Distributed Machine Learning for Natural Hazard Applications Using PERMON

M. Pecha[1,2], Zachary Langford[3], David Horák[1,2], Richard T. Mills[4]

June 7, 2023

2023 Annual PETSc Meeting, Chicago

[1]Department of Applied mathematics, FEECS, VŠB –Technical University of Ostrava
[2]Institute of Geonics, Czech Academy of Sciences
[3]Oak Ridge National Laboratory, Tennessee, USA
[4]Argonne National Laboratory, Illinois, USA

Outline

- Maximal-margin classifier (SVM)
- Model calibration (Platt scaling)
- Data processing
- Wildfires localization in the Alaska region
- Summary

Maximal-margin classifier (SVM)

Let $\boldsymbol{X}$ be a matrix of features associated with samples and $\boldsymbol{y}$ be a vector of labels:

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{bmatrix}, \quad \boldsymbol{y} = \begin{bmatrix} +1 \\ -1 \\ \vdots \\ +1 \end{bmatrix}.$$

We look for a hyperplane

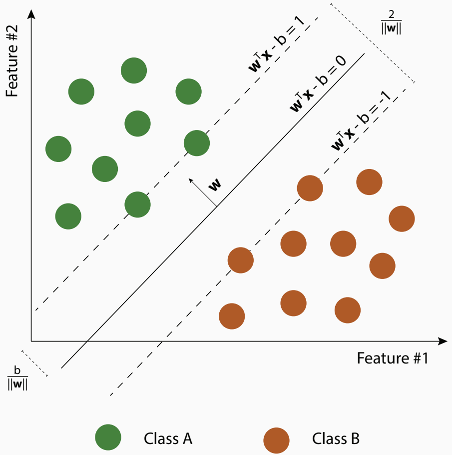
$$H : \langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0, \tag{1}$$

such that

$$\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b \geq +1 \ \ldots \ (\text{Class A}),$$

and

$$\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b \leq -1 \ \ldots \ (\text{Class B}).$$

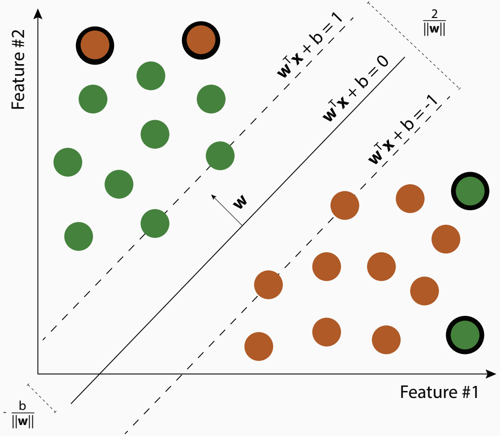


Real world data are not linearly separable!

We introduce a miclassification error term (hinge loss function) for each sample $\boldsymbol{x}_i$ such that:

$$\xi_i = \max\{0, 1 - y_i\left(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle - b\right)\}. \tag{2}$$

This function quantifies error between predicted and right classification of sample $\boldsymbol{x}_i$ as **distance between hyperplane and misclassified sample**.



Relaxed-bias classifier

The standard soft-margin SVM solves a problem of finding a classification model in the form of the maximal-margin hyperplane; the dual formulation of the primal $\ell 1$-loss SVM takes a following form:

$$\arg\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\boldsymbol{\alpha}^T \underbrace{\mathbf{Y}^T\mathbf{K}\mathbf{Y}}_{=:\mathbf{H}}\boldsymbol{\alpha} - \boldsymbol{\alpha}^T\boldsymbol{e} \ \text{ s.t. } \begin{cases} \boldsymbol{o} \le \boldsymbol{\alpha} \le C\boldsymbol{e}, \\ \boldsymbol{y}^T\boldsymbol{\alpha} = 0. \end{cases} \tag{3}$$

In the case of the relaxed-bias classification, we do not consider bias b in a classification model, but we include it into the problem by means of augmenting the vector $\boldsymbol{w}$ and each sample sample $\boldsymbol{x}_i$ with an additional dimension so that:

$$\widehat{\boldsymbol{w}} \leftarrow \begin{bmatrix} \widehat{\boldsymbol{w}} \\ B \end{bmatrix}, \quad \widehat{\boldsymbol{x}}_{\boldsymbol{i}} \leftarrow \begin{bmatrix} \boldsymbol{x}_i \\ \gamma \end{bmatrix}, \tag{4}$$

where $b \in \mathbb{R}$, and $\gamma \in \mathbb{R}^+$ is a user defined variable, which is typically set to 1. In a fact, we consider the bias B as a user-defined parameter (similar to the Deep Neural Networks).

Relaxed-bias classifier

Let $p \in \{1,2\}$ for purposes related to our application, then the problem of finding hyperplane $\widehat{H} = \langle \widehat{\boldsymbol{w}}, \widehat{\boldsymbol{x}} \rangle$ can be formulated as a constrained optimization problem in the following primal formulation:

$$\underset{\widehat{\boldsymbol{w}},\ \xi_i}{\arg\min}\ \frac{1}{2}\langle \widehat{\boldsymbol{w}}, \widehat{\boldsymbol{w}} \rangle \ + \ \frac{C}{p}\sum_{i=1}^{n} \widehat{\xi}_i^p \text{ s.t. } \begin{cases} y_i \langle \widehat{\boldsymbol{w}}, \widehat{\boldsymbol{x}}_i \rangle \geq 1 - \widehat{\xi}_i, \\ \widehat{\xi}_i \geq 0 \text{ if } p = 1,\ i \in \{1, 2, \ldots, n\}. \end{cases} \tag{5}$$

For both $p = 1$ and $p = 2$, we can dualize the primal formulation (5) using the Lagrange duality so that:

$$\underset{\boldsymbol{\alpha}}{\arg\min}\ \frac{1}{2}\boldsymbol{\alpha}^T \boldsymbol{H}\boldsymbol{\alpha} - \boldsymbol{\alpha}^T \boldsymbol{e} \text{ s.t. } \boldsymbol{o} \leq \boldsymbol{\alpha} \leq C\boldsymbol{e}, \tag{6}$$

$$\underset{\boldsymbol{\alpha}}{\arg\min}\ \frac{1}{2}\boldsymbol{\alpha}^T \left(\boldsymbol{H} + C^{-1}\boldsymbol{I}\right)\boldsymbol{\alpha} - \boldsymbol{\alpha}^T \boldsymbol{e} \text{ s.t. } \boldsymbol{o} \leq \boldsymbol{\alpha}, \tag{7}$$

respectively.

Model calibration (Platt scaling)

An approximation of a posterior probability using a parametric form of a sigmoidal function such that:

$$P(y = 1 \mid \boldsymbol{x}) \approx P_{A,B}(y = 1 \mid \boldsymbol{x}) = \frac{1}{1 + \mathrm{e}^{Ah_\theta(\boldsymbol{x}) \ + \ B}}, \qquad (8)$$

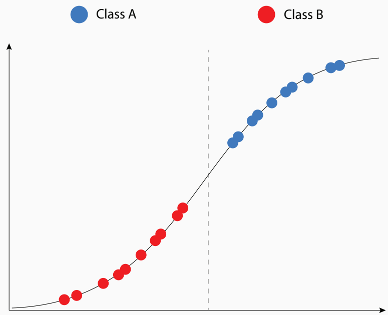
where $h_\theta(\boldsymbol{x}) = \langle \widehat{\boldsymbol{w}}, \widehat{\boldsymbol{x}} \rangle$ is a relaxed SVM model.

The parameters are determined by means of minimizing a binary cross-entropy so that:

$$\underset{A,B}{\arg\min} - \sum_{j=1}^{l} t_j \ln p_j + (1 - t_j) \ln(1 - p_j), \qquad (9)$$

where $p_j = P_{A,B}(y = 1 \mid \boldsymbol{x}_j)$, and t_j is a target probability associated with the sample $\boldsymbol{x}_j$:

$$t_j = \begin{cases} \frac{N_p+1}{N_p+2} & \dots \ y = +1, \\ \frac{1}{N_n+2} & \dots \ y = -1. \end{cases} \qquad (10)$$



This is not the QP problem!

For solving underlying unconstrained optimization, NLS implemented in TAO is directly used.

Wildfires localization

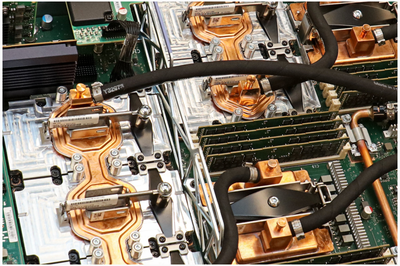
The 2004 fire season in Alaska and western Canada.

Sources downloaded from nasa.gov and nbcnews.com.

Data processing (ALASKA, 2004)

- Sources were downloaded from Google Earth Engine (multispectral MODIS images and corresponding labels)
- The time series data was converted into a 7-dimensional time series.
- The dimensions represent the spectral Bands (red, blue, green and NIR and $3\times$ SWIR) collected from January to December.
- Not observed pixels are removed from data set.
- Additional feature engineering such as standardization or PCA was processed.

Facility for training Models [Summit IBM AC922 system at ORNL]



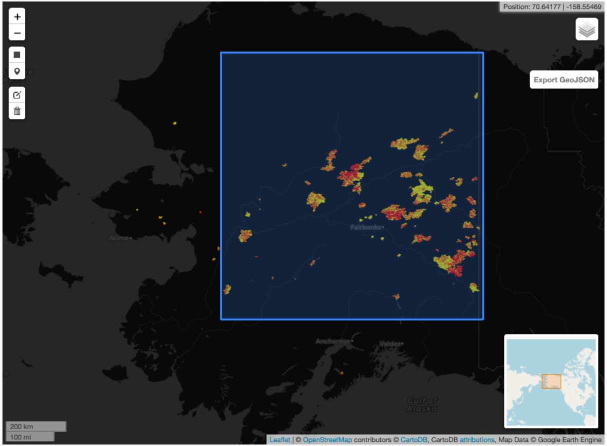
Summit System totals

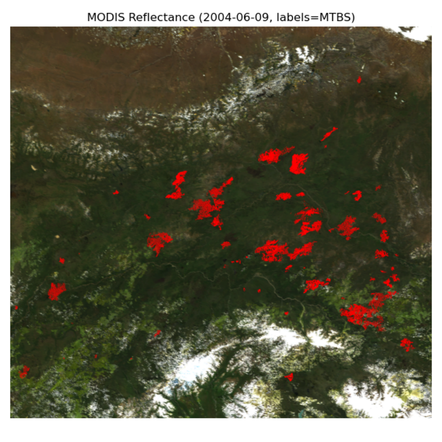
- $\sim$ 200 PFlop/s theoretical peak
  143 PFlop/s LINPACK—#5 in TOP500
- 4,608 compute nodes

Node configuration

- Compute:
  - Two IBM Power9 CPUs, each 22 with cores, 0.5 DP TFlop/s
  - Six NVIDIA Volta V100 GPUs, each with 80 SMs–32 FP64 cores/SM, 7.8 DP TFlop/s
- Memory:
  - 512 GB DDR4 memory
  - 96 (6×16) GB high-bandwidth GPU memory
  - 1.6 TB nonvolatile RAM (I/O burst buffer)

Wildfires localization

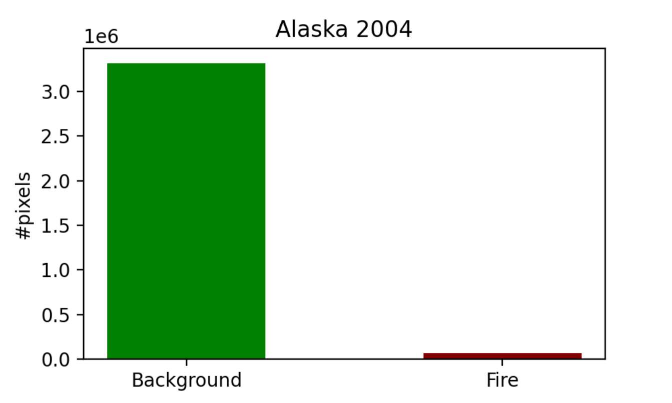




A center of area N65° 44′ 55.259″ E149° 53′ 50.859″, **area** $\approx 722,500\text{km}^2$, **projection** EPSG3338

multispectral images collection MOD09A1, **image size** $\underbrace{1918 \times 1780}_{\text{space domain}} \times \underbrace{(46 \times 7)}_{\text{time domain}}$ px

Wildfires localization: data processing pipelines with feature engineering



Highly unbalanced data set **3,317,870** (97.92%) of background pixels and **70,631** (2.08%) of wildfire pixels. A data set was **shuffled** and split into training and test data set (ratio 3 : 1). Time series length 1 year (46 time steps).

Wildfire localization: data processing pipelines with feature engineering (a non-sparse model)

Computation using $\ell 2$-loss failed on model perfomance scores! Feature selection required!

Tool	Transformation	#features	Sen.	Prec.	F1	Training time [s]
PermonSVM	z-score* (23.23s)	7×46 (322)	0.03	0.97	0.08	$2.39^\dagger$
	PCA* (83.40s)	7×27 (189)	0.03	0.97	0.07	$2.33^\dagger$

Table 1: Solver: MPGP, an expansion step is performed using projected CG step, $\Gamma = 1$ in a proportion criterion. Penalty $C = 0.01$ and a loss type is set to ℓ2-loss. rtol $= 0.1$ **Double precision**.

- **Since feature vectors related to pixels are entirely dense, we use a dense format for distributed matrices, i.e. MATMPIDENSECUDA in PETSc.**
- PCA latent factors were determined by means of a cumulative sum of explainable variances related to factors at 95% confidence level.

Symbols:
† 6x NVidia Volta V100 *Sequential run on an one CPU core (i7 SB, 32GB RAM DDR3, Debian).

It works much better employing a feature selection approach!

Tool	Transformation	#features	Sen.	Prec.	F1	Training time [s]
PermonSVM	z-score* (23.23s)	7×46 (322)	0.79	0.80	0.80	58.03†
XGBoost			0.83	0.83	0.83	8662.10*
PermonSVM	PCA* (83.40s)	7×27 (189)	0.78	0.75	0.77	20.33†
XGBoost			0.85	0.74	0.79	4266.96*

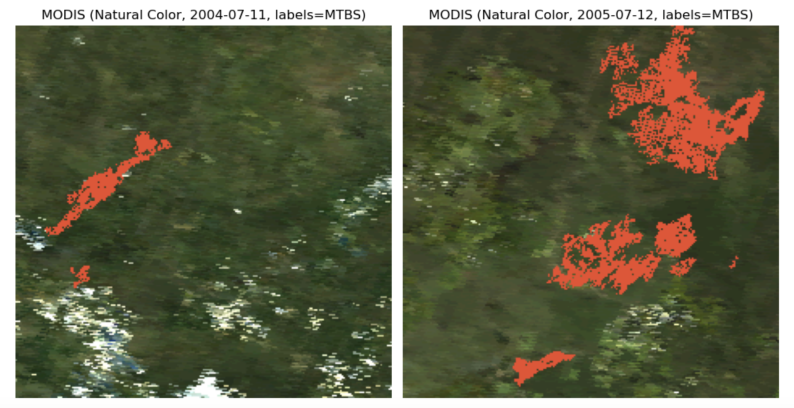
Table 2: Solver: MPGP, an expansion step is performed using projected CG step, $\Gamma = 1$ in a proportion criterion. Penalty $C = 0.01$ and a loss type is set to ℓ1-loss. rtol $= 0.1$ **Double precision**.

- **Since feature vectors related to pixels are entirely dense, we use a dense format for distributed matrices, i.e. MATMPIDENSECUDA in PETSc.**
- PCA latent factors were determined by means of a cumulative sum of explainable variances related to factors at 95% confidence level.

Symbols:

† 6x NVidia Volta V100 $\quad$ *Sequential run on an one CPU core (i7 SB, 32GB RAM DDR3, Debian).

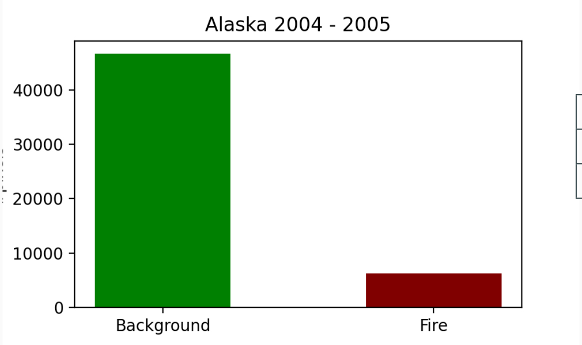
Wildfire localization: ALASKA 2004–2005



A center of area N67° 21′ 54.875″ E142° 40′ 6.4459″, **area** $\approx 13,450\text{km}^2$, **projection** EPSG3338

multispectral images collection MOD09A1, **image size** $\underbrace{231 \times 233}_{\text{space domain}} \times \underbrace{(92 \times (7 \text{ or } 8))}_{\text{time domain}}$ px

Wildfire localization: ALASKA 2004–2005 (data processing)



Data set	#background pixs.	#fire pixs.
Training	29,444	5,585
Test	17,223	717

Unbalanced data set **46,667** (88.10%) of background pixels and **6,302** (11,90%) of wildfire pixels. An image was split horizontally into training and test data set (ratio 2 : 1). Time series length equals 2 years (92 time points).

Wildfire localization: ALASKA 2004–2005, REFLECTANCE (calibrated SVM model)

Tool	Transformation	#features	Sen.	Prec.	$F1$
PermonSVM*	z-score	7×92 (644)	0.92	0.86	0.89
XGBoost			0.91	0.82	0.86
PermonSVM*	PCA	7×61 (427)	0.86	0.88	0.87
XGBoost			0.91	0.81	0.86

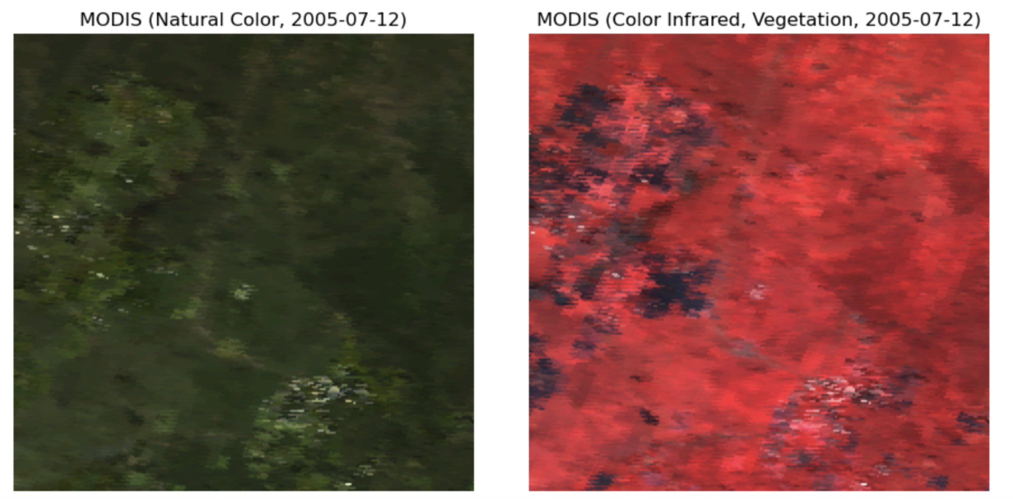
Table 3: Solver: MPGP, an expansion step is performed using projected CG step, $\Gamma = 10$ in a proportion criterion. Penalty $C = 0.01$ and a loss type is set to ℓ1-loss. rtol $= 0.1$ **Double precision**.

- **Since feature vectors related to pixels are entirely dense, we use a dense format for matrices, i.e. MATSEQDENSE in PETSc (A SEQUENTIAL RUN ON A LAPTOP)**
- PCA latent factors were determined by means of a cumulative sum of explainable variances related to factors at 99% confidence level.

Symbols:

*a decision threshold was set to 0.4 $\quad ^\dagger$ default parameter settings (not run hyper-parameter searching)

Wildfire localization: vegetation in color IR (infra red)



Wildfire localization: ALASKA 2004–2005, REFLECTANCE+EVI (calibrated SVM model)

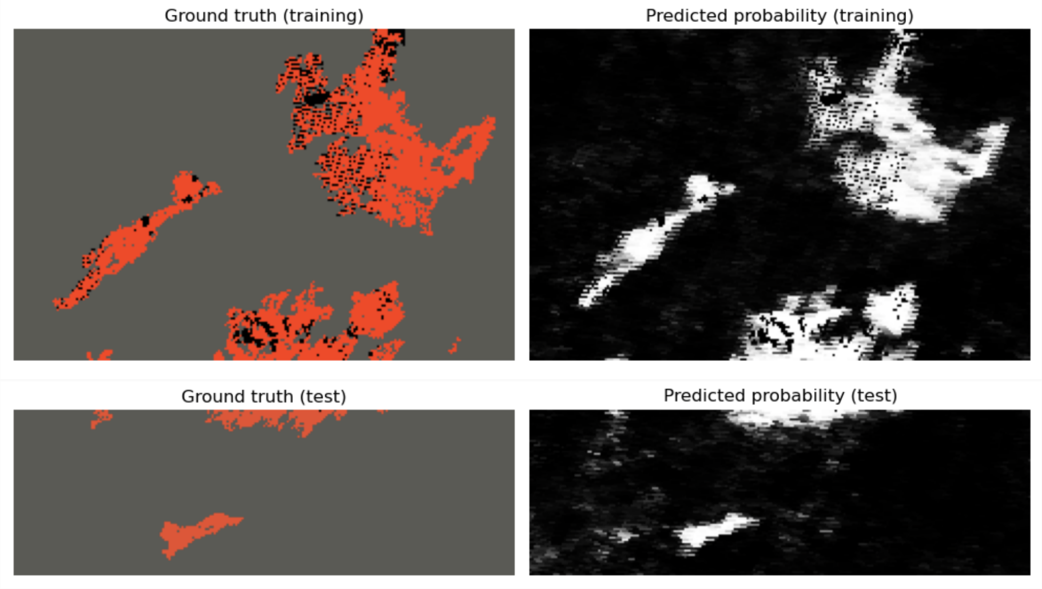
Tool	Transformation	#features	Sen.	Prec.	$F1$
PermonSVM*	z-score	8×92 (644)	0.87	0.88	0.88
XGBoost†			0.92	0.82	0.86
PermonSVM*	PCA	8×61 (488)	0.90	0.85	0.87
XGBoost†			0.92	0.79	0.84

Table 4: Solver: MPGP, an expansion step is performed using projected CG step, $\Gamma = 10$ in a proportion criterion. Penalty $C = 0.01$ and a loss type is set to $\ell 1$-loss. rtol $= 0.1$ **Double precision**.

- **Since feature vectors related to pixels are entirely dense, we use a dense format for matrices, i.e. MATSEQDENSE in PETSc (A SEQUENTIAL RUN ON A LAPTOP).**
- PCA latent factors were determined by means of a cumulative sum of explainable variances related to factors at 99% confidence level.

Symbols:
*a decision threshold was set to 0.4 $\quad ^\dagger$ default parameter settings (not run hyper-parameter searching)

Wildfire localization: a posterior probability (calibrated SVM model)



Summary

- SVM models obtained using PermonSVM show good performance for wildfire localization with MODIS data comparable with the Boosted Trees approach (XGBoost).
- Communication efficiency should be improved if we can use a non-buggy implementation of GPU-aware MPI.
- Focus on solving standard SVM model formulation, i.e. without relaxed bias, and batch processing.
- Experiments with other feature extraction such as a visual dictionary or feature extraction using the VGG16/VGG19/RESNET backbone.
- Increasing a model complexity using a hybrid approach, e.g. calibrated SVM could be used as a last classification layer in the UNet type network.
- Tools for processing MODIS data will be available soon on `https://github.com/natural-hazards/wildfires`.

Thank you for your kind attention. Any questions?

Interested? Please visit us on

`permon.vsb.cz`, `github.com/permon`

This work was supported by Project of the Czech Academy of Sciences – Strategy AV21

R. T. Mills was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two US Department of Energy organizations (Office of Science and the National Nuclear Security Administration). This research used resources of the Oak Ridge Leadership Computing Facility, a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.