Supplementary material for the DAGM paper: A simple extension of stability feature selection

A. Beinrucker, Ü. Dogan and G. Blanchard

University of Potsdam, Institute of Mathematics

1 Algorithmic description of the method

Extended stability selection algorithm SFS($L, V$). $\Pi_F S$ denotes the projection of the covariates of the sample $S$ on the space spanned by feature set $F$ only. The original stability selection algorithm is recovered for $L = 2, V = 1$.

Parameters:

• Number $L$ of sample slices
• Number $V$ feature slices
• Number $R_{rf}, R_{rs}$ of feature/sample splits
• Threshold $\tau \in (0, 1)$

Input: sample $S$

Init: $N_i = 0, i = 1, \ldots, d$

for $r = 1$ to $R_{rf}$ do

Split feature indices $F = \{1, \ldots, d\}$ randomly into $V$ slices $F^{(1)}, \ldots, F^{(V)}$ of equal size $\lfloor \frac{d}{V} \rfloor$

for $r = 1$ to $R_{rs}$ do

Split the sample $S$ randomly into $L$ slices $S^{(1)}, \ldots, S^{(L)}$ of equal size $\lfloor \frac{N}{L} \rfloor$

for $i = 1, \ldots, L; j = 1, \ldots, V$ do

$G := $ BaselineFS($\Pi_{F^{(j)}} S^{(i)}$)

for all $k \in G$ do

$N_k \leftarrow N_k + 1$

end for

end for

end for

return set of indices $k$ such that $N_k / (L R_{rs}) \geq \tau$.

2 Proof of Theorem 1 and Corollary 1

We denote $S(\cdot)$ the baseline feature selection method. We split the sample randomly in $L$ non-overlapping subsamples of size $\lfloor n/L \rfloor$. Let $I_1, \ldots, I_L$ be random subsets of $\{1, \ldots, n\}$ such that $I_i \cap I_j = \emptyset \forall i \neq j$. This is repeated $R$ times, we denote correspondingly $I^{(r)}_k$ the random subsamples for repetition $r \in \{1, \ldots, R\}$.

Let

$$II^{SS}_L(k) := \frac{1}{RL} \sum_{(r, k)\in(1,1)} \mathbf{1}\{k \in S(I^{(r)}_k)\}$$
denote the empirical selection frequency of feature \( k \). The output of the stability selection algorithm for a threshold \( \tau \in (0, 1) \) is the set
\[
S_{L,\tau}^{SS} := \{ k : \Pi_{L}^{SS}(k) \geq \tau \}.
\]

We denote the random variable
\[
N_{L}^{(r)}(k) := \sum_{\ell=1}^{L} 1 \{ k \in S(I^{(r)}_{\ell}) \}
\]
counting for how many of the disjoint slices feature \( k \) has been selected in the \( r \)-th repetition; and define
\[
\Pi_{L,\ell_0}^{\text{simult}}(k) := \frac{1}{R} \sum_{r=1}^{R} 1 \{ N_{L}^{(r)}(k) \geq \ell_0 \}
\]
the number of repetitions out of \( R \) where feature \( k \) has been selected in at least \( \ell_0 \) slices.

**Lemma 1.** It holds for any \( k \in \mathcal{F} \):
\[
\left( \frac{L - \ell_0 + 1}{L} \right) \Pi_{L,\ell_0}^{\text{simult}}(k) + \frac{\ell_0 - 1}{L} \geq \Pi_{L}^{SS}(k).
\]

**Proof.** We have for all data repetitions \( r = 1, \ldots, R \):
\[
\frac{1}{L} \sum_{\ell=1}^{L} 1 \{ k \in S(I^{(r)}_{\ell}) \} \leq \left( \frac{\ell_0 - 1}{L} \right) 1 \{ N_{L}^{(r)}(k) \leq \ell_0 - 1 \} + 1 \{ N_{L}^{(r)}(k) \geq \ell_0 \}.
\]

Averaging over the repetitions \( r = 1, \ldots, R \), we obtain
\[
\Pi_{L}^{SS}(k) \leq \frac{\ell_0 - 1}{L} (1 - \Pi_{L,\ell_0}^{\text{simult}}(k)) + \Pi_{L,\ell_0}^{\text{simult}}(k) = \left( \frac{L - \ell_0 + 1}{L} \right) \Pi_{L,\ell_0}^{\text{simult}}(k) + \frac{\ell_0 - 1}{L}.
\]

**Lemma 2.** The following inequality holds for any \( k \in \mathcal{F} \), \( \xi > 0 \), and \( \ell_0 \in \{1, \ldots, L\} \) such that \( p_0 := \frac{\ell_0}{L} \geq p_{k,L} \):
\[
\mathbb{P} \left[ \Pi_{L,\ell_0}^{\text{simult}}(k) \geq \xi \right] \leq \frac{1}{\xi} \exp \left( -L \frac{(p_0 - p_{k,L})}{D} \right).
\]

**Proof.** We have
\[
\mathbb{E} \left[ \Pi_{L,\ell_0}^{\text{simult}}(k) \right] = \mathbb{P} \left[ \sum_{\ell=1}^{L} 1 \{ k \in S(I^{(1)}_{\ell}) \} \geq \ell_0 \right] = \mathbb{P} \left[ \text{Bin}(L, p_{k,L}) \geq \ell_0 \right] \leq \exp \left( -L \frac{(p_0 - p_{k,L})}{D} \right),
\]
where the last inequality is the Chernoff binomial bound. Using Markov’s Inequality we get (2).
Proof of the theorem:
For any \( k \in A_{\theta,L} \), it holds by definition of this set and the assumptions on \( p_0 \) that \( p_{k,L} \leq \theta \leq p_0 \), hence (2) holds, by Lemma 2. Now using Lemma 1 it follows that
\[
P[\Pi_L^{SS}(k) \geq \tau] \leq P\left[ \left( \frac{L - \ell_0 + 1}{L} \right) \Pi_{L,\ell_0}^{\text{simult}}(k) + \frac{\ell_0 - 1}{L} \leq \tau \right]
\]
\[
= P\left[ \Pi_{L,\ell_0}^{\text{simult}}(k) \geq \frac{L \tau - \ell_0 + 1}{L - \ell_0 + 1} \right]
\]
\[
\leq \frac{1 - p_0 + L^{-1}}{\tau - p_0 + L^{-1}} \exp(-LD(p_0,p_{k,L})) ,
\]
where we have used \( \xi := \frac{L \tau - \ell_0 + 1}{L - \ell_0 + 1} \). Hence
\[
E[|S_{L,\tau,\ell_0}^{SS} \cap A_{\theta,L}|] \leq \frac{1}{|A_{\theta,L}|} \sum_{k \in A_{\theta,L}} P[\Pi_L^{SS}(k) \geq \tau]
\]
\[
\leq \frac{1 - p_0 + L^{-1}}{\tau - p_0 + L^{-1}} \sum_{k \in A_{\theta,L}} \frac{1}{|A_{\theta,L}|} \exp(-LD(p_0,p_{k,L})) .
\]
We obtain the first part of the result by upper bounding for all \( k \in A_{\theta,L} \):
\[
\exp(-LD(p_0,p_{k,L})) \leq \exp(-LD(p_0,\theta)) .
\]
For the second part, we use instead the upper bound
\[
\exp(-LD(p_0,p_{k,L})) = \frac{\exp(-LD(p_0,p_{k,L}))}{p_{k,L}} p_{k,L}
\]
\[
\leq \frac{\exp(-LD(p_0,\theta))}{\theta} p_{k,L} ,
\]
since it can be checked that the function \( t \mapsto \frac{\exp(-LD(p_0,t))}{t} \) is nondecreasing for \( t \leq p_0 - L^{-1} \).

Proof of the Corollary: this follows the same argument as in [2]. If feature selection were completely random, the marginal selection probability of any given feature would be \( \frac{\theta}{d} \), where we recall \( q = E[|S_{L}^{\text{base}}|] \) is the average number of features selected by the baseline. Under Assumption (A), we assume that the selection probability of a relevant feature (set \( S \)) is better than random; it therefore entails that for any \( k \in S \), we must have \( p_{k,L} > \frac{\theta}{d} \). Conversely, and by exchangeability of features, for any \( k \in N \) (noise features), one has \( p_{k,L} < \frac{\theta}{d} \). Therefore, with \( \theta := \frac{\theta}{d} \) we must have \( A_{\theta,L} = N \) and \( A_{\theta,L}^c = S \) in this case. Inequality (1) therefore implies (5), wherein we have taken a minimum over the range of the bound parameter \( \ell_0 \) allowed in Theorem 1.

3 Heuristic analysis of feature subsampling

We restrict ourselves here to a heuristic analysis, whose aim is merely to illustrate how feature subsampling can help with the problem of “mutual masking” of
interesting features. For this we will consider a toy model similar to that proposed in [1] in the discussion of the original stability selection paper and exhibiting strong correlation of relevant variables. Let us assume a linear relationship \( Y = \sum_{i=1}^{p} \beta_i X_i + \varepsilon \), where the \( p \) first variables out of \( K \) are relevant, and \( \beta_1 \geq \beta_2 \geq \ldots \geq \beta_p \). We assume furthermore that the covariate vector \((X_i)\) has a \( N(0, \Sigma) \) distribution, with off-diagonal terms of \( \Sigma \) being 0 except for \( \Sigma_{12} = \Sigma_{21} = \rho \).

In [1] it is shown experimentally that there is a mutual masking effect of the two first variables when \( \rho \) is large, in that the selection of one variable by Lasso tends to prevent selection of the other. As a consequence, the original stability selection does not work so well on that model.

If we consider a simpler baseline given by greedy orthogonal matching pursuit (OMP) (or forward variable selection), we can heuristically understand this effect by considering the population version of the algorithm. In the population version, the first selected variable is \( X_1 \) since it has maximum correlation with \( Y \); the residual after removing the orthogonal projection of \( Y \) on \( X_1 \) takes the form \( \tilde{Y}_1 = -\rho X_1 + \sum_{i=2}^{p} \beta_i X_i + \varepsilon \), and at the second step of the algorithm we have \( \text{Cov}(\tilde{Y}_1, X_2) = \beta_2 (1 - \rho^2) \). Therefore, the apparent relevance of the second variable is severely downweighted, and remaining variables such that \( \beta_i > \beta_2 (1 - \rho^2) \) will be selected first. On real data, we expect a similar trend up to some random variations, and in particular that the two first variables have low probability to be selected simultaneously, so that at least one of them will have a relatively low (below 0.5) selection probability.

Now, if we consider what happens with feature subsampling into \( V \) non-overlapping groups, the two first features only have a probability \( 1/L^2 \) to be in the same group (in which case masking occurs as above), but probability \( 1 - 1/L^2 \) to be in two separate groups, in which case they both will be selected first in their respective groups (here again in the sense of the population version – on real data we expect a similar, if more noisy, behavior). This simple example shows the benefits of feature splitting in this situation. We note that [1] proposed the use of elastic net regularization instead of Lasso the baseline to counteract the masking effect. Here, our aim is to suggest instead a general-purpose approach that can potentially be applied to any baseline. While the above reasoning clearly is adapted to a specific baseline, the masking effect it likely to occur with many other popular feature selection methods that work in a greedy-iterative fashion (such as CMIM considered in the experimental section).

4 Supplementary results on ESS for MASH datasets

The setting and notation considered here is the same as in the main body of the paper, Section 3.2: the dataset considered is a subset of the MNIST dataset; predictor variables are given by a large-dimensional library of image feature extractors; and we are using (early stopped) CMIM as a baseline feature selection. Multiclass AdaBoost is used for learning after feature selection. Globally, feature subsampling appears to improve final classification rate while data subsampling appears to be performance neutral.
Table 1. The effect of feature subsample size on SFS\((L, V)\). The number of sample splits is fixed to \(L = 2\) and 1000 features are used.

<table>
<thead>
<tr>
<th># iterations</th>
<th>SFS(_1)(2, V)</th>
<th>SFS(_10)(2, V)</th>
<th>SFS(_{100})(2, V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>7.9 (0.4)</td>
<td>7.6 (0.4)</td>
<td>7.5 (0.3)</td>
</tr>
<tr>
<td>100</td>
<td>5.6 (0.2)</td>
<td>5.3 (0.2)</td>
<td>5.3 (0.2)</td>
</tr>
<tr>
<td>200</td>
<td>3.3 (0.1)</td>
<td>3.0 (0.2)</td>
<td>3.0 (0.2)</td>
</tr>
<tr>
<td>400</td>
<td>3.1 (0.1)</td>
<td>2.9 (0.2)</td>
<td>2.9 (0.2)</td>
</tr>
<tr>
<td>800</td>
<td>2.9 (0.1)</td>
<td>2.7 (0.2)</td>
<td>2.7 (0.2)</td>
</tr>
<tr>
<td>1600</td>
<td>2.2 (0.1)</td>
<td>2.1 (0.2)</td>
<td>2.1 (0.2)</td>
</tr>
</tbody>
</table>

Table 2. The effect sample splitting to SFS\(_{10}\). The number of feature splits is fixed to \(V = 2\) and 1000 features are used.

<table>
<thead>
<tr>
<th># iterations</th>
<th>SFS(_{10})(L, 2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>25.8 (1.6)</td>
</tr>
<tr>
<td>50</td>
<td>8.1 (0.2)</td>
</tr>
<tr>
<td>100</td>
<td>5.0 (0.2)</td>
</tr>
<tr>
<td>200</td>
<td>3.3 (0.1)</td>
</tr>
<tr>
<td>400</td>
<td>2.8 (0.1)</td>
</tr>
<tr>
<td>800</td>
<td>2.4 (0.1)</td>
</tr>
<tr>
<td>1600</td>
<td>2.0 (0.1)</td>
</tr>
</tbody>
</table>
References