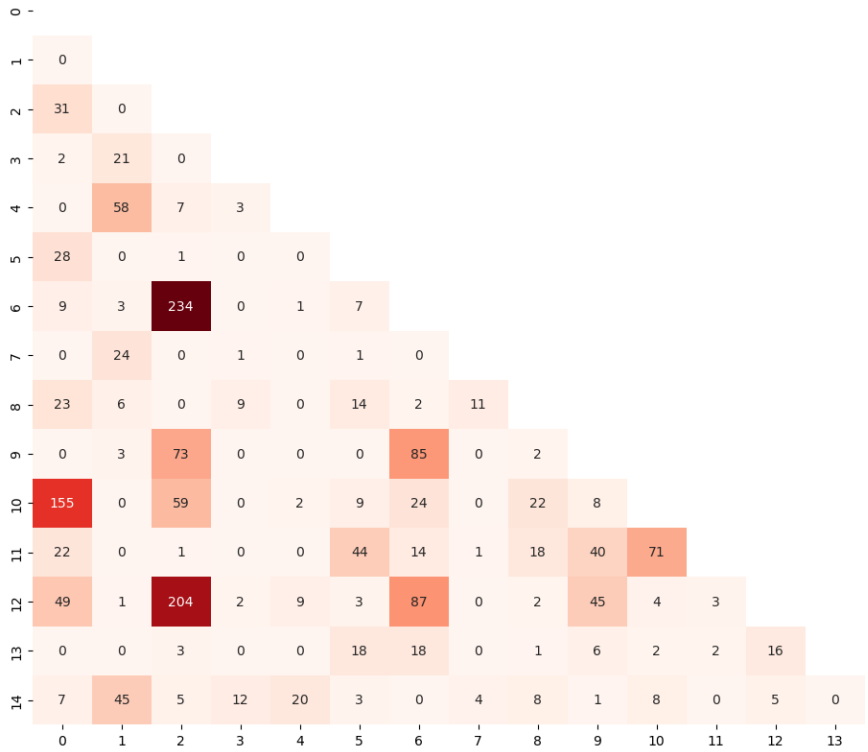


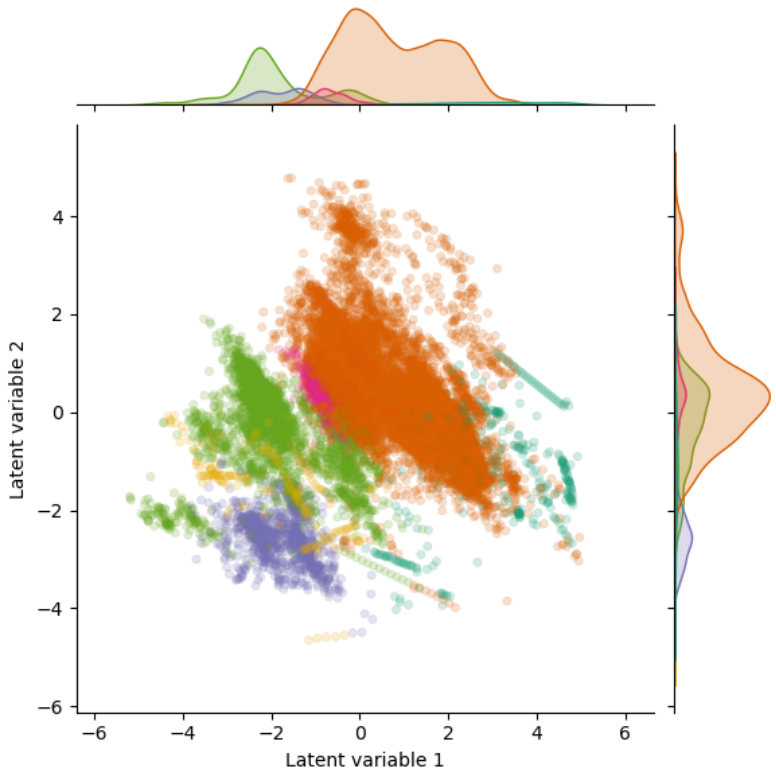
Case study: iron ore flotation data

https://www.kaggle.com/datasets/edumagalhaes/quality-prediction-in-a-mining-process

- Partial Least Squares for dimensionality reduction
- K-means clustering with cluster refinement

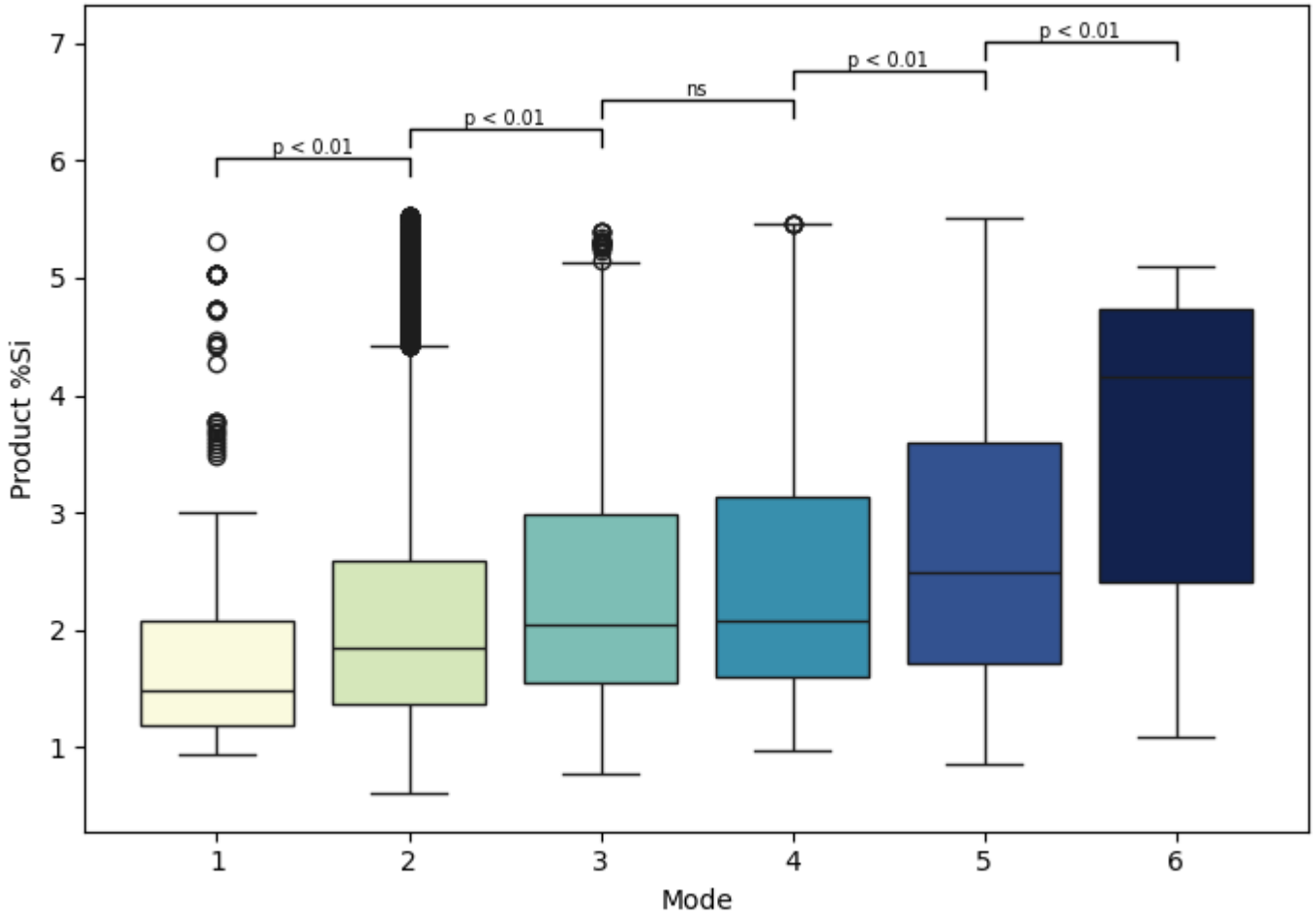


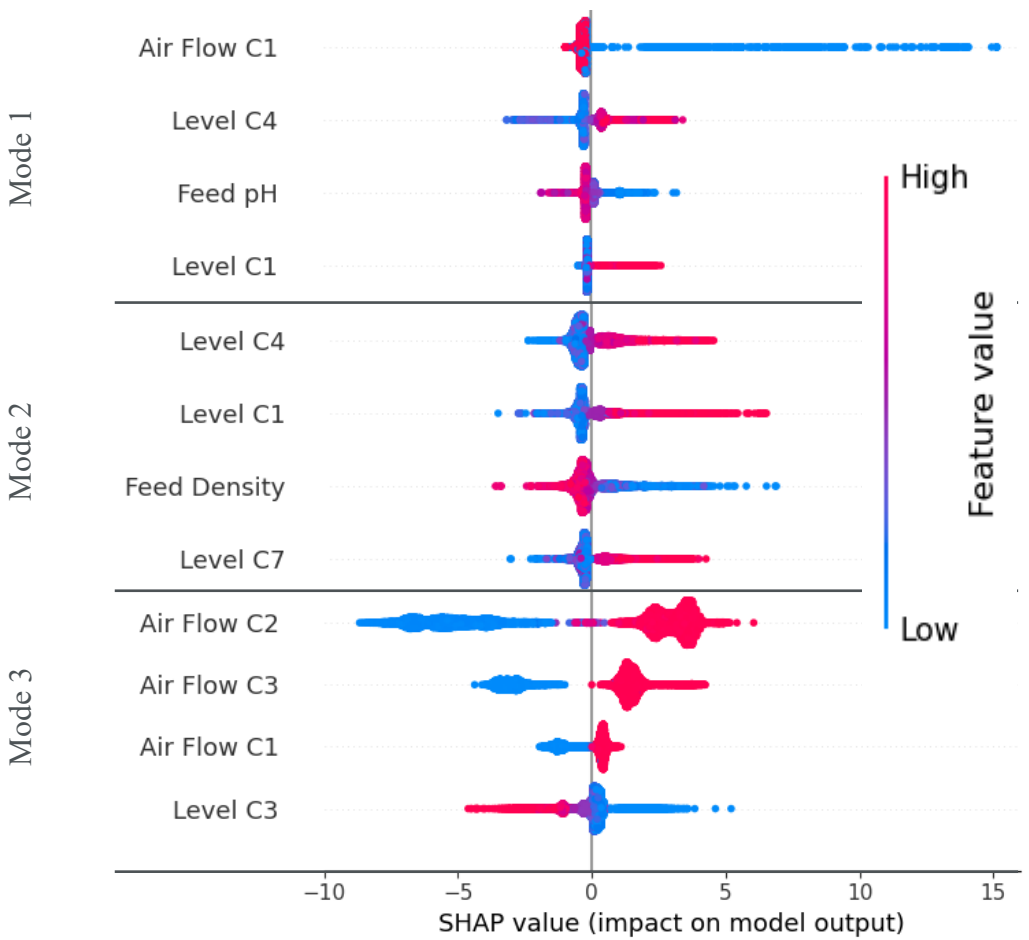




Case study: iron ore flotation data

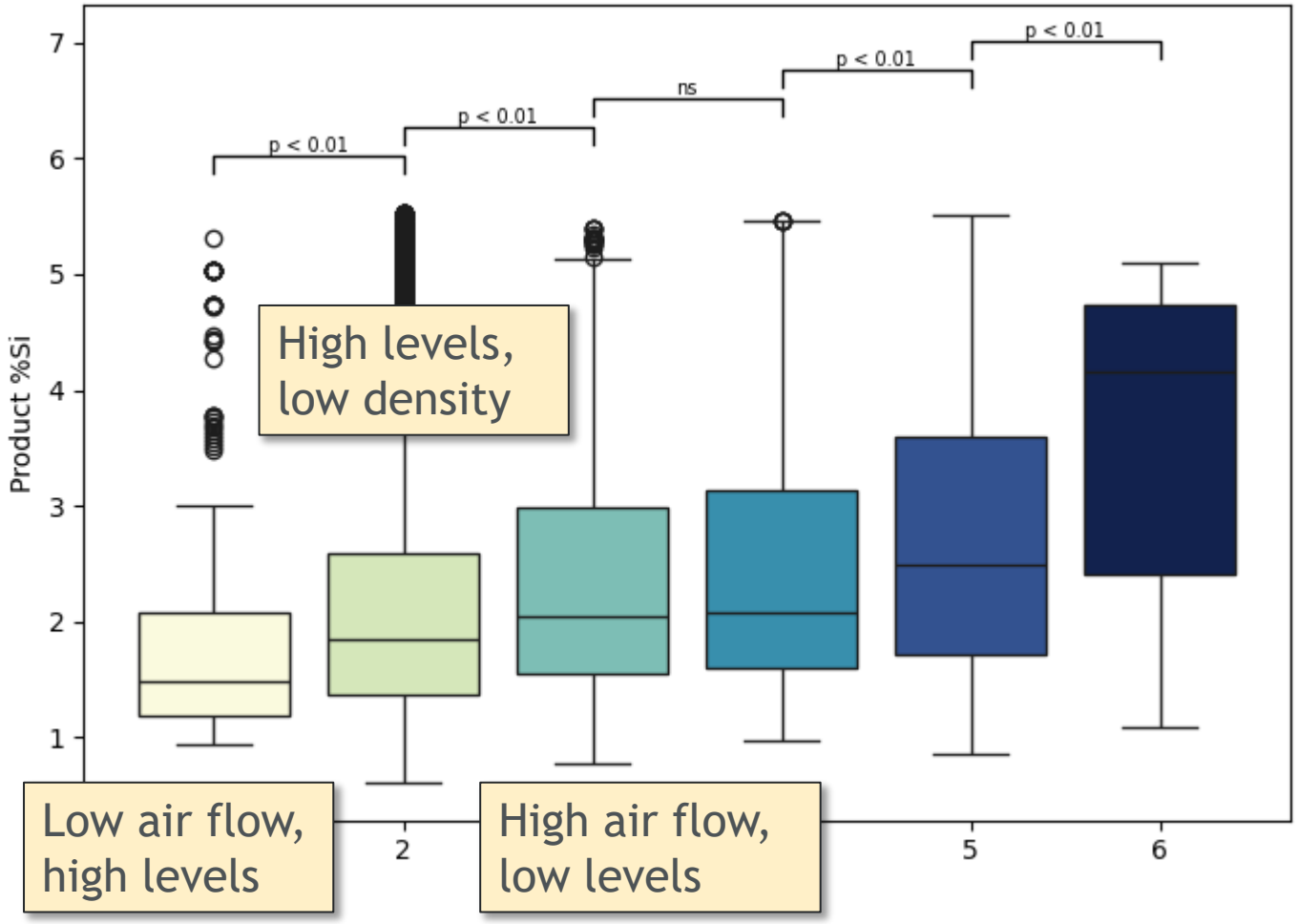
https://www.kaggle.com/datasets/edumagalhaes/quality-prediction-in-a-mining-process

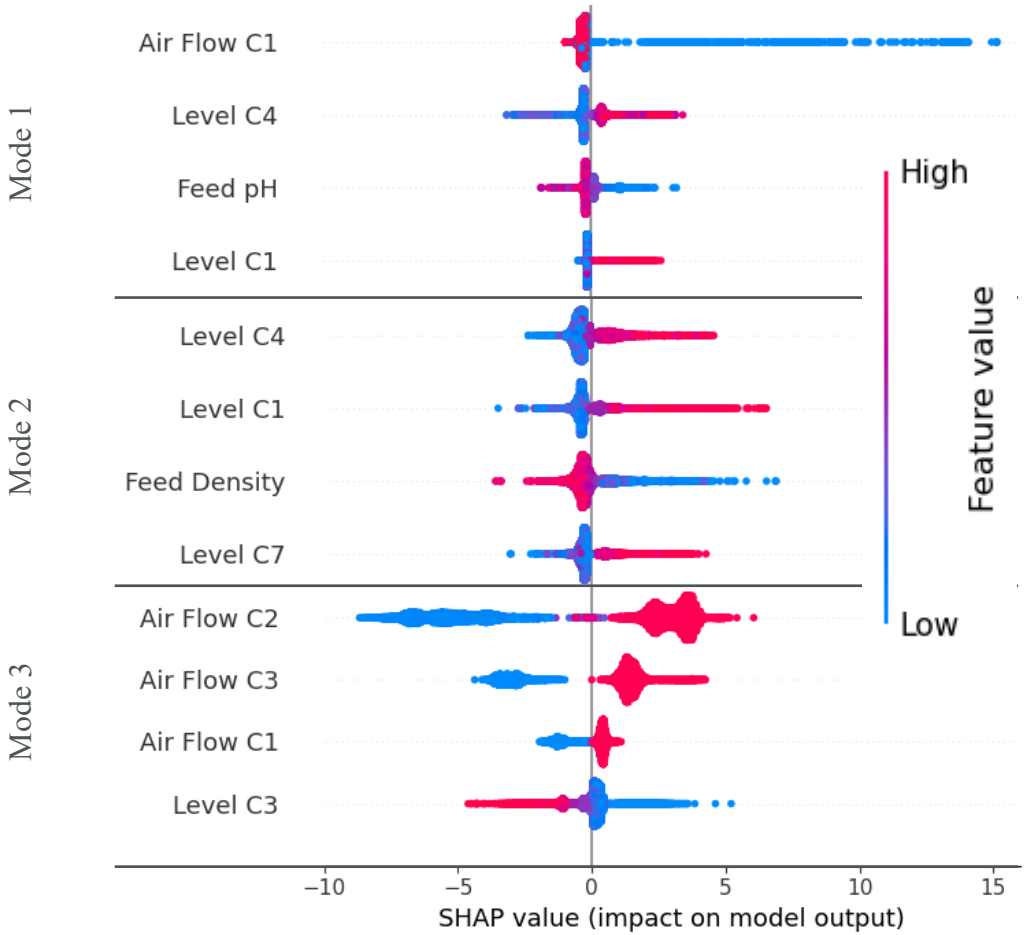




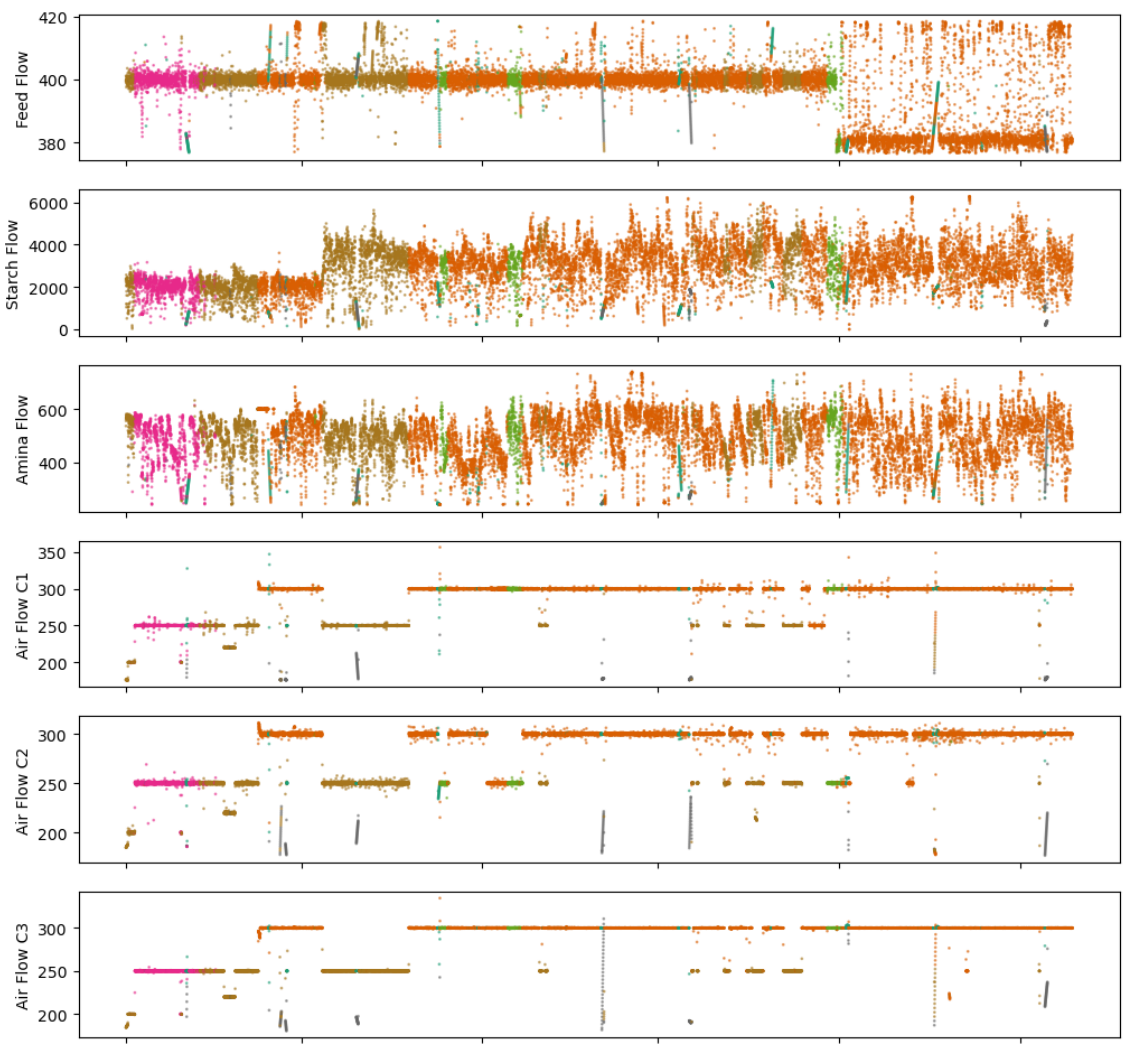
Case study: iron ore flotation data

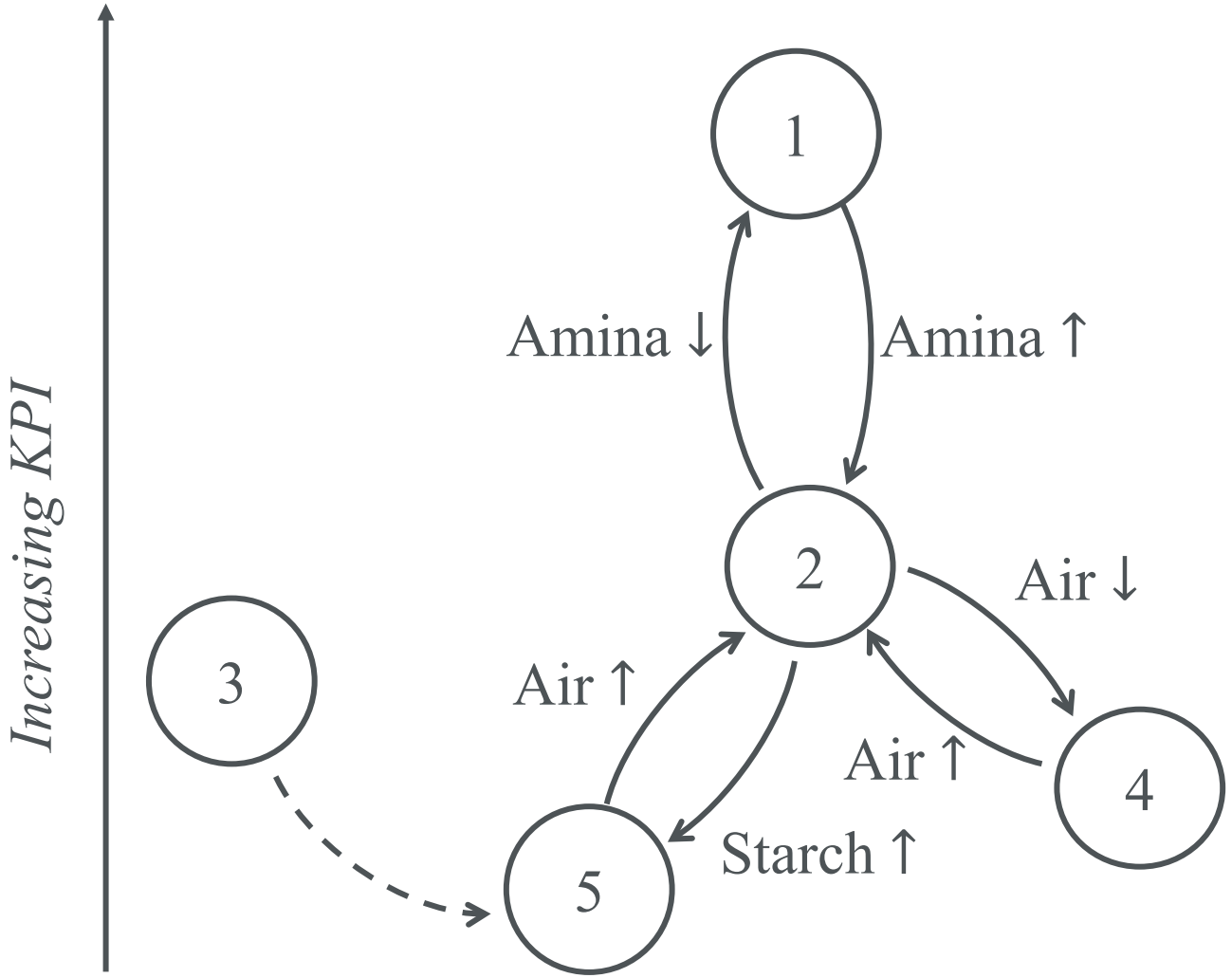
https://www.kaggle.com/datasets/edumagalhaes/quality-prediction-in-a-mining-process





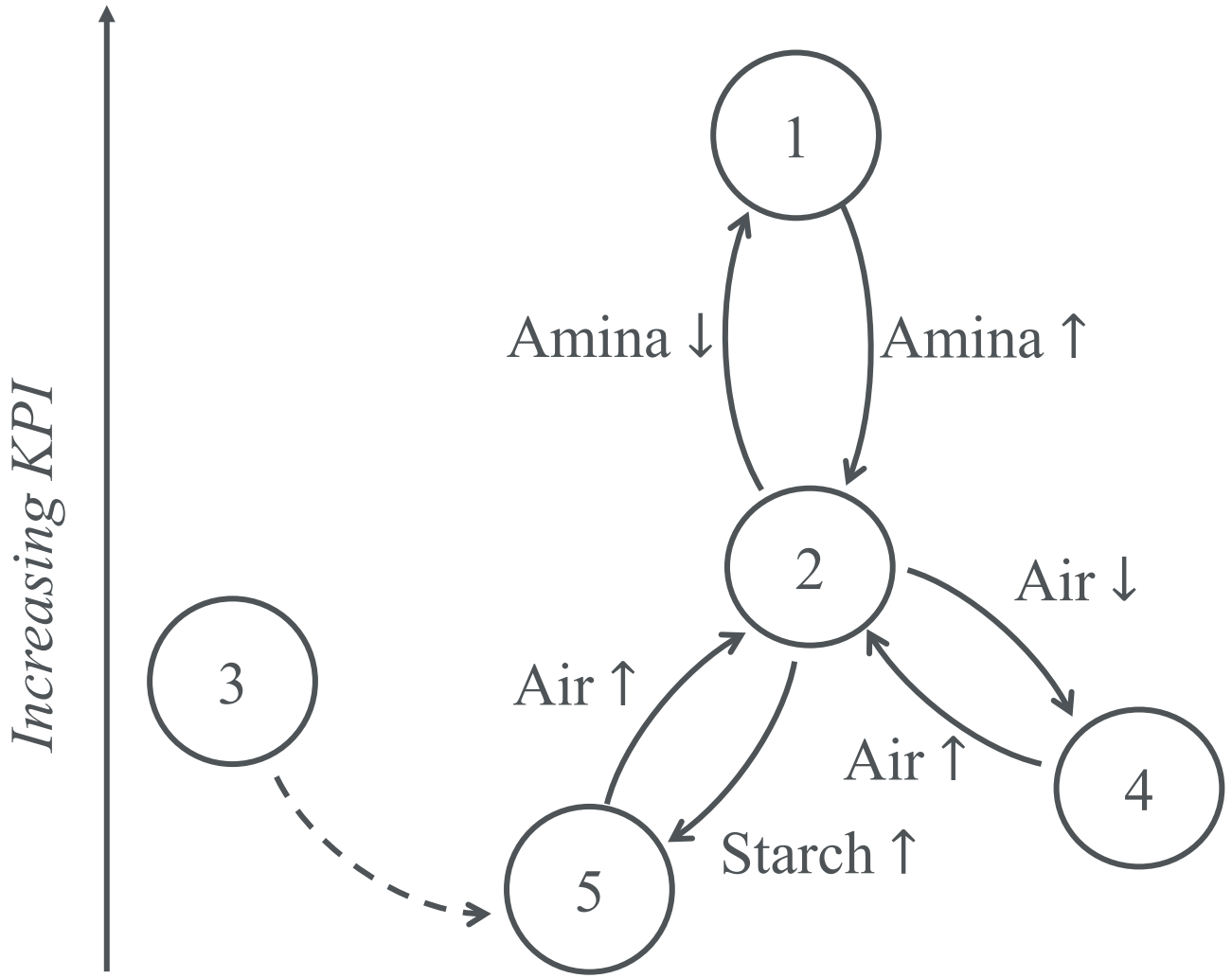
Conclusions





Conclusions

- Prescriptive analytics generated in the form of an optimality graph
- No YAMLA: only standard ML libraries utilised throughout
- Rapidly generated preliminary results not entirely convincing, but indicates potential value
- Suitable for immediate industry adoption



Stellenbosch
UNIVERSITY
IYUNIVESITHI
UNIVERSITEIT

Thank you
Enkosi
Dankie

Photo by Stefan Els