Distance Covariance Optimization on a Stiefel Manifold for Fast Sequential Sufficient Dimension Reduction of n << p Datasets

Joshua White
University of Central Florida, Florida, USA

December 1, 2022

Abstract. Multiple sufficient dimension reduction (SDR) techniques have been recently proposed to retain the information associated with the variance of feature values. This variance is critical to regression analysis and prediction. However, several SDR methods suffer from one of the following side-effects: inability to handle large n efficiently, inability to handle problems where n < p, or important information needed for regression is not retained. This proposed method takes a hybrid approach between existing methods such as the MM Algorithm for distance covariance (DCOV) SDR and sequential sufficient dimension reduction (SSDR) to handle the aforementioned issues. In addition, Stiefel learning is leveraged to more efficiently solve for the optimal solution of the reduced subspace.

Keywords: Sufficient Dimension Reduction, MM Algorithm, Stiefel Manifold Optimization, Minimum Average Variance, Large p

1 Introduction

Regression models are an important appli
Sufficient Dimension Reduction (SDR) is an important concept in reducing the dimension of a dataset while retaining important characteristics of the dataset and features for regression [1]. This relatively new methodology, introduced in 1991, has been heavily researched and many theoretical papers have been published on the topic. However, there are several limitations on the theoretical approach of SDR.

SDR provides tremendous value to the applications of high dimensional regression models. Ensuring that the variance between features is maintained is of the upmost importance as the variance between these features provide important information in the regression model. Several papers were researched in determining the optimal method for a sparse data set with more features than observations but both dimensions are relatively large. The predictor of the regression model is also multidimensional.

One of the more common approaches of SDR is Sliced Inverse Regression [1]. This involves taking slices of the features and reducing the features in such a way that they are then invertible. From here traditional regression methods can be performed such as Least Squares to calculate the coefficients of the remaining variables. This, method can also be applied by finding the sufficient subspace such that $E(Y|X) = E(Y|P_S X)$. Thus the central mean subspace, $S_{E(Y|X)}$, is equal to the conditional subspace $S_{Y|X}$. This provides a method to reduce the dimensionality of X while maintaining the conditional relationship of X and Y for regression. However it is noted that other important properties, such as the variance is lost in this dimension reduction. While this does provide a sufficient method of removing redundant features but does have a significant time complexity, and is unable to handle cases in which $n < p$. In addition it has a requirement that features belong to the exact same vector space in order to be removed providing a high-barrier to remove features. Other methods have been introduced to be more flexible in dimension reduction, while retaining important information.

Another way to extract the subspace associated with the conditional moments, i.e. $E(Y^k|X) = E(Y^k|P_S X)$. Thus the characteristic function of the conditional describes the central mean subspaces $E(e^{itY}|X)$. This method lends itself to the minimum average variance estimator (MAVE) [2]. MAVE takes the gradient of $E(Y|X)$ and constructs a subspace based on this value. This approach estimates the central mean subspaces without requiring details on the distribution of X with a quadratic computation time. However the direction of the variance is lost for $E(Y|X)$ given that the projection is taken. Thus a refined MAVE (RMAVE) method [3], is used to recapture the direction of the variance of the conditional. They leverage a family of distributions to estimate the direction of this variance without adding any computational complexity.

With MAVE, for predictor variables X, response Y, information associated with $Y|X$ is retained while the dimension of X is reduced. This method relies on subsets of the family of the distributions of the model ($\mathcal{F}$) to characterize the central subspace, L_1 and L_2. $L_1(F_Y)$ is a subset of $\mathcal{F}$ in which the expected value of $f(Y)$ is finite and $L_2(F_Y)$ is a subset of $\mathcal{F}$ where the variance of $f(Y)$ is finite where the inner product of two functions in the family is the expected value of the product of the functions. There are 5 important theorems within the paper, each of which shows the assumptions that this ensemble model works under and the reasoning on why this ensemble model estimates the central subspace. This method is best for bounded functions, but can be used for unbounded functions.

MAVE is a flexible model that allows for prediction of discrete and continuous variables and reduces the dimensionality of the features without sacrificing the variance of the features allowing for a more accurate prediction. The convergence rate of this model quadratic but does not work well with high n, nor when $n < p$, and only handles $Y \in \mathbb{R}$.

Minimum Average Deviance Estimator (MADE) [4] is another SDR method that is closely related to that of MAVE, but instead of leveraging the average variance, the average deviance is calculated instead. This model leverages the canonical link function of the distribution of Y to calculate the average deviance leveraging the mean parameter to calculate the Hes-

sian and Jacobian of the deviance function in the optimal direction of the high-dimensional space. The optimal direction on a stiefel manifold to determine that optimal function that maximizes the deviance from the subspace. This overcomes the large n encumbrance, but requires a known distribution (and one that is in the exponential family) and is only intended for $Y \in \mathbb{R}$. Also does not implicitly handle $n < p$.

MM Algorithm for Distance Covariance (MMDCOV) [5] goes one step further to remove the requirement of a known distribution as established in MADE. Removes the known distribution assumption and can be applied to $Y \in \mathbb{R}^j$ for sufficiently small j. However does not implicitly handle $n < p$ and utilizing Newton's Method on the high-dimensional space is very costly.

None of the afforementioned methods are capable of handling cases of SDR where $n < p$ directly. Sequential SDR (SSDR) iterates through sections of the dataset to extract features of cardinality less than the number of observations for each section. Though SSDR is just a method of taking subsets of features itself and does not directly perform SDR. The dimension reduction occurs on each one of the sections where SIR is performed on each section. The reduced dataset is then stored to another dataset, as well as each other reduced section. This occurs multiple times until all sections are reduced. SSDR handles dimension reduction where $n << p$ utilizing Sliced Inverse Regression, which also only predicts $Y \in \mathbb{R}$.

Both MADE and MMDCOV calculate the direction of the multi-dimensional variance on a Stiefel manifold. And both methods calculate the value using an estimate of the direction on the stiefel manifold via creating kernel matrices and taking the Kroeneker product to estimate the direction on the stiefel manifold. However, there are more efficient ways to calculate the direction on the stiefel manifold directly using Stiefel Optimization [6]. This process creates $p \times d$ orthonormal matrices to create a hypersphere; the stiefel manifold. Instead of optimizing the goal function on the entire $\mathbb{R}^{p \times d}$ vector space, the optimization is only applied along the stiefel manifold. Since the observed dimension is higher than the true rank, this allows for a more efficient solving of the optimization problem.

This paper is structured in the following manner: Section 2 is the Approach. In this section, the implementation of a hybrid approach combining the features of MMDCOV, SSDR, and Stiefel Optimization. Section 3 is Analysis. In this section a dataset containing essays of students is used to produce a high-dimensional sparse data matrix in which the number of features is higher than the number of observations. The hybrid approach outlined in section 2 will then be utilized on this dataset. Section 4 is the Discussion. This section has a link to the code that was created, a summary of the results, and future steps to improve this method.

2 Approach

MAVE, MADE, and MMDCOV were researched methods for handling such a dataset but each of them had their own advantages, but each of them fell short of being applied to

such a dataset. Both MAVE and MADE had restrictions on the dimension of the regression predictor. Both of these had a requirement that the predictor be a one-dimensional vector. Additionally MAVE is not efficient in running on a dataset where n is adequately large, thus is mostly based in theory.

MADE can be ran on larger sized datasets, but there is a requirement that the distribution of Y is known and must belong to the exponential family. However, this provides limitations on the data sets that can be used and requires additional restrictions on the regression model and adds additional complexity in a high-dimensional space, which is why this method is only intended to be used to predict a vector.

MMDCOV overcomes the restrictions of MAVE and MADE by introducing an algorithmic approach for reducing the dimension of the dataset while maximizing the retained value of the DCOV. This approach allows for the prediction to be a multi-dimensional matrix and puts no restrictions on the distribution. This approach is the most flexible and can be applied on the aforementioned sparse data set. However, there are limitations on the method itself. First, the approach does not implicitly handle cases where n¡p. Second, MMDCOV relies on computing the maximum DCOV between variables using Newton's Method over the entire cartesian plane and then condensing that to a Stiefel manifold, which is very computationally expensive. Two methods can be implemented on MMDCOV to overcome these encumbrances and those methods are SSDR and Stiefel Optimization respectively.

SSDR provides the ability to handle regression models where $n << p$. It works by looping through sections of the dataset selecting a number of features less than the number of observations and then performing SIR on the section. On its own this SSDR using SIR is computationally expensive and runs into the same obstacles as MAVE and MADE. But, by replacing the substep of using SIR with using MMDCOV, all of the advantages of MMDCOV can be utilized and can now be extended to cases where $n << p$.

Stiefel Optimization provides exceptional efficiencies over optimizing over the entire cartesian plane. The goal function of MMDCOV is implemented using Stiefel Optimization to determine the optimal features that maximize the DCOV of the data set. These values are retained, and provide a much more efficient method of solving for this optimization. All together the hybrid approach of MMDCOV, SSDR, and Stiefel Optimization provide an excellent approach to reduce the dimensions of a large and sparse regression model where the number of features is greater than the number of observations. The methods defined in the papers have tremendous applied value and can be applied directly to the described data.

2.1 Goals

There are many obstacles in reducing the dimensions in a regression model and predicting a multidimensional response variable for a high-dimensional and sparse dataset. Like many dimension reduction models, the goal is two-fold. The first objective is to reduce the dimensions of the dataset to find only relevant features in determining the outcome via a regression

model. The second objective is to reduce the dimension to reduce the computation time to make predictions. It is important to ensure that a minimal amount of the information is lost in this dimension reduction.

But this dataset adds additional complications. This dataset is very sparse. Considering the fact that the features are count based occurrences of words being included in an essay, many of the observations have a value of 0. It is pivotal that the chosen method be able to handle such sparsity in making predictions. In addition, the features of the validation dataset can vary significantly from the features of the training dataset. Since words included in each essay are aggregated and counted as features within the model, these datasets can differ greatly and that needs to be incorporated in the model. Finally, the dataset contains significantly more features than observations. This inhibits taking the inverse of the dataset, and the low rank of the dataset in relation to the number of features needs to be accounted for in the model selection.

Time complexity is also a factor in the application of predictions on this dataset. This is the case for both the training of the model as well as the validation of the model. While there is no exact time barrier for reducing the and training the model for the training dataset, it does need to be completed in a reasonable amount of time. Many SDR models have high computational complexity, thus large values for n and/or p may result in exhaustive computation times. Once trained, the model should be able to be validated in near-real time. The primary purpose of the essays dataset is to provide students with instant results to their submitted essays. Thus the essays should be able to be incorporated and predicted with a sufficiently small computation time.

2.2 Implementation

Since MMDCOV requires $n > p$, for each segment, s, select a sequential subset of features where $p^{(s)} \leq (n-1)$ where the last segment contains the remaining features. Then perform MMDCOV on each $p^{(s)}$. The non-selected features are removed from the analyzed data set: p^*. This is repeated until $X^* \in S_{Y|X}$ is of full rank. DCOV is defined by the following equation below as defined by [5].

$$\mathcal{V}^2_{n,\varepsilon}(a, B) = \left(\frac{1}{n^2} \sum_{k,l=1}^{n} \left(a_{kl}\gamma - \varepsilon \log\left(1 + \frac{a_{kl}\gamma}{\varepsilon}\right) B_{kl} I(B_{kl} > 0) \right) \right) \tag{1}$$

The goal is to maximize the value: $max_\gamma \frac{1}{2} tr(\gamma' Q \gamma) + tr(\gamma' L)$. This is maximized when the minimum result of the negative value of this function is calculated utilizing the conjugate gradient solver from the MANOPT package [7]. The Hessian and Gradient are calculated from this goal function $g(\gamma)$. Utilizing the Stiefel Manifold Optimization a value for the unit matrix direction ξ is found. The optimal DCOV is calculated utilizing this direction and assigned for a new value of γ. This is continued until γ converges. The value of DCOV is

calculated via the line-search implementation of each iteration of γ.

$$g(\gamma) = -Re\left(\frac{1}{2}tr(\gamma' Q\gamma) + tr(\gamma' L)\right) \tag{2}$$

$$gradg(\gamma) = -(Q\gamma + L) \tag{3}$$

$$Hessg(\gamma)[\xi] = -Q\xi \tag{4}$$

During each iteration of SSDR and MMDCOV, the rank (k) of the subsection of X is calculated. The values with the k largest average values of γ are retained $S_{Y|X}$, where the values that are removed are stored to an indexed list of values $\bar{S}_{Y|X}$. The coefficient value of this regression can be calculated as such $\hat{\beta} = \Sigma^{-\frac{1}{2}}_{X^* \in S_{Y|X}}\hat{\gamma}$. Thus $\hat{Y} = \hat{\beta}' X^*$ where $\hat{Y}, \hat{\beta} \in \mathbb{R}^j$ for sufficiently small j.

Utilizing SSDR, the features (from $n << p$) are broken up into maximal sections of predictors ($n > p$) in sequential order. From here MMDCOV is performed, in which the subsection of features is taken and features that maximize the goal function are retained. The values that are retained are the top k features with the highest corresponding value for γ. SSDR is used to iterate through sections of of the data where $n << p$ leveraging MMDCOV in lieu of the SIR for the $n > p$ dimension reductions on each of the sections. This can be seen in Algorithm 1.

Algorithm 1 Hybrid Approach of MMDCOV, SSDR, and Steifil Optimization

procedure BIG SDR(X,Y)
  Center X, Y
  Initialize $S_{Y|X}$
  for each $section \in X$ where each section is of size $n-1$ or the remaining features **do**
    $X_{section} \leftarrow X[section]$
    for each $section \in X$ where each section is of size $n-1$ or the remaining features **do**
      Calculate Q, L from MMDCOV Algorithm 1 [5]
      Caluclate $k = rank(X_{section})$
      Create a $p \times d$ stiefel manifold
      Optimize γ using conjugate gradient with equations 1-4 using on the stiefel manifold
      $S_{Y|X}$ is the index of the k largest values of the row mean of $gamma$
    end for
  end for
  $X^* \leftarrow X \in S_{Y|X}$
  $\hat{\beta} \leftarrow \gamma\Sigma^{-\frac{1}{2}}_{X^*}$
  features $\leftarrow$ features $\in S_{Y|X}$ **return** $S_{Y|X}$, $\hat{\beta}$, features
end procedure

3 Analysis

3.1 Data Collection and Preparation

Kaggle has provided a dataset of student submitted essays and the corresponding scores for their essays [8]. This can be modeled as a regression model to predict essay scores of English Language Learners in grades 8th-12th. The purpose of this is to use Natural Language Processing to determine the scores on a computer for more instant feedback. The scores are graded from 1-5 for 6 categories: cohesion, syntax, vocabulary, phraseology, grammar, and conventions. Each dataset comprises of the raw text for each student with a student id. The training dataset also has each of the scores for that essay accompanying the raw text. This information can be used to accurately predict each of the scores of a students essay score in a time efficient manner.

To begin analysis, raw text is converted into a text data matrix. This results in a matrix of 3,911 students (rows) and 13,228 terms (columns). There are 6 different scores that need to be predicted, this will be done either by treating the prediction variable as a 6-dimensional vector for each student. There are six values that need to be predicted: cohesion, syntax, vocabulary, phraseology, grammar, and conventions (Y is multinomial). Each of these scores are a value from 1 to 5. The raw text cells were cleansed which entailed the removal of non-letters, de-stemming words, and removing stop words prior to putting the data in a data matrix. No other words were removed in an attempt to ensure minimal information is removed from the regression model for the prediction of the score.

3.2 Modelling

The hybrid approach of MMDCOV, SSDR, and Stiefel Optimization allow for an SDR approach to reduce any dataset from $n << p$ to p^* dimensions where $p^* < n$. It also results in a coefficient matrix that predicts a multi-dimensional predictor in which the reduced dataset that preserves the distance covariance amongst features. The labels are also retained so this can be utilized in the prediction model. The resulting dataset is $X^* \in S_{Y|X}$ such that $S_{Y|X}$ is the reduced subspace.

Some data manipulation needs to be conducted on the dataset to accurately predict the resulting essay scores. First, all features from the prediction matrix must be removed that are not in the subspace ($X_{pred} \notin S_{Y|X}$). Next, the index values of $X^* \cap X_{pred}$ where $X^*, X_{pred} \in S_{Y|X}$ should be removed from $\hat{\beta}$ so the prediction matrix can be calculated where $\hat{Y}_{pred} = \hat{\beta}' X^*$. For the essay scores, the stored centrality measurement of Y is then added back in and rounded to the nearest half-integer value (bounded by 1 and 5) to calculate the predicted score. Since the predictor variables is a six-dimensional categorical variable, Categorical Cross-Entropy is used as the loss function, and can be used to determine the model validity.

MMDCOV is able to handle SDR in a reasonable time complexity, albeit as currently structured, it is not optimal. The time complexity of MMDCOV is improved on the training

dataset by utilizing Stiefel Optimization. This is performed by implementing the MANOPT package [7]. The MMDCOV does estimate a Stiefel manifold for calculating the direction of the reduced dimensional hyperplane, but first requires optimization for the goal function via Newton's Method over the entire cartesian hyperplane. The Stiefel Optimization method and MANOPT package provide a more efficient way for calculating the optimal value to maximize the DCOV value.

3.3 Model Discussion

The number of rows is 3911 while the number of features is 13230 (n ¡¡ p). There are six values that need to be predicted: cohesion, syntax, vocabulary, phraseology, grammar, and conventions (Y is multinomial). Requirement to apply the regression to future data that will likely have different features, thus the variable selection cannot happen in a black-box (i.e. PCA). Computation time needs to be minimal to provide instant feedback. All values are count based and positive. Expected high correlation amongst different values. Text data is treated for this regression model similar to how image data is treated in regression for image detection. While the relationship between one word to another is not intrinsicly calculated, it is expected that the regression model will learn the relationship to one another and determine the corresponding weights of their relationships. The hybrid implementation handles all of this criteria, while the methods below have their limitations.

3.4 Methods

This approach is still a work in progress and still requires rigorous testing and procedural improvements. Next steps include partitioning the data into a test and training set and calculating the Categorical Cross-Entropy of this method versus competing methods. The dataset is very sparse, thus only methods that handle sparsity can be used to retain the predictive power of the regression model. In addition, traditional dimension reduction methods such as Lasso, Sparse PCA, and Laplacian Eigenmaps cannot be used due to predictors having a different number of p than the test data set. Thus a method where the labels of the dataset need to be retained over the dimension reduction. This greatly reduces the number of methods that could be used, however two comparative models will be introduced. Those methods are SSDR utilizing SIR and a random removal of k number of features and then utilizing linear regression. However, the later method may fail due to the sparsity of the dataset.

The purpose of this method is to introduce an applicative method for SDR that minimizes assumptions of the dataset to provide an accurate model for regression. In order to show this value many other datasets should be used to ensure validity. The datasets to be used will be both real-world and synthetic datasets with a focus on datasets that are explicitly intended for regression purposes. MADE, MAVE, SSDR with SIR, and SSDR with MMDCOV (but without Stiefel Optimization) will be used as points of comparison within these synthetic datasets. Additional research will be conducted to determine when certain methods can be used, and this hybrid method will be emphasized for examples of when it is the only method that can be used. Baselines for accuracy and time complexity will be measured empirically,

as well as mathematical conclusions will be determined.

3.5 Results

The computer this method was ran has an 11th Gen Intel Core Processor running at 3.00 GHz on a 64-bit Windows Operating System with 16.0 GB of RAM. There were no performance enhancements made to utilize the GPU, so those specs are irrelevant. While the entire method was unable to be fully implemented on this computer, several iterations were ran. Below is one such example.

```
 iter                   cost val     grad. norm
    0   -6.9324385300919502e+00 1.58518474e+01
    1   -1.8448611077950911e+01 8.86827430e+01
    2   -7.2888268047894087e+01 1.24105813e+03
    3   -7.2888268047894087e+01 1.24105813e+03
Last stepsize smaller than minimum allowed; options.minstepsize = 1e-10.
Total time is 283.539120 [s] (excludes statsfun)
```

Each iteration of solving for γ utilizing the conjugate gradient solver converged in nearly 4 iterations on average. The computation time exceeded 200 seconds for such an iteration on average. This implementation was significantly more effective that implementing MMDCOV without Stiefel Optimization as that process took approximately 4 hours to solve for the same step of γ.

4 Discussion

In lieu of an **appendix**, all of the code and instructions can be found here: `https://github.com/joshuaderekwhite/dcov-sdr-manifoldoptimization`. While this approach is still a work in progress there appears to be significant novelty to this solution. First, this appears to be one of the only methods to predict a multi-dimensional regression model in cases where $n < p$ while p changes. The implementation is slow, but produces a fairly rapid prediction. The accuracy still needs to be verified, but appears that there is a way to reduce such a high-dimensional dataset to a lower-dimensional subspace to predict a multi-dimensional value, while retaining the most pivotal information about its variance.

While this hybrid approach leverages several highly effective methods for SDR, there may still be areas of improvement. Currently MMDCOV requires a line-search method to determine the optimal value of γ in determining which features maximize the distance covariance. This may be able to be directly incorporated into the goal function. Further research needs to be conducted to see if this is possible. Additionally, SSDR is aptly named based on the process of sectioning of features sequentially, but this is not optimal if highly correlated features are not in the same section. There may be algorithmic or mathematical methods to section these features together rather than proceeding sequentially. Better yet, there may be approaches

to eliminate features before even beginning SSDR. Further research will be conducted on these approaches to improve the efficiency of this hybrid SDR approach.

References

[1] L. Li, R. Cook, and C. Nachtsheim, "Model-free variable selection," *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, vol. 67, pp. 285–299, Apr. 2005.

[2] Y. Xia, H. Tong, W. Li, and L. ZHU, *An adaptive estimation of dimension reduction space*, pp. 299–346. United States: World Scientific Publishing Co., Jan. 2009. Publisher Copyright: © 2002 Royal Statistical Society. Copyright: Copyright 2016 Elsevier B.V., All rights reserved.

[3] X. Yin and B. Li, "Sufficient dimension reduction based on an ensemble of minimum average variance estimators," *The Annals of Statistics*, vol. 39, no. 6, pp. 3392–3416, 2011.

[4] K. P. Adragni, "Minimum average deviance estimation for sufficient dimension reduction," *Journal of Statistical Computation and Simulation*, vol. 88, no. 3, pp. 411–431, 2018.

[5] R. Wu and X. Chen, "Mm algorithms for distance covariance based sufficient dimension reduction and sufficient variable selection," *Computational Statistics & Data Analysis*, vol. 155, p. 107089, 2021.

[6] X. Li, S. Chen, Z. Deng, Q. Qu, Z. Zhu, and A. Man-Cho So, "Weakly convex optimization over stiefel manifold using riemannian subgradient-type methods," *SIAM Journal on Optimization*, vol. 31, no. 3, pp. 1605–1634, 2021.

[7] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, "Manopt, a Matlab toolbox for optimization on manifolds," *Journal of Machine Learning Research*, vol. 15, no. 42, pp. 1455–1459, 2014.

[8] "Feedback prize - english language learning."