# The Winning Solution for the Mars Spectrometry: Detect Evidence for Past Habitability Challenge

Dmytro Poplavskiy

[redacted]

## Abstract

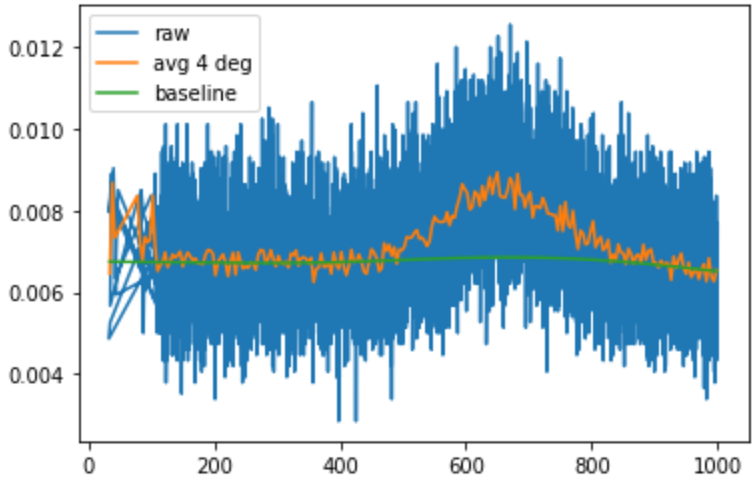
The solution is based on the ensemble of diverse CNN, RNN and transformer-based deep models over 2d representations (m/z over temperature) of the mass spectrometry data. Data augmentation is applied during training and the simple ensemble is used, averaging the results of the different architecture models, trained on the differently preprocessed data.

## Data Preprocessing

The mass spectrometry data used for training and prediction is restricted to the most common subset: temperature range of 28-1000 degrees and m/z range of 0-100.

### SAM testbed preprocessing

The next step is applied only to the SAM testbed data: the data looks much noisier, with a larger number of samples, so the SAM testbed samples are averaged over the 4 degrees bins:



All the next steps are common for both commercial and SAM testbed samples.

### Background subtraction

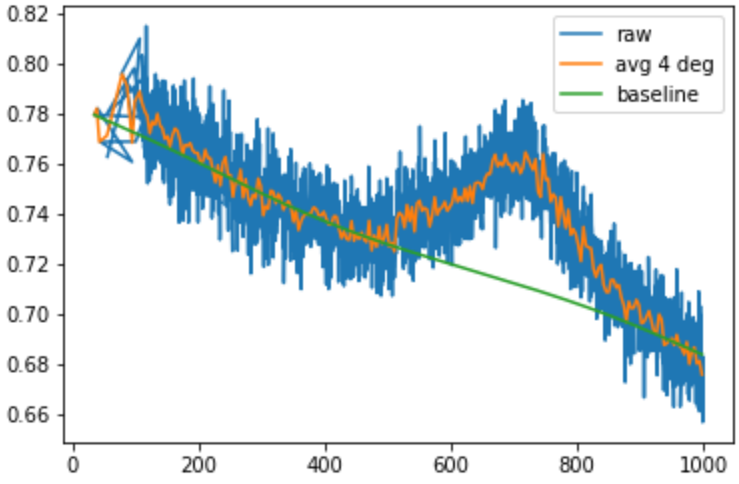
Background baseline is estimated using the "Adaptive smoothness penalized least squares smoothing (asPLS)"[1] method from the pybaselines package: pybaselines.whittaker.aspls.

Models are trained with either the minimum value subtracted as the background or with the estimated baseline subtracted or trained on both. Instead of subtracting the absolute minimum value, I used the random combination of 0.1%, 5%, 10% and 20% percentiles as data augmentation.  When comparing the performance on cross-validation, I have not seen a

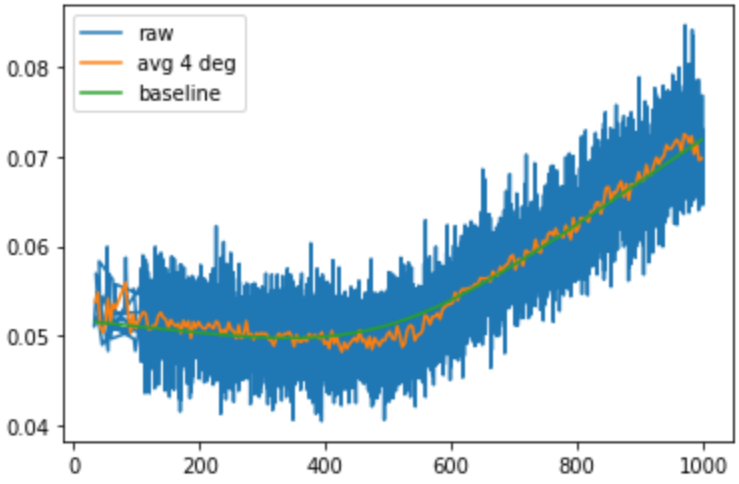
[1] Zhang, F., et al. Baseline correction for infrared spectra using adaptive smoothness parameter penalized least squares method. Spectroscopy Letters, 2020, 53(3), 222-233.

significant difference between just subtracting the minimum value and using a more complex estimated baseline, but I kept both preprocessing approaches for models diversity.

The adaptive background subtraction seems to work well on some samples, like:



But may potentially produce an incorrect result on other samples:



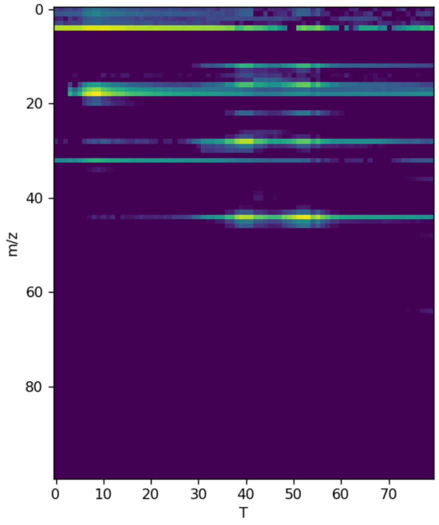
If it’s practical to collect more training samples, I’d suggest using only the simple minimum/percentile background subtraction and applying the possible effect of the SAM testbed background values as data augmentations for commercial samples.

## 2D representations

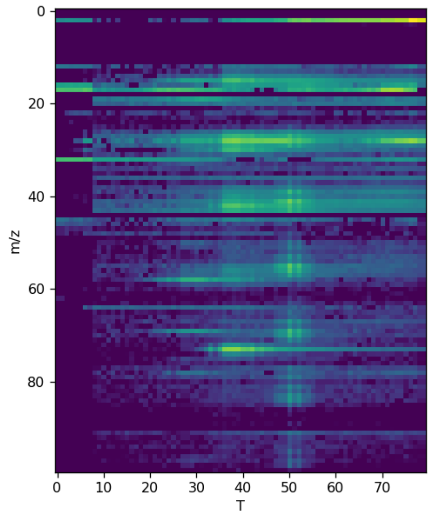
Unlike the baseline solution, with and abundances values averaged over 100℃ bands, I decided to use the smaller 12℃ step with only one abundance value randomly sampled from the overlapping 32℃ bands. This allows to better describe the peak shape while the weighted random sampling acts like an extra augmentation.

The value is normalized to 1 (He value is reduced 0.1x before), clamped to 1e-4 - 1.0 range and converted to the log space in the -4..0 range.

Example of 2D mass spectrogram from the commercial sample:



2D spectrogram representation from the SAM testbed sample:



## Data augmentation

To avoid overfitting to the relatively small dataset, the following augmentations have been applied during training and prediction:

- Instead of the minimum background, the random combination of 0.1%, 5%, 10% and 20% percentiles is subtracted with and without background baseline estimation.
- Random query of the abundance values from the overlapping temperature bands.
- Every m/z plot is randomly scaled by $2^{\text{random\_normal(0 mean, sigma)}}$ with 1.0 or 0.5 sigma values.
- For some models, instead of normalizing data by maximum value over all measurements, m/z plots are normalized to 1 individually. Surprisingly, the models performance was similar to the maximum scaling. So some models in the ensemble were trained with the max normalization and some with per m/z band normalization.
- Mixing two samples. 2D representations of two samples are mixed together, either as an average or maximum value with the label combined as maximum. Samples with zero labels are excluded from the mixing since after being scaled they may add too much noise to the combined non-empty label sample.

## Models

All models used the 80x100 2D mass spectrogram as input and produced 10 logits for individual labels. The performance for all listed models has been comparable.

- CNN models, pre-trained on imagenet. Some models used the spectrogram plus encoded value for the temperature and m/z, another only the spectrogram and relied on zero-padding information leak for position encoding.
    - Resnet34
    - SeResnext50
    - DPN68
    - EfficientNet B2
- RNN based models, with all m/z values at a certain temperature combined to the single sample and sequence dimension applied to the temperature. The temperature value is added as one of the inputs.
    - LSTM
    - GRU
- Visual Transformer. Instead of 16x16 patches of the original visual transformer model, the 8x1 patch of 8 temperature readings at every m/z value is used. The abundance value is converted to one-hot encoding.
- Visual Transformer with all m/z values combined to the single patch, in a similar way to RNN models. The temperature value is added as one of the inputs.

All the models have been trained on 4 folds for cross-validation, with some models trained with the different data pre-processing and augmentation parameters. Most of the models are very lightweight with a small size input of 80x100 values, so training usually takes a few hours per model.

For the next submissions I have also used lightgbm and catboost based models, the single model performance was similar or slightly better but the results between different GBM models were much more correlated compared to the neural network based models, so improvement from ensembling was lower.

## Ensembling approach

The ensembling approach is very simple - to average predictions of a number of diverse models, trained on 4 different folds with different data pre-processing and augmentation configurations. Since the data preparation pipeline involves random data sampling and augmentations, the same test time augmentations have been applied over 16 samples.

The most improvement was for ensembling the first few models, ensembling more than  8 best models started to slightly degrade the score.

| Ensemble size, models (TTA 16) | Model added | Out of fold validation score,ensemble | Out of fold validation score,single model |
|---|---|---|---|
| 1 | 100_vis_tr_16_3_512_v0.5_ps8 | 0.1277 | 0.1277 |
| 2 | 145_vis_trans_1d_norm_m_sbg5 | 0.1054 | 0.1295 |
| 3 | 172_cls_enet_b2_mix0.25_sbg5 | 0.0996 | 0.1284 |
| 4 | 140_vis_trans_1d | 0.0976 | 0.1435 |
| 5 | 111_cls3_seresnext50 | 0.0963 | 0.1217 |
| 6 | 144_vis_trans_1d_norm_m | 0.0955 | 0.1445 |
| 7 | 143_vis_trans_1d_v1_mix0.5 | 0.0951 | 0.1313 |
| 8 | 163_cls_resnet34_sbg5 | 0.0950 | 0.1342 |

| | | | |
|---|---|---|---|
| 9 | 141_vis_trans_1d_v1 | 0.0950 | 0.1368 |
| 10 | 162_cls3_resnet34_mix0.25_clip3 | 0.0950 | 0.1515 |
| 11 | 120_lstm_1024_3_v0.5 | 0.0951 | 0.1759 |
| 12 | 150_dpn68b_v1 | 0.0951 | 0.1262 |
| 13 | 113_cls3_seresnext50_sbg5_norm_m | 0.0952 | 0.1286 |
| 14 | 130_gru | 0.0954 | 0.1542 |
| 15 | 167_cls_resnet34_norm_m | 0.0955 | 0.1383 |
| 16 | 151_dpn68b_v1_mix0.25 | 0.0957 | 0.1318 |
| 17 | 160_cls3_resnet34 | 0.0959 | 0.1456 |
| 18 | 131_gru_mix0.25 | 0.0960 | 0.1422 |
| 19 | 164_cls_resnet34 | 0.0962 | 0.1291 |

Model name suffixes: *norm_m* - normalize every m/z band independently, *mix0.25* - 25% of samples augmented with the mix of two samples, *sbg5* - 50% of samples used min background subtraction and 50% - estimated baseline, v1, v0.5 - sigma of random bands scale augmentations.

With a larger number of models ensembled, the impact of multiple samples TTA becomes negligible but still positive:

| Test time augmentations, samples | Best ensemble score | Full ensemble score | Best single model score |
|---|---|---|---|
| 1 | 0.0978 | 0.0997 | 0.1588 |
| 2 | 0.0954 | 0.0984 | 0.1512 |
| 4 | 0.0963 | 0.0986 | 0.1419 |
| 8 | 0.0949 | 0.0979 | 0.1231 |
| 16 | 0.0950 | 0.0978 | 0.1277 |

## Data postprocessing

The only post processing was to scale the 0 .. 1 output range to smaller clip .. 1.0-clip range, to avoid a large penalty for the very confident but incorrect predictions by logloss metrics.

During the cross validation, the impact of such post processing was almost negligible - improvement from 0.0982 to 0.0978 for the clip value of 0.002.

For the SAM testbed samples the clip value was 0.01, due to unknown and likely much worse performance of the model on SAM samples (even while on the 12 validation samples the performance was comparable to the commercial machine samples).

Next submissions used even bigger clip values to be on more conservative side at the cost of the lower score, with more models added. The submissions scored slightly lower.