Experimenting with ML/AI in Science Operations

Francisco Montenegro-Montes, Paulina Venegas & Claudio Agurto

ESO Chile - Atacama PathFinder Experiment (APEX)

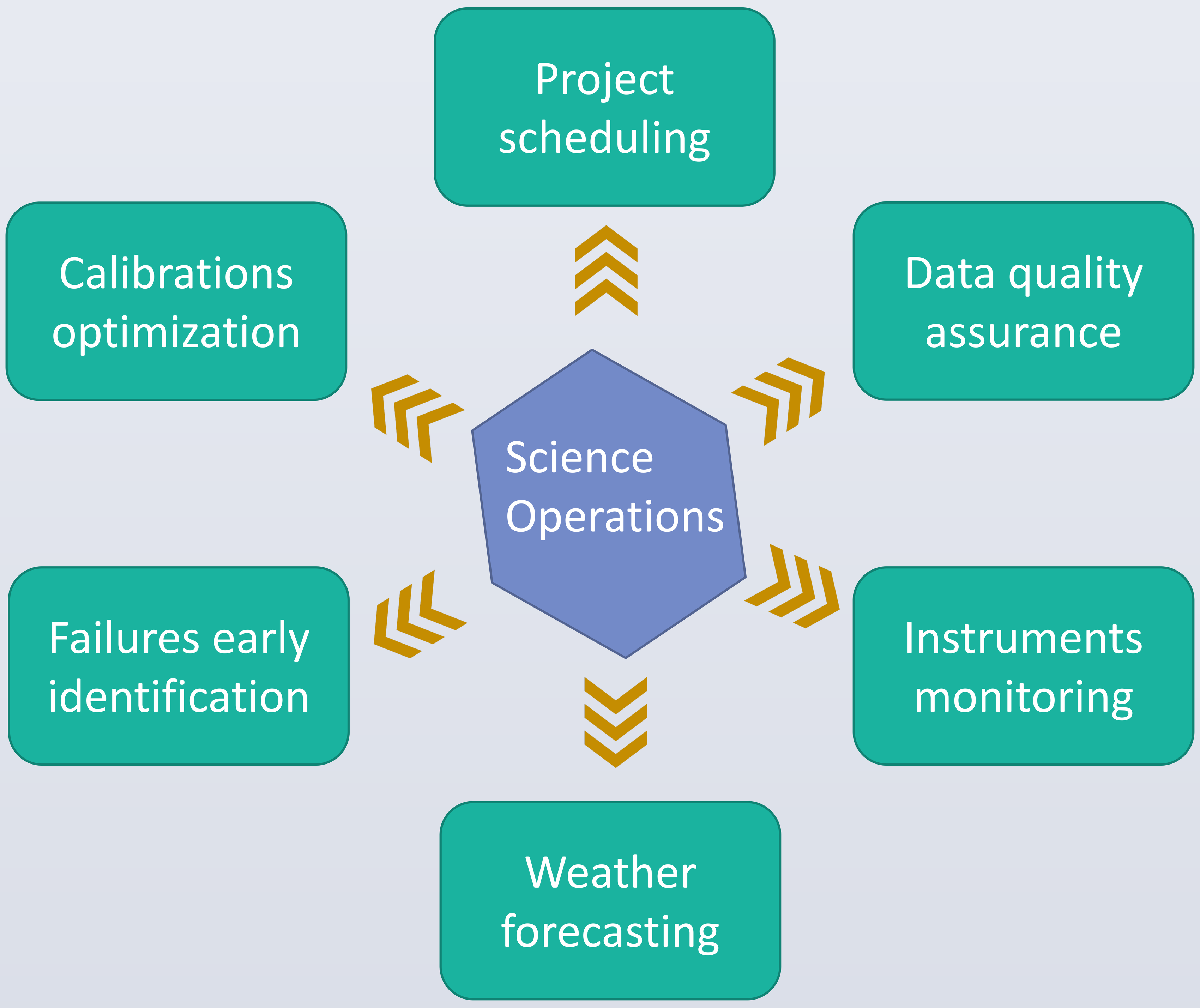
MOTIVATION

Operational efficiency and reliability are two performance indicators that modern astronomical observatories try to maximize. These depend on multiple factors like readiness of equipment (telescope, instruments, auxiliary systems), program scheduling practices, the ratio of calibration/system health measurements versus the amount of time spent on the scientific targets, identification and troubleshooting of problems, etc. In many of these areas the decisions and actions taken by astronomers and operators are crucial, and they are based on an increasingly large amount of information and judgement. By nature, these decisions are subject to errors, personal impressions, inexperience, bad habits or misconceptions not always compatible with achieving the highest efficiency or reliability.

We believe Machine Learning (ML) and Artificial Intelligence (AI) techniques can help in this direction and therefore in our APEX Science Operations team we have started learning and experimenting with some ML/AI concepts. The final goal would be developing a small prototype system to assist astronomers and operators taking better decisions on specific operational aspects.

APEX Science Operations

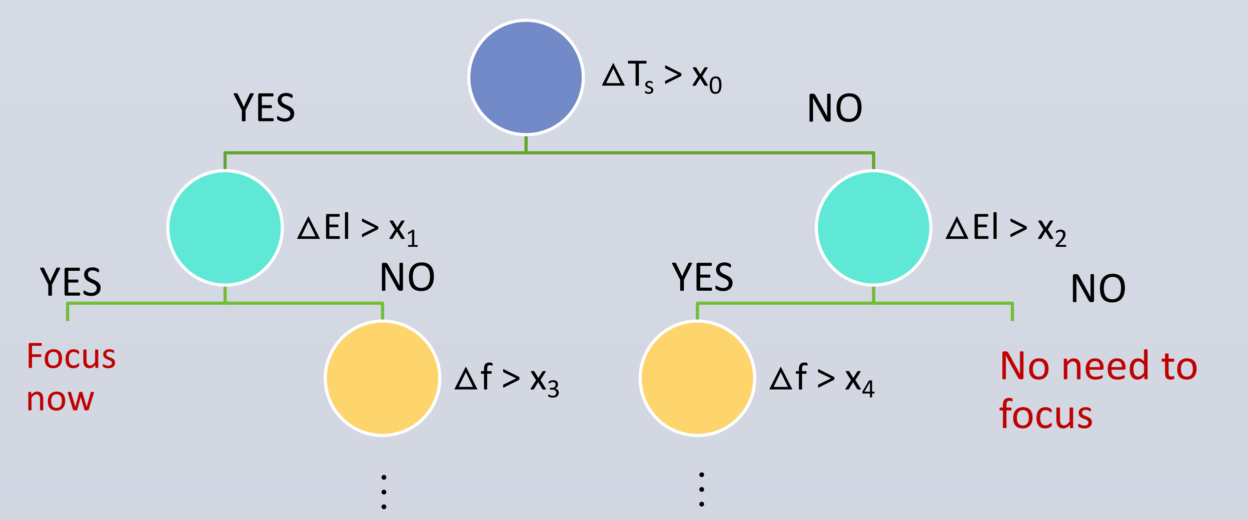
There are several areas in Science Operations where automatic learning and artificial intelligence can potentially be applied to improve the efficiency and the reliability of the observations:



We outline below some examples belonging to some of these areas:

Optimizing number of calibration scans: Decision trees can be easily implemented to decide whether a pointing or focus scan is needed at a certain moment, avoiding doing too much or too little. Features like tuning frequency changes (affecting the beam size), variations in sub-reflector temperature or in source elevation can be used to derive optimal decisions to maximize the efficiency.

Example of a Z-focus decision tree for SEPIA660 (beam size 9 arcsec)

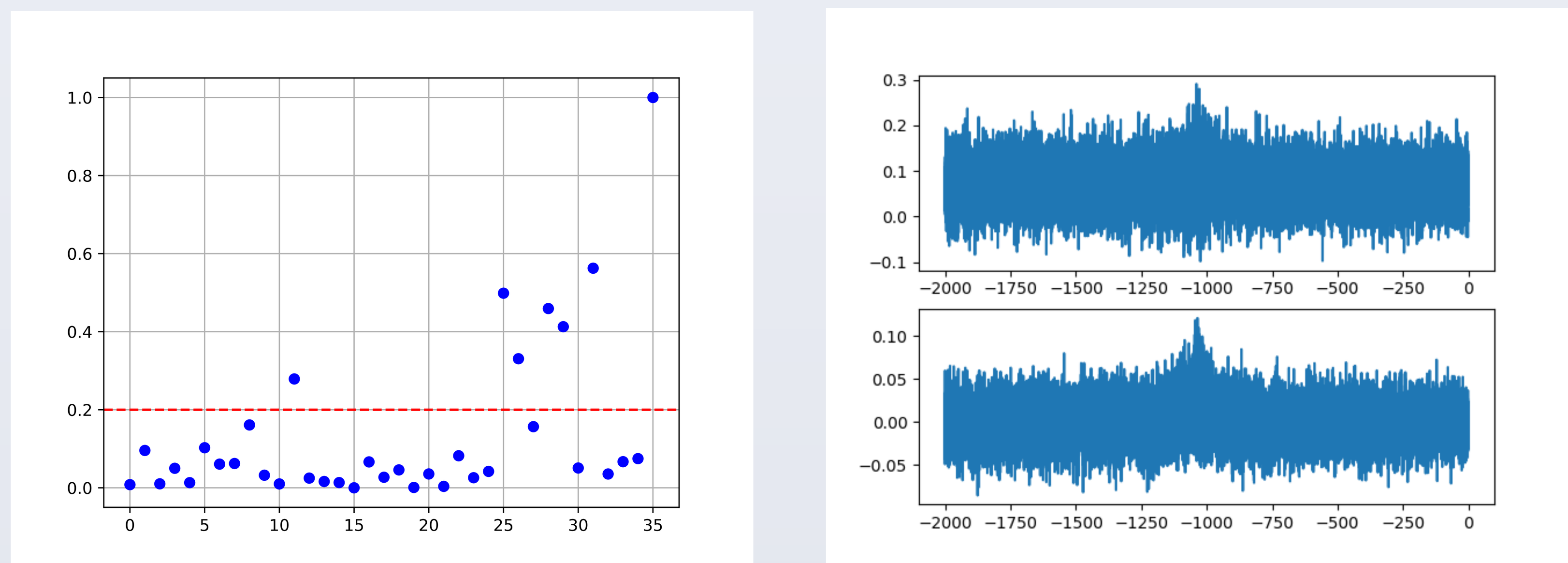


Thresholds x_0, x_1, x_2,… for all nodes can be adjusted to optimal values based on the running history of focus variation measurements together with the record of the corresponding features. Similar trees can be designed for pointing scans based on features like positional variations. These can also be adjusted with the running quality of the pointing model in place.

Optimal project scheduling: This is a complex task depending on multiple variables like: scientific ranking by the TAC, current weather conditions and short-term forecast, probability of successful project completion in a certain time, LST pressure, partner share quota, availability and performance of relevant instrumentation, amount of overheads required, etc. Tree-based models are probably better to assist in this kind of problem, since neural networks tend to work better in homogeneous datasets when all features have similar meanings.

Data quality assurance: Various quality assurance tasks can be automated for a better efficiency. One could be, for instance, assessing the current status of the pointing and focus performances based on the latest measurements and on an extrapolated model from historical measurements.

Also, outlier detection algorithms (unsupervised) can be used to detect sub-scans with anomalies (spikes, unstable baselines, wrong calibration). We are experimenting with the Local Outlier Factor algorithm (`scikit-learn`) to detect outliers in a series of sub-spectra taken on the same source/line. Careful preparation of the input dataset is needed before running the model to avoid meaningless outlier detections. A threshold needs to be defined to separate "outliers" from normal cases.



Example of anomaly detection among spectral sub-scans. Scores higher than 0.2 are classified as outliers. On the right, averaged anomalies (top) and normal cases (bottom) are shown, suggesting a slightly worse definition of the astronomical line in cases.

Failure identification and instrument monitoring: The monitoring system implemented at APEX sounds alarms triggered by values exceeding certain allowed ranges (min-max), normally pointing to issues, sometimes critical (loss of communication, power failure, etc). The full system used in Sciops-R (remote APEX operations, Klein et al. 2018) monitors almost 2000 parameters at rates from 1 second to minutes. Regression algorithms can likely be used to identify the various parameters associated to a certain issue, with the aim of hopefully allowing its earlier detection.

Web front-end of the APEX engineering Sciops-R monitoring system

Another technique that could lead to early failure identification is the unsupervised classification of messages contained in the APEX Control System (APECS, Muders et al. 2006) log files, by using natural language processing-based (NLP) algorithms. Some of the obtained classes could be identified with hardware or software failures.

On this front, we are experimenting with the Weka software (Waikato environment for Knowledge analysis) but we plan to try out also well-established NLP Python packages such as `nltk`, `spacy` or `gensim`.

References

- D. Muders, H. Hafok, et al. 2006, "APECS: the APEX control system", A&A 454, L25
- T. Klein, F.M. Montenegro-Montes et al. 2018, "APEX beyond 2016: the evolution of an experiment into an efficient and productive Submillimeter Wavelength Observatory", SPIE Vol. 10704, 107041V, p16.