ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414669

# OPTIMUM FEATURE ORDERING FOR DYNAMIC INSTANCE–WISE JOINT FEATURE SELECTION AND CLASSIFICATION

*Yasitha Warahena Liyanage*    *Daphney–Stavroula Zois*

Electrical and Computer Engineering Department
University at Albany, SUNY, Albany, NY, USA
Emails: {yliyanage, dzois}@albany.edu

## ABSTRACT

We introduce a supervised machine learning framework to perform joint feature selection and classification individually for each data instance during testing. In contrast to our prior work, we decide both the order and the number of features for each data instance. Specifically, our proposed solution dynamically selects the feature to review at each stage based on the already observed features and stops the selection process to make a prediction once it determines no classification improvement can be achieved. To gain insights, we analyze the properties of the proposed solution. Based on these properties, we propose a fast algorithm and demonstrate its effectiveness compared to the state–of–the–art using 4 publicly available datasets.

***Index Terms***— instance–wise feature selection, feature ordering, classification, dynamic programming

## 1. INTRODUCTION

In many real world applications (e.g., medical diagnosis, disaster prediction), features are not freely available to acquire, while accurate, time–sensitive and interpretable decisions are needed. For instance, consider the case where a doctor aims to diagnose a patient (classification decision) as quickly as possible by conducting the minimum number of tests (features) due to both the cost of the tests and the time sensitivity of the decision. Note that a *different* set of tests may be appropriate for *each* individual patient (data instance). For example, relevant features for predicting heart failure may differ across patient subgroups [1]. At the same time, the order by which test are conducted for each patient does not only affect the cost, but also the classification decision [2].

In our prior work [3, 4], we studied the problem of instance–wise dynamic feature selection and classification for independent/correlated features. The goal was to determine the number of features needed to classify an instance and the classification strategy when the order by which the features are reviewed is fixed and common for all instances. Herein, we remove this restrictive assumption and consider the more general problem of determining also the order by which features must be reviewed. Thus, we derive the optimum feature ordering, the number of features and the classification strategy that needs to be adopted for each data instance individually when features sequentially arrive one at a time during testing. We also analyze the theoretical properties of this optimum solution. Specifically, we show that the corresponding functions are continuous, concave, and piece–wise linear on the domain of a sufficient statistic, and use these properties to derive an efficient implementation. In our experiments, we observe that both the order by which our framework reviews features and the number of features used for each test instance are varying. In addition, less number of features on average is needed to achieve comparable performance with the state–of–the–art.

Next, we briefly summarize the most relevant work. Traditional feature selection methods [5, 6] assume all features are available during training, while streaming feature selection methods [7–9] are designed to handle features arriving sequentially during model training. Both such methods suffer from a key limitation; the features discovered during model training and used during testing are the *same* for all test instances. Instance–wise feature selection methods [10–12], instead, identify a subset of relevant features that explains/predicts the output of a machine learning model individually for each test instance, but, must first reveal all feature assignments and do not scale for large feature spaces. Similar to our method, classification with costly features methods [13, 14] reveal features one at a time and make a prediction based only on the observed features for each test instance. However, these methods define the problem globally to limit the number of features used on average for all data instances (i.e., do not optimize instance–wise), and do not scale for large feature spaces.

## 2. PROBLEM DESCRIPTION

We consider a standard supervised machine learning setting, where a data instance $s \in \mathcal{S}$ is represented using the realiza-

This material is based upon work supported by the National Science Foundation under Grants ECCS–1737443 & CNS–1942330. We would like to thank Dr. Charalampos Chelmis for his suggestions on this work.

tion $f$ of a set $F$ of $K$ features, i.e., $\{F = f\} \triangleq \{F_1 = f_1, \ldots, F_K = f_K\}$. Each data instance $s$ belongs to one of $L$ classes with associated prior probability $P(\mathcal{C} = c_i) = p_i$ for each assignment $c_i, i \in \{1, \ldots, L\}$, of the class variable $\mathcal{C}$. Furthermore, the cost of acquiring feature $F_k$ is denoted by $e(F_k) > 0, k \in \{1, \ldots, K\}$, while $Q_{ij} \geqslant 0$ represents the cost of selecting class $c_j$ when the true class is $c_i$, $i, j \in \{1, \ldots, L\}$. Based on the above, our goal is to assign each data instance $s$ to one out of $L$ possible classes. At the same time, we wish to select the most informative features to use to reach such a classification decision, under the constraint that features arrive sequentially one at a time during testing. In this context, at each time, we have to decide between: i) **stopping** and optimally classifying the current data instance based on the reviewed features, or ii) **continue** and selecting the next feature from the remaining set of available features. Next, we introduce three random variables, $\sigma, \sigma(R), D_{\sigma(R)}$ that will help us describe our proposed approach. Specifically, variables $\sigma$ and $\sigma(R)$ represent the feature selection strategy, since they denote the order by which features are selected and the feature at which the sequential selection process stops, respectively. For instance, if $K = 3$, then $\sigma = (F_3, F_1, F_2)$ denotes a valid feature ordering, while $\sigma(R = 2) = F_1$ indicates that our framework stops after acquiring the second feature $F_1$ in the ordering $\sigma$. On the other hand, variable $D_{\sigma(R)}$ represents the classification strategy, which depends both on $\sigma$ and $\sigma(R)$. For instance, if $K = 3, L = 2$ and $\sigma = (F_3, F_1, F_2)$, then the event $\{D_{\sigma(R=2)} = 1\}$ represents deciding in favor of class $c_1$ based on the feature assignments $\{f_3, f_1\}$.

In order to find the optimum ordering $\sigma^*$, the optimum stopping feature $\sigma^*(R^*)$ and the optimum classification strategy $D^*_{\sigma^*(R^*)}$ that can lead to an accurate classification decision for each data instance $s \in \mathcal{S}$, we propose to solve the following optimization function:

$$\underset{\sigma, \sigma(R), D_{\sigma(R)}}{\text{minimize}} \; J\big(\sigma, \sigma(R), D_{\sigma(R)}\big), \tag{1}$$

where $J\big(\sigma, \sigma(R), D_{\sigma(R)}\big) = \mathbb{E}\Big\{ \sum_{k=1}^{R} e\big(F_{\sigma(k)}\big) \Big\} + \sum_{j=1}^{L} \sum_{i=1}^{L} Q_{ij} P\big(D_{\sigma(R)} = j, \mathcal{C} = c_i\big)$. The first term represents the cost of reviewing features, and the second term penalizes the misclassification cost of decision $D_{\sigma(R)}$.

## 3. SOLUTION

To solve the optimization problem of Eq. (1), we begin by simplifying the probability $P\big(D_{\sigma(R)} = j, \mathcal{C}_i\big)$. Specifically, since $x_R = \sum_{k=0}^{K} x_k \mathbb{1}_{\{R=k\}}$ for any sequence of random variables $\{x_k\}$, where $\mathbb{1}_A$ is the indicator function for event $A$, the probability $P\big(D_{\sigma(R)} = j, \mathcal{C} = c_i\big)$ takes the form:

$$P\big(D_{\sigma(R)} = j, \mathcal{C} = c_i\big) = \mathbb{E}\left\{ \pi^i_{\sigma(R)} \mathbb{1}_{\{D_{\sigma(R)} = j\}} \right\}. \tag{2}$$

The term $\pi^i_{\sigma(k)} \triangleq P\big(\mathcal{C} = c_i | F_{\sigma(1)}, \ldots, F_{\sigma(k)}\big)$ denotes the probability of the data instance under examination belonging to class $c_i$ given the accumulated information until $k$th feature for ordering $\sigma$. We observe that using Eq. (2), the optimization function in Eq. (1) can be rewritten as follows:

$$\begin{aligned} J\big(\sigma, \sigma(R), D_{\sigma(R)}\big) = \mathbb{E}\Bigg\{ &\sum_{k=1}^{R} e\big(F_{\sigma(k)}\big) \\ &+ \sum_{j=1}^{L} Q_j^T \pi_{\sigma(R)} \mathbb{1}_{\{D_{\sigma(R)} = j\}} \Bigg\}, \end{aligned} \tag{3}$$

where $Q_j \triangleq [Q_{1,j}, Q_{2,j}, \ldots, Q_{L,j}]^T$ and $\pi_{\sigma(k)} \triangleq [\pi^1_{\sigma(k)}, \pi^2_{\sigma(k)}, \ldots, \pi^L_{\sigma(k)}]^T$. We can recursively update the posterior probability vector $\pi_{\sigma(k)} \in [0,1]^L$ via Bayes' rule as follows:

$$\pi_{\sigma(k)} = \frac{\operatorname{diag}\Big(\Delta\big(F_{\sigma(k)} | F_{\sigma(1)}, \ldots, F_{\sigma(k-1)}, \mathcal{C}\big)\Big) \pi_{\sigma(k-1)}}{\Delta^T(F_{\sigma(k)} | F_{\sigma(1)}, \ldots, F_{\sigma(k-1)}, \mathcal{C}) \pi_{\sigma(k-1)}}, \tag{4}$$

where $\Delta\big(F_{\sigma(k)} | F_{\sigma(1)}, \ldots, F_{\sigma(k-1)}, \mathcal{C}\big) \triangleq [P(F_{\sigma(k)} | F_{\sigma(1)}, \ldots, F_{\sigma(k-1)}, c_1), \ldots, P(F_{\sigma(k)} | F_{\sigma(1)}, \ldots, F_{\sigma(k-1)}, c_L)]^T$, $\operatorname{diag}(A)$ represents a diagonal matrix with elements of vector $A$, and $\pi_{\sigma(0)} \triangleq [p_1, p_2, \ldots, p_L]^T$.

To obtain the optimum classification strategy $D^*_{\sigma(R)}$ for any stopping feature $\sigma(R)$ and ordering $\sigma$, we search for a lower bound for the last term in Eq. (3). Specifically, we note that $\sum_{j=1}^{L} Q_j^T \pi_{\sigma(R)} \mathbb{1}_{\{D_{\sigma(R)} = j\}} \geqslant g\big(\pi_{\sigma(R)}\big)$, where $g\big(\pi_{\sigma(R)}\big) \triangleq \min_{1 \leqslant j \leqslant L} \big[Q_j^T \pi_{\sigma(R)}\big]$ for any stopping feature $\sigma(R)$ and ordering $\sigma$. Thus, the optimum classification strategy is:

$$D^*_{\sigma(R)} = \arg \min_{1 \leqslant j \leqslant L} \big[Q_j^T \pi_{\sigma(R)}\big]. \tag{5}$$

Since $J\big(\sigma, \sigma(R), D_{\sigma(R)}\big) \geqslant J\big(\sigma, \sigma(R), D^*_{\sigma(R)}\big)$, where $J\big(\sigma, \sigma(R), D^*_{\sigma(R)}\big) = \min_{D_{\sigma(R)}} J\big(\sigma, \sigma(R), D_{\sigma(R)}\big)$, Eq. (3) can be rewritten to depend only on feature ordering $\sigma$ and the stopping feature $\sigma(R)$ as follows:

$$\widetilde{J}\big(\sigma, \sigma(R)\big) = \mathbb{E}\left\{ \sum_{k=1}^{R} e\big(F_{\sigma(k)}\big) + g\big(\pi_{\sigma(R)}\big) \right\}. \tag{6}$$

Next, we solve the optimization problem $\min_{\sigma, \sigma(R)} \widetilde{J}\big(\sigma, \sigma(R)\big)$ to obtain both the optimum ordering $\sigma^*$ and the optimum stopping feature $\sigma^*(R^*)$ simultaneously. At stage $k$, let $N_k = \{F_{\gamma_1}, \ldots, F_{\gamma_k}\}, \gamma_k \in \{1, \ldots, K\}, \forall k$, be the $k$ features reviewed thus far, and $Z_k = F - N_k$ be the remaining set of features. Theorem 1 summarizes the optimum solution found by *dynamic programming* [15].

**Theorem 1.** *For $k = K-1, \ldots, 0$, the function $\widehat{J}_k(\pi_{\gamma_k})$ is related to $\widehat{J}_{k+1}(\pi_{\gamma_{k+1}})$ through the equation:*

$$\begin{aligned} &\widehat{J}_k(\pi_{\gamma_k}) = \min\big[g(\pi_{\gamma_k}), \widehat{\mathcal{A}}_k(\pi_{\gamma_k})\big] \textit{ with} \\ &\widehat{J}_K(\pi_{\gamma_K}) = g(\pi_{\gamma_K}), \end{aligned} \tag{7}$$

*where* $\widehat{\mathcal{A}}_k(\pi_{\gamma_k}) \triangleq \min_{F_{k+1} \in Z_k} \Big[e(F_{k+1}) + \sum_{F_{k+1}} \Delta^T(F_{k+1}$ $|F_{\gamma_1}, \ldots, F_{\gamma_k}, \mathcal{C})\pi_{\gamma_k} \widehat{J}_{k+1}\left(\frac{\operatorname{diag}\left(\Delta(F_{k+1}|F_{\gamma_1},\ldots,F_{\gamma_k},\mathcal{C})\right)\pi_{\gamma_k}}{\Delta^T(F_{k+1}|F_{\gamma_1},\ldots,F_{\gamma_k},\mathcal{C})\pi_{\gamma_k}}\right)\Big]$.

Therefore, the optimum feature ordering is $\sigma^* \triangleq (F_{\gamma_1}, \ldots, F_{\gamma_K})$, while the optimum stopping feature $\sigma^*(R^*)$ is equal to the first $k < K$ feature for which $g(\pi_{\gamma_k}) \leqslant \widehat{\mathcal{A}}_k(\pi_{\gamma_k})$, or $\sigma^*(R^* = K)$ if there are no more features to be reviewed. Equivalently, we stop the feature review process at stage $k$, if the cost of stopping $g(\pi_{\gamma_k})$ is no greater than the minimum expected cost of continuing $\widehat{\mathcal{A}}_k(\pi_{\gamma_k})$ given all information accumulated up to stage $k$. In Theorem 1, the optimum feature $F_{\gamma_{k+1}}$ is obtained from the set $Z_k$ of remaining features given $\pi_{\gamma_k}$ such that the total cost until termination is minimized. Hence, the optimum feature ordering $\sigma^* = (F_{\gamma_1}, \ldots, F_{\gamma_K})$ derived from Theorem 1 is not a global ordering common for all data instances, but rather *varies* based on the assigned feature values $\{f_{\gamma_1}, \ldots, f_{\gamma_K}\}$ for each data instance $s \in \mathcal{S}$ (see Section 6 for a demonstration).

## 4. THEORETICAL RESULTS

We discuss some important properties of the optimum solution in Section 3. Consider a general form of the function $g(\pi_{\gamma_k})$, used to derive the optimum classification strategy in Eq. (5), given by $g(\varpi) \triangleq \min_{1 \leqslant j \leqslant L}\left[Q_j^T \varpi\right], \varpi \in [0,1]^L$, where $\varpi \triangleq [\omega_1, \ldots, \omega_L]^T$, such that $\omega_i \in [0,1], \sum_{i=1}^L \omega_i = 1$. The domain of $g(\varpi)$ is the probability space of $\varpi$, which is an $L-1$ dimensional unit simplex. Function $g(\varpi)$ has some interesting properties as described in Lemma 1.

**Lemma 1.** *Function $g(\varpi)$ is continuous, concave, and piecewise linear, and consists of at most L hyperplanes represented by the set $\{Q_j^T\}_{j=1}^L$ of L vectors. Each hyperplane denotes a unique classification decision, while the optimum classification strategy $D^*(\varpi) = \arg\min_{1 \leqslant j \leqslant L}\left[Q_j^T \varpi\right]$.*

Next, we consider the general form of the function $\widehat{\mathcal{A}}_k(\pi_{\gamma_k})$ in Eq. (7) given by:

$$\widehat{\mathcal{A}}_k(\varpi) = \min_{F_{k+1} \in Z_k}\Bigg[e(F_{k+1}) + \sum_{F_{k+1}} \Delta^T(F_{k+1}|F_{\gamma_1}, \ldots, F_{\gamma_k}, \mathcal{C}) \times \varpi \widehat{J}_{k+1}\left(\frac{\operatorname{diag}\left(\Delta(F_{k+1}|F_{\gamma_1},\ldots,F_{\gamma_k},\mathcal{C})\right)\varpi}{\Delta^T(F_{k+1}|F_{\gamma_1},\ldots,F_{\gamma_k},\mathcal{C})\varpi}\right)\Bigg]. \quad (8)$$

Lemma 2 summarizes the key properties of this function.

**Lemma 2.** *The functions $\widehat{\mathcal{A}}_k(\varpi), k = 0, \ldots, K-1$, are continuous, concave, and piecewise linear. In particular, we can write $\widehat{\mathcal{A}}_k(\varpi) \triangleq \min_{F_{k+1} \in Z_k}\left[\beta_k^{F_{k+1}} \varpi\right]$, where $F_{\gamma_{k+1}} \triangleq \arg\min_{F_{k+1} \in Z_k}\left[\beta_k^{F_{k+1}} \varpi\right]$, and $\beta_k^{F_{k+1}} \in \mathbb{R}^{1 \times L}$.*

The properties of functions $g(\varpi)$ and $\widehat{\mathcal{A}}_k(\varpi)$ stated in Lemmas 1 and 2, respectively, allow for a parsimonious representation of the function related to the optimum feature selection strategy in Eq. (7), as shown in Theorem 2.

**Theorem 2.** *At every stage $k \in \{0, \ldots, K\}$, there exists a finite set $\{\alpha_k^i\}, \alpha_k^i \in \mathbb{R}^{1 \times L}$, of vectors such that $\widehat{J}_k(\varpi) = \min_i[\alpha_k^i \varpi]$, where $\{\alpha_k^i\} \triangleq \left\{\left\{\beta_k^{F_{\gamma_{k+1}}}\right\} \cup \{Q_j^T\}_{j=1}^L\right\}, k \in \{0, \ldots, K-1\}$ with $\{\alpha_K^i\} \triangleq \{Q_j^T\}_{j=1}^L$.*

Note that each vector $\alpha_k^i$ defines a region in the probability space of $\varpi$ for which this vector maximizes the function $\widehat{J}_k(\varpi)$. These regions form a partition of the probability space induced by the piecewise linearity of $\widehat{J}_k(\varpi)$. Hence, the finite set $\{\alpha_k^i\}$ can be used as a compact representation of $\widehat{J}_k(\varpi)$, instead of computing $\widehat{J}_k(\varpi)$ at every realization of $\varpi$ in the probability space. This property can be used to derive an efficient algorithmic implementation of the optimum solution as described in Section 5.

## 5. IFCO ALGORITHM

Lemma 1 and Theorem 2 allow for an efficient implementation of the optimum solution described by Eqs. (5) and (7). Specifically, the decision to stop or continue the feature review process depends only on $\alpha_k^* = \arg\min_{\alpha_k^i}[\alpha_k^i \varpi]$, such that if $\alpha_k^* \in \{Q_j^T\}_{j=1}^L$, we stop the feature review process, else if $\alpha_k^* \in \left\{\beta_k^{F_{\gamma_{k+1}}}\right\}$, we continue with feature $F_{\gamma_{k+1}}$. This is based on the fact that $\widehat{J}_k(\varpi) = \min\left[g(\varpi), \widehat{\mathcal{A}}_k(\varpi)\right]$. Namely, if $\alpha_k^* \in \{Q_j^T\}_{j=1}^L$, Lemma 1 implies that $g(\varpi) \leqslant \widehat{\mathcal{A}}_k(\varpi)$. On the other hand, if $\alpha_k^* \in \left\{\beta_k^{F_{\gamma_{k+1}}}\right\}$, Lemma 2 implies that $g(\varpi) > \widehat{\mathcal{A}}_k(\varpi)$. Based on the above, we present IFCO, an algorithm for Instance–wise Feature selection and Classification with optimum feature Ordering. Initially, $\varpi$ is set to $\pi_0$, and $\alpha_0^* = \arg\min_{\alpha_0^i}[\alpha_0^i \varpi]$ is computed. If $\alpha_0^* \in \{Q_j^T\}_{j=1}^L$, IFCO classifies the instance under examination, else if $\alpha_0^* \in \left\{\beta_k^{F_{\gamma_1}}\right\}$, feature $F_{\gamma_1}$ is reviewed. IFCO repeats these steps until either it decides to classify the instance using $< K$ features, or exhausts all features. The input vector sets $\{\alpha_k^i\}$ can be computed using a standard point–based value iteration algorithm [16] during training. For simplicity, we opted for the Perseus algorithm [17], among the several point–based value iteration algorithms in the literature [18]. Specifically, we fed a fixed number $\zeta$ (e.g., ∼1000 [17]) of reachable $\varpi$ vectors from each stage, conditional probability distributions $P(F_k|\mathcal{C}_i)$, misclassification costs $Q_{ij}$ and feature review costs $e(F_k)$ into the Perseus algorithm to obtain $\{\alpha_k^i\}, k \in \{0, \ldots, K-1\}$.

## 6. EXPERIMENTS

In this section, we assess the performance of IFCO on 3 DNA microarray datasets [19], MLL (72 instances / $5,848$ features / 3 classes), Lung2 (203 instances / $3,312$ features / 2 classes), Car (174 instances / $9,182$ features / 11 classes),

**Table 1**. Performance comparison with baselines. "Acc", and "Feat" stands for accuracy, and the average number of features per data instance, respectively. "All" represents using all features.

| Method | MLL | | Spambase | | Lung2 | | Car | |
|---|---|---|---|---|---|---|---|---|
| | Acc | Feat | Acc | Feat | Acc | Feat | Acc | Feat |
| IFCO | **1.00** | **3.20** | **0.813** | **3.01** | **0.887** | **3.94** | **0.857** | **8.64** |
| MB [4] | **1.00** | **4.88** | 0.741 | 3.08 | 0.842 | 3.96 | 0.539 | 5.63 |
| ASSESS [3] | **1.00** | **5.07** | 0.847 | 7.47 | 0.882 | 15.6 | 0.810 | 12.91 |
| OFS-Density [7] | 0.960 | 11.0 | 0.787 | 7.60 | **0.912** | **16.2** | 0.597 | 6.80 |
| SAOLA [8] | 0.867 | 28.0 | 0.824 | 24.6 | 0.882 | 28.2 | 0.798 | 41.4 |
| OSFS [9] | 0.800 | 3.00 | 0.801 | 33.8 | 0.847 | 5.80 | 0.556 | 5.20 |
| FAST–OSFS [9] | 0.800 | 5.00 | 0.801 | 33.8 | 0.842 | 9.40 | 0.608 | 8.40 |
| Lasso | **1.00** | **4.00** | 0.902 | 29.6 | 0.685 | 9.40 | 0.551 | 28.8 |
| Tree [5] | 0.933 | 100 | 0.947 | 18.2 | 0.897 | 207 | 0.752 | 429 |
| PCA | 0.667 | 36.0 | 0.693 | 1.00 | 0.897 | 88.4 | 0.391 | 91.0 |
| SVM–G | **1.00** | **All** | 0.834 | All | 0.788 | All | 0.563 | All |
| R–Forest | **1.00** | **All** | 0.940 | All | 0.911 | All | 0.758 | All |
| XG–Boosting | 0.733 | All | **0.955** | **All** | 0.906 | All | **0.844** | **All** |

and an email dataset [20], Spambase ($4,601$ instances / 57 features / 2 classes). For MLL, we use the originally provided training and validation sets, while for Spambase, Lung2, and Car, we report five–fold cross validated results.

We use a smoothed maximum likelihood estimator to estimate $p(F_k|\mathcal{C}_i), k = 1, \dots, K, i = 1, \dots, N$, after quantizing the feature space. Specifically, $\hat{p}(F_k|\mathcal{C}_i) = \frac{S_{k,i}+1}{S_i+V}$, where $S_{k,i}$ denotes the number of instances that satisfy $F_k = f_k$ and belong to class $\mathcal{C}_i$, $S_i$ denotes the total number of instances belonging to class $\mathcal{C}_i$, and $V$ is the number of bins considered. We estimate the prior probabilities as $P(\mathcal{C}_i) = \frac{S_i}{\sum_{i=1}^{N} S_i}, i = 1, \dots, N$. In our experiments, $Q_{ij} = 1, \forall i \neq j, Q_{ii} = 0, i, j \in \{1, \dots, L\}$, $V = 4$, and feature cost $e(F_k) = 0.01, \forall k$.

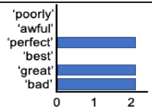
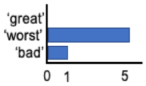
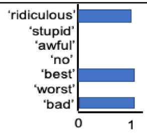
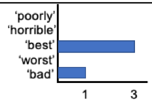
We compare the performance of IFCO to (i) 4 online feature selection methods: OFS–Density [7], SAOLA [8], OSFS [9], FAST–OSFS [9], (ii) 2 instance–wise joint feature selection and classification methods: ASSESS [3], MB [4], (iii) 3 offline feature selection and dimentionality reduction methods: L1–norm based feature selection (Lasso), Tree–based feature selection (Tree) [5], Principal Component Analysis with SVM classifier (PCA), and (iv) 3 state–of–the–art classifiers: Support Vector Machines with Gaussian kernel (SVM–G), Random Forest (R–Forest), XG Boosting (XG–Boosting). We summarize the results in Table 1 next. IFCO achieves $100\%$ accuracy using just 3.2 features on average on MLL dataset. MB, ASSESS, Lasso, SVM–G and R–Forest achieve the same accuracy, but use between $25\%$ and $1.8 \times 10^5\%$ more features compared to IFCO. For Spambase, XG–Boosting achieves the highest accuracy with 19 times more features compared to IFCO for a difference of $14\%$ in accuracy. For Lung2, OFS–Density achieves the highest accuracy (i.e., $2.8\%$ better than IFCO), but requires 4 times more features. Finally, in the Car dataset, IFCO achieves the highest accuracy using 8.6 features on average. This is an improvement of $1.3\%$ in accuracy with $\sim$ 1000 times less features compared to XG–Boosting, which achieves the highest accuracy among the baselines.

In Table 2, we demonstrate IFCO using 4 examples from the IMDB movie reviews dataset ($50,000$ instances / $89,523$ features / 2 classes) [21]. We use the original provided training and validation sets with bag–of–words features. Note that IFCO selects *different features* (*order*, *number*) for *different data instances* in a dynamic setting and predicts the class label based on the observed features.

## 7. CONCLUSION

We presented an instance–wise dynamic joint feature selection and classification framework that selects both the order and the number of features for each data instance individually. We showed that the functions related to the optimum solution are continuous, concave and piece–wise linear on the domain of a sufficient statistic. Using these properties, we proposed IFCO, and validated its effectiveness compared to prior work using real–world datasets. In our future work, we plan to extend this work to regression settings.

**Table 2**. Words (features) picked by IFCO are highlighted in yellow. The true/predicted label is given at the end of each review. The second column reports features selected in ascending order (Y–axis) versus feature value (X–axis).

| IMDB Review Text **(True Label, Predicted Label)** | |
|---|---|
| I had read up on the film . . . I wasn't expecting anything great, figured it would be mostly fluff but hopefully not a totally bad experience. I have to admit I was pleasantly surprised. The dialogue was pitch perfect, most of the actors were exceptionally good and it flowed nicely. Ash Christian was perfect, . . . Ashley Fink is gem, a great young character actress that hopefully will get more work. There are moments in the film that could have used some work, but all in all not a bad time at the cinema. . . . **(positive, positive)** |  |
| This movie is the worst thing ever created by humans. You think manos is the worst movie ever? It doesn't even come close to this garbage. I dont even know where to begin. The "russian" commander and the rebel chic are the worst "actors" ever to appear in a movie. . . . the goofiest rape scene ever filmed, and the worst acting ever put on film. This movie deserves to be more well known among bad movie fans. Definitely the worst movie ever made. **(negative, negative)** |  |
| Do-It-Yourself indie horror auteur Todd Sheets . . . and a trio of hottie sisters all have to do their best to survive this harrowing ordeal. . . . Let's not forget the ridiculous ending in which several of our survivors stumble across a few vials of flesh-eating bacteria to use on the shambling undead hordes. Sure, this flick is pure dreck, but it has a certain endearingly abominable quality to it that in turn makes it a great deal of so-awful-it's-awesome Grade Z fun for hardcore aficionados of bad fright fare. **(positive, negative)** |  |
| Tommy Lee Jones was the best Woodroe and no one can play Woodroe F. Call better than he. Not only was he the first and best, he was the only person that could portray his grief and confusion. It was a bad let-down and I'm surprised I even made myself watch it. . . . The first movie was the best and the only . . . **(negative, positive)** |  |

## 8. REFERENCES

[1] David P Kao, James D Lewsey, Inder S Anand, Barry M Massie, Michael R Zile, Peter E Carson, Robert S McKelvie, Michel Komajda, John JV McMurray, and JoAnn Lindenfeld, "Characterization of subgroups of heart failure patients with preserved ejection fraction with possible implications for prognosis and treatment response," *European Journal of Heart Failure*, vol. 17, no. 9, pp. 925–935, 2015.

[2] National Academies of Sciences Engineering, Medicine, et al., *Improving Diagnosis in Health Care*, National Academies Press, 2015.

[3] Yasitha Warahena Liyanage, Daphney-Stavroula Zois, Charalampos Chelmis, and Mengfan Yao, "Automating the classification of urban issue reports: an optimal stopping approach," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3137–3141.

[4] Yasitha Warahena Liyanage, Daphney-Stavroula Zois, and Charalampos Chelmis, "On–the–fly feature selection and classification with application to civic engagement platforms," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3762–3766.

[5] Pierre Geurts, Damien Ernst, and Louis Wehenkel, "Extremely randomized trees," *Machine Learning*, vol. 63, no. 1, pp. 3–42, 2006.

[6] Isabelle Guyon and André Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[7] Peng Zhou, Xuegang Hu, Peipei Li, and Xindong Wu, "Ofs-density: A novel online streaming feature selection method," *Pattern Recognition*, vol. 86, pp. 48–61, 2019.

[8] Kui Yu, Xindong Wu, Wei Ding, and Jian Pei, "Towards scalable and accurate online feature selection for big data," in *2014 IEEE International Conference on Data Mining*. IEEE, 2014, pp. 660–669.

[9] Xindong Wu, Kui Yu, Wei Ding, Hao Wang, and Xingquan Zhu, "Online feature selection with streaming features," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 5, pp. 1178–1192, 2012.

[10] Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan, "Learning to explain: An information-theoretic perspective on model interpretation," in *International Conference on Machine Learning*, 2018, pp. 883–892.

[11] Jinsung Yoon, James Jordon, and Mihaela van der Schaar, "INVASE: Instance-wise variable selection using neural networks," in *International Conference on Learning Representations*, 2018.

[12] Qi Xiao and Zhengdao Wang, "Mixture of deep neural networks for instancewise feature selection," in *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 2019, pp. 917–921.

[13] Gabriel Dulac-Arnold, Ludovic Denoyer, Philippe Preux, and Patrick Gallinari, "Datum-wise classification: a sequential approach to sparsity," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2011, pp. 375–390.

[14] Jaromír Janisch, Tomáš Pevnỳ, and Viliam Lisỳ, "Classification with costly features as a sequential decision-making problem," *Machine Learning*, pp. 1–29, 2020.

[15] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1, Athena Scientific, 2005.

[16] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[17] Matthijs TJ Spaan and Nikos Vlassis, "Perseus: Randomized point-based value iteration for pomdps," *Journal of Artificial Intelligence Research*, vol. 24, pp. 195–220, 2005.

[18] Guy Shani, Joelle Pineau, and Robert Kaplow, "A survey of point-based pomdp solvers," *Autonomous Agents and Multi-Agent Systems*, vol. 27, no. 1, pp. 1–51, 2013.

[19] Kun Yang, Zhipeng Cai, Jianzhong Li, and Guohui Lin, "A stable gene selection in microarray data analysis," *BMC Bioinformatics*, vol. 7, no. 1, pp. 228, 2006.

[20] Mark Hopkins, Erik Reeber, George Forman, and Jaap Suermondt, "Spambase Data Set," 1999, [Online]. Available: `http://archive.ics.uci.edu/ml/datasets/Spambase`.

[21] Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 142–150.