Sparse and reduced-rank family of generalized regressions with transformation from pca or autoencoder

Rodolphe Priam

Sparse and reduced-rank family of generalized regressions with transformation from pca or autoencoder

R. Priam *

January 10, 2023

Abstract

Linear regression is one of the most studied methods after descriptive statistics, and univariate tests because it aims at understanding a target variable as a function of explaining or predictive variables. The interest in pca regression is how to improve the estimation of the output by new algorithms reducing the design matrix, also relevant for generalized linear models. By an association of several criteria, a family of new objective functions is proposed and the results are compared with pca regression and regression.

1 Introduction

There exists alternative approaches to a glm within a neural network which remains nevertheless the more usual way to do currently. In all cases, a penalization or constraints may be added, as they are worth to try because regularizations improve the models. The three different cases of models are listed as follows.

- Direct: Classical model embedding a reduction
- Plugin: Reduction followed by glm via a consecutive minimization of their losses
- Combined: Reduction associated to glm by simultaneous minimization of a combination of their losses

These model may have also a penalty term added to the objective function for regularization. They are given in a more formal way in the table below.

Approach	Criterion
Direct	$\text{argmin}_w\ F(x;w) + \alpha R(w)$
Plugin	$\text{argmin}_w\ F(\text{argmin}_h G(x;h);w) + \alpha R(w)$
Combined	$\text{argmin}_{w,h}\ G(x;h) + \lambda\ F(h;w) + \alpha R(w)$

*rpriam@gmail.com.

Such models with a reduction or a variable selection are considered herein. For reduction, there exists two main families of methods, the one of type penalization which allows to reduce the columns of the design matrix by removing some useless ones, and the one of type pca which allows a prereduction to construct new columns before the pca. By associating both and with a simultaneous estimation, the final regression is expected to be even more relevant. The aim of this paper is to compare such algorithms with the state of art.

The plan of the paper is as follows. Section 1 is for the introducing the problematic. Section 2 is for presenting the generalized models with reduction from pca or autoencoder. Next sections are for the experiments and for the conclusion with the perspectives.

2 Models with reduction

In this section, two main cases are presented with the previous architecture just above. They are compared in a next section with experiments.

2.1 Prediction loss functions and autoencoder

The model for the basic autoencoder with one hidden layer is as follows:

$$f_\theta(\mathbf{x}_i) = g_x(\mathbf{W}_x^T g_h(\mathbf{W}_h\mathbf{x}_i, b_h), b_x)\,.$$

This is with two stages:

$$\mathbf{h}_i = \mathbf{h}(\mathbf{x}_i) = g_h(\mathbf{W}_h\mathbf{x}_i, b_h)$$

$$\hat{\mathbf{x}}_i = \hat{\mathbf{x}}(\mathbf{h}_i) = g_x(\mathbf{W}_x\mathbf{h}_i, b_x)\,.$$

This leads to the three architectures from autoencoder to regression, with same expression for the predictions,

$\hat{y}_i = \hat{\beta}^T \hat{\mathbf{h}}_i + \hat{\mathbf{b}}_y = \hat{\beta}^T g_h(\hat{\mathbf{W}}_h \mathbf{x}_i, \hat{\mathbf{b}}_h) + \hat{\mathbf{b}}_y$, but three different loss functions to optimize. In this case, the parameters estimated are $(\hat{\beta}, \hat{\mathbf{b}}_y, \hat{\mathbf{W}}_h, \hat{\mathbf{b}}_h)$ for the direct approach, $\hat{\beta}, \hat{\mathbf{b}}_y$ and $(\hat{\mathbf{W}}_x, \hat{\mathbf{W}}_h, \hat{\mathbf{b}}_h, \hat{\mathbf{b}}_x)$ for the plugin approach and $(\hat{\mathbf{W}}_x, \hat{\mathbf{W}}_h, \hat{\mathbf{b}}_h, \hat{\mathbf{b}}_x, \hat{\beta}, \hat{\mathbf{b}}_y)$ for the combined approach. Another "bias term" (or "intercept") may be added in some cases for x_i. The plugin and combined approaches count nearly each one two times the number of parameters of the direct approach because of the decoding step, hence they may ask for more data to train. They seem to be tested in the literature with more advanced autoencoders such as for image processing for medical images and clinical data for instance. This is because, one may have even additional data for the combination or plugin approach, by extending the vector $\mathbf{x}_i$ but with a part not reduced by the autoencoder. Let denote:

$$\hat{y}_{ik}(\mathbf{h}(\mathbf{x}_i)) = \boldsymbol{\beta}_k^T \mathbf{h}(\mathbf{x}_i) - \boldsymbol{\gamma}_k^T \mathbf{s}_i - b_y^k \,.$$

Here, the index k is removed in case of univariate target variable. An additional independent variables is denoted $\mathbf{s}_i$. This leads to three general models with $\boldsymbol{\gamma}_k$ the coefficients vectors for the additional vector. These models are very different because the informations for the hidden layer are not the same, either it is a reduction for the space of the independent variables without knowledge of the target variable, either a reduction for the space of the independent variables but with the knowledge of the target variable. When y_i is a vector, the functions minimized are written with $\hat{\mathbf{y}}_i$ with components $\hat{y}_{ik}$, such as the new solutions look for a minimization from quadratic function Table 2.1, just below.

Table 1: Extended example with one hidden layer

Approach	Criterion
Direct	$\underset{\mathbf{B},\boldsymbol{\Gamma},b_y,\mathbf{W}_h,b_h}{\operatorname{argmin}} \sum_i L^{\text{out}}\left(\mathbf{y}_i,\ \hat{\mathbf{y}}_i(\mathbf{h}(\mathbf{x}_i))\right)$
Plugin	$\underset{\mathbf{W}_x,\mathbf{W}_h,b_h,b_x}{\operatorname{argmin}} L^{\text{in}}\left(\mathbf{x}_i,\ \hat{\mathbf{x}}(h(\mathbf{x}_i))\right)$ $\underset{\mathbf{B},\boldsymbol{\Gamma},b_y}{\operatorname{argmin}} \sum_i L^{\text{out}}\left(\mathbf{y}_i,\ \hat{\mathbf{y}}_i(\hat{\mathbf{h}}(\mathbf{x}_i))\right)$
Combined	$\underset{\mathbf{W}_x,\mathbf{W}_h,b_h,b_x,\mathbf{B},\boldsymbol{\Gamma},b_y}{\operatorname{argmin}} \sum_i L^{\text{in}}\left(\mathbf{x}_i,\ \hat{\mathbf{x}}(\mathbf{h}(\mathbf{x}_i))\right)$ $+\lambda \sum_i L^{\text{out}}\left(\mathbf{y}_i,\ \hat{\mathbf{y}}_i(\mathbf{h}(\mathbf{x}_i))\right)$

Note that here $\mathbf{B}$ aggregates the $\boldsymbol{\beta}_k$ as next subsection, and $\boldsymbol{\Gamma}$ also for the vectors $\boldsymbol{\gamma}_k$, because there are matrices in the multivariate case. In the more usual case, one may have the loss just quadratic, as a mean squared error, with a sum on the index k for each component of $\mathbf{y}_i$, or more generally the logarithm of gaussian distribution with unknown covariance matrix $\boldsymbol{\Omega}$, with $K(\boldsymbol{\Omega})$ the remaining terms,

$$\begin{aligned} L^{\text{in}}\left(\mathbf{x}_i,\ \hat{\mathbf{x}}(h(\mathbf{x}_i))\right) &= \| \mathbf{x}_i - \hat{\mathbf{x}}(h(\mathbf{x}_i)) \|^2 \\ L^{\text{out}}\left(y_i,\ \hat{y}_i(\hat{\mathbf{h}}(\mathbf{x}_i))\right) &= -0.5 \| \mathbf{y}_i - \hat{\mathbf{y}}_i(\hat{\mathbf{h}}(\mathbf{x}_i)) \|_\Omega^2 + K(\Omega) \,. \end{aligned}$$

An alternative loss is a multivariate copula in order to model more complex dependences. For one component and a quadratic term, one ends with the least squared regression associated to the classical autoencoder.

2.2 Linear regression and pca

Regression is widely used and studied since many decades, it is even more useful today with deep models. For the selection of the columns, many methods have been proposed but pca is studied despite its known limits. A selection of the colums which is separated from the regression inference is risky and may lead to lower predictive properties. Following [1], let's have the matrix of responses denoted $Y = [y_1, \cdots, y_n]^T$, a response vector of length q denoted y_i, the $p \times q$ matrix of coefficients denoted $B = [\beta_1, \cdots, \beta_p]T$ and the $n \times q$ error matrix denoted E, the usual design matrix denoted $X = [x_1, \cdots, x_p]$ where each column variable vector $n \times 1$ is denoted x_j. For lighter notations, the columns of X are supposed to be centred. The classical regression model and the principal component regression are presented below.

- **Regular regression:** The multivariate regression is defined as the following model:

  $$\hat{y}_{ik}(\mathbf{x}_i) = \boldsymbol{\beta}_k^T \mathbf{x}_i$$

  In a matricial notation, one gets that $Y = XB + E$. The loss function is just,

  $$\begin{aligned} \ell(B) &= \sum_i L^{\text{out}}\left(y_i,\ \hat{y}_i(\mathbf{x}_i)\right) \\ &= \sum_i \| \mathbf{y}_i - \hat{\mathbf{y}}_i(\mathbf{x}_i) \|^2 \ . \end{aligned}$$

  With an estimator for B denoted $\hat{B}$ and a new single p-dimensional datum x_0, one gets the usual solution $\hat{B} = (X^T X)^{-1} X^T Y$ by solving the least squared criterion, called M_0 with estimated parameter $\hat{B}$, is computed for test samples. The predicted $q \times 1$ dimensional response is as follows.

  $$\hat{y}_0^T = x_0^T (X^T X)^{-1} X^T Y$$

If $p < n$ and the inverse $(X^TX)^{-1}$ exists, this leads to a relevant solution. If the inverse of (X^TX) does not exist or numerically ill, then a penalization is the usual correction, such as in ridge or lasso regressions. Alternative approaches are clustering the columns of the design matrix or constructing new synthetic columns with the more important information from the columns in order to help the regression.

- **Regression following pca plus cross-validation:** For reducing the number of variables, principal component regression or pca regression is introducing a pca of the data matrix. This is possibly denoted, $Z_k = XR_k$. Here Z_k is an $n \times k$ matrix with only k columns instead of p, thus $k \leq p$ and $Z_k = [z_1, \cdots, z_k]$. The reduction matrix is R_k may be any able to preserve the information from X while allowing the regression. This can be a clustering matrix, a random projection or the first k eigenvectors of the sample covariance matrix from the columns when this is pca regression [1]. At the end, the regression has the same form of solution than just above, when replacing X by Z_k in the criterion, $Y = Z_kB + E$. For this model, the component with largest eigenvalues are kept. Their exact number is found via cross-validation when the penalized criterion, called M_k with estimated parameter $\hat{B}$, is computed for test samples.
- **Regression following pca plus lasso:** The problem of pca regression is that Z_k is found after X without the knowledge of Y, hence may ask for additional components which are not useful but have more weights in the pca. This can be found by a full pca of the design matrix, followed by a lasso regression, in order to find the best components for the regression. This is written as follows, with $Z_p = XR_p$. Such that one may get a solution with the corresponding penalized least squared criterion, called M_p with estimated parameter $\hat{B}$. This is written with a l1 distance for the full matrix B, but it would be more exact to rewrite B as S_pB, where $S_p = Diag(s_p)$ is a diagonal matrix from diagonal binary elements $s_p = (s_{(1)}, \cdots, s_{(k)}, \cdots, s_{(p)})$ with $s_{(p)} \in \{0, 1\}$. Let also add the penalization for s_p, such the criterion is named M_p^{01} with estimated parameter $\hat{B}$ and $\hat{s}_p$. Note that bayesian models exist for this optimization when B is a vector, in the literature. The limit with this approach is this double parameterization which asks for a dedicated algorithm, and to require to compute the full pca of the design matrix, which is not practical currently for large matrices. A way to relax the binary variables is to parameterize with $s^u_{(p)} = 1 - \sigma(u_{(p)})$ with $u_{(p)} \in \mathbb{R}$ aggregated in the vector s^u_p, such as the new matrix involved is $S^u_p = Diag(s^u_p)$. Here $u_{(p)}$ is continuous and aggregated in the vector u_p, while the penalization is now for this vector. With $\sigma(.)$ the usual sigmoidal function, it is clear that if $u_{(p)}$ is near zero, then $u_{(p)}$ is also small, and it kept bounded for other values. The new solution solve for the corresponding criterion named M^σ_p with estimated parameter $\hat{B}$ and $\hat{u}_p$. An alternative and way around to such a selection of the component is by combining the two criteria instead of having one following the other.
- **Regression combined with pca:** For the combination of regression with pca, following the approach with autoencoder, we get that:

$$
\begin{aligned}
(\hat{Z}_k, \hat{B}, \hat{R}_k) &= \underset{Z_k, B, R_k}{\operatorname{argmin}} D(Z_k, B, R_k) \\
D(Z_k, B, R_k) &= A(Z_k, B) + \tau\, B(Z_k, R_k) \\
&\qquad + C(B, R_k) \\
A(Z_k, B) &= \|Y - Z_kB\|^2_F \\
B(Z_k, R_k) &= \|Z_k - XR_k^T\|^2_F \\
C(B, R_k) &= \lambda_B|B| + \lambda_R|R_k|\,.
\end{aligned}
$$

  For this criterion we supposed that R_k is at least approximatively orthogonal, it is called $\tilde{M}^c_k$ with estimated parameter $\hat{B}$, $\hat{R}_k$ and $\hat{Z}_k$.

It is recognized the three approaches (direct, plugin and combined) which have been introduced at the beginning of first section. Alternative solutions have been given for better completeness. The models are summarized in the table below.

Name	Criterion
M_0	$\operatorname{argmin}_B \|\|Y - XB\|\|^2$
M_k	$\operatorname{argmin}_B \|\|Y - Z_kB\|\|^2 + \lambda \|B\|$
M_p	$\operatorname{argmin}_B \|\|Y - Z_pB\|\|^2 + \lambda \|B\|$
M^{01}_p	$\operatorname{argmin}_{B,s_p} \|\|Y - Z_pS_pB\|\|^2 + \lambda \|B\| + \lambda_s \|s_p\|$
M^σ_p	$\operatorname{argmin}_{B,u_p} \|\|Y - Z_pS^u_pB\|\|^2 + \lambda \|B\| + \lambda_u \|u_p\|$
M^c_k	$\operatorname{argmin}_{Z_k,B,R_k} D(Z_k, B, R_k)$

Note that the model M_p recalls the model M_0 when the same lasso penalization is added: this is a regular lasso regression except that the matrix columns are not dependent. If $B = B_1B_2$ with $B_1 \in \mathbb{R}^{p\times k}$ and $B_2 \in \mathbb{R}^{k\times q}$, it is retrieved the pca regression if the matrix B_1 is orthogonal with $B_1 = R^T_k$, this justified a reduced-rank procedure. Nextafter, it is discussed how to introduced other constraints in order to improve the model, in particular for new data.

2.3 Constraints

The main constraints we are interested on are penalization from pca, lasso regression and reduced-rank regression. In the first case, there is pca with lasso regression which is available and regression with l1 penalization also, hence this is not discussed further. In the other hand, it is required herein also a reduced-rank approach: a solution is written in the literature at the reference [2] for the multivariate regression, while some variants of the idea exist.

3 Inference procedures for linear regression associated to pca

The model is fitted when combining pca and regression as explained in the introduction. It is compared with the existing method from the literature. The algorithm for parameters estimation is proposed after derivation w.r.t. to the matrix in the model.

3.1 Inference

The inference just ask for derivative w.r.t. the matrices in the model as follows. First, let approximate the L1 distance in the penalization with the trick for majorization minimization which provides a L2 distance with weights instead. The criterion is rewritten as follows.

$$\begin{aligned}\tilde{D}(Z_k, B, R_k) &= A(Z_k, B) + \tau\, B(Z_k, R_k) + \tilde{C}(B, R_k)\\ \tilde{C}(B, R_k) &= \lambda||W \odot B||_F + \lambda_k||W_k \odot R_k||_F\,.\end{aligned}$$

Thus, one may get the following results from some usual algebra tricks, equivalence of the norm and a product plus trace, and linearity or other properties of the trace operator.

- First, let solve without orthogonal or reduced-rank constraints.

$$\begin{aligned}A(Z_k, B) &= tr\left\{(Y - Z_k B)^T (Y - Z_k B)\right\}\\ &= trY^TY - trY^TZ_kB - trB^TZ_k^TY\\ &\quad + trB^TZ_k^TZ_kB\,.\end{aligned}$$

$$\begin{aligned}B(Z_k, R_k) &= tr\left\{(Z_k - XR_k)^T (Z_k - XR_k)\right\}\\ &= trZ_k^TZ_k - Z_k^TXR_k - R_k^TX^TZ_k\\ &\quad + R_k^TX^TXR_k\,.\end{aligned}$$

After that, let find the derivatives:

$$\begin{aligned}\frac{\partial}{\partial B}\tilde{D}(Z_k, B, R_k) &= -Z_k^TY - Z_k^TY + 2Z_k^TZ_kB\\ &\quad -2Z_k^TY + 2Z_k^TZ_kB + \lambda W\\ \frac{\partial}{\partial Z_k}\tilde{D}(Z_k, B, R_k) &= -YB^T - YB^T + 2Z_kB^TB\\ &\quad +2\tau Z_k - 2\tau XR_k\\ \frac{\partial}{\partial R_k}\tilde{D}(Z_k, B, R_k) &= -\tau X^TZ_k - \tau X^TZ_k\\ &\quad +2\tau X^TXR_k + \lambda_k W_k\end{aligned}$$

This may lead to the following updates, which need to be validated numerically:

$$\begin{aligned}B &= (Z_k^TZ_k - \lambda W)^{-1}Z_k^TY\\ R_k &= (X^TX + \lambda_k W_k)^{-1}X^TZ_k\\ Z_k &= (YB^T + \tau XR_k)(B^TB + \tau I)^{-1}\,.\end{aligned}$$

- After this, the sparse lasso, the orthogonality constraints or the constraints for a reduced-rank model may be added in order to improve the results.

The extension to glm for the criteria above replaces the sum of squares by the logarithm of the exponential function, a new algorithm for fitting is deduced here from the resulting weighted regression after a taylor serie in the case of univariate, otherwise with an additional sum in case of independence of the components.

3.2 Related approaches

In the literature, some methods share some common foundation for clustering or regression but look like different. These methods are listed just after.

- The method named Discriminative K-SVD (D-KSVD) [3] is an extension of the K-SVD algorithm for dictionary learning which introduces an error term from the classification into the criterion as follows,

$$\underset{D,W,X}{\operatorname{argmin}} C_{\text{DKS}}(D, W, X)\,,$$

where,

$$C_{\text{KS}}(D, W, X) = ||Y^T - X^TD^T||^2 + \gamma||H - WX||^2\,,$$

and the norm constraints,

$$\forall i,\ ||x_i||_0 < T_0\,.$$

Here the matrix H is from the a priori classification, this is the label matrix with one if the datum is in the class and zero otherwize. Instead of be part of the inference procedure it remains constant and known.

- The method in [4], as a unified framework for discrete spectral clustering improves the first given criterion, formula (3) at page 2274,

$$\min_{F,Q,Y} C_{\text{USC}}(F,Q,Y)\,,$$

where, the constraints are,

$$F^TF = I, F^TF = I, \text{ and } Y \in \{0,1\}^{n\times k}\ ,$$

while,

$$\begin{aligned} C_{\text{USC}}(F,Q,Y) &= Tr(F^TLF) + \alpha||F - YQ^T||_F^2 \\ &\doteq ||L - FF^T||_F^2 + \alpha||F - YQ^T||_F^2\,. \end{aligned}$$

This criterion was extended in this research for a better results via advanced algebra, which may be used here as a perspective.

- The method from signal processing literature named deeply transformed subspace clustering [5] is defined after the idea of learning transform (TL) with a deep version associated to terms from subspace clustering (SC) in order to cluster the new matrix obtained from TL. This is written in the non deep version,

$$\begin{aligned} &\underset{T,Z,C}{\text{argmin}} ||TX - Z||_F^2 + \lambda(||T||_F^2 - \log\det T) + \mu||Z||_1 \\ &+\gamma\textstyle\sum_i ||z_i - Z_{i^c}c_i||_2^2 + T(C)\,. \end{aligned}$$

In the deep case, the matrix T is replaced with the matrix $T_3T_2T_1$ from the idea of a previous model such as deep semi-nmf from the machine learning literature. A sum then replaces the term for the regularisation of T for each T_i separately. Note that each T_i has a different dimensionality which leads to the deep transformation by a non linear constraint at each level of the product for the new matrix T.

Several clustering methods with matricial sum in two parts from the statistical literature are presented in [6] for instance.

4 Conclusion

Herein, it has been proposed a family of models in order to improve generalized linear models when the independent variables need a reduction or a nonlinear transformation. This family is expected to include several models recently introduced in deep learning, such that a perspective is further experiments with real data and a better understanding of the bias an variances. Informative complements are a gradient descent for the optimization and a boostraping for the bias and the mse. Reduced rank semi-nmf may also reduce the latent matrix with no information for clustering in large dimensions, as another perspective.

References

[1] Inge Koch and Kanta Naito, "Prediction of multivariate responses with a selected number of principal components", *Computational Statistics & Data Analysis*, vol. 54, no. 7, pp. 1791–1807, 2010.

[2] Alan Julian Izenman, "Reduced-rank regression for the multivariate linear model", *Journal of Multivariate Analysis*, vol. 5, no. 2, pp. 248–264, 1975.

[3] Igor Kviatkovsky, Moshe Gabel, Ehud Rivlin, and Ilan Shimshoni, "On the equivalence of the lc-ksvd and the d-ksvd algorithms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 411–416, 2017.

[4] Yang Yang, Fumin Shen, Zi Huang, and Heng Tao Shen, "A unified framework for discrete spectral clustering", in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. 2016, IJCAI'16, pp. 2273–2279, AAAI Press.

[5] Jyoti Maggu, Angshul Majumdar, Emilie Chouzenoux, and Giovanni Chierchia, "Deeply transformed subspace clustering", *Signal Processing*, vol. 174, pp. 107628, 2020.

[6] Angelos Markos, Alfonso D'Enza, and Michel van de Velden, "Beyond tandem analysis: Joint dimension reduction and clustering in r", *Journal of Statistical Software, Articles*, vol. 91, no. 10, pp. 1–24, 2019.