International Journal of Data Science and Analytics (2024) 17:443–460
https://doi.org/10.1007/s41060-023-00494-6

REGULAR PAPER

A stable model for maximizing the number of significant features

Eunkyung Park[1] · Raymond K. Wong[1] · Junbum Kwon[2] · Victor W. Chu[3]

Received: 23 August 2021 / Accepted: 7 December 2023 / Published online: 16 January 2024
© The Author(s) 2024

Abstract
Identifying significant features (SFs) is important because they are driving factors of a target outcome. However, it is difficult when they have much more features than observations. The problem becomes more challenging when there are multicollinearity and infrequent common features. In such case, standard explainable methods such as OLS and Lasso often fail to identify many SF. To tackle these problems, we propose a stable model to maximize the number of SFs using selective inference called SFLasso-SI. First, in each point in the regularization path, SFLasso-SI conducts selective inference for conservative significance test. Then, it chooses the optimum value of regularization that maximizes the number of SFs. Our extensive experiments across different types of data - text, image, and video show that our SFLasso-SI can find the biggest number of SFs while maintaining similar prediction accuracy as the benchmarking methods.

Keywords Explainable models · Statistical significance test · Lasso · Selective inference

1 Introduction

Machine learning has received massive attention due to its learning and prediction abilities. However, most recently, there is a growing demand for explanatory artificial intelligence (XAI) [1]. XAI provides users explanations of their decisions in some level of detail [2]. To meet the real-world demands, researchers could focus on explainability in different angles to improve their models. With the increase the ability of of users to understand the behavior from machine learning or data mining, there are many aspects and perspectives we can focus [3]. In our work, we want to examine one specific aspect: helping decision-makers to fully understand what factors drive the outcomes (target variables) in many practical applications. An example is that explainable models are used to explain the relationship between explanatory variables x and a target outcome y [4]. Differently, we focus on how to maximize the number of significant features, and use it as a measure of explainability.

For instance, advertising professionals want to know as many effective content features (e.g., visual objects, spoken words, and text descriptions) to explain ad popularity (e.g., number of likes from YouTube). These compelling features can become key inputs to come up with better content ideas for their next ads. The more significant content features, the more insight they get. In this paper, we aim to identify the maximum number of statistically significant content features that affect a target outcome (i.e., the number of likes in YouTube videos and Instagram pictures, the number of useful reviews).

In this paper, we propose SFLasso-SI to maximize the explainability of a target outcome by maximizing the number of significant features. Considering two opposing factors: model size (i.e., the number of active variables) and the correlations among the active variables, SFLasso-SI chooses a regularization parameter λ, generating the largest number of significant features via statistical significance test using selective inference [5, 6] in training data. On the other hand, Lasso chooses λ, which gives the highest prediction accuracy in validation data.

✉ Eunkyung Park
eunkyung.park@unsw.edu.au

Raymond K. Wong
ray.wong@unsw.edu.au

Junbum Kwon
junbum.kwon@unsw.edu.au

Victor W. Chu
wchu@ntu.edu.sg

[1] Computer Science and Engineering, University of New South Wales, Sydney, Australia

[2] School of Marketing, University of New South Wales, Sydney, Australia

[3] Computer Science and Engineering, Nanyang Technological University, Singapore, Singapore

Lately, there are renewed interests in feature selection to explain a target outcome in the machine learning community by making the second model after running Deep Neural Network (DNN) [7–10]. However, these recent approaches are not ideal for our datasets due to their small sample size. It is well-known that deep learning requires many observations to learn a model with acceptable accuracy.

Instead, we use statistical test to deliver our solution. A conservative test is required in feature selection because some features might be chosen by random chance in a small sample. However, most existing methods only rely on the magnitude of estimated coefficients. They choose features with large coefficients [11] or nonzero coefficients from Least Absolute Shrinkage and Selection Operator (Lasso) [12]. As an alternative method, Hastie et al. [13] suggests determining the statistical strength of the selected features using p-values. The estimated coefficients need to have a sizable dataset with small enough variance to survive in this statistical test. Unfortunately, recent post-hoc feature selection methods do not provide the framework for statistical tests using p values yet.

Lasso is a variable selection method by making some of the coefficients to be exactly zero. When we have more variables than observation, there is no unique least squares coefficients estimation which means the variance is infinite therefore, we cannot use it. To tackle this issue, we can constrain or shrink the estimated coefficients. We can significantly reduce the variance at the cost of a negligible increase in bias. This can result in substantial improvement in the accuracy [14]. Similarly Elastic-Net selects variables too. However, Elastic-net shrinks together the coefficients of correlated predictors and has considerable computational advantages [4].

Therefore, we employ explainable linear models in this paper. Ordinary least square (OLS), Lasso, and Elastic-net are commonly used to estimate linear models. However, they have difficulty in identifying significant features from certain types of data which we describe below. In all our content analysis, we find that benchmarking methods can only find a few or even nil significant content features. As such, they offer immaterial insight to data scientists in identifying significant content features.

Unfortunately, our datasets exhibit the following unfavorable properties making their analysis a challenging task:

1. **Multicollinearity**
   Small-n-large-l, which means that the number of features (l) is greater than the number of observations (n). As creating videos or pictures is expensive, the number of videos or pictures is limited. On the other hand, videos, pictures, and text have many content features. For example, World Vision US has only 529 YouTube videos, but those videos have 7,370 unique content features. In this scenario, explanatory variables exhibit high correlations. As a result, OLS does not identify many significant features well as multicollinearity increases the variances of the estimated coefficients.
2. **Infrequent common features across samples**
   Since the videos, images, and text have extensive contents, the content features vary substantially across different samples. When content heterogeneity is high, there are many infrequent content features. Let us use zero to represent a missing feature value. Some features have many zeros or even all zeros across ads in the validation set. We call them “unmatched" variables because they have nonzero values in the training set but all zero values in the validation set. For example, in Apple Instagram data, 83% of variables have all zeros in their validation set.

Using two datasets of YouTube videos, seven datasets of Instagram pictures, and one dataset of drug review, our proposed SFLasso-SI shows significantly improved results. It identifies much more significant features over all the ten datasets across video, image, and text content data while OLS, Lasso, and Elastic-net obtain much less or even nil significant features. By comparing with “true" appealing features in World Vision videos from human evaluators, we find that SFLasso-SI shows up to 25 times (US), 14 times (Canada) better accuracy than the best bench marking models and some other bench marking models obtain zero F1 scores. Importantly, the prediction error of SFLasso-SI is similar to those of Lasso and Elastic-net, although SFLasso-SI is not designed to maximize prediction accuracy unlike Lasso and Elastic-net.

In summary, SFLasso-SI can suggest much more significant content features while maintaining a similar level of prediction ability with Lasso and Elastic-net. Notably, our approach can be easily extended to other Lasso variants, which opens an opportunity for more extensive applications.

This paper is organized as follows. In Sect. 2, we provide existing work. The limitations of existing work are described in Sect. 3. Sections 4 and 5 presents our proposed model with the respective results of an extensive experimental evaluation. Finally, Sect 6 provides the main conclusion.

2 Related work

Our work is closely related to three lines of research: explainable models, Lasso, and significance test for Lasso.

2.1 Explainable models

Intrinsic explainable models such as linear models provide a straightforward relationship between input features and a target outcome based on coefficients’ magnitude.

In contrast, post-hoc explainability approach typically runs complex Deep Neural Networks (DNN) first for high prediction accuracy and then runs the second model to explain the first model's prediction by selecting input features.

For example, Deep Learning Important Features Through Propagating Activation Differences (DeepLIFT) [15] decomposes the output of DNN on each input For feature selection. Learning to Explain (L2X) [7] maximizes the mutual information between selected features and the response variable. Agglomerative Contextual Decomposition (ACD) [9] is a hierarchical algorithm for identifying clusters of features.

Given that our data have small observations, it would not be enough to train DNN. Instead, in our work, we extend Lasso, which is an intrinsic explainable model.

2.2 Lasso and its variants

Many of Lasso's [12] variants such as Elastic-net[16] have been proposed to improve prediction accuracy. Random Lasso [17] adopts bootstrapping to deal with high-dimensional data. Precision Lasso [18] selects more stable and consistent active variables by employing an additional hyperparameter (γ). Enumerate Lasso [19] enumerates solutions with different supports based on its objective values rather than finding a single optimal solution. Independently Interpretable Lasso (IILasso) [20] is a new regularization model by excluding correlated variables. In this model, each active variable is independently affecting target variable. There have been attempts to explain black-box models by incorporating Lasso. Neural Lasso [21] applies $L1$ and $L2$ penalties to their input gradient.

To summarize, most Lasso variants focus on improving prediction accuracy or finding more or better active variables. However, in our work, we model to find more significant features which has been drived from SFLasso [22].

2.3 Selective inference for Lasso

To test the significance of the estimated coefficients, OLS post-Lasso [23] proposed to rerun OLS with the active variables resulted from Lasso. However, this naive approach results in many false significant features [5, 6, 13]. Covariance Test [24] pioneered for significance test after variable selection when signal variables (having true nonzero coefficients) are not too correlated with noise variables (having true zero coefficients) [13]. Soon later, solutions for more general explanatory variables were proposed. For the linear regression model, Lee et al. [5] drive closed-form p values for selected active variables after fitting Lasso with a fixed value of hyperparameter λ. Taylor and Tibshirani [6] extend these results to generalized regression models such as logistic regression. In this paper, we incorporate selective inference in the framework of SFLasso to investigate further and do more traditional significance tests. While [25] reports the preliminary result of our initial test using very few datasets, in this paper, we have incorporated selective inference, a state-of-the-art statistical testing method for variable selection, in the framework of SFLasso to find more stable solutions, and they were not discussed in [25]. Also, this paper employs more diverse and extensive datasets (i.e., numeric, image, and video content) to generalize our approach for varieties of data. Furthermore, this paper makes a significant extension by using a more extended coordinate decent search along regularization paths for a comprehensive solution search. Finally, different from [25], this paper presents the detailed algorithms for integrating selective inference in SFLasso.

3 Intrinsic explainable linear models

In this section, we present why conventional linear methods fail in identifying significant features. This paper is to identify which content features explain their popularity significantly. While neural network methods are often used if the goal is prediction, linear models are employed if the goal is to explain the effect of explanatory variables $X \in \mathbb{R}^{n\times l}$ (e.g., visual objects, text) on a response variable $y \in \mathbb{R}^n$ (e.g., the number of likes of YouTube videos, Instagram pictures, and useful count of drug reviews) as follows.

$$y = X\beta + \epsilon$$

where $\beta \in \mathbb{R}^l$ is a vector of regression coefficients and $\epsilon \in \mathbb{R}^n$ represents noises.

3.1 Predictability maximized methods

Among many solutions to fit the above linear model, by far, the most popular methods are OLS and Lasso. We also add Elastic-net to compare with other models. Given the values of regularization parameters, those methods estimate regression parameters (β) by purely minimizing prediction error using the following objective function of Elastic-net:

$$\min_{\beta} \parallel y - X\beta \parallel_2^2 + \lambda[\alpha \parallel \beta \parallel_1 + (1-\alpha) \parallel \beta \parallel^2], \qquad (1)$$

where $\lambda > 0$ is a regularization parameter and $0 \leq \alpha \leq 1$ is a mixing parameter between Ridge ($\alpha = 0$) [26] and Lasso ($\alpha = 1$). It becomes OLS when $\lambda = 0$.

However, in this paper, we are more interested in maximizing explainability than predictability. In the next section, we illustrate why these "Predictability Maximized" methods may fail in identifying significant features with different reasons.

3.1.1 OLS ($\lambda = 0$ in Eq. 1)

Since we can assume that OLS's standardized coefficient follows t-distribution, one can test whether the estimated coefficient is significantly different from zero easily. Its t-statistics and p-value are

$$t = \frac{\hat{\beta}_i}{\sqrt{\text{Var}(\hat{\beta}_i)}}, \text{ and} \tag{2}$$

$$p\text{-value} = \begin{cases} 2 \times (1 - P(T \leq t)) \text{ if } \hat{\beta}_i > 0, \\ 2 \times P(T \leq t), \text{ otherwise.} \end{cases} \tag{3}$$

As t-statistics deceases, the p value increases. If p-value is greater than a significance level (e.g., 0.05), one cannot reject the null hypothesis that $\hat{\beta} = 0$. Although the estimated coefficient $\hat{\beta}$ is not zero, t-statistics can be close to zero if its variance is big. In this sense, a statistical test using p value is a stronger test than other feature selection methods such as Lasso, which selects active variables as long as their coefficients are not zero.

One could throw away some x variables (e.g., frequent low variables) to make $p < n$ as our X is not full rank, and the estimated coefficients are no longer unique [4]. However, OLS still suffers from multicollinearity, albeit a high correlation among x variables. It is because the number of x variables is still close to the number of observations. The amount of redundancy of x_i with the rest of x variables can be measured by R-squared statistics (R_i^2), which is the proportion of the variance in x_i that is predictable from the other x variables. R_i^2 never decreases as the number of x increases. Hence, when there are many x variables, the chance of multicollinearity is high.

Variance Inflation Factor (VIF) can be calculated by

$$\text{VIF}_i = \frac{1}{(1 - R_i^2)}, \tag{4}$$

where $i = 1, 2, \ldots, p$ and R_i^2 is the square of the coefficient of the multiple correlations for the regression of x_i on the other x variables. The variance of the estimated coefficient is

$$\text{Var}(\hat{\beta}_i) = \frac{\sigma^2}{(n-1)\text{Var}(x_i)} \text{VIF}_i. \tag{5}$$

where $sigma$ is a standard deviation value. Therefore, one can see that multicollinearity (i.e., high R_i^2) causes high VIF as in equation (4) and so increases variance of the estimated coefficient according to equation (5). This finally decreases t statistics as in equation (2) and so increases p-value as in equation (3). With a high p-value (>0.05), we cannot reject the null hypothesis that the estimated coefficient is zero. When one x variable has a high VIF, many other x variables are likely to have high VIFs because redundancy is interdependently measured. Therefore, when there are many x variables and multicollinearity happens, the variance of many estimated coefficient increases together. As a result, one can not find many significant explanatory variables.

3.1.2 Lasso ($\alpha = 1$ in Eq. 1)

When several x variables are highly correlated with each other, Lasso is likely to select only one or, at most, a few active variables and exclude others [27]. As a result, active variables resulted from Lasso tend to have low VIFs. Therefore, Lasso is free from multicollinearity among active variables and can identify more significant features than OLS.

However, Lasso, which focuses on prediction accuracy from unseen validation data, may not achieve our goal to identify many significant x variables. The justification to use validation data for prediction accuracy is originated from the assumption that validation data do not possess too different patterns from training data. However, our training and validation data are quite different. Because "unmatched variables" have all zeros in the validation set, their coefficients do not contribute to the prediction value at all in the validation set. Thus, when choosing λ, "matched variables" are forced to do more significant roles. For the only subset of variables to have attention, Lasso needs to penalize parameters more (i.e., bigger λ), which, in turn, forces some of the "matched variables" inactive as well. Thus, Lasso ends up with very few active variables in the existence of many "unmatched variables". In other words, there is "inflated regularization" due to the "unmatched variables". This is problematic because it reduces the number of candidates of significant features in training data.

3.1.3 Elastic-net (Eq. 1)

For small-n-large-l data, unlike Lasso, Elastic-net can choose more than n variables including a group of correlated variables by using both $L1$ and $L2$ penalties in equation (1) [16]. Thus, Elastic-net has potential to identify more significant features.

However, if there are many "unmatched" variables due to infrequent content features, like Lasso, Elastic-net will also suffer from "heavy" regularization, as seen in equation (1), and it results in only a few active variables and a small number of significant features.

In summary, OLS does not obtain many significant features due to multicollinearity. Lasso and Elastic-net fail in identifying significant features because of "heavy" regularization. To tackle those problems, we introduce our novel model in the next section.

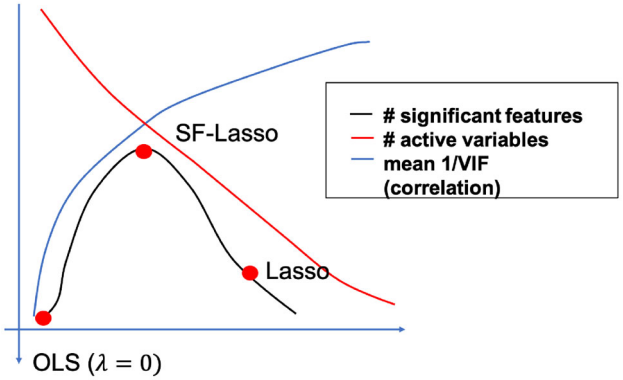


Fig. 1 # significant features is affected by # active variables and correlation among the active variables

4 Proposed explainability maximized method

In this section, we present a framework of SFLasso-SI to maximize explainability (Sect. 4.1). Extended gradient descent method to decide regularization paths is addressed in the Sect. 4.2 and details of Selective Inference for more conservative significance test is provided in the Sect. 4.3.

4.1 SFLasso-SI

Fig. 1 shows that as λ decreases, the number of active variables increases, but the correlation among the active variables also increases. This relationship suggests that too big or too small values of λ are not good at identifying significant features. When λ is very big (e.g., Lasso, Elastic-net), only a small number of active variables is generated, so multicollinearity is not likely to happen. However, only a few active variables result in even fewer significant features. On the other hand, When λ is very small (e.g., λ = 0 for OLS), there are many active variables, so the correlation among active variables is high. This multicollinearity reduces the number of significant features.

Considering those two competing forces, one can expect that the maximum number of significant features is likely to occur at some point of λ where there is a good enough number of active variables while the correlation among x variables is not too high. Based on the above behaviors, we propose a novel SFLasso-SI to maximize the number of significant features. The objective function for our proposed model is defined as

$$f(\lambda) = \max \sum_{i=1}^{p} I(p\text{-value}_{\beta_i} \leq 0.05),$$

where $I = 1$ if p-value$_{\beta_i} \leq 0.05$ or $I = 0$ otherwise.

Algorithm 1 SFLasso-SI (X, y, K)

1: $\mathcal{P} \leftarrow$ Extended coordinate descent(X, y, K)
2: **for** $\lambda \in \mathcal{P}$ **do**
3: A = Lasso (X, y, K)
4: Conduct the selective inference (A)
5: Count number of significant explanatory variables
6: **end for**
7: $\lambda^*_{\text{SFLasso-SI}} = \arg\max_\lambda \sum_{i=1}^{p} I(p\text{-value}_{\beta_i} \leq \alpha)$
8: **return** The optimal solution $\lambda^*_{\text{SFLasso-SI}}$

For the efficient search of regularization path, we adopt the regularization path of coordinate descent [28]. As smaller λ generates more active variables, we search more for small λ values (Sect. 4.2).

For a given λ, we select active variables (A) with nonzero coefficients that minimize the objective function with α=1 in equation (1) for Lasso.

For testing each selected variable's significance, we apply the selective inference [5, 6] (Sect. 4.3). Then, we count the number of significant features. We repeat this process within a range of λ values. Finally, we choose λ that generates the maximum number of significant features. The whole algorithm of SFLasso-SI is summarized in Algorithm 1.

One potential concern about our proposed SFLasso-SI is over-fitting because it uses only training data to find the optimum λ. However, note that SFLasso-SI also uses $L1$ regularization in estimating their coefficients in equation (1), and hence, can also choose a small set of variables by excluding ineffective or highly correlated variables. Furthermore, statistical testing via selective inference has the power to exclude potentially noise variables. Therefore, SFLasso-SI is not likely to suffer from over-fitting problem. We will address this in the Sect. 5 later.

4.2 Regularization paths via extended coordinate descent

We extend coordinate descent [28] to find the selected λ values. We define a regularization path $\mathcal{P}$ as a set of all solutions for $\lambda_i > 0$ and it is a decreasing sequence

$$\mathcal{P} = \{\lambda_{\max}, \ldots, \lambda_{\min}\}.$$

$\lambda_{\max}$ that can be easily computed from the input X and y; it is the smallest value for λ such that all the coefficients are zero (entire vector $\hat{\beta} = 0$) to serve as a warm start and

$$n\lambda_{\max} = \max_l |\langle x_l, y \rangle|,$$

where $\langle x_j, y \rangle = \sum_{i=1}^{n} x_{ij} y_i$. Therefore, we need to compute inner products of each feature with y initially, and then each time a new feature x_k enters the model, we need to compute and store its inner product with all the rest of the features.

Algorithm 2 Extended coordinate descent (X, y, K)

$\mathcal{P} = \emptyset$
$\lambda_{\max} = \frac{1}{n} \max_l |\langle x_l, y \rangle|$
$\lambda_{\min} = \epsilon \lambda_{\max}$
$r = \epsilon^{1/K}$
$\lambda_0 = \lambda_{\max}$
$\mathcal{P} = \mathcal{P} \cup \{\lambda_1\}$
for $i \in \{1, \ldots, K\}$ **do**
  if $\frac{dev(\lambda_{i-1})}{dev(\lambda_i)} <$ threshold or $dev(\lambda_i) <$ limit **then**
    $\lambda_i = \lambda_{\max} \times r^i$
    $\mathcal{P} = \mathcal{P} \cup \{\lambda_i\}$
  end if
end for
d=$\lambda_{\min+1} - \lambda_{\min}$
$\lambda_{\text{new}} = \lambda_{\min} - d$
while $\lambda_{\text{new}} > 0$ **do**
  $\mathcal{P} = \mathcal{P} \cup \{\lambda_{\text{new}}\}$
  $\lambda_{\text{new}} = \lambda_{\text{new}} - d$
end while
return Regularization paths $\mathcal{P}$

Next, we compute $\lambda_{\min}$ by selecting a minimum value $\lambda_{\min} = \epsilon \lambda_{\max}$ and construct a sequence of K values of λ decreasing from $\lambda_{\max}$ to $\lambda_{\min}$ on the log scale. Typical values are $\epsilon = 0.001$ and $K = 100$.

As smaller λ generates more active variables which can provide more candidates for significant features. Significant features are from the significant test of active variables. Therefore, we search more λ values from $\lambda_{\min}$ to 0 with an interval, which is the difference of the last two λ values (i.e., $\lambda_{\min+1}$ - $\lambda_{\min}$). We don't include zero value as $lambda$=0 means OLS rather than Lasso. See Algorithm 2 for details. In Algorithm 2, dev is defined by the fraction of deviation explained on the training data:

$$D^2_\lambda = \frac{dev_{\text{null}} - dev_\lambda}{dev_{\text{null}}},$$

where the deviation $dev_\lambda = -2logL(\hat{\beta}) + 2logL$(saturated model) with parameter λ and saturated model (i.e., having $\hat{y} = y_i$). dev_{null} is the null deviation computed at the constant model [13].

4.3 Selective inference

We conduct statistical testing to find significant variables using selective inference [5, 6]. The main idea of the selective inference is to account for the fact that active variables selected from adaptive methods such as Lasso have nonzero coefficients in doing a significance test [13]. The event that the Lasso selects a certain model M (i.e., a set of active variables) by assigning nonzero coefficients and certain signs to those selected active variables can be written as the polyhedral form with a set of linear constraints $\{Ay \leq b\}$, where $y = X\beta$; the matrix A and vector b depend on X and the selected model M. The details of A and b are spelled out in the papers [5, 6]. $\{y|Ay \leq b\}$ is the set of random response vectors y that would yield the same active variables and their coefficient signs with the selected model M.

Now suppose that y is sampled from a normal distribution with a population mean μ and a variance $\sigma^2 I$.

$$y \sim N(\mu, \sigma^2 I),$$

where $\mu = X\beta$ and σ^2 is assumed known. With the reduced model M that has selected active variables, the estimated coefficient vector is

$$\hat{\beta} = (X_M^T X_M)^{-1} X_M^T y.$$

Its corresponding population coefficient parameter is

$$\beta = (X_M^T X_M)^{-1} X_M^T \mu,$$

which is the coefficients in the projection of μ on X_M. Individual population coefficient can be expressed as the linear combination of the population mean μ as follows.

$$\beta_i = \eta^T \mu,$$

where η is one of the columns $X_M(X_M^T X_M)^{-1}$. Its estimated value is $\hat{\beta}_i = \eta^T y$. The main purpose of selective inference is to make inference on linear combination of μ, which is $\beta_i = \eta^T \mu$ conditional on the selection event $Ay \leq b$. Conditional on the selected model, $\hat{\beta}_i | Ay \leq b = \eta^T y | Ay \leq b$ follows Truncated Normal (TN) distribution distribution with support range $[v^-(y), v^+(y)]$ determined by A and b

$$\eta^T y | \{Ay \leq b\} \overset{\text{d}}{=} TN(\eta^T \mu, \sigma^2 \|\eta\|_2^2, [v^-(y), v^+(y)]),$$

where

$$\{Ay \leq b\} = \{v^-(y) \leq \eta^T y \leq v^+(y), v^0(y) \geq 0\}$$
$$v^-(y) = \max_{j:\alpha<0} \frac{b_j - (Ay)_j + \alpha_j \eta^T y}{\alpha_j}$$
$$v^+(y) = \min_{j:\alpha>0} \frac{b_j - (Ay)_j + \alpha_j \eta^T y}{\alpha_j}$$
$$v^0(y) = \min_{j:\alpha=0} (b_j - (Ay)_j)$$
$$\alpha = \frac{A\eta}{\|\eta\|_2^2}.$$

The Cumulative Distribution Function (CDF) of a random variable, evaluated at the value of that random variable, has

uniform distribution as follows

$$
\begin{aligned}
&G(\beta_i = \eta^T y) \\
&= F^{\nu_-,\nu_+}_{\eta^T\mu,\sigma^2\|\eta\|_2^2}(\eta^T y)|\{Ay \le b\} \\
&= \frac{\Phi((\eta^T y - \eta^T\mu)/\sigma\|\eta\|_2) - \Phi((\nu^- - \eta^T\mu)/\sigma\|\eta\|_2)}{\Phi((\nu^+ - \eta^T\mu)/\sigma\|\eta\|_2) - \Phi((\nu^- - \eta^T\mu)/\sigma\|\eta\|_2)} \\
&\sim U(0,1),
\end{aligned}
$$

where F denotes the CDF of a random variable from $N(\mu, \sigma^2)$ truncated to the interval $[\nu_-, \nu_+]$. Φ is the CDF of a random variable from $N(0,1)$.

As coefficients $\beta_i = \eta^T\mu$ follow a uniform distribution, it is easy to test whether the ith estimated coefficient is significantly different from zero using its p-value.

$$
p\text{-value} = \begin{cases} 2 \times (1 - G(\eta^T y)), & \text{if } \hat{\beta}_i > 0 \\ 2 \times G(\eta^T y)). & \text{otherwise} \end{cases} \tag{6}
$$

Because the closed-form for p-value calculation is available in equation (6), after active variables are identified, the additional calculation time in each lambda value is trivial.

One potential concern about our proposed SFLasso-SI is over-fitting because it uses only training data to find the optimum λ. However, note that SFLasso-SI also uses $L1$ regularization in estimating their coefficients in equation (1), and hence, can also choose a small set of variables by excluding ineffective or highly correlated variables.

Another advantage of SFLasso-SI is that it could use more data for training coefficients as SFLasso-SI does not require validation set in optimizing λ, particularly when sample size (n) is small as in our video and picture ad data. However, for a fair comparison, we use the same training set for all the models.

5 Empirical evaluation

We conduct experiments using two datasets of YouTube videos, seven datasets of Instagram pictures, and one text dataset of UCI Drug review. Our datasets are from diverse industries such as charity, cafe, beer, fashion and pharmaceutical industry. We divide our datasets into the training (70%), validation (15%), and test (15%) sets. The validation set is used to tune hyperparameters for Lasso and Elastic-net, optimized by prediction accuracy on unseen data. The test set is used to assess the prediction abilities of all models on unseen data. Table 1 shows the size of training, validation and test datasets.

Our datasets have bigger number of features than the number of observations and high unmatched ratio (Table 2). As ad creatures need to require new contents, each video content has high heterogeneity. Therefore, we found that high unmatched ratio from the datasets. We will explain details of features in the next section. Table 3 shows summary statistics of our target variables.

Table 1 Size of training, validation, and test sets

	# Train	# Validation	# Test	# Total
World Vision US	370	79	80	529
World Vision CA	318	68	69	455
Starbucks	1,216	261	261	1,738
Corona	561	120	121	802
Nike	709	152	152	1,013
Apple	313	67	68	448
AWS	377	81	81	539
Google	695	149	150	994
Microsoft	364	78	79	521
Drug Review	700	150	150	1,000

Table 2 Unmatched ratio of datasets

	Unmatched ratio (%)
World Vision US	68.9
World Vision CA	73.0
Starbucks	74.7
Corona	78.1
Nike	67.2
Apple	83.0
AWS	77.5
Google	76.8
Microsoft	74.7
Drug Review	69.3

5.1 Data collection

Our datasets of videos and picture are unique. We scrape videos datesets from YouTube and pictures from Instagram. We also use UCI Drug review dataset [29]. We download the dataset from UCI Machine Learning repository (https://archive.ics.uci.edu).

5.1.1 YouTube videos

Table 4 shows feature information of YouTube video datasets. From YouTube videos, we extract visual objects, speech tones (e.g., happy, sad), and spoken words. We collect the videos of World Vision US and World Vision Canada (which are Global Non-Profit Organizations) from YouTube. Figure 2 is a playlist of World Vision US video ads from YouTube. The collect data include the number of likes, published date, description, duration, genre, tag, and URL. The number of likes is used as a response variable for our

Table 3 Summary statistics of the number of likes (y)

	N	Mean	SD	Min	Max
Vision US	370	19.7	48.8	0	536
Vision CA	318	32.8	148.0	0	2,205
Starbucks	1,216	161,468.0	98,811.2	481	649,634
Corona Beer	561	4,239.7	2,291.1	556	21,751
Nike	709	32,383.2	21,873.3	4,748	296,636
Apple	313	225,310.1	124,112.8	49,889	970,632
AWS	377	198.1	244.6	5	1,478
Google	695	44,899.3	32,609.9	2,831	336,796
Microsoft	364	9,145.4	8,379.2	824	45,407
Drug Review	700	9,145.4	8,379.2	824	45,407

Table 4 The number of features of YouTube video datasets

	Visual objects	Speech tones	Spoken words	Total
Vision US	402	2	6,966	7,370
Vision CA	357	2	5,848	6,207

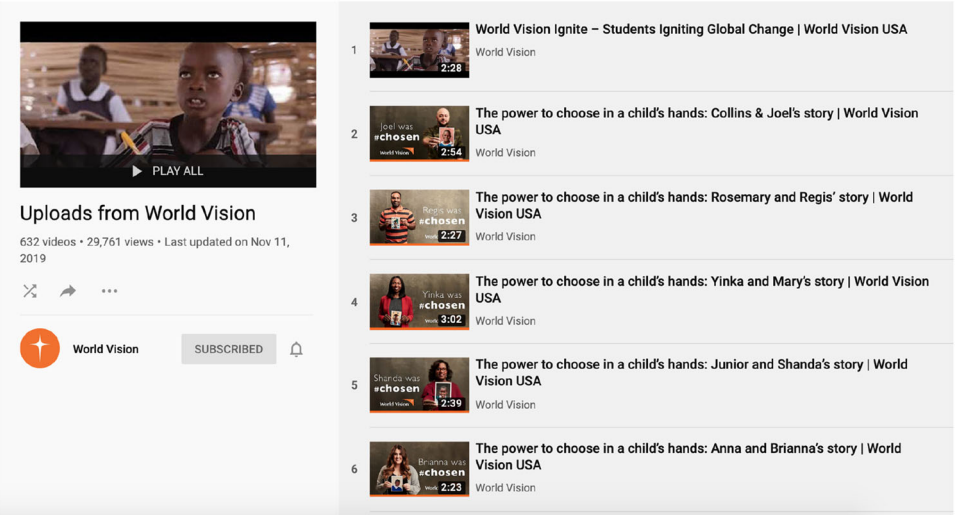


Fig. 2 A playlist of videos of World Vision US from YouTube

regression model. With the URLs from YouTube, we download all videos. To extract the video information we use the Microsoft Video Indexer API (https://azure.microsoft.com/en-au/services/media-services/video-indexer). For example, we obtain visual content features such as sky, plant, and orange (Table 10). Also we collect a spoken work list using video scripts such as clean and water (Fig. 5).

5.1.2 Instagram pictures

Table 5 shows feature information of Instagram picture datasets. To show that our proposed model can be applied to many different contents well, we used diverse sources across different types of products (e.g., drink, shoes, mobile phone, IT service). We scrape Instagram data using Phantombuster's Instagram Posts Extractor (https://phantombuster.com). Figure 3 is an example of Instagram of Apple. The data contain the number of likes, image URLs, and text descriptions. Using the URLs, we download the ad pictures. Then we use the following Google Vision API to extract picture infor-

Table 5 The number of features of Instagram picture datasets

	Visual objects	Text description	Total
Starbucks	410	3,461	3,871
Corona Beer	146	2,577	2,723
Nike	222	1,849	2,071
Apple	159	1,854	2,013
AWS	200	1,753	1,953
Google	230	3,731	3,961
Microsoft	104	3,640	3,744

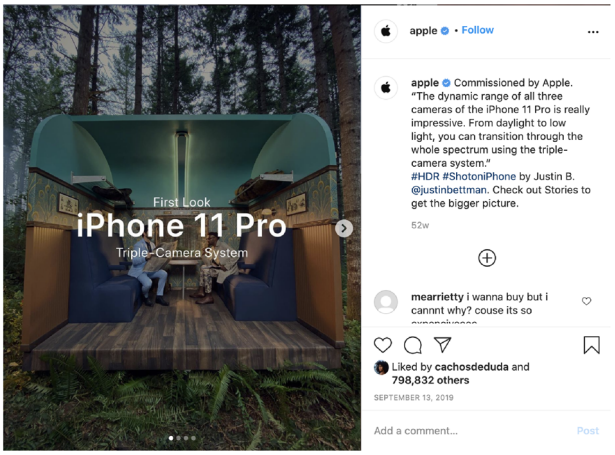


Fig. 3 A screenshot of Apple's Instagram

mation (https://cloud.google.com/vision/docs/features-list#image-properties-5):

- Optical Character Recognition (OCR)
  Text in the pictures. Use text recognition and conversion to machine-coded text.
- Face Emotion Detection
  Returns likelihood rating for emotions (joy, sorrow, anger, and surprise).
- Landmark Detection
  Provides the name of the landmark.
- Safe Search Detection (Explicit content detection)
  Provides likelihood ratings for the following content categories: adult, spoof, medical, violence, and racy.
- Multiple Object Detection
  Provides general labels of identified multiple objects.
- Image Properties Detection
  Returns dominant colors in an image.
- Web Entities Detection
  Provides inferred entities and best guesses as to the topic of an image from a similar image search on the web.
- Logo Detection
  Provides brand logo text identified with a confidence score.

For example, from the Starbucks dataset, we collect content features like ocr-text (love) and visual object (e.g., shoe) (Fig. 6 and Table 12).

5.1.3 UCI drug review data

Table 6 shows feature information of Drug review datasets. As other video and picture datasets are small-n-large-l, we randomly sample 1,000 observations from the original dataset. We use a useful counts as our target outcome and use names of drug, the condition from which the patient is suffering, reviews written by the patients, and the numerical rating values as explanatory features. For processing text data, we follow the same pre-processing steps which we have done for the textual data in the other datasets.

5.2 Evaluation settings

In our experiments, we have three kinds of features:

- Explanatory variables—content features such as visual objects, speech tones, spoken words (for videos), various visual information, text descriptions (for pictures), drug name, condition, review, rating (for drug review).
- Active variables—the nonzero coefficient variables for Lasso, Elastic-net, and SFLasso-SI. For OLS, we select frequently used n (# observations of the training set)-2 variables, the maximum number of explanatory variables. These active variables become inputs for the significant test.
- Significant features—our target measure to assess the level of explainability. We test each coefficient of active variables using p-value ≤ 0.05.

When the goal is better prediction, one needs to evaluate prediction accuracy using unseen test data. On the contrary, the relationship between x and y cannot be evaluated using test data because the model is made from training rather than test data. Notably, a significant relationship between the x and y does not entirely depend on the given training data. Note that the change in training data leads to the change in estimated coefficients. In other words, there is distribution (or uncertainty) in the estimated coefficients. To deal with this issue, statistical inference plays a vital role in the significance test. Inference means guessing the pattern of population data from sample data. Although the model is trained in given one sample, statistical test accounts for the possibility of the variation of sample data. p-value ≤ 0.05 is usually used as the criteria of the significance test. Significant feature with p value ≤ 0.05 means that if training data is sampled from a population, 95% of the time, the confidence interval of coefficients does not include zero. Therefore, significant features are likely to affect an outcome variable y, although training data varies.

To measure aggregate amount of correlations among active variables, we use the mean of VIF. It has a range from 0 to infinity, and a large value means high collinearity.

5.3 Illustration of SFLasso-SI

Figure 4 is the result of World Vision US YouTube Videos. It shows that our SFLasso-SI is mainly affected by two opposing factors: model size and correlations among explanatory variables. The # of active variables, and the mean of VIF,

Table 6 The number of features of Drug review datasets

	Rating	Drug Name	Condition	Review	Total
Drug review	1	417	214	4,083	4,714

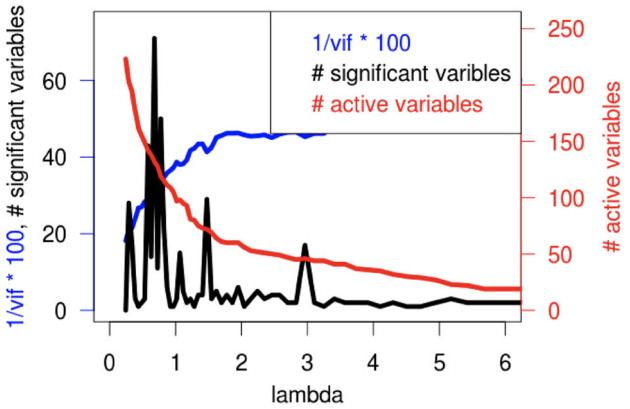


Fig. 4 Results: World Vision US

and # of significant features resemble theoretical prediction in Fig. 1. As λ decreases, because of the two opposing forces, the peaks in the # of significant features increase and then decrease after the optimum values of λ (0.68), at which SFLasso-SI finds the maximum number of significant features. One exception is that # of significant features does not have a smooth function of λ. Instead, there are many low values close to zero between peaks due to the selective inference screening of some features with insufficient significance.

5.4 Experiment results

Table 7 shows that our SFLasso-SI can identify the maximum number of significant features over nine ad datasets, while others detect a small number of significant features. As seen in Fig. 1 and 4, our SFLasso-SI has the optimum value of λ between OLS (λ=0, which means zero regularization) and the other two models. Here, as OLS has no regularization parameter λ, we can regard optimum λ for OLS is zero.

For Elastic-net, we experiment with the same ranges of λ varying α from 0.1 to 0.9 with 0.1 intervals.

As OLS uses the biggest number of active variables, it has the highest average VIF (multicollinearity), resulting in less significant features. On the other hand, Lasso and Elastic-net produce a relatively small number of active variables, thus less significant features despite the smallest average VIF. As discussed, Lasso and Elastic-net lose many active variables while optimizing for prediction accuracy, mainly due to heavy regularization resulted from many unmatched variables. Therefore, ad practitioners cannot get much insight from the standard models. On the contrary, our proposed SFLasso-SI identifies many active variables, much more than Lasso and Elastic-net. However, it has a much lower correlation among active variables than OLS. Because of the low correlation among relatively many active variables, SFLasso-SI could identify many significant features.

5.5 Prediction accuracy

Table 7 shows the prediction error on the test set with Root Mean Square Error (RMSE). OLS has the highest error on the test set and lowest error on the training set. This suggests that OLS (no regularization) suffers from over-fitting. The other three lasso variants have much lower RMSE on prediction error, confirming that regularization helps avoid over-fitting. Although SFLasso-SI uses only training data in searching the maximum number of significant features, both regularization and selective inference prevent SFLasso-SI from over-fitting. Notably, although SFLasso-SI is not intented to maximize prediction accuracy unlike Lasso and Elastic-net, our proposed SFLasso-SI has a similar prediction error as Lasso and Elastic-net. SFLasso-SI has the lowest prediction error in four datasets (Starbucks, Corona Beer, AWS, Microsoft), while Lasso and Elastic-net have the lowest one in four and two test data, respectively.

As we already mentioned in the Sect. 4, SFLasso-SI is not likely to suffer from over-fitting problem because it uses $L1$ regularization in estimating their coefficients in equation (1), and hence, can also choose a small set of variables by excluding ineffective or highly correlated variables.

In our experiments, SFLasso-SI generates active variables whose average VIF is much lower than that of OLS. Furthermore, statistical testing via selective inference has the power to exclude potentially noise variables. As a result, although SFLasso-SI is not designed to maximize prediction accuracy, SFLasso-SI achieves the similar level of prediction accuracy as Lasso or Elastic-net.

In summary, across all 10 datasets, our proposed SFLasso-SI finds more “true" significant video content features than the others while maintaining comparable prediction accuracy with Lasso and Elastic-net.

5.6 Complexity analysis

Compared to Lasso, the proposed approach needs to additionally calculate p-values for each active variable given λ value. However, since the closed-form for p-value calculation is available in equation 6, the additional calculation time is trivial. Furthermore, SFLasso-SI does not require validation

Table 7 Results from models

		λ	α	#Active Features	VIF	RMSE (train)	RMSE (test)	#Significant **features**
World	OLS	0		368	1407.0	0.7	2486.5	**0**
Vision	Lasso	34.9		4	1.3	44.4	203.0	**2**
US	Elastic-net	276.4	0.1	7	1.6	44.0	204.0	**1**
	SFLasso-SI	0.68		132	3.2	20.8	209.8	**71**
World	OLS	0		316	1326.6	1.7	3895.4	**0**
Vision	Lasso	135.6		4	1.5	149.5	280.1	**0**
CA	Elastic-net	1356.1	0.1	3	1.1	148.8	277.4	**1**
	SFLasso-SI	2.2		117	2.9	78.4	315.4	**40**
Starbucks	OLS	0		1,214	55.5	121.3	21474.5	**10**
	Lasso	3.24		149	1.6	1323.9	1428.3	**25**
	Elastic-net	4.05	0.8	155	1.7	1328.8	1432.9	**0**
	SFLasso-SI	0.57		275	6.8	1083.3	1423.5	**126**
Corona	OLS	0		559	20.3	53.3	5947.6	**21**
	Lasso	33.50		4	2.4	521.2	1483.4	**1**
Beer	Elastic-net	200.85	0.1	8	4.0	525.5	1485.2	**2**
	SFLasso-SI	0.77		143	2.0	431.4	1450.4	**46**
Nike	OLS	0		707	127.3	25.0	12213.7	**4**
	Lasso	2.29		39	1.8	325.9	315.9	**5**
	Elastic-net	2.80	0.9	39	1.8	332.6	322.9	**11**
	SFLasso-SI	0.34		239	2.3	356.2	369.3	**45**
Apple	OLS	0		311	23.5	353.8	137657.2	**37**
	Lasso	21.30		27	1.3	2251.0	2004.5	**1**
	Elastic-net	32.34	0.6	32	1.4	2313.6	2058.2	**10**
	SFLasso-SI	0.57		275	6.8	2025.6	2169.0	**126**
AWS	OLS	0		375	34.1	37.8	13459.4	**16**
	Lasso	1.29		83	1.9	155.3	376.4	**3**
	Elastic-net	1.36	0.9	92	2.0	149.5	382.9	**1**
	SFLasso-SI	0.27		230	3.9	66.0	306.5	**69**
Google	OLS	0		693	154.3	30.0	478739.8	**4**
	Lasso	2.84		51	1.5	404.2	422.7	**9**
	Elastic-net	3.15	0.9	52	1.5	400.2	418.7	**1**
	SFLasso-SI	0.45		324	2.5	289.2	418.8	**115**
Microsoft	OLS	0		362	4964.3	10.4	13499.6	**0**
	Lasso	2.98		17	1.4	169.7	184.7	**5**
	Elastic-net	4.72	0.3	44	1.4	179.7	186.5	**0**
	SFLasso-SI	0.42		140	2.0	167.9	181.5	**59**
UCI	OLS	0		698	82.2	5.6	352.0	**10**
	Lasso	0.60		40	1.2	30.3	36.4	**0**
Drug Review	Elastic-net	1.74	0.3	54	1.4	28.7	37.9	**0**
	SFLasso-SI	0.26		107	1.4	25.8	39.3	**45**

The bold values mean our proposed measures to quantify explainability

Table 8 Run-time from World Vision US and World Vision CA

		Train time (s)
World	OLS	0.119
Vision	Lasso	0.284
US	Elastic-net	2.233
	SFLasso-SI	0.652
World	OLS	0.056
Vision	Lasso	0.174
CA	Elastic-net	1.532
	SFLasso-SI	0.518

Table 9 Accuracy from human evaluation

	World prec	Vision Recall	US **F1**	World prec	Vision Recall	CA **F1**
OLS	0	0	**0**	0	0	**0**
Lasso	100	0.7	**1.5**	0	0	**0**
Elastic-net	100	0.4	**0.7**	100	0.6	**1.1**
SFLasso-SI	71.8	19.0	**37.8**	70.0	7.9	**14.1**

set in optimizing λ value, leading to the saving of computational time. Note that VIF requires calculating R-squared for every input dimension as a function of others. However, this is not part of SFLasso-SI. We report VIF in Table 7 only for illustration purposes to explain multicollinearity issues among benchmarking models. Table 8 shows the comparison of computational time. As expected, the computational times are similar across benchmarking models.

5.7 Human evaluations

We validate significant features from the two video datasets. To validate whether each model detects the right ad content features significantly appealed to ad viewers, we employ a human evaluation to validate the significant features resulted from all the four methods. One hundred video ads were randomly selected. The survey includes a URL and a list of visual objects, speech tones, and spoken words for given videos. Twenty marketing students were asked to watch video ads and circle video content features that make them like the video.

We use the set of content features from the survey as our ground-truth sets and compare them with significant features resulted from each model. Note that there are many video content features: 7,370 and 6,207 features across all the train set videos. The human evaluators chose 269 (4%) and 356 (6%) unique features from each country's videos. Therefore, accuracy between the evaluators' answers and each model's outcomes is likely to be small.

Table 10 Significant content features—World Vision US

OLS (0/0)	
Lasso	⟨visual objects⟩ **mammal(1)**
(2/2)	⟨spoken words⟩ **water(1)**
Elastic-net (1/1)	⟨spoken words⟩ **mammal(1)**
SFLasso-SI	⟨visual objects⟩ **outdoor(10) person(8)**
(51/71)	**sky(8) women(7) plant(3) watersport(3) yellow(3) young(3) chair(2) cow(2) food(2) house(2) little(2) orange(2) standing(2) brick(1) chicken(1) clothing(1) group(1) people(1) rock(1) rollingcredits(1) suit(1) sheep(1) text(1)** car cup curtain floor glasses militaryuniform shelf sitting tattoo television
	⟨spoken words⟩ **day(5) life(5) love(5)**
	make(5) food(4) people(4) believe(3) community(3) gift(2) healthy(2) mom(2) program(2) really(2) break(1) clean(1) continue(1) end(1) happen(1) follow(1) important(1) information(1) letter(1) parent(1) stuff(1) time(3) Anna Celisa close listen local midwife night photo Rwanda Sabina student

The features in bold mean the truly significant features verified by human evaluation

From the survey (Table 9), SFLasso-SI obtains 37.8 and 14.1 in F1 for the World Vision US and CA, respectively, which are almost from 14 times to 25 times bigger than Lasso and Elastic-net, respectively. As OLS (for both US and CA) and Lasso (for CA) do not detect any significant feature, precision, recall, and F1 are zero. Lasso in the US videos and Elastic-net in both US and CA show 100% in precision. However, they record very low recall and F1 as they identify only one or two significant feature(s). In other words, Lasso and Elastic-net identify the right features that are appealed to ad viewers, but a too small number of content features.

5.7.1 Validation of significant features

Table 10 shows statistically significant features from each model and 'true' content features by the survey participants. Our SFLasso-SI hits 51 variables among 71. The numbers within parentheses mean the number of videos where the evaluators like the video feature. The most frequently selected (in more than five videos) visual objects are 'outdoor'(10), 'person'(8), 'sky'(8), and 'women'(7), and spoken words are 'day'(5), 'life'(5), 'love'(5), and 'make'(5) and only SFLasso-SI identifies these features, suggesting that

Table 11 Significant content features—World Vision CA

OLS (0/0)	
Lasso (0/0)	
Elastic-net	⟨visual objects⟩
(2/2)	**grass(2) sky(2)**
SFLasso-SI	⟨visual objects⟩
(28/40)	**outdoor(4) child(3) people(3) building(2) wearing(2) clothing(1) scene(1) desert(1)** little looking road dirt way umbrella airplane library
	⟨spoken words⟩
	child(13) kid(5) school(5) like(5) vision(4) center(3) month(3) change(2) clean(2) government(2) sick(2) source(2) coffee(1) back(1) need(1) stay(1) talk(1) time(1) wait(1) today(1) belly Delhi force Wilfred

The features in bold mean the truly significant features verified by human evaluation

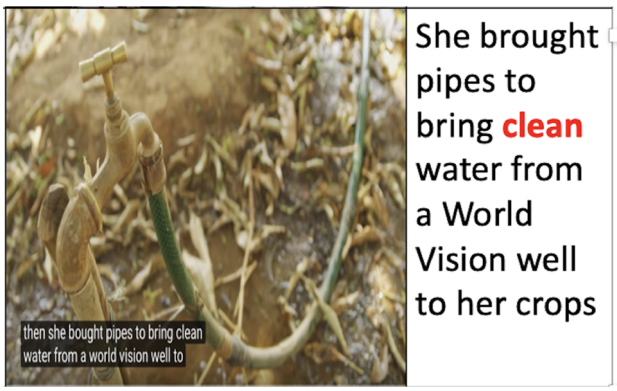


Fig. 5 An example of significant content feature of World Vision Canada YouTube video—clean (spoken words)

SFLasso-SI has better ability to identify more important features to ad viewers than other methods. Regarding the meaning of content features, note that World Vision helps children and families in need. Significant features are the visual objects or spoken words about family ('mom', 'parent'), health ('healthy', 'clean', 'water'), and food ('orange', 'food','chicken').

Table 11 shows significant content features from World Vision Canada. As the results from World Vision US, only SFLasso-SI identifies the most frequently selected content features (more than five) such as 'child', 'kid', 'school', and 'like'. Figure 5 is a screenshot and transcript of the World Vision US video, including a true significant feature, clean. Figure 6 and Table 12 show another example of true significant features from Starbucks Instagram.



Fig. 6 An example of significant content feature of Starbucks—love (ocr-text in the picture)

Table 12 Significant content features—Starbucks

OLS (10)	⟨objects⟩watch ⟨text⟩coldbrew home pick welcome cooky care coffeeathome monthly football
Lasso (25)	⟨ocr⟩love roasting ⟨web⟩caffèAmericano icedcoffee pumpkinspicelatte Teavana ⟨best⟩icecream ⟨objects⟩baggedpackagedgoods mug ⟨text⟩Canada cookie espresso fall flavor gingerbreadcafe home icedcaramelmacchiato icedtea lemonade meet nomnom plastic strawberry today watch
Elastic-net (0)	
SFLasso-SI (72)	⟨logo⟩BiFi ⟨ocr⟩frappuccino latte love via ⟨web⟩almondmilk coffeebean frappécoffee ⟨best⟩ coffeecup latte ⟨objects⟩ dessert packagedgoods person shoe ⟨text⟩ began bistrobox butterscotch cakepop changing chocolaty ck classic coffee cooper cream dreamy dunk emojis enjoying especially fact far favor friendship gluten icedcaramelmacchiato icedtea kinda look meet Mexico mocha mug mummycakepop naturally office psl reason recipe redcupart responder reusablecup roaster roll sale sauce seasonalsips Seattle sharejoy spiced spring start sunset sweet thing time tuesday Verona WA warm weekend white

6 Conclusion and future work

State-of-the-art machine learning methods and black-box algorithms have attracted remarkable attention. However, it becomes necessary for them to explain how to get to their conclusion. Explaining the relationship between the target variable and explanatory features is important for decision-makers. To tackle the issue, we model to maximize the number of significant features in our work.

However, this approach can be challenging. Data from many applications may have small-n-large-l problem and contain heterogeneous features. Such data often have multicollinearity and infrequent common variables across samples. When explainability matters, traditional methods that estimate a linear model do not work well. This paper has proposed SFLasso-SI to address these problems and verified its effectiveness by experiments. Our proposed model is found to be able to discover many significant content features, and our human survey has also verified the identified ad features. While providing better explainability, SFLasso-SI maintains a similar degree of prediction accuracy with Lasso and Elastic-net.

In this paper, our goal is to maximize the number of significant features rather than maximize prediction accuracy. Our proposed SFLasso-SI achieves this goal. However, our study has a limitation that an explainability is hard to quantify and the significant features is not the only way to quantify the explainability. In our further research, we will add more criteria to compare the explainability of different methods.

Acknowledgements This research is supported by the Australian Government Research Training Program Scholarship.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

A Appendix

This technical appendix includes further examples of the experiments. Figures 7, 8, 9, 10, 11 and 12 illustrate significant content features from our SF-Lasso. Tables 13, 14, 15, 16, 17, 18 and 19 show the lists of significant features from our experiments.



Fig. 7 An example of significant content feature of Corona beer—person (objects)



Fig. 8 An example of significant content feature of Nike—footwear (objects)



Fig. 9 An example of significant content feature of Apple—water (best)



Fig. 11 An example of significant content feature of Google—joy (face)



Fig. 10 An example of significant content feature of AWS—Amazon (ocr)



Fig. 12 An example of significant content feature of Microsoft—Microsoft Corporation (logo)

Table 13 Significant content features—Corona Beer

OLS (21)	⟨objects⟩PackagedGoods ⟨text⟩cerveza paradise place sand sun blue enjoy change cool horizon wo vibe embrace hat live luck number rise rolling walking
Lasso (1)	⟨text⟩legal
Elastic-net(2)	cerveza extra
SFLasso-SI (46)	⟨logo⟩RipCurl ⟨objects⟩boat person surfboard swimwear ⟨web⟩watercourse ⟨text⟩bay beach beautiful cliff create findyourbeach fishing foot ha hat home ibiza kick looking mauritius mount noronha number opportunity option phangan recommend resort ritual session shark sunset tourist trail underwaterworld waiting walk walking water waterfall wave wedge weekend white zaknoyle

Table 14 Significant content features—Nike

OLS (4)	⟨text⟩breathe innovation detailing aloha
Lasso (5)	⟨logo⟩Nike ⟨ocr⟩the ⟨text⟩Nikefc Nikesportswear stay
Elastic-net(11)	⟨logo⟩M4Carbine ⟨text⟩mike arrives august bold Nikefc Nikesportswear shadow sneakerboot snkrs stay
SFLasso-SI (45)	⟨logo⟩M4carbine⟨web⟩Sneakers ⟨best⟩cobaltblue orange sportswear ⟨objects⟩footwear hat jacket luggagebags necklace pants shoe top ⟨text⟩add arriving bio color drop dynamic era february flyknit ice inspired inspires jacquard june lunarlon mean new niketiempo process radiant rift rugged season shop sneakerboots sole soon stay track vapormax water

Table 15 Significant content features—Apple

OLS (37)	⟨text⟩ apple commissioned like just love life place work support want camera culture cloud voice wall come learn picture York art carry contrast east right travel adrienne dynamic emotion fell game holi Martin music range remind represent worth
Lasso (1)	⟨text⟩eye
Elastic-net(10)	⟨object⟩ Hat Wheel ⟨text⟩eye landscape learn nature snake thing using world
SFLasso-SI (126)	⟨logo⟩OverTheTopWrestling ⟨best⟩monochromephotography orange water ⟨object⟩animal bird hat lighting luggagebags person wheel ⟨text⟩absolutely allows anindito area banker beachside belarus bird boudhanath Brooklyn building calm changing chaos Christopher closer Columbia cowboy cranberry culture Daniel David destination drama eduardo etna Eugene existing exploration film fleeting forget fragment fun gaucho gujarat ha hauling help hesitation history human ice image important inevitability ivy Japan Jason Jenny Jesse John Jun kid kodoli learn life Liliana live long love lulu Martín mean memory Michael middle morning mystery new nuance outside park passer photography picture place playfulness plus potential pro rashed ready right safe shape sharing Sheldon short shotoniphone sicily space special spiderweb stand stop sweet Taipei tell thing thought tree turmeric tushar UAE unique unusual water way wildlife win work worthwhile çağla

Table 16 Significant content features—AWS

OLS (16)	⟨ocr⟩hereataws dogsofaws technical week ⟨text⟩
Lasso (3)	⟨object⟩Top ⟨text⟩amp day
Elastic-net(1)	⟨web⟩ Convention
SFLasso-SI (69)	⟨color⟩green pink ⟨logo⟩Audient GWK ⟨ocr⟩Amazon AWS deepracer services startups ⟨face⟩joy ⟨objects⟩Outerwear Packaged-Goods Pants Person ⟨safe⟩racy ⟨text⟩acceptance apps AWS awsdeepracer awsstartupday bepeculiar bite bootcamps builder business cash catch community cozy Dallas doe ear environment event family fluffy fun garden grab great hard historic info IOT jogging joined minute night online Paris prize run scene session sharing shepherd slideshare spooky spot started são tech wait walker way week woof year

Table 17 Significant content features—Google

OLS (4)	⟨object⟩Animal ⟨text⟩big stay animal
Lasso (9)	⟨logo⟩Google ⟨text⟩captured data juan learn mysuperg perfect teampixel week
Elastic-net(1)	⟨object⟩ Glasses
SFLasso-SI (115)	⟨ocr⟩and for hey the with ⟨face⟩joy ⟨objects⟩Animal Bracelet Clothing Glasses Houseplant Pants Tire Window ⟨best⟩sky ⟨text⟩adorable Android adventures away awesome beach big black browser building card computer crumbles cuddling data daydream designer doodle explore exploring fall feast featuring floor foggy forever game German goal handmade hat hi holiday home honor iconic India Juan key kicked lab launched lead lightning like link little live location London long look lot makermonday man meet month mural music national Nationaldogday new newest old oven parade participating past pattern pedal pencil phone photographer place plus pond reflecting reimagine Romania run science self smell snap sock splash standing star stayhome story street sweet technology thanks thing time tour winning word worldwide YouTube

Table 18 Significant content features—Microsoft

OLS (0)	
Lasso (5)	⟨logo⟩MicrosoftCorporation ⟨objects⟩Person ⟨text⟩domore learn visit
Elastic-net(0)	
SFLasso-SI (59)	⟨color⟩pink white ⟨logo⟩MicrosoftCorporation ⟨face⟩joy ⟨objects⟩animal pants shoe top ⟨text⟩access app approach bio brain breadcrumb camera coast comedy console domore event follow game group hand help home James joey journey life located lot medal met month need new old perfect photo picture program read Satya science Seattle showcase solution stop tech track tree using visit way winner year youth

Table 19 Significant content features—UCI Drug Review

OLS (10)	⟨drug name⟩Lisinopril Duloxetine ⟨condition⟩depression ⟨review⟩use dose recommend probably loose prior procedure
Lasso (0)	
Elastic-net(0)	
SFLasso-SI (45)	⟨condition⟩birth ⟨review⟩acne adderall ago amp bed birth bone celexa chantix completely do erectile food gabapentin go hack help hurt important low make mood night period pill prescribed prozac quality quot sack seizure shot start stomach sweat syndrome taken think use way went worse worst zoloft

References

1. Harder, F., Bauer, M., Park, M.: Interpretable and differentially private predictions. In: AAAI, pp. 4083–4090 (2020)
2. Gilpin, L.H., Bau, D., Yuan, B.Z., Bajwa, A., Specter, M.A., Kagal, L.: Explaining explanations: an overview of interpretability of machine learning. In: 5th IEEE International Conference on Data Science and Advanced Analytics, DSAA 2018, Turin, Italy, October 1-3, 2018, IEEE, pp. 80–89 (2018)
3. Areosa, I., Torgo, L.: Explaining the performance of black box regression models. In: 2019 IEEE International Conference on Data Science and Advanced Analytics, DSAA 2019, Washington, DC, USA, October 5-8, 2019, IEEE, pp. 110–118 (2019)
4. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics. Springer, Berlin (2001)
5. Lee, J.D., Sun, D.L., Sun, Y., Taylor, J.E.: Exact post-selection inference, with application to the lasso. Ann. Stat. **44**(3), 907–927 (2016)
6. Taylor, J., Tibshirani, R.: Post-selection inference for *l*1-penalized likelihood models. Can. J. Stat. **46**(1), 41–61 (2018)
7. Chen, J., Song, L., Wainwright, M.J., Jordan, M.I.: Learning to explain: an information-theoretic perspective on model interpretation. In: ICML, Vol. 80, pp. 882–891 (2018)
8. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: ICML, p. 3145-3153 (2017)
9. Singh, C., Murdoch, W.J., Yu, B.: Hierarchical interpretations for neural network predictions. In: ICLR (2019)
10. Tu, M., Huang, K., Wang, G., Huang, J., He, X., Zhou, B.: Select, answer and explain: interpretable multi-hop reading comprehension over multiple documents. In: AAAI, pp. 9073–9080 (2020)
11. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. Mach. Learn, arXiv (2017)
12. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B (Methodol.) **58**(1), 267–288 (1996)
13. Hastie, T., Tibshirani, R., Wainwright, M.: Statistical Learning with Sparsity: The Lasso and Generalizations (2015)
14. James, G., Witten, D., Hastie, T., Tibshirani, R.: An Introduction to Statistical Learning: with Applications in R. Springer (2013). https://faculty.marshall.usc.edu/gareth-james/ISL/
15. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: ICML, Vol. 70, pp. 3145–3153 (2017)
16. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. R. Stat. Soc. Ser. B **67**, 301–320 (2005)
17. Wang, S., Nan, B., Rosset, S., Zhu, J.: Random lasso. Ann. Appl. Stat. **5**, 468–485 (2011)
18. Wang, H., Lengerich, B.J., Aragam, B., Xing, E.P.: Precision Lasso: accounting for correlations and linear dependencies in high-dimensional genomic data. Bioinformatics **35**(7), 1181–1187 (2018)
19. Hara, S., Maehara, T.: Enumerate lasso solutions for feature selection. In: AAAI, p. 1985-1991 (2017)
20. Takada, M., Suzuki, T., Fujisawa, H.: Independently interpretable lasso: a new regularizer for sparse regression with uncorrelated variables. In: AISTATS (2017)
21. Ross, A.S., Lage, I., Doshi-Velez, F.: The neural lasso: Local linear sparsity for interpretable explanations. In: NIPS (2017)
22. Park, E., Wong, R.K., Kwon, J., Chu, V.W., Rutz, O.J.: Video ads content analysis using significant features lasso. In: The 43rd ISMS Marketing Science Conference (2021)
23. Belloni, A., Chernozhukov, V.: Least squares after model selection in high-dimensional sparse models. In: Bernoulli, vol. 19, pp. 521–547 (2013)
24. Lockhart, R., Taylor, J., Tibshirani, R.J., Tibshirani, R.: A significance test for the lasso. Ann. Stat. **42**(2), 413–468 (2014)
25. Park, E., Wong, R.K., Kwon, J., Chu, V.W.: Maximizing explainability with sf-lasso and selective inference for video and picture ads. In: The 25th Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 566–577 (2021)
26. Hoerl, A.E., Kennard, R.W.: Ridge regression: biased estimation for nonorthogonal problems. Technometrics **12**(1), 55–67 (1970)
27. Kim, Y., Bum Kim, S.: Collinear groupwise feature selection via discrete fusion group regression. Pattern Recognit. **83**, 1–13 (2018)
28. Friedman, J.H., Hastie, T.J., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. J. Stat. Softw. **33**(1), 1–22 (2010)
29. Gräßer, F., Kallumadi, S., Malberg, H., Zaunseder, S.: Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. In: DH, pp. 121–125 (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.