CHARACTERIZING UNCERTAINTY IN LOW-DIMENSIONAL MODEL SELECTION

by

Nicholas Kissel

B.S., University of Pittsburgh, 2018

Submitted to the Graduate Faculty of the Dietrich School of Arts and Sciences in partial fulfillment of the requirements for the degree of

Master of Science

University of Pittsburgh

2019
CHARACTERIZING UNCERTAINTY IN LOW-DIMENSIONAL MODEL SELECTION

Nicholas Kissel, M.S.
University of Pittsburgh, 2019

In the context of big and often high-dimensional data, valid procedures for assessing variable importance and identifying accurate model representations are essential tools, especially in the presence of substantial instability. Instead of seeking to find only a single set of covariates that form the empirically optimal model, we propose an automated procedure for identifying an entire collection of stable and predictively similar models. Within each iterate of the selection method, we develop a procedure to identify covariates that are predictively similar with regard to a chosen loss function, thereby providing multiple options as to which covariate should be added to the final model. By construction, our procedure acts a wrapper method that can be applied to any statistical or machine learning technique. Furthermore, we provide a natural and intuitive graphical display of these model paths that makes apparent potential underlying relationships between covariates as well as the relative importance of the covariates selected.
# Table of Contents

1.0 Introduction .................................................. 1

2.0 Background and Motivation .................................. 5
   2.1 Motivating Example ....................................... 7

3.0 Methods .................................................... 9
   3.1 Forward Selection ...................................... 9
   3.2 Forward Stability Selection ............................ 10
   3.3 Model Path Selection .................................. 11

4.0 Graphical Visualization .................................... 14

5.0 Computational Efficiency ................................... 16
   5.1 High-Dimensional Multinomial Testing ............... 17

6.0 Simulations ................................................ 20

7.0 Applications ............................................... 23
   7.1 Application to Public Movie Ratings .................. 23
   7.2 Application to Parkinson’s Classification .......... 27
   7.3 Application to Riboflavin Production Rate .......... 29

8.0 Conclusion ................................................ 32

Appendix ....................................................... 33

Bibliography .................................................. 35
List of Tables

A1 Description of the covariates in the movie dataset. 33
A2 Description of the covariates in the Parkinson’s dataset. 34
List of Figures

4.1 Example graphs of model path selection. .......................... 15
5.1 Power curves of various tests against a sparse alternative hypothesis 18
6.1 Test error ratio of different model selection techniques .................. 21
7.1 Repeated 10-fold cross validation of forward selection mean squared errors. 24
7.2 Variable paths found by model path selection for the movie dataset ........ 25
7.3 Intermediary steps of model path selection for the movie dataset ........... 26
7.4 Paths for each of the procedures described for the Parkinson’s dataset .... 28
7.5 Repeated 5-fold cross validation with respect to mean squared error .......... 30
7.6 Model path selection on gene dataset up to a depth of 9. .................. 31
1.0 Introduction

With the ever-increasing size and availability of data, efficient variable and model selection methods capable of efficiently sorting through the data to uncover meaningful relationships remain among the most important statistical tools. Given data $Z_1, \ldots, Z_n \sim F_Z$ that comes in the form of ordered pairs $Z = (X, Y)$ of a vector of covariates $X = (X_1, \ldots, X_p)$ and a response, it’s common to imagine a generic relationship of the form

$$Y = f(X) + \epsilon$$

where $\epsilon$ is well-behaved noise. Suppose that we have some list of candidate models defined as

$$\mathcal{M}(Z) = \bigcup_{\omega \in \Omega} M_\omega(Z)$$

where all candidate models are organized through some index $\omega \in \Omega$. In this framework, model selection procedures are designed to select some $\hat{\omega}$ such that the corresponding model $M_{\hat{\omega}}(Z)$ provides a good approximation or representation of $f(X)$. The quality of each available model in $\mathcal{M}(Z)$ can be measured via some loss function $\mathcal{L}$ so that

$$\hat{\omega} = \arg\min_{\omega \in \Omega} \mathcal{L}(M_\omega(Z)).$$

Therefore, in performing model selection, we ultimately select one final “best” model. Using this framework, vast amounts of literature have investigated the properties of various model selection procedures, focusing on whether or not it is possible to identify the true best model under various assumptions (Zhao and Yu, 2006; Bach, 2008; Zhang, 2009).

It is common to consider the set of all possible models, but in applied research areas it is often the case where researchers wish to impose some model size. Put more explicitly, the list of candidate models that they consider is some set of models

$$\mathcal{M}^d(Z) = \bigcup_{\omega \in \Omega} M^d_\omega(Z)$$
in which all models are of size $d$. It, however, is still the case that model selection seeks to find the best model among the candidate models. We choose to think about model selection from this perspective.

An important and practical consideration that seems to have received less attention is the inherent variability and uncertainty associated with choosing a model. Among the most popular approaches to addressing this issue is stability selection from Meinshausen and Buhlmann (2010). Here the authors envision a setting in which the original collection of covariates $\{X_1, ..., X_p\}$ can be partitioned into subsets $S$ and $N$ of signal and noise covariates, respectively, where $|S| = s < p$. In many cases, a linear model of the form

$$Y = X\beta + \epsilon$$

is assumed so that $S = \{X_k : \beta_k \neq 0\}$ and $N = \{X_k : \beta_k = 0\}$. The general procedure, discussed in more detail in future chapters, involves taking $B$ resamples of the original data, and fitting a collection of models on each for some large grid of possible regularization parameters $\lambda \in \Lambda$ that control model size. Each $\lambda$ induces some collection of selected variables $\hat{S}_i(\lambda) \subseteq \{X_1, ..., X_p\}$ on each resample $Z_i^*$ and for a particular choice of $\lambda$, a final estimate of the signal features $\hat{S}_{stable}$ is produced consisting of variables that appeared in a large fraction of the models.

In a regression context however, stability selection is designed with the explicit purpose of recovering the true set $S$ of signal variables, not necessarily obtaining the best model in some original collection $\mathcal{M}^d(Z)$ – we define this alternative approach as variable selection. From a practical standpoint, this is evidenced in numerous biological applications (Alexander and Lange, 2011; He and Yu, 2011; He and Lin, 2011). It also remains quite common though that scientists and applied researchers are specifically interested in finding easily-interpretable, low-dimensional models that retain some degree of predictive accuracy. A situation we’ve encountered in the past, for example, involved attempting to find a straightforward model to help identify individuals in the U.S. Armed Forces at elevated risk for injury. Although hundreds of covariates were available, because the final model could potentially be utilized in the field, it needed to contain only a very small number of variables and those variables needed to be easily measurable without advanced equipment.
In situations such as these, it is very likely that the true number signal variables \( s \) far exceeds our desired model size \( d \). Thus, while stability selection may be helpful for reducing the number variables that should be considered from \( p \) to \(|\hat{S}_{\text{stable}}|\), in order to find the best collection of \( d \) variables to include in our model, we must take an \textit{ad hoc} approach such as choosing the \( d \) variables selected most often or reverting back to methods like lasso or forward selection. We thus develop a two step model selection process thereby increasing the variability of model selection.

Our first and more minor contribution in this work is to formalize and demonstrate the benefits of embedding a stability framework within model selection procedures that are more easily tailored to produce models of a given form. We focus in particular on forward selection and refer to this modified procedure naturally as \textit{forward stability selection}.

While the particular model selected via this stabilized process may be preferable to that obtained by direct minimization of the loss function, there remains a great deal of variability in this process that is concealed by supplying only a single empirically optimal model for each element of \( \mathcal{M}^d(Z) \). It is from this perspective that we seek to make a potentially more impactful contribution. In lieu of a single model estimate for a model of a given size, we propose a novel procedure referred to as \textit{model path selection} designed to provide a set of models \( \tilde{\mathcal{M}}^d(Z) \subseteq \mathcal{M}^d(Z) \) with similar predictive accuracy. In much the same spirit as stability selection, rather than propose a particular selection method or robust model class, both the forward stability selection and model path selection procedures proposed here can act as a wrapper for user-defined model classes and loss functions. We focus the development of these ideas around a general forward selection scheme, though we stress up front that analogous tools can be designed to work within other general selection strategies.

Lastly, we will note that much work has also been conducted outside this general framework, instead opting to operate in the realm of artificial variables. To our knowledge, the oldest such article to investigate this was written by Wu et al. (2007), in which the authors compare the covariates composing \( X \) to some other large vector of randomly generated covariates. Similarly, Barber and Candes (2015) also takes the artificial variable route, but changes the procedure by which artificial variables are generated. In a much more strict environment, knockoff copies of covariates composing \( X \) are produced so to allow for ad-
vantageous properties when comparing covariates composing $X$ with their knockoff copies. Both of these works are well-used for their ability to limit the false discovery rate, which is less sought after in the model selection framework developed in this paper. These artificial variable methods share the same goal as stability selection, in that they aim to find all covariates in the set $S$ rather than attempt to find some optimal model.

The remainder of this paper is laid out as follows. In Chapter 2, we provide relevant background information on previous works and more detailed motivating examples for our methods. Then in Chapter 3, we provide the technical details of our procedure. One key benefit to model path selection is the easily interpretable graphical displays that summarize the model-fitting process. These graphical details are discussed in Chapter 4. In high-dimensional settings, these procedures can become computationally burdensome and we therefore introduce multiple ideas in Chapter 5 that greatly improve the computational efficiency. In Chapter 6, we provide a number of simulations to demonstrate the utility of our method before concluding with several in-depth walkthroughs of our procedure on real data in Chapter 7.
2.0 Background and Motivation

Assume we have vector-valued data composed of random elements $Z_1, \ldots, Z_n \sim F_Z$ such that $Z = (X, Y)$, where $X = (X_1, \ldots, X_p)$ is a vector of covariates, and $Y$ is the response. For $s < p$, let $X^*$ be a $s$-dimensional vector composed of covariates from $X$. Also, let $\epsilon^*$ be well-behaved noise. In a variable selection framework, we assume that the true relationship between $X$ and $Y$ is sparse, meaning that $Y$ does not depend on all covariates in $X$, rather it only depends on the covariates composing $X^*$. Therefore, we assume there exists a function $f^*$ that takes the form

$$Y = f^*(X^*) + \epsilon^*$$

In performing variable selection, we seek to identify the set of covariates composing $X^*$, which we denote with $S$. Additionally, all covariates comprising $X$ that are not included in $X^*$ are noise covariates – the set of which is denoted by $N$. Generically, variable selection procedures depend on some regularization (tuning) parameter $\lambda \in \Lambda$ – where $\Lambda$ is the regularization region – that control the size of the model built so to produce a good estimator $\hat{S}(\lambda)$ of $S$.

Suppose the current modeling environment is such that typical variable selection techniques are incredibly unstable – that is, for $B$ resamples of equal size $Z^*_1, \ldots, Z^*_B$, variable selection yields models with drastically different selected covariates, $|\cap_{i=1}^B \hat{S}_i(\lambda)| \ll s$. This can be satisfied by a number of different set-ups, most notably $p \gg n$ and $p \gg s$. Stability selection defined by Meinshausen and Buhlmann (2010) is among a small number of methods that can handle these highly unstable set-ups. Stability selection aims to generate a more stable estimate $\hat{S}_{\text{stable}}$ of $S$ by aggregating results from many estimates of $S$.

Stability selection operates as follows. First, we draw $B$ resamples $Z^*_1, \ldots, Z^*_B$, then on each resample $Z^*_i$ a collection of models $\{M^\lambda(Z^*_i) : \lambda \in \Lambda'\}$ are fit across a grid of regularization parameters $\Lambda' \subseteq \Lambda$. By generating a collection of models $\{M^\lambda(Z^*_i) : \lambda \in \Lambda'\}$, we also generate a collection of selected covariate sets $\{\hat{S}_i(\lambda) : \lambda \in \Lambda'\}$. Then for each $\lambda$ we calculate the selection proportions for all covariates $X_1, \ldots, X_p$.

**Definition 1. (Selection proportion SS)** Among $B$ total resamples, the selection propor-
tion for covariate $X_j$ is defined as

$$\hat{\theta}_j(\lambda) = \frac{1}{B} \sum_{i=1}^{B} 1\{X_j \in \hat{S}_i(\lambda)\}$$

After calculating the selection proportions for all covariates and regularizations, we must choose an appropriate $\lambda$ and threshold $\pi_{thr} \in [0, 1]$ to determine our final set of selected covariates $\hat{S}^{stable}$. Upon selecting a $\lambda$ and $\pi_{thr}$, we can generate a set of selected covariates, which is defined as

$$\hat{S}^{stable} = \{X_j : \hat{\theta}_j(\lambda) > \pi_{thr}\}$$

for all $j \in \{1, \ldots, p\}$. However, choosing these values $\lambda$ and $\pi_{thr}$ is non-trivial. The authors recommend using the following theorem,

**Theorem 1.** Assume the base selection method is better than random guessing. For all $i \in \{1, \ldots, B\}$ and $\lambda \in \Lambda' \subseteq \Lambda$ assume that the distribution of $1\{X_j \in \hat{S}_i(\lambda)\}$ for all $X_j \in N$ is exchangable, the expected number of falsely selected covariates $V$ is bounded by

$$\mathbb{E}(V) \leq \frac{1}{2\pi_{thr} - 1} \frac{q_{\Lambda'}}{p}$$

where $q_{\Lambda'}$ is the mean number of selected covariates, $q_{\Lambda'} = \mathbb{E}(|\cup_{\lambda \in \Lambda'} \hat{S}_i(\lambda)|)$.

We note that $\pi_{thr}$ can be thought of as a tuning parameter of which the results do not vary much within the range $(0.6, 0.9)$. After specifying the threshold, a regularization region $\Lambda' \subseteq \Lambda$ that satisfies the exchangeability assumption and some desired error control can be found. Then, we can choose any $\lambda \in \Lambda'$. Alternatively, one could estimate $q_{\Lambda'}$ and solve for $\pi_{thr}$, but this approach is much less common, as providing a threshold is more intuitive than $q_{\Lambda'}$. Finally, after specifying these parameters, all covariates satisfying $\hat{\theta}_j(\lambda) > \pi_{thr}$ are chosen by stability selection and together create the set $\hat{S}^{stable}$. Additional work has been done to soften the exchangeability assumption and make the selection of $\pi_{thr}$ and $\lambda$ more straight-forward, but the extensions and reformulations still very much follow the same line of thinking (Shah and Samworth, 2012; Beinrucker et al., 2016; Hofner et al., 2015).
2.1 Motivating Example

We wish to develop a scenario so to demonstrate the limitations of stability selection and motivate our work. Suppose our regression function consists of real-valued variables of the form

$$\mathbb{E}[Y|X] = \alpha \sum_{i=1}^{s} X_i + 0 \cdot \sum_{j=s+1}^{p} X_j$$

for some $\alpha \neq 0$ and we wish to estimate a linear model of size $d_{\text{max}} < s$. Depending on the correlation structure of the predictors and the form of the error $\epsilon$ associated with the data generating process, there may be many linear models with $d_{\text{max}}$ terms that model the signal equally well. For example, let the covariates be independent and identically distributed (and independent of $\epsilon$) so that $\text{cov}(X) = \sigma^2 I_p$, thus there are $\binom{s}{d_{\text{max}}}$ equally accurate models with respect to any reasonable loss function.

If we were to employ stability selection, we would obtain an estimate $\hat{S}^{\text{stable}}$ that estimates the entire set $S = \{X_1, \ldots, X_s\}$, not some subset of size $d_{\text{max}}$. For stability selection, we have no direct method of controlling the size of its selected set – given the assumptions of Theorem 1 with some fixed $\mathbb{E}(V)$, it is difficult to change the parameters so to guarantee a reduction in $|\hat{S}^{\text{stable}}|$. This sentiment is even reiterated by the authors in their writing that the results of stability selection do not strongly depend on the choice of regularization $\lambda$ or the regularization region $\Lambda'$. Additionally, they note that results are often similar for $\pi_{\text{thr}} \in (0.6, 0.9)$ (Meinshausen and Buhlmann, 2010). Therefore, using the error control theorem, there is no direct method of obtaining an estimate of size $d_{\text{max}}$.

To rebut this point, one could argue that we can then use $\hat{S}^{\text{stable}}$ to form a new predictor space in which we employ some other model selection procedure so to achieve a final estimate of size $d_{\text{max}}$ if $d_{\text{max}} < |\hat{S}^{\text{stable}}|$. However, given this example, all possible $\binom{s}{d_{\text{max}}}$ models should be equally accurate, thus our secondary model selection step would likely be highly unstable. Moreover, whatever result is obtained from such a procedure would be highly conditional on the data, meaning that given some other initial sample, model selection may result in a different set of $d_{\text{max}}$ predictors. This illustrates a source of instability that is unable to be addressed by stability selection – upon finding our estimate $\hat{S}^{\text{stable}}$, selecting a model of size
$d_{\text{max}} < |\hat{S}_{\text{stable}}|$ is a high variance procedure. In this paper, we therefore seek to produce a new formulation of stability selection that allows for a straight-forward approximation of a selection set of restricted size.

Furthermore, this example also demonstrates a novel concept missing from the model selection literature. All selection methods produce one model, which is an element of our set of candidate models. By extension, traditional model selection procedures can be seen as providing the equivalent of a point estimate but without a natural measure of uncertainty – we may obtain a “best” model, but we have no way of knowing how many other models of equal size are equivalently accurate. Even stability selection, which acknowledges the instability of model selection and aggregates many model estimates only produces a singular set of variables. This basic sentiment motivates our larger contribution: instead of producing a singular element, we seek to produce an entire set $\tilde{M}^d(Z) \subseteq M^d(Z)$ of model estimates so to uncover a structure for representing the variability associated with selection procedures.
3.0 Methods

We again assume we have vector-valued data composed of random elements $Z_1, \ldots, Z_n \sim F_Z$ such that $Z = (X,Y)$, where $X = (X_1, \ldots, X_p)$ is a vector of covariates, and $Y$ is the response. Also, let $\epsilon$ be well-behaved noise. Then we assume data is of the form

$$Y = f(X) + \epsilon$$

Generically, selection procedures depend on some regularization parameter $\lambda \in \Lambda$ – where $\Lambda$ is the regularization region – that control the size of the model built so to produce a good estimator $\hat{S}(\lambda)$ of $S$, where $S$ is the set of signal covariates.

The basic structure of stability selection is of substantial interest with regard to our following contributions. It is the combination of selection methods with resampling procedures that is of particular interest, as their synthesis results in substantially more stable selection procedures. We therefore take the basic formulation put forth in Meinshausen and Buhlmann (2010) to develop new stable selection methods. Our procedures, forward stability selection and model path selection, are defined generally so to afford them generality with respect to loss and model type – much like stability selection, our contributions act as wrapper methods but for existing model types and loss functions.

3.1 Forward Selection

As mentioned in Chapter 1, traditional model selection aims to find some $M^d(Z)$ that minimizes the loss for a model of size $d$. In model selection, the set $S^{(d)}$ is defined as the set of variables comprising $M^d(Z)$, which is not necessarily the set of signal covariates $S$ – note that this is a departure from our stability selection formulation, as we are no longer interested in $S$.

To carry out model selection, a best subset selection could be performed in which all models of size $d$ are built and the model yielding the minimal loss is selected, however, as
$p$ increases this becomes computationally expensive. Instead, we can decrease computation time by performing a forward selection. Traditionally, forward selection is a process in which covariates are sequentially added to a model until the desired number of covariates is reached. In this process, each covariate is specifically chosen so to minimize the loss of the model.

Generically, in step $k$ of forward selection, we define $\mathcal{X} = \{X_1, \ldots, X_p\}$ and $A = \mathcal{X} \setminus \hat{S}^{(k-1)}$. Next, for any $X_j \in A$, we define $C$ as the vector of all covariates in $\hat{S}^{(k-1)} \cup X_j$. Then, the covariate $X_i$ selected in step $k$ of forward selection has index

$$
  i = \arg\min_{j: X_j \in A} \mathcal{L}(\hat{f}(C), Y)
$$

meaning $\hat{S}^{(k)} = \hat{S}^{(k-1)} \cup X_i$. Established in this way, we define $\hat{S}^{(0)} = \emptyset$. This process of estimating some $\hat{S}^{(k)}$ iteratively continues until $k = d$.

### 3.2 Forward Stability Selection

To modify forward selection to handle unstable data, we need only add a resampling procedure. Therefore, within step (iterate) $k$ of forward selection – where with stable data we’d seek to find $\hat{S}^{(k)}$ – we generate $B$ resamples of equal size $Z_1^*, \ldots, Z_B^*$ and generate estimates $\hat{S}_1^{(k)}, \ldots, \hat{S}_B^{(k)}$ by performing one step of forward selection on each resample. Then, we calculate selection proportions.

**Definition 2. (Selection Proportion FSS)** Denote $\mathcal{X} = \{X_1, \ldots, X_p\}$. Within step $k$ of forward selection, let $A = \mathcal{X} \setminus \hat{S}^{(k-1)}$. Among $B$ total resamples, the selection proportion for covariate $X_j \in A$ is

$$
  \hat{\theta}_j^{(k)} = \frac{1}{B} \sum_{i=1}^{B} \mathbb{1}\{X_j \in \hat{S}_i^{(k)}\}
$$

After calculating selection proportions for every $X_j \in A$, the covariate to be appended to $\hat{S}^{(k-1)}$, thus forming $\hat{S}^{(k)}$, must be determined. The selected covariate $X_i$ corresponds to index

$$
  i = \arg\max_{j: X_j \in A} \hat{\theta}_j^{(k)}.
$$
Therefore, the covariate that attains the highest selection proportion is selected to be an element of \( \hat{S}^{(k)} \). As with forward selection, this process continues in a recursive and forward fashion, thus finishing when the desired number of covariates \( d \) is reached. The method just defined is referred to as *forward stability selection*. Given how the stability selection is redefined, forward stability selection finds the best variable at each step of the forward procedure with respect to the loss function and model type specified (as is with forward selection), instead of finding all signal covariates across multiple resamples.

### 3.3 Model Path Selection

By redefining forward/stability selection this way, we open ourselves to a new question, “how much better is a selected variable than an unselected variable?” With stability selection, there is no such problem since the authors provide some information regarding how \( \pi_{thr} \) can be used to limit false discoveries. When assessing the selection proportions calculated in forward stability selection, its natural to note the insignificant difference between two closely valued proportions. We use this line of thinking to generate a new branching forward selection procedure which allows for alternative variable paths. We wish take the same forward stability selection procedure just referenced, except now we will assess whether or not other variables have similar selection proportions at each step.

Before testing proportions in our specific context, we must first observe a framework in which we can test multiple proportions simultaneously. To do so, we consider the multinomial distribution.

**Theorem 2.** Suppose we have \( p \) categories and the probability of an item belonging to category \( j \) is \( \theta_j \). Also, suppose that we take a sample of size \( n \) and let \( x_j \) represent the number of items that belong to each category. We model this data as a multinomial \( \Theta(\theta_1, \ldots, \theta_p) \) with probability distribution function,

\[
g(x_1, \ldots, x_p) = \frac{n!}{x_1! \ldots x_p!} \theta_1^{x_1} \ldots \theta_p^{x_p}
\]
where \( x_i \in \{0, \ldots, n\}, i \in \{1, \ldots, p\}, \sum_i x_i = n, \) and \( \theta_i \in [0, 1], \sum_i \theta_i = 1. \) For \( n \) sufficiently large and \( \min_i \{n\theta_i\} \geq 10, \) then with \( q \leq p \)

\[
T_{\chi^2} = \sum_{k=1}^{q} \frac{(n\hat{\theta}_k - n\theta_k)^2}{n\theta_k} \rightarrow \chi^2_q
\]

except if \( q = p, \) then \( T_{\chi^2} \rightarrow \chi^2_{p-1}. \)

This theorem can be used to set up a hypothesis test for equality of proportions as compared to a fixed multinomial \( \Theta_0(\eta_1, \ldots, \eta_p). \) For a set of \( q \leq p \) proportions, such a hypothesis test is of the form

\[
H_0 : \theta_1 = \eta_1, \theta_2 = \eta_2, \ldots, \theta_q = \eta_q
\]

\[
H_1 : \theta_i \neq \eta_i \text{ for some } i \in \{1, \ldots, q\}.
\]

Note that if the asymptotic conditions are not met, an empirical test can be performed with relatively high power – more on this in Chapter 5.

At each step of the forward stability selection procedure, we assess the empirical selection proportions and determine which proportion is the highest. In doing this, we assume that the variable with the highest empirical selection proportion is the most appropriate variable to be added at each step of the procedure; however, with the tool just referenced, we can now test whether any set of true selection proportions are equal. Within step \( k \) of forward stability selection suppose there are \( r \leq p \) available covariates – thus \( r \) selection proportions – and let \( \{\hat{\theta}^{(k)}_1, \ldots, \hat{\theta}^{(k)}_r\} \) be order statistics such that \( \hat{\theta}^{(k)}_1 > \cdots > \hat{\theta}^{(k)}_r, \) then the test is defined with the hypotheses

\[
H_0 : \hat{\theta}^{(k)}_1 = \hat{\theta}^{(k)}_2 = \cdots = \hat{\theta}^{(k)}_r
\]

\[
H_1 : \text{at least one } \hat{\theta}^{(k)}_i \text{ not equal for } i \in \{1, 2, \ldots, r\}
\]

where under the null, \( \hat{\theta}^{(k)}_i = \frac{1}{r} \sum_{j=1}^{r} \hat{\theta}^{(k)}_{(j)} \) for all \( i \in \{1, \ldots, r\}. \) Using these hypotheses, the set of predictively equivalent covariates is defined as follows.

**Definition 3.** Let \( \{\hat{\theta}^{(k)}_1, \ldots, \hat{\theta}^{(k)}_r\} \) be order statistics calculated in step \( k \) of forward stability selection. Given the largest \( u \leq r \) such that the null hypothesis

\[
H_0 : \hat{\theta}^{(k)}_1 = \hat{\theta}^{(k)}_2 = \cdots = \hat{\theta}^{(k)}_u
\]
is not rejected, the set of predictively equivalent covariates only contains the covariates corresponding to order statistics \( \{ \hat{\theta}^{(k)}_{(1)}, \ldots, \hat{\theta}^{(k)}_{(u)} \} \).

With this information, we can now continue forward stability selection with any predictor from the set of predictively equivalent covariates.

Our ultimate method, *model path selection*, is defined as the process by which all equivalent covariates are each individually added to the model being built by forward stability selection, thereby creating splits in the original forward stability selection procedure. After these splits are made within the procedure, we can continue selection on multiple different sets of selected covariates. Then, future steps of the forward procedure will then draw upon these sets to produce even larger variable sets, perhaps branching out even further. These variable sets, or variable paths, increase in both breadth (number of paths) and depth (number of covariates) until the maximum depth is reached on all paths. We organize these variable paths as graphs, which are further discussed in Chapter 4.

In crafting this procedure, we develop an idea that is often ignored in model selection literature – that being the possibility of selecting numerous “best” models. Instead of generating a single model estimate, we produce a set of model estimates \( \tilde{M}^d(Z) \subseteq M^d(Z) \) that all have similar predictive accuracy. In doing so, we uncover the inherent instability of performing model selection in specific modeling environments – that is, those which produce unstable model estimates. This fact can be summarily seen in the number of variable sets produced by a model path selection; one with many paths suggests that model selection is generally unstable, while one with few paths would suggest traditional model selection may still be acceptably stable. Furthermore, model path selection allows us to see the depth (or model size) at which model selection becomes unstable – perhaps the number of paths explodes at some depth \( d^* \leq d \).
4.0 Graphical Visualization

Model path selection is best displayed as a graph or collection of graphs, wherein the depth describes the iteration at which a covariate was added to the final model and sibling nodes are deemed proportionally equivalent in Definition 3. The entire group of branching diagrams found by model path selection is called a model path selection set. Additionally, the diagrams are left justified, meaning the leftmost covariate is the most stable; that is, it is the covariate with the highest selection proportion. All sibling nodes to the right of the leftmost node are organized by decreasing proportion even though we determined all of the corresponding covariates were equivalent with respect to selection proportion. We do this for a couple of reasons: (1) if the number of resamples is increased sufficiently, we may see some of the rightmost paths disappear; (2) if the number of resamples is sufficiently small and model path selection is performed again, the rightmost paths may swap places. Due to both scenarios seeming rather likely, we believe that knowing the relative proportions within each equivalent group is valuable information.

An example model path selection set is shown in Figure 4.1; it has a total of 10 variable paths spread across two separate graphs. First, a single forward stability selection iterate was performed against an empty model and two covariates, X10 and X21, were selected as proportionally equivalent. Because X10 and X21 were chosen at a depth of 1, they become the top-most node of two separate graphs. At a depth of 2, X32 and X33 were determined to be equivalent with respect to selection proportion when being added to a model containing only X10; similarly, X24 and X15 were determined to be equivalent with respect to selection proportion when being added to a model containing only X21. This same pattern persists for all further depth nodes.

Additionally, this plot can easily be adjusted to suit a given research scenario. Suppose, each covariate has an associated price of collection. Researchers could then apply some sort of color gradient associated with price to easily display the cost of collecting all covariates in each path—thus providing them with the information as to which paths are the least costly. A similar idea could be applied to missingness, which is also particularly relevant in many
applied research areas. In Chapter 7, we will look at an example in which missingness is a consideration. Lastly, for particularly error-conscious researchers, this idea could be applied to the error of each path.
5.0 Computational Efficiency

In comparison with other selection procedures, forward stability selection is incredibly computationally expensive. Because forward stability selection embeds stability selection within a forward selection framework, there are potentially many more computations needed to fulfill its variant of stability selection in comparison to the original stability selection procedure. This is true for stability selections in which the base selection method is not a stepwise selection method—however, if the base selection method is forward selection, then it can be easily shown that forward stability selection and stability selection are computationally equivalent for a given number of forward selection steps.

For a single fixed regularization, consider a some resample $Z_i^*$ within stability selection. In the traditional stability selection procedure in which the base selection method is the lasso, only one model is built, while forward stability selection builds as many models as there are available covariates. For a given depth $d$ the number of sub-models built within forward stability selection is

$$B \sum_{j=0}^{d-1} (p - j)$$

whereas stability selection only builds $B$ models for a given regularization. For model path selection, the number of models built is at least equal to the number of models built for forward stability selection.

To decrease computation time, the user may feel inclined to lower the value of $B$, but such a decision comes at the cost of accuracy. Too few resamples could result in covariates with relatively low true selection proportions being selected in forward stability selection. In the case of model path selection, too few resamples could result in large sets of predictively equivalent covariates, thus increasing the number of variable paths. We will therefore consider increasing computational efficiency in a scenario with fixed $B$. In order to do so, we wish to check whether the covariate with the highest true selection proportion has been selected more than 0 times after taking some $b < B$ resamples. If we can infer that the covariate with the true highest selection proportion is likely to have been selected more than
0 times, we can then eliminate all covariates with 0 selections (after \( b \) resamples) from the broader selection procedure. Then, for the remaining \( B - b \) resamples, we only consider the covariates that had been selected at least once during the first \( b \) resamples. In order for this procedure to effectively cut computation time, we need a high-power test so to limit the number of remaining candidate covariates to a minimum.

5.1 High-Dimensional Multinomial Testing

As just stated, our primary consideration is to find some generally high-powered test for proportions. To accomplish the task of carrying out such a procedure, we must draw from recent work in the analysis of high dimensional multinomials. The first such extensive review of the topic was done by Balakrishnan and Wasserman (2018) in which they highlight powerful hypothesis testing procedures for high dimensional multinomials. Here, high dimensional multinomials refers to a multinomials with many categories but very few counts—often, this is taken to mean that the number of categories is much larger than the sample size.

Recall the multinomial distribution defined in Theorem 2. For a multinomial \( \Theta(\theta_1, \ldots, \theta_p) \) we have hypotheses

\[
H_0 : \Theta = \Theta_0 \quad \text{versus} \quad H_1 : \Theta \neq \Theta_0
\]

where \( \Theta_0(\eta_1, \ldots, \eta_p) \) is a fixed multinomial with uniform proportions, meaning \( \eta_i = \frac{1}{p} \) for all \( i \in \{1, \ldots, p\} \). We consider multiple different test statistics and observe their power against the most relevant alternative, a sparse multinomial. In the context of multinomials, sparsity means some \( q < p \) proportions are much larger than the remaining \( p - q \) proportions. We look at the following test statistics against a sparse alternative distribution.

\[
T_{\chi^2} = \sum_{j=1}^{p} \frac{(n\hat{\theta}_j - n\theta_j)^2}{n\theta_j} \\
T_{\text{trunc}} = \sum_{j=1}^{p} \frac{(n\hat{\theta}_j - n\theta_j)^2 - n\hat{\theta}_j}{\max \{\theta_j, \frac{1}{p}\}} \\
T_{l_1} = \sum_{j=1}^{p} |n\hat{\theta}_j - n\theta_j| \\
T_{l_2} = \sum_{j=1}^{p} (n\hat{\theta}_j - n\theta_j)^2
\]

Power curves of each of these test statistics with \( p = 1000 \) are displayed in Figure 5.1. We see that sparse multinomials have reasonably high power on all test statistics even in
difficult scenarios, like when $s = n = 50$. As one may expect, as $s$ decreases, the power increases. These power curves show that under the most relevant alternative, the previously stated test statistics have surprisingly high power.

Therefore, after $b < B$ resamples, we can employ the above hypothesis test and be confident that we will have relatively high power. It should be noted that all of the hypothesis tests summarized in Figure 5.1 are performed using the empirical distribution of the null instead of asymptotic distributions. As mentioned previously, $T_{\chi^2}$ does approach a $\chi^2$ dis-

Figure 5.1: Power curves of various tests against a sparse alternative hypothesis with $s$ categories having large probabilities and $n$ counts.
tribution with a sufficiently large \( b \), but this is not necessary for our test as calculating a null empirical distribution is comparatively (to forward stability selection) computationally inexpensive.

We will now take the reader through a sketch as to why we believe that this test is appropriate. A worst case scenario in which we would be unable to differentiate any proportions from one another would be a scenario in which all proportions are equal, therefore this is naturally our null hypothesis. If we can reject this null hypothesis, that implies that there exists some \( \theta_i > \theta_j \) for \( i \neq j \). This necessarily implies that the true highest selection proportion \( \theta_h \) is greater than the true lowest selection proportion \( \theta_\ell \). Since we know that \( \theta_h > \theta_\ell \), we can confidently throw out all 0 proportioned covariates because it is unlikely that \( \hat{\theta}_h = 0 \).

For the hypothesis test’s \( \alpha \)-level, our recommendation does not guarantee that the covariate with \( \theta_h \) is correctly kept \((1 - \alpha) \times 100\% \) of the time. We can however control the rate at which we incorrectly discard \( \theta_h \) (denoted as \( \alpha_0 \)) through an adjustment of \( b \). Ultimately, we wish to for \( P(\hat{\theta}_h = 0) < \alpha_0 \) and since \( \hat{\theta}_h \) depends on \( b \), control \( \alpha_0 \) through \( b \). Trivially, the probability that the covariate corresponding to \( \theta_h \) has been selected 0 times after \( b \) resamples is

\[
\alpha_0 = (1 - \theta_h)^b.
\]

Using this expression, we can adjust the value of \( b \) to ensure the hypothesis test is not discarding the true highest selected covariate more than a desired \( \alpha \)-level. In practice, this greatly increases computation time using \( \hat{\theta}_{(1)} \) as an estimator of \( \theta_h \).
6.0 Simulations

Much of what’s been discussed thus far has been tailored towards the practical uses of model path selection and forward stability selection, but we will now focus on their performances using simulated data to gain insight on how well they construct accurate models. Considering forward stability selections namesake, we will be comparing forward stability selection and model path selection with stability selection. We take the basic construction of these simulations from Meinshausen and Buhlmann (2010); data are of the form $Y = X \beta + \epsilon$ and we study 3 different design matrix set-ups that were generated in the following ways.

1. Low correlation. All $p$ predictor variables are i.i.d. $N(0, 1)$.

2. Moderate correlation block structure with 10 blocks. We generate a total of $p$ predictor variables that follow a $N(0, \Sigma)$ distribution, where $\Sigma_{i,j} = 0$ for all pairs $(i, j)$ except if $\text{mod}_{10}i = \text{mod}_{10}j$, for which $\Sigma_{i,j} = 0.45$.

3. High correlation Toeplitz design. We generate a total of $p$ predictor variables that follow a $N(0, \Sigma)$ distribution, where $\Sigma_{i,j} = 0.90| i - j |$.

We first create sparse regression vectors by setting $\beta_k = 0$ for all $k = 1, \ldots, p$, except for a randomly chosen set of covariates $S$. The values of non-zero $\beta_k$ are chosen independently from $Unif(0, 1)$ and the noise vector $\epsilon$ is chosen i.i.d. $N(0, \sigma^2)$ such that $\sigma^2$ corresponds to some desired signal-to-noise ratio (SNR). We also set $|S| = s = 5$. The variant of stability selection used in these simulations utilizes forward as its base selection method. In order to compare forward stability selection against its computational equivalent, we force SS to only select a total of 5 covariates on each resample – this seems to be the most straight forward comparison.

We choose to use the test error as a measurement of each model selection technique’s ability to identify an accurate model. Often, researchers prefer to study a model selection technique’s ability to identify the set of signal variables, but given that the correlation structure of the third design is so complex, we believe it would be more informative to study
Figure 6.1: Test error ratio of different model selection techniques across a range of different SNR and $p = 1000$. The designs are in order from top to bottom.

the test error. In order to study the test error, during the data generation process we also generate a test dataset of equal size to our training dataset. We then consider a ratio of test errors

$$RTE = \frac{MSE_{method}}{MSE_{true}}$$

where $MSE_{method}$ is the mean squared error (MSE) of the model built using the covariates
found by each model selection method and $MSE_{true}$ is the MSE of the true model.

All setups are constructed such that we vary the $SNR$ from 0.25 to 8. Each selection algorithm is designed such that it only selects $s$ covariates, except model path selection which is built to a depth of $s$; thus, the number of paths is restricted only through the number of resamples $B$. In these simulations, forward stability selection, model path selection, and stability selection were all performed with $B = 500$ bootstrap samples. Also, note that for model path selection, only the best performing paths are summarized in the corresponding plots. Finally, each simulation is repeated about 250 times and then we plot the mean and standard error of the RTE calculated across all repetitions. Results are shown in Figure 6.1.

We see across all simulation settings that model path selection outperforms all other tested model and variable selection methods. This shows that some of the alternative paths identified by MPS are indeed more accurate than those found by forward stability selection – that is, there are paths beyond the first that yield a lower test error. It is possible that the success of model path selection could be attributed to its number of paths, however, in these simulation settings, the average number of paths are below 6 for all designs with low SNR. When we compare such a small number to the total number of possible paths $\binom{1000}{6}$, we see that model path selection does a good job at cutting down the set of possible 5 variable models to a relatively small set of candidates. Our plots also suggest that stability selection makes a considerable improvement upon forward selection, while forward stability selection appears to be equivalent to forward selection.
7.0 Applications

Forward stability selection and model path selection are both highly applicable to “real data” problems, thus a large portion of the remainder of this paper will demonstrate how to effectively use these tools. Two of the datasets used in this chapter were found at the UCI Machine Learning Repository (Dua and Graff, 2019).

7.1 Application to Public Movie Ratings

We apply the model path selection procedure to regression trees using the mean squared error as the loss function on a dataset downloaded from the UCI Machine Learning Repository. For \( n = 231 \) instances, there is a continuous response variable measuring public movie rating taken from the Internet Movie Database (IMDb) along with \( p = 12 \) covariates that specify movie production characteristics and public reception markers (Ahmed et al., 2015). Many websites, like IMDb, allow for public persons to voluntarily score/rate a movie on a scale ranging from 1 to 10. In the past, these movie scores were not taken very seriously as they were limited by a low number of participants, were completely voluntary, and were heavily vulnerable to ballot-stuffing. Now, many websites filter ratings such that singular users cannot seriously alter a movie’s score by repeated submissions, thus curbing the ballot stuffing issue. In addition, public movie ratings have garnered a larger audience – perhaps due to increased accessibility of internet streaming services – and therefore are becoming increasingly influential in both the film industry and the public. Furthermore, services providers like Netflix and Amazon (the parent company of IMDb) have vastly increased participation of movie and television viewer scoring by fluidly integrating rating systems with their applications – in fact, ratings given through Amazon’s internet streaming service are automatically submitted to IMDb. Since movie scoring is a popular topic with many potential relationships underlying the relevant covariates, we decided to apply model path selection to this dataset. Additional information describing the relevant covariates can be
Although movie ratings may not seem to be an unstable problem at first glance, the presence of numerous related covariates makes differentiating novel information between them rather difficult. This means that the variables themselves are not necessarily unstable, but rather their relationships destabilize selection algorithms. This example is used to demonstrate forward stability selection’s ability to find small stable models in the presence of highly related covariates. Finding such relationships without model path selection would require researchers to know the underlying relationships between the covariates.

Before model path selection was performed, the appropriate depth was calculated by way of a repeated 10-fold cross validation with 100 repeats. On each training fold, the covariates that attained the lowest loss on the training fold were added to the model in a forward fashion and the loss was calculated on the hold-out fold at each step in this forward procedure. The results for each fold were then averaged according to the number of covariates in the model, and the same was done for all 100 cross validations; the results are shown in Figure 7.1. This led to the determination that a model formed with 5 covariates would likely yield the lowest loss. After deciding on a 5 covariate model, the model path selection procedure was performed up to a depth of 5 with \( B = 1000 \) bootstrap samples. The resultant paths are shown in Figure 7.2, and intermediary steps are summarized in Figure 7.3.
Figure 7.2: Variable paths found by model path selection for the movie dataset. This is used to demonstrate the convenience of using model path selection in settings where some important variables have a high amount of missing values.

The intermediary steps of model path selection demonstrate its ability to merge paths. For example, in the middle right tree of Figure 7.3, Aggregate Followers is connected to 2 terminal nodes forming 2 variables paths, \{Likes, Dislikes, Aggregate Followers, Genre\} and \{Likes, Dislikes, Aggregate Followers, Budget\}, then in the next step of model path selection, these two paths merge to form \{Likes, Dislikes, Aggregate Followers, Genre, Budget\}. This occurs because model path selection found Budget as one of the next covariates to add to the path \{Likes, Dislikes, Aggregate Followers, Genre\}, and similarly the only covariate added to \{Likes, Dislikes, Aggregate Followers, Budget\} was Genre. Since the ordering of the previously selected covariates does not matter – \{Likes, Dislikes, Aggregate Followers, Genre, Budget\} is the same as \{Likes, Dislikes, Aggregate Followers, Budget, Genre\} – the paths are merged such that the more stable path is shown. Another example of merging paths can be observed the rightmost model path selection sets of the middle right and bottom trees in Figure 7.3.

From the results of the model path selection up to a depth of 5, we see that the paths
Figure 7.3: Intermediary steps of model path selection for the movie dataset.

indexed \{1,2,4,7\} (ordered left to right) located by model path selection contain the covariate listed as Aggregate Followers, which has a total of 35 missing values; these missing values were imputed with multivariate imputation by chained equations using regression trees. The remaining paths indexed by \{3,5,6,8\} do not contain this variable, but they otherwise contain similar covariates (with the exception of Path-6 which uniquely contains Comments). In practice, the large drawback to using a model constructed by paths \{1,2,4,7\} is the potential difficulty of effectively utilizing the covariate Aggregate Followers – whether that difficulty be collecting the data or the influence imputation has on model interpretations. However,
model path selection fixes this problem by offering alternative models to those including Aggregate Followers while still maintaining a similarly low loss value. Without model path selection, one would have to select alternative covariates to Aggregate Followers one-by-one and test their relative changes in loss to determine if other models are comparable.

### 7.2 Application to Parkinson’s Classification

Model path selection using forward stability selection can also be used in the context of classification problems. Also from the UCI Machine Learning Repository, this dataset consists of \( n = 195 \) observations and \( p = 23 \) covariates on a response class denoting whether or not patients have Parkinson’s disease (Little et al., 2007). Response variable, status, has 2 unbalanced classes composed of 147 observations with Parkinson’s disease and 48 without. For the purposes of this example, the class denoting observations with Parkinson’s disease will be considered the positive result (1), while the class denoting disease-free observations is the negative result (0). Information describing the covariates can be found in Table A2.

We chose to apply model path selection on this dataset using two different model types along with two different loss functions. First, the model path selection procedure was performed on classification trees using a misclassification rate, wherein false negatives were weighted 3 times higher than false positives – chosen in accordance with the imbalanced response variable.

Before we discuss the second loss function, we must redefine the way in which we produce a classification tree. To do so, we first build a regression tree such that the positive response class is given a value of (1) and negative response class is given value of (0), which therefore produces a model that outputs probabilities. After the regression tree is built, we apply a decision boundary which classifies all results above said boundary as (1) and all results equal to or below it as (0). For example, one may choose a decision boundary to be whatever probability yields the lowest misclassification rate on the training dataset, or in the case of balanced data, it could be fixed at 0.50. Now, suppose the decision boundary is defined as such: the minimal false positive rate such that the true positive rate is at least 0.80. This
particular decision boundary was created to help generate predictive models on data with an imbalanced response, and this dataset fits such a criterion with 75% positive response values and 25% negative values. After redefining the decision boundary, we then define the loss function as the false positive rate. These definitions of loss and decision boundary are supposed to yield a model with a minimal false positive rate conditioned on the true positive rate being at least 0.80. This second loss function was applied to both trees and logistic regressions, and the results with $B = 2000$ bootstrap samples are shown in Figure 7.4.

In Figure 7.4, the top yellow paths are for classification trees; the middle blue paths are for trees with our custom loss function; the bottom lavender paths are for logistic regressions with our custom loss function. This shows how paths can be unique with respect to the loss function and model type employed. As can be readily observed, the differing model types and loss functions yield completely unique paths. No path contained in one model path selection set is observed in either of the remaining sets. Therefore, using model path selection and forward stability selection in these three scenarios shows forward stability selection’s agnosticism to both loss and model type. Model path selection can generate different paths and whole new variable subsets depending these parameters – this is because

Figure 7.4: Paths for each of the procedures described for the Parkinson’s dataset.
forward stability selection is finding the most stable models with respect to the loss function, model type, and variable depth (number of desired covariates). Moreover, this shows the importances of correctly selecting the loss function and model type to suit the user’s needs.

### 7.3 Application to Riboflavin Production Rate

A rather notable example from the original stability selection paper regards the use of stability selection on a dataset describing the riboflavin production rate of a particular bacterium, *bacillus subtilis*. Downloaded from the *hd* package in R, this dataset contains a total of 4088 genes pertaining to the continuous response variable measuring the logarithmic riboflavin production rate of the bacteria species (Dezeure et al., 2015). In the original example, all but 6 signal covariates were permuted such that the permuted covariates maintained their dependence structure in order to demonstrate stability selection’s ability to locate true signal covariates among numerous noise covariates. It should be noted that the version of this dataset used in our analyses contains only $n = 71$ instances, while the version used by Meinshausen and Buhlmann (2010) is larger with 44 more instances.

We too were interested in how our methods would perform in this context, however, this has a considerably different interpretation than put forth in the original stability selection research article. Since we are interested in demonstrating model selection, not variable selection, we do not need to alter the dataset in any way to demonstrate our method. Instead, we ran the model path selection procedure on the entire original dataset to identify multiple predictive models. Stability selection undoubtedly fails to produce a usable result on the original dataset as no covariate gets selected overwhelmingly more often than others.

To find the appropriate depth for model path selection, 5-fold cross validations were performed 100 times and averaged in similar fashion to the repeated cross validation in Chapter 7.1, and the corresponding results are shown in Figure 7.5. Additionally, the results of a model path selection using $B = 5000$ bootstrap samples per iterate executed up to a depth of 9 are shown in Figure 7.6. It should be noted that throughout the model path selection procedure no paths converged, therefore the paths shown in Figure 7.6 are complete.
Figure 7.5: Repeated 5-fold cross validation with respect to mean squared error. Cross validations were performed at each variable depth and averaged according to those values.

for any variable depth – meaning that we can interpret the results for any covariate depth without ignoring paths that converge at future depths. Given the massive size of this dataset, the lack of paths (only 3 in total) is staggering.
Figure 7.6: Model path selection on gene dataset up to a depth of 9.
8.0 Conclusion

Forward stability selection and model path selection have shown themselves to be powerful exploratory modeling tools. Both methods are agnostic to the model type and loss function specified by the user; in this way, these methods conveniently act as wrapper algorithms for all structures. This however casts a large burden on the researcher to select an appropriate model type, loss function, and desired variable depth. Still, if the researcher selects all of these components correctly, model path selection appears to be relatively accurate in some of its selections.

Model path selection brings a new idea to the model selection literature – a method of characterizing the uncertainty associated with model selection procedures. This novel tool provides researchers with insight into whether or not the results of model selection are unstable. Furthermore, in the construction of model path selection, we explicitly address the fact that on some datasets there are likely many collections of variables that yield models with similar error. Among these multiple models, we showed that model path selection contains models which yield a lower error than the most relevant competing model selection methods.
## Appendix

Table A1: Description of the covariates in the movie dataset.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Missing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genre</td>
<td>19 levels of movie genres</td>
<td>0</td>
</tr>
<tr>
<td>Budget</td>
<td>Continuous value of total USD spent to produce and market movie</td>
<td>1</td>
</tr>
<tr>
<td>Number of Screens</td>
<td>Integer of US theaters showing movie</td>
<td>10</td>
</tr>
<tr>
<td>Sequel</td>
<td>Integer of movie’s sequence in series</td>
<td>0</td>
</tr>
<tr>
<td>Ratings</td>
<td>Continuous score from 1 to 10</td>
<td>0</td>
</tr>
<tr>
<td>Gross Income</td>
<td>Continuous value of USD made by movie</td>
<td>0</td>
</tr>
<tr>
<td>Aggregate Actor Followers</td>
<td>Continuous value of total followers of all actors in movie</td>
<td>35</td>
</tr>
<tr>
<td>Number of views</td>
<td>Integer of times viewed</td>
<td>0</td>
</tr>
<tr>
<td>Number of likes</td>
<td>Integer of total likes on social media platforms</td>
<td>0</td>
</tr>
<tr>
<td>Number of dislikes</td>
<td>Integer of total dislikes on social media platforms</td>
<td>0</td>
</tr>
<tr>
<td>Number of comments</td>
<td>Integer of total comments made on social media platforms</td>
<td>0</td>
</tr>
<tr>
<td>Sentiment Score</td>
<td>Integer score measuring viewer anticipation</td>
<td>0</td>
</tr>
<tr>
<td>Movie Name</td>
<td>Character value of movie title</td>
<td>0</td>
</tr>
</tbody>
</table>
Table A2: Description of the covariates in the Parkinson’s dataset.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>ASCII subject name and recording number</td>
</tr>
<tr>
<td>MDVP.Fo.Hz.</td>
<td>Average vocal fundamental frequency</td>
</tr>
<tr>
<td>MDVP.Fhi.Hz.</td>
<td>Maximum vocal fundamental frequency</td>
</tr>
<tr>
<td>MDVP.Flo.Hz.</td>
<td>Minimum vocal fundamental frequency</td>
</tr>
<tr>
<td>MDVP.Jitter...</td>
<td>Several measures of variation in fundamental frequency</td>
</tr>
<tr>
<td>MDVP.Jitter.Abs.</td>
<td></td>
</tr>
<tr>
<td>MDVP.RAP</td>
<td></td>
</tr>
<tr>
<td>MDVP.PPQ</td>
<td></td>
</tr>
<tr>
<td>Jitter.DDP</td>
<td></td>
</tr>
<tr>
<td>MDVP.Shimmer</td>
<td>Several measures of variation in amplitude</td>
</tr>
<tr>
<td>MDVP.Shimmer.db.</td>
<td></td>
</tr>
<tr>
<td>Shimmer.APQ3</td>
<td></td>
</tr>
<tr>
<td>Shimmer.APQ5</td>
<td></td>
</tr>
<tr>
<td>MDVP.APQ</td>
<td></td>
</tr>
<tr>
<td>Shimmer.DDA</td>
<td></td>
</tr>
<tr>
<td>NHR</td>
<td>Two measures of ratio of noise to tonal components in the voice</td>
</tr>
<tr>
<td>HNR</td>
<td></td>
</tr>
<tr>
<td>RPDE</td>
<td>Two nonlinear dynamical complexity measures</td>
</tr>
<tr>
<td>D2</td>
<td></td>
</tr>
<tr>
<td>DFA</td>
<td>Signal fractal scaling exponent</td>
</tr>
<tr>
<td>spread1</td>
<td></td>
</tr>
<tr>
<td>spread2</td>
<td>Three nonlinear measures of fundamental frequency variation</td>
</tr>
<tr>
<td>PPE</td>
<td></td>
</tr>
</tbody>
</table>


